



POLITECNICO
MILANO 1863

DIPARTIMENTO DI ARCHITETTURA,
INGEGNERIA DELLE COSTRUZIONI
E AMBIENTE COSTRUITO

DOCTORAL PROGRAM IN
ARCHITECTURE, BUILT ENVIRONMENT AND CONSTRUCTION ENGINEERING

POINT CLOUD PROCESSING
FOR PHYSICAL ACCESSIBILITY EVALUATION
IN HISTORICAL URBAN ENVIRONMENTS.
THE CASE OF UNESCO SITE OF SABBIONETA.

Doctoral Dissertation of:
Daniele Treccani

Supervisor:

Prof. Andrea Adami

Co-Supervisor:

Prof. Lucía Díaz Vilariño

Tutor:

Prof. Marco Scaioni

The Coordinator of ABC Doctoral Program:

Prof. Marco Scaioni

Year 2022 – Cycle XXXIV

*To my family,
who know I live with my head in the clouds... of points.*

Preface

Abstract

The management of physical accessibility in the urban environment is a topic of great interest at national and international level, where it is addressed by both United Nations and European Commission. At national level, the topic is addressed by Italian legislative framework, but also by specific legislation issued by Regions through the predisposition of Plans for the Elimination of Architectural Barriers (PEBA). It is also worth mentioning the recent push given by the PNRR National Recovery and Resilience Plan towards sustainability and inclusion, where accessibility is recalled both in Mission 2 and 3.

The management of accessibility could be fostered by the use of Information Technology. The support of Geomatics could play a crucial role while dealing with physical accessibility, intended as the ability to move in a physical environment independently of the motor abilities of the individuals (e.g., the users of baby prams, wheelchairs, walking sticks). Plus, it is precisely when managing physical accessibility in historic urban settings that the topic becomes even more interesting and challenging. Indeed, the non-standard organisation of historical urban scenarios is a feature that makes more difficult to identify approaches that exploit artificial intelligence to analyse spatial data and generate an inventory of urban elements and issues specifically related to accessibility.

Specifically, in this thesis two different approaches were proposed in order to analyse and process a point cloud of a historical city, in order to extract useful information for a better management of physical accessibility. The historic city analysed was Sabbioneta, UNESCO site since 2008, and the element automatically detected were the roads and the sidewalks of the city. The work started with a survey of the physical form of the city, conducted with a Mobile Mapping System: Leica Pegasus:Two. The accuracy of the resulting point cloud was validated and it was used as the beginning dataset for the subsequent steps of the research. Two processing methods were defined, a first one, developed by implementing different algorithms and original pieces of code, named knowledge-based method, and a second one that exploited a Random Forest classifier through a Machine Learning (ML) approach. In both the cases the point cloud of Sabbioneta was semantically segmented into urban ground elements. The two methods were also tested on other datasets, acquired on small portions of two Italian cities, Mantova and Domodossola, and on a Portuguese city, Porto. In all cases, the ML method showed better results; an average accuracy of 89% could be assessed for the ML method, while 76% could be assessed for the knowledge-based method. ML method was also considered to be the most promising due to its better adaptability to the different urban contexts tested and the greater simplicity in calibrating the operating parameters.

The focus of the research shifted at this point to the greater characterization of sidewalks, which were identified as the preferred routes of movement for all city users, and thus the objects on which to develop accessibility management. The computa-

tions were made on consequent portions of sidewalk, one every 2 metres. Specifically attributes computed were: the width, the elevation respect the road, the slopes in the two main directions, and the material used for the paving. From the spatial position of sidewalks, a vector network made of nodes and edges was generated, representing the sidewalks of the city. The test area was only Sabbioneta. A total of 1780 nodes and 1720 edges were generated, and the attributes were stored within the edges representing sidewalks in the network.

The shapefile generated in the last phase of the work was then used to demonstrate its multiple possible uses. In fact, it was used to create thematic maps, it was exploited for the calculation of accessible routes within the city, it was tested for its possible use on a webGIS platform, it was evaluated by OpenStreetMap Italian community and will be uploaded to OpenStreetMap to update the existing map data. The various uses demonstrated the flexibility of the generated file, and its great usefulness for different users at various levels (public entities, urban planners, social associations, citizens, tourists) and for different purposes (city management, plan interventions and maintenance, develop touristic routes, etc.).

This research work presented a complete method for the inventory and management of the sidewalk network of a historic city, with a focus on accessibility related analysis, starting from spatial data. From this work emerges the importance and benefits that can be generated by interdisciplinary research as a meeting point of different fields. In this specific case, it was necessary to involve experts from the field of Geomatics, but also data processing and accessibility experts in order to develop processes that can lead to a technological and methodological improvement of some urban management procedures.

Future developments in this research could investigate its use for automated implementation of urban management plans (e.g., PEBA); its benefits for BIM/GIS integration processes and for the development of 3D City Models; the use of Deep Learning techniques for the characterization of urban pavements; the possibilities of inventorying other urban elements as well and their impact on the management of urban physical accessibility.

Abstract in Italiano

La gestione dell'accessibilità fisica nell'ambiente urbano è un tema di grande interesse, trattato a scala internazionale sia dalle Nazioni Unite che dalla Commissione Europea. A livello nazionale, il tema è affrontato dal quadro legislativo italiano, ma anche da normative specifiche emanate dalle Regioni e attraverso la predisposizione di Piani per l'Eliminazione delle Barriere Architettoniche (PEBA). Da segnalare anche la recente spinta del PNRR (Piano Nazionale di Recupero e Resilienza) verso la sostenibilità e l'inclusione, dove l'accessibilità è richiamata sia nella Missione 2 che nella 3.

Appare evidente che la gestione dell'accessibilità potrebbe essere favorita dall'uso delle Tecnologie dell'Informazione (IT). Il supporto della Geomatica potrebbe giocare un ruolo fondamentale per quanto riguarda l'accessibilità fisica, intesa come capacità di muoversi in un ambiente fisico indipendentemente dalle capacità motorie degli individui (ad esempio, gli utilizzatori di passeggini, sedie a rotelle, bastoni da passeggio). Inoltre, è proprio nella gestione dell'accessibilità fisica in contesti urbani storici che l'argomento

diventa ancora più interessante e stimolante. Infatti, l'organizzazione non standard degli scenari urbani storici è una caratteristica che rende più difficile l'individuazione di approcci che sfruttino l'intelligenza artificiale per analizzare i dati spaziali e generare inventari di elementi urbani e problematiche specificamente legate all'accessibilità.

Nello specifico, in questa tesi sono stati proposti due approcci diversi per analizzare e processare una nuvola di punti di una città storica, al fine di estrarre informazioni utili per una migliore gestione dell'accessibilità fisica. La città storica analizzata è Sabbioneta, sito UNESCO dal 2008, e gli elementi individuati automaticamente sono legati all'impianto urbano della città.

Il lavoro è iniziato con un rilievo della forma fisica della città, condotto con un sistema di mobile mapping: Leica Pegasus:Two. L'accuratezza della nuvola di punti ottenuta è stata validata, la nuvola di punti stessa è poi stata utilizzata come set di dati iniziale per le fasi successive della ricerca. Sono stati definiti due metodi di elaborazione: un primo, sviluppato implementando diversi algoritmi e linee di codice originali, denominato metodo knowledge-based, e un secondo che ha sfruttato un Random Forest classifier attraverso un approccio Machine Learning (ML). In entrambi i casi la nuvola di punti di Sabbioneta è stata segmentata semanticamente individuando i diversi elementi urbani presenti sulla superficie orizzontale della città. I due metodi sono stati testati anche su altri dataset, ottenuti con diversi strumenti di rilievo, acquisiti su piccole porzioni di due città italiane, Mantova e Domodossola, e su una città portoghese, Porto. In tutti i casi, il metodo ML ha mostrato risultati migliori; è stato possibile valutare un'accuratezza media dell'89% per il metodo ML, mentre il 76% per il metodo knowledge-based. Il metodo ML è stato considerato il più promettente anche per la sua migliore adattabilità ai diversi contesti urbani testati e per la maggiore semplicità nella calibrazione dei parametri operativi.

L'attenzione della ricerca si è spostata a questo punto sulla maggiore caratterizzazione dei marciapiedi, che sono stati identificati come le vie di circolazione preferenziali per tutti gli utenti della città, e quindi gli oggetti su cui impostare la gestione dell'accessibilità. In particolare, gli attributi calcolati sono stati: la larghezza, la quota rispetto alla strada, le pendenze nelle due direzioni principali e il materiale utilizzato per la pavimentazione. Ad oggi, i suddetti dati restano più fruibili se, anziché strutturati come una nuvola di punti, vengono gestiti in ambiente GIS. Pertanto dalla posizione spaziale dei marciapiedi, è stata generata una rete vettoriale composta da nodi e linee, che rappresenta i marciapiedi della città. L'area di test in questo caso è stata limitata alla città di Sabbioneta. In totale sono stati generati 1780 nodi e 1720 linee, e gli attributi sono stati memorizzati all'interno delle linee che rappresentano i marciapiedi nella rete.

Lo shapefile generato nell'ultima fase del lavoro è stato poi utilizzato per dimostrare le sue molteplici possibilità di utilizzo. Infatti, è stato utilizzato per creare mappe tematiche, è stato sfruttato per il calcolo dei percorsi accessibili all'interno della città, è stato testato per il suo possibile utilizzo su una piattaforma webgis, è stato valutato dalla comunità italiana di OpenStreetMap e sarà caricato su OpenStreetMap per aggiornare i dati cartografici esistenti. I vari utilizzi hanno dimostrato la flessibilità del file generato e la sua grande utilità per diversi utenti a vari livelli (enti pubblici, urbanisti, associazioni sociali, cittadini, turisti) e per diversi scopi (gestione della città, pianificazione di interventi e manutenzioni, sviluppo di percorsi turistici, ecc.).

Questo lavoro di ricerca ha presentato un metodo completo per la gestione dell'accessibilità a partire da dati spaziali, tale metodo è stato testato qui su marciapiedi, ma può essere esteso, con opportune modifiche, a molti altri aspetti legati a una gestione inclusiva del patrimonio urbano storico. Da questo lavoro emergono l'importanza e i benefici che possono essere generati dalla ricerca interdisciplinare come punto d'incontro di campi diversi. In questo caso specifico, è stato necessario coinvolgere esperti di Geomatica, ma anche di elaborazione dati e di accessibilità, per sviluppare processi che possano portare a un miglioramento tecnologico e metodologico di alcune procedure di gestione urbana.

Gli sviluppi futuri di questa ricerca potrebbero riguardare il suo utilizzo per l'implementazione automatizzata di piani di gestione urbana (ad esempio, i PEBA); i suoi vantaggi per i processi di integrazione BIM/GIS e per lo sviluppo di 3D city models; l'uso di tecniche di Deep Learning per la caratterizzazione delle pavimentazioni urbane; le possibilità di inventariare anche altri elementi urbani e il loro impatto sulla gestione dell'accessibilità fisica urbana.

Note for the reader

The term: accessibility

The term accessibility can refer to a multitude of different scenarios, which go beyond just physical accessibility. However, throughout this thesis work, the term "accessibility" will be related just to physical accessibility. This work addressed, described and developed methods that collected and extracted information that was related to physical accessibility.

The reference system

For all spatial representation in this thesis work, from the point clouds database to the generated shapefile, and also referring to any map here presented, the coordinates are referenced in the World Geodetic System WGS84, with cartographic projection UTM zone 32N (EPSG code 32632). The only exception is the data for the Porto dataset, described and tested in section 4.6.4 and 5.3.2, which are referenced in the geodetic system with Datum: European Terrestrial Reference System 1989 (ETRS89) and projection Portugal TM06 (EPSG code 3763).

The Images and maps of this thesis

All the images, photos, and maps provided in this thesis work are intended as taken, made and elaborated by the author unless otherwise specified in the caption.

Acknowledgements

When I look back at these almost four years of my life, I realize that the PhD program has been an experience that required commitment and sacrifice by my part, but it has allowed me to realize some of my life goals. First and foremost, since high school the disciplines of Geomatics have fascinated me, and since then I had in my intentions to deepen my knowledge with a doctorate. I am glad I had the chance to fulfill this ambition. Secondly, this experience has been enriched by the possibility of carrying out architectural surveys with state-of-the-art instrumentation, dealing with cultural heritage of the highest level in Italy and the rest of the world, and being in contact with high-profile international researchers. For this, I certainly have to thank the HeSuTech group of the MantovaLAB at Mantova Campus.

I would like to start with a special thanks to my supervisor, professor Andrea Adami. His constant presence, guidance, and suggestions have been invaluable and fundamental to the conduct of this thesis. The opportunity to have a continuous peer-to-peer discussion with him, and his encouragements have been decisive in allowing me to achieve objectives that I thought I could not achieve.

I would like to express then my warmest gratitude to my co-supervisor, professor Lucía Díaz Vilariño. Her support was crucial, both in virtual mode during the pandemic period and in presence, when I was hosted at the University of Vigo. She has always allowed me to exchange views with her, managing to find solutions together in the most challenging moments of the research. For this reasons, I want to thank her very much.

I would also like to thank the members of the HeSuTech group, professor Luigi Fregonese, professor Andrea Adami, Jacopo Helder, Olga Rosignoli and Laura Taffurelli, for hosting me in the laboratory, for the many ideas exchanged about work and life topics, and for the advice given to me throughout the thesis.

Many thanks to Sebastiano Marconcini, whose expertise supported me as I was approaching the topic of accessibility in historic environments.

I would also like to thank Leica Geosystem Italia, in the person of Valerio Brunelli, for providing the Leica Pegasus:Two instrument.

Finally, I would like to thank my parents, Gianpaolo and Sandra, and my brother Nicola for always being there. Lorenzo, for walking with me all the way and always making me feel his support. Fabiana, for the help with English grammar and spell check. And lastly, I would like to thank all the people who have always encouraged me and without whom I would not be the man I am today.

Contents

List of Acronyms	XI
List of Figures	XIII
List of Tables	XIX
1 Introduction	1
1.1 The context: urban physical accessibility	4
1.2 Research problem: 3D point clouds and accessibility	7
1.3 The research project	9
1.3.1 The workflow	10
1.3.2 General contributions and advancements respect the state of the art	13
1.4 Organization of the contents	13
2 State of the art	15
2.1 Theoretical background on Point Clouds	15
2.1.1 Point cloud data	15
2.1.2 Point cloud Geometric features	16
2.2 Point cloud semantic segmentation	19
2.2.1 Segmentation-only methods	20
2.2.2 Classification methods, including Machine Learning	22
2.2.3 Deep Learning methods	23
2.2.4 Applications	24
2.3 Semantic segmentation of point clouds of urban areas	24
2.3.1 Applications	24
2.3.2 Strategies	25
2.3.3 Detected objects	26
2.3.4 Benchmark Datasets	28
2.4 Semantic segmentation of point clouds for urban accessibility	29
3 Data acquisition with a Mobile Mapping System	35
3.1 Case study: UNESCO site of Sabbioneta	35
3.1.1 Urban structure of Sabbioneta and its historical evolution	38
3.1.2 Urban paving in Sabbioneta	40
3.2 Theoretical background: Mobile mapping technologies	43
3.2.1 On-board sensors	44
3.2.2 Direct georeferencing	47

3.2.3	Portable Mobile Mapping Systems	47
3.3	Exploitation and validation of the geometric survey	49
3.3.1	The instrument	51
3.3.2	On-field survey	53
3.3.3	Survey data processing	56
3.3.4	Data validation	58
4	Knowledge-based approach for segmentation	67
4.1	Introduction to the method	67
4.2	Pre-processing	70
4.2.1	Geometric features computation and selection	70
4.2.2	Corner identification	72
4.2.3	Region Of Interest selection	74
4.3	Segmentation	79
4.3.1	Feature selection	80
4.3.2	Number of clusters	83
4.3.3	Candidate selection and refinement	85
4.4	Semantic labelling	87
4.4.1	"Sidewalks" class candidate check	87
4.4.2	Labelling 'road' and 'other' class points	96
4.5	Performances evaluation	97
4.6	Results	98
4.6.1	Sabbioneta	99
4.6.2	Mantova	103
4.6.3	Sacro Monte di Domodossola	103
4.6.4	Porto	104
4.7	Discussion	106
5	Segmentation exploiting Machine Learning	109
5.1	Introduction to the method	109
5.1.1	Machine Learning, theoretical references	110
5.2	Preliminary phases	112
5.2.1	The Ground Truth for Sabbioneta	112
5.2.2	Training dataset	118
5.2.3	Peculiarities of the Sabbioneta dataset	119
5.2.4	Model selection	121
5.2.5	Feature selection	122
5.3	Test phases	127
5.3.1	Preliminary tests	128

5.3.2	First approach: ML model for the classification of urban elements	129
5.3.3	Second approach: ML model for the classification of paving materials	132
5.4	Discussion and comparison between knowledge-based and ML methods	134
6	Sidewalk characterization: attributes computation	137
6.1	Introduction to the attributes computations	137
6.2	Paving material detection	138
6.2.1	Method I: thresholds-based	139
6.2.2	Method II: Machine Learning model	139
6.2.3	Results and comparison	140
6.2.4	Final considerations	140
6.3	Other geometric attributes	143
6.3.1	Sidewalk width	143
6.3.2	Sidewalk relative elevation	144
6.3.3	Sidewalk slope	145
6.4	Results	146
6.4.1	Attributes	146
6.4.2	Validation	148
6.5	Discussion	150
7	Sidewalk network vectorization and some possible implementations	151
7.1	Motivation for the vectorization	151
7.2	Vectorization method	152
7.2.1	Representation: points or lines	153
7.2.2	Generation of the Sidewalk network	156
7.2.3	Linking the attributes	157
7.2.4	Results	158
7.3	Possible uses	160
7.3.1	Thematic maps	160
7.3.2	Path computation	165
7.3.3	WebGIS platform	169
7.3.4	Upload on OpenStreetMap	172
7.4	Discussion	173
8	Conclusions	177
8.1	Critical discussion	178
8.1.1	The survey	178
8.1.2	The knowledge-based segmentation	179

Contents

8.1.3	The segmentation exploiting supervised ML	180
8.1.4	Comparison of the two segmentation methods	181
8.1.5	The sidewalks attributes computation	183
8.1.6	The vectorisation of sidewalk network and its applications	184
8.1.7	Summary of the achievements	185
8.2	Perspectives for the future	186
8.2.1	Survey opportunities	186
8.2.2	Accessibility-related possibilities	187
8.2.3	Further accessibility attributes to treat	188
8.2.4	Other implementations for the method	189
8.3	Final comments	190
Bibliography		193
A Semantic Segmentation results		203
A.1	Sabbioneta - Track A	204
A.2	Sabbioneta - Track B	208
A.3	Sabbioneta - Track C	211
A.4	Sabbioneta - Track D	215
A.5	Sabbioneta - Track E	219
A.6	Sabbioneta - Track F	223
A.7	Sabbioneta - Track H	227
A.8	Sabbioneta - Track I	231
A.9	Mantova	235
A.10	Sacro Monte	240
A.11	Porto - 1	245
A.12	Porto - 2	250
A.13	Porto - 3	255

List of Acronyms

BDS	BeiDou navigation Satellite System
GLONASS	GLObal NAvigation Satellite System
NavIC	Navigation with Indian Constellation
QZSS	Quasi-Zenith Satellite System
ADA	American with Disability Act
AI	Artificial Intelligence
ALS	Airborne Laser Scanning
BB	Bounding Box
BIM	Building Information Modeling
CC	CloudCompare
CH	Cultural Heritage
CNN	Convolutional Neural Networks
CRS	Coordinate Reference System
DL	Deep Learning
EPSG	European Petroleum Survey Group
GIS	Geographic Information System
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
GT	Ground Truth
HBIM	Heritage Building Information Modeling
HSV	Hue Saturation Value (color space)
ICT	Information and Communication Technology
IMU	Inertial Measurement Unit
KNN	k-Number of Neighbour
LiDAR	Light Detection and Ranging
MLS	Mobile Laser System / Mobile Laser Scanning
ML	Machine Learning
MMS	Mobile Mapping System
OSM	OpenStreetMap
PAU	Piano di Accessibilità Urbana (Urban Accessibility Plan)

List of Acronyms

PCA	Principal Component Analysis
PEBA	Piano di Eliminazione delle Barriere Architettoniche (Architectural Barrier Removal Plan)
PMMS	Portable Mobile Mapping System
PNRR	Piano Nazionale di Ripresa e Resilienza (National Recovery and Resilience Plan)
POI	Point of Interest
RF	Random Forest
RGB	Red Green Blue (color space)
ROI	Region of Interest
RTK	Real Time Kinematic
SLAM	Simultaneous Localization and Mapping
TLS	Terrestrial Laser Scanning
TS	Total Station
UAS	Unmanned Aerial System
VGI	Volunteered Geographic Information

List of Figures

1.1	Evolution of research interest in Point Cloud processing and accessibility	8
1.2	Evolution of research interest in Point Cloud processing	8
1.3	Scheme of the thesis workflow	11
1.4	Scheme of the thesis in terms of Working Packages of processes . . .	12
2.1	Types of neighbourhood selection in point clouds	17
2.2	Generic point cloud supervised classification workflow	22
3.1	Some photos of the case study: the historical city of Sabbioneta. . . .	36
3.2	A schematic representation of Sabbioneta, its roads and the historical buildings.	37
3.3	Map of the previous surveys and works developed in Sabbioneta by HeSuTech Group.	38
3.4	Map of the pavings in Sabbioneta.	40
3.5	Photos taken in Sabbioneta in several areas around the city centre with a focus on the pavings.	41
3.6	Photos of urban pavings taken in historical centres of some Italian cities.	42
3.7	Picture of the paving of a road in the historical centre of Santiago de Compostela, Spain.	43
3.8	List of some MMS on the market and their characteristics.	45
3.9	An exemplification of a MMS mounted on a car.	46
3.10	An exemplification of how it is conducted a survey with a MMS. . . .	46
3.11	Scheme and equation of georeferencing approach for a MMS.	48
3.12	Comparison of point clouds from three PMMS.	49
3.13	Leica Pegasus:Two and typically mounted sensors.	51
3.14	Technical datasheet of Leica Pegasus:Two.	52
3.15	Leica Pegasus:Two on-site during the survey.	53
3.16	Map of the 10 tracks (or missions) of acquisition in Sabbioneta. . . .	55
3.17	A screenshot from the software Pegasus Manager.	58
3.18	Data validation workflow followed in this work.	59
3.19	A screenshot from the software Pegasus Manager.	60
3.20	The point cloud of the project coloured by Pegasus Manager according to a quality parameter.	60
3.21	Some slices of the point cloud, made in Cyclone, to check the alignment between different tracks.	61
3.22	Map of the GNSS-measured points used in the quality check by HeSuTech laboratory, coloured respect their registration error.	63
3.23	Snapshots of the point cloud, different point density is clearly visible.	64
3.24	Snapshots of the point cloud, cone of shadows in corners.	64
3.25	Snapshots of the point cloud, colour projection errors are visible. . . .	65
3.26	Snapshots of the point cloud, the complete survey	66
4.1	A brief scheme of the workflow presented in the chapter.	68
4.2	A complete scheme of the workflow presented in this chapter. From The survey raw data to the semantically segmented point cloud. . . .	69

List of Figures

4.3	Sample of point clouds of some paving materials: cobblestone, bricks, sampietrini, stone.	71
4.4	A crop of the point cloud with three different features used to colour the points	72
4.5	Crossings analysed in the method: T-shape, X-shape, L-shape.	73
4.6	COmparison of trajectories, from the instrument and from OSM.	75
4.7	Exemplification of the method to project a trajectory point on the ground.	76
4.8	Method to detect a curve on the trajectory and adapt ROI shape.	77
4.9	Exemplification of the generation of a Bounding Box using trajectory points.	78
4.10	First test on Bounding Boxes creation	79
4.11	Second test on Bounding Boxes creation	79
4.12	Example of some ROIs with distance computation	82
4.13	Graph of the Elbow method applied to a ROI	83
4.14	Visual method to define the most suitable k-number of clusters.	84
4.15	The subsequent result of the segmentation method on the same ROI.	87
4.16	Pictures of some paving materials in the case study.	90
4.17	Cumulative probability density functions of several samples of the 4 materials analysed.	91
4.18	Workflow of the paving material characterisation methodology.	92
4.19	Map of the buildings' footprint in Sabbioneta	94
4.20	Scheme of the application of Criterion 2: building presence.	94
4.21	Scheme of the application of Criterion 3: vertical boundary surfaces.	95
4.22	Scheme of the application of Criterion 4: continuity between ROIs.	96
4.23	Example of performance evaluation on a ROI.	98
4.24	Map of the 10 tracks (or missions) of acquisition in Sabbioneta.	99
5.1	Types of ML and their characteristics.	111
5.2	Photos of the 12 classes of materials for ML.	113
5.3	Several screenshots of the point cloud with reference images for the GT for urban element	114
5.4	Point cloud of the whole city, points rendered according to the GT for paving materials.	115
5.5	Comparison of different methods for the selection of training dataset.	118
5.6	Comparison between RGB and HSV colours	123
5.7	Features importance diagram computed for 2 classes: 'sidewalk' and 'road'.	125
5.8	Features importance diagram computed for 4 classes: 'cobblestone', 'stone', 'sampietrini', 'bricks'.	126
5.9	Top view of predictions for track-C of preliminary tests.	134
5.10	Scheme of the updated approach for segmentation, exploiting ML methods.	135
5.11	Top view of the point cloud of the whole Sabbioneta dataset after semantic segmentation with ML method.	136
6.1	Graphic scheme of the computation of the attribute "paving material".	140

6.2	Visual result of the paving material computation with both methods.	142
6.3	Graphic scheme of the computation of the attribute "width".	143
6.4	Graphic scheme of the computation of the attribute "elevation difference".	144
6.5	Graphic scheme of the computation of the attribute "slopes".	145
6.6	Results of the Attributes computed for each sidewalk segment	147
6.7	Map of the position of sidewalks used for the comparison between computed width and measured width.	149
6.8	An example of direct comparison between computed and measured width.	149
7.1	Results from the vectorization of sidewalks on Track-C using PCA to transform each sidewalk segment in line.	154
7.2	Results from the vectorization of sidewalks on Track-C representing each sidewalk segment with a point.	154
7.3	Result of the complete vectorization of Track-C into a sidewalk network.	155
7.4	Graphic scheme of the vectorization of sidewalk segments into points (i.e. edges).	156
7.5	Scheme of the regularization workflow for three types of crossing.	157
7.6	A portion of the city of Sabbioneta, with network edges also across the road.	158
7.7	A screenshot of QGIS with the attributes of a selected edge highlighted.	159
7.8	Result of the vectorization of the whole Sabbioneta dataset.	159
7.9	Thematic map of sidewalks width	161
7.10	Thematic map of sidewalks relative elevation	161
7.11	Thematic map of sidewalks longitudinal slope	162
7.12	Thematic map of sidewalks transverse slope	162
7.13	Thematic map of sidewalks paving materials	163
7.14	Thematic map of sidewalks width accessible or inaccessible	163
7.15	Thematic map of level of severity of sidewalks, in terms of accessibility.	164
7.16	Computation of path considering widths of sidewalks as constraints.	167
7.17	Computation of path considering Z differences between sidewalks and roads as constraints.	168
7.18	Screenshot of the main page of the webGIS platform.	170
7.19	Screenshot of the webGIS platform with a pop-up with sidewalk attributes.	170
7.20	Screenshot of the webGIS platform with information about POI.	171
7.21	Workflow of the sidewalk attributes extraction and vectorization.	174
8.1	General workflow of the thesis.	177
A.1	Confusion matrix and precision metrics for track-A	204
A.2	Confusion matrix and precision metrics for track-A	204
A.3	Confusion matrix and precision metrics for track-A	204
A.4	Map of the path of Track-A	205
A.5	Top view of the point cloud of Track-A after semantic segmentation with the knowledge-based method.	206

List of Figures

A.6	Top view of the point cloud of Track-A after semantic segmentation with ML method.	207
A.7	Confusion matrix and precision metrics for track-B	208
A.8	Confusion matrix and precision metrics for track-B	208
A.9	Map of the path of Track-B	209
A.10	Top view of the point cloud of Track-B after semantic segmentation with the knowledge-based method.	210
A.11	Confusion matrix and precision metrics for track-C	211
A.12	Confusion matrix and precision metrics for track-C	211
A.13	Map of the path of Track-C	212
A.14	Top view of the point cloud of Track-B after semantic segmentation with the knowledge-based method.	213
A.15	Top view of the point cloud of Track-C after semantic segmentation with ML method.	214
A.16	Confusion matrix and precision metrics for track-D	215
A.17	Confusion matrix and precision metrics for track-D	215
A.18	Map of the path of Track-D	216
A.19	Top view of the point cloud of Track-D after semantic segmentation with the knowledge-based method.	217
A.20	Top view of the point cloud of Track-D after semantic segmentation with ML method.	218
A.21	Confusion matrix and precision metrics for track-E	219
A.22	Confusion matrix and precision metrics for track-E	219
A.23	Map of the path of Track-E	220
A.24	Top view of the point cloud of Track-E after semantic segmentation with the knowledge-based method.	221
A.25	Top view of the point cloud of Track-E after semantic segmentation with ML method.	222
A.26	Confusion matrix and precision metrics for track-F	223
A.27	Confusion matrix and precision metrics for track-C	223
A.28	Map of the path of Track-F	224
A.29	Top view of the point cloud of Track-F after semantic segmentation with the knowledge-based method.	225
A.30	Top view of the point cloud of Track-F after semantic segmentation with ML method.	226
A.31	Confusion matrix and precision metrics for track-H	227
A.32	Confusion matrix and precision metrics for track-H	227
A.33	Map of the path of Track-H	228
A.34	Top view of the point cloud of Track-H after semantic segmentation with the knowledge-based method.	229
A.35	Top view of the point cloud of Track-H after semantic segmentation with ML method.	230
A.36	Confusion matrix and precision metrics for track-I	231
A.37	Confusion matrix and precision metrics for track-I	231
A.38	Map of the path of Track-I	232

A.39	Top view of the point cloud of Track-I after semantic segmentation with the knowledge-based method.	233
A.40	Top view of the point cloud of Track-I after semantic segmentation with ML method.	234
A.41	Confusion matrix and precision metrics for Mantova	235
A.42	Confusion matrix and precision metrics for Mantova	235
A.43	Confusion matrix and precision metrics for Mantova	235
A.44	Map of the path of Mantova	236
A.45	Top view of the point cloud of Mantova after semantic segmentation with the knowledge-based method.	237
A.46	Top view of the point cloud of Mantova after semantic segmentation with the ML method.	238
A.47	Top view of the point cloud of Mantova after semantic segmentation with the ML method.	239
A.48	Confusion matrix and precision metrics for Sacro Monte	240
A.49	Confusion matrix and precision metrics for Sacro Monte	240
A.50	Confusion matrix and precision metrics for Sacro Monte	240
A.51	Map of the path in Domodossola.	241
A.52	Top view of the point cloud of Domodossola after semantic segmentation with the knowledge-based method.	242
A.53	Top view of the point cloud of Sacro Monte after semantic segmentation with the ML method.	243
A.54	Top view of the point cloud of Sacro Monte after semantic segmentation with the ML method.	244
A.55	Confusion matrix and precision metrics for Porto-1	245
A.56	Confusion matrix and precision metrics for Porto-1	245
A.57	Confusion matrix and precision metrics for Porto-1	245
A.58	Map of the path in Porto-1.	246
A.59	Top view of the point cloud of Porto-1 after semantic segmentation with the knowledge-based method.	247
A.60	Top view of the point cloud of Porto-1 after semantic segmentation with the ML method.	248
A.61	Top view of the point cloud of Porto-1 after semantic segmentation with the ML method.	249
A.62	Confusion matrix and precision metrics for Porto-2	250
A.63	Confusion matrix and precision metrics for Porto-2	250
A.64	Confusion matrix and precision metrics for Porto-2	250
A.65	Map of the path in Porto-2.	251
A.66	Top view of the point cloud of Porto-2 after semantic segmentation with the knowledge-based method.	252
A.67	Top view of the point cloud of Porto-2 after semantic segmentation with the ML method.	253
A.68	Top view of the point cloud of Porto-2 after semantic segmentation with the ML method.	254
A.69	Confusion matrix and precision metrics for Porto-3	255
A.70	Confusion matrix and precision metrics for Porto-3	255

List of Figures

A.71	Confusion matrix and precision metrics for Porto-2	255
A.72	Map of the path in Porto-3.	256
A.73	Top view of the point cloud of Porto-3 after semantic segmentation with the knowledge-based method.	257
A.74	Top view of the point cloud of Porto-3 after semantic segmentation with the ML method.	258
A.75	Top view of the point cloud of Porto-3 after semantic segmentation with the ML method.	259

List of Tables

1.1	Some reference values for physical accessibility	7
2.1	Summary of scientific articles related to the topic of semantic segmentation of point clouds for urban accessibility.	33
3.1	Description of length and number of points in the point cloud for each acquisition Track.	55
3.2	Registration errors of GNSS-measured points and the point cloud.	62
4.1	Scheme of application of the voting system.	89
4.2	Description of the 4 classes for paving material criterion.	90
4.3	Performance metrics for all the Tracks in Sabbioneta, subdivided into the three classes, with average values.	107
4.4	Performance metrics for the dataset in Mantova and Domodossola, subdivided into the three classes, with average values.	107
4.5	Performance metrics for the three dataset in Porto, subdivided into the three classes, with average values.	107
5.1	Description of the 12 classes for paving materials.	116
5.2	Description of pavings for the 10 Tracks of Sabbioneta.	117
5.3	Results of the tests made for the definition of the most suitable ML classifier.	121
5.4	Selected features for the two ML models.	124
5.5	Summary of all the ML tests made.	127
5.6	Results of the preliminary test	129
5.7	Results of the tests made using only two classes for prediction and focusing on urban elements.	130
5.8	Performance metrics for the ML prediction of 2 classes ('road' and 'sidewalk') for the Tracks.	130
5.9	Accuracies of the tests made using the ML model for the recognition of urban classes ('road' and 'sidewalk') on other datasets	131
5.10	Result of the tests made using only 4 classes for the training dataset.	133
6.1	Results of the computation of paving materials for each Track, using Method I (thresholds-based)	141
6.2	Results of the computation of paving materials for each Track, using Method II (ML)	141
6.3	Results of the Attributes computed for each sidewalk segment.	147
6.4	Comparison between the attribute "width" computed by the method presented in this thesis work and measured on-site by an expert.	150
8.1	Comparison of performances of knowledge-based and ML methods	182

1

Introduction

Physical accessibility in urban centres is a topic of general interest and primary importance, addressed on several levels by both local and national authorities, but also by the European Union, and by international organisations such as the United Nations. At the National level, the relevance of the topic is also remarked by the policy directions provided within the *Piano Nazionale di Ripresa e Resilienza*¹ (PNRR, Recovery and Resilience Plan). PNRR addresses inclusion, accessibility and mobility in several Missions. In Mission 1, Component 3 (M1C3) "*Tourism and Culture*", accessibility of cultural places is recalled at Investment 1.2, that says: "*increasing access to cultural heritage also goes through the full accessibility of cultural places*" and "*the intervention aims to remove architectural, cultural and cognitive sense-perceptual barriers in a set of Italian cultural institutions*". In Mission 2, Component 2 (M2C2) "*Renewable energy, hydrogen, sustainable network and sustainable mobility*", and in Mission 3 "*Infrastructure for sustainable mobility*", there are interventions for mobility, local public transport and rail lines that promote the improvement and accessibility of infrastructure and services for all citizens.

Coherent and comprehensive management of accessibility is not simply a matter of providing ramps next to sidewalks or designing solutions to overcome differences in level without steps. Coherent management must necessarily take into account a series of parameters that allow any user to fully enjoy the urban space. Therefore, the presence or absence of lighting elements, resting areas (e.g., benches), differences in level, but also the use of correct materials and their state of preservation should be considered useful for accessibility evaluation. An assessment based on the elements actually present in an urban area can then lead to the definition of a set of routes that can be defined as accessible.

The management of urban accessibility is in itself a complex matter, which can rely on geographic data that identify problematic elements or architectural barriers, and are a basis for decision-making and a first fundamental step towards inclusive design. Such data, organized as maps, GIS or point clouds, can be useful not only for public administrations or for urban designers, but also for citizens who use urban space daily, or even for tourists who wish to plan their trips in advance. Both the definition of which elements are useful for an accessibility map (whether it serves for the management or the fruition of urban space), and the definition of how such a map can also be easily accessible, usable and correctly interpretable by everybody, are topics of great interest.

¹<https://www.governo.it/sites/governo.it/files/PNRR.pdf>

The first step in the management of urban accessibility is certainly the definition and subsequent census of all those elements and issues to be taken into consideration. The definition of which elements to identify is of great interest and was a topic already addressed and investigated by PhD theses (e.g., [1]) and researches at Architecture, Built environment and Construction engineering department at Politecnico di Milano, and also in other Italian and worldwide universities. The topic of inclusion in historical cities was also addressed by a group of researchers "*Think Tank inclusione e città storica*" established in 2021 at Politecnico di Milano². The research work presented here was part of the same line of research and focused on how to automatize the data collection and how to produce documents that could make the data collected spatially identifiable.

Accessibility data collection can take place in different ways. For example, referring to an expert technician, who physically goes to the urban area of interest, and moving around the environment compiles a series of tables and questionnaires. Once back in the office, these notes can be useful for the drafting of technical data documentation and the subsequent production of maps. Another approach, which involves the end-users of the urban environment to a greater extent, is the use of crowdsourcing, i.e. general users (non-experts) can send reports on urban problems related to accessibility. The expert technician then carries out an evaluation and validation of the collected issues, which can then be used to prepare reports and maps. It is clear that urban accessibility data collection can benefit from the implementation of Information and Communication Technology (ICT) techniques and methodologies. Specifically, the Geomatics techniques that allow for the rapid geometric survey of an urban environment, could be useful for the creation of an objective and geometrically correct base on which to carry out measures and a census of the problems related to accessibility.

The topic of processing 3D Point clouds can be intended as the set of actions that can transform a raw 3D point cloud into a meaningful collection of information, extracted from the point cloud itself and delivered in a format that is meaningful for the final user. In fact, even if a raw point cloud can provide a useful and highly detailed visual impression of the recorded scene, the points that compose it do not contain any kind of additional information related to the elements of the real scene they are representing. It is here that a set of automatized processes can be useful to add semantic meaning to points. The semantic segmentation of a 3D point cloud can be defined as the job of linking each point of the point cloud to a semantic label. This process, ideally as automatic as possible, gives the possibility of adding significant attributes to 3D data. The point cloud semantic segmentation methodologies, also implementing Artificial Intelligence (AI), allow the census of accessibility issues to be carried out in an automated and objective manner. If there is a rule to identify a specific problem of accessibility, this rule can be implemented in specific algorithms so that the said problem can be correctly identified by analysing the point cloud.

Although with some exceptions, in a modern urban environment the organisation of elements typically follows a simple pattern repeated similarly in every area of the city. As an example, it can be noted that the street tends to be at a lower level, while the

²<https://www.dabc.polimi.it/event/think-tank-inclusione-e-citta-storica/>

sidewalks are at a higher level and are separated from the road by curbs. There are, of course, exceptions, such as areas exclusively for pedestrian use, zone only for cars and vehicles, mixed-use urban squares, and car parks. The paving material most commonly found is asphalt. It is very interesting to note that when the urban environment under consideration is the historical centre of a city or even a historical town, the situation changes considerably. Indeed, in a historical city, it becomes more difficult to give a standardisation for the elements of the urban space. To deal with the different layers that have occurred over time and to circumvent any obstacles or constraints, ad-hoc solutions are often implemented, which are not so standard. Often the materials used are also diversified and it is much more difficult to find asphalt pavings.

This research intends to perform the semantic segmentation of 3D point clouds that is not generalized but focused on one specific category of objects: urban accessibility items. In such a way, the point cloud can be used as a sound basis and an informative element, useful to make a decision. Furthermore, the subject of the analysis is a historical urban environment. The choice of the historical environment is related to the challenging aim of dealing with the discussed peculiar urban configurations. The results of the segmentation include a classified point cloud where all points of specific urban features are labelled together. The collected information is then vectorized and lately stored in maps, so that it can be used to produce accessible routes. The definition of which element to identify is out of the purpose of the thesis. The general aim, instead, is to define an automatized strategy that automatically extracts useful information.

The case study selected to test the approaches is the historical city of Sabbioneta, located near Mantova, in Northern Italy. It was built in the 16th century as a fortified city, following the "ideal city" principle. Since 2008 it is a UNESCO site together with Mantova. The replicability and scalability of the method are of primary importance. For this reason, all the work is conducted using and implementing rules as general as possible. Having defined the correct strategy, it can then be applied in several different conditions, for this reason the developed method was also tested on datasets coming from other cities.

This thesis, presenting applied research, showed the possibility of developing procedures for semantic segmentation of the point cloud, and using it for the extraction of additional information and attributes of elements found within it and previously identified by the proposed workflow. Although the point cloud could serve as a container and a dissemination element for the extracted information, in this work a vector representation was chosen, to easily reach a wider range of end users. The use of a point cloud representation as a medium for conveying information to end users is considered a topic that can be discussed in the further course of this research. A second theme of importance of this applied research is its interdisciplinarity, and thus the fact that for its development it was necessary to merge skills, sensibilities, and practices that come from different fields, such as Geomatics, accessibility, and computer science. This shows how increasingly important it is to have a broad-based, multi-skilled approach to tackling problems.

1.1 The context: urban physical accessibility

The International Classification of Functioning, Disability and Health (ICF), states that disability can be intended as a dynamic interaction between health conditions and contextual factors, where contextual factors can be either personal or environmental [2]. Furthermore, in accordance with the *World report on disability* by the World Health Organization [3], "*environment has a huge impact on the extent and experience of disability: inaccessible environments create disability by creating barriers to participation and inclusion. With the aim of improving health conditions, preventing impairments and improving outcomes for persons with disabilities, the environment may be changed*".

It is clear that environmental factors should be addressed when dealing with accessibility and urban space. In addition, accessibility cannot be only associated with the overcoming of a physical barrier but should be related also to the quality of the space and the cognitive dimension involved [4]. In particular, for design purposes, two main categories should be considered: physical needs (moving and using objects in the built environment), and sensorial and cognitive needs (concerning orientation and localization within the space) [5].

Within this framework, the identification of accessibility of objects (e.g., urban space, buildings) is most important and data collection and its dissemination to city users become crucial [5] because data are used either by people with disabilities or by experts (planners, decision-makers). While technicians use the data for simulations and design purposes, people with disabilities use them to plan their trips in advance and to recognize possible obstacles along their routes. Accessibility data collection can be performed in several ways: from municipality databases (even if it is not always available), from geo-spatial datasets, implementing crowdsourcing and Volunteered Geographic Information (VGI), integrating data from open source geo-technologies like OpenStreetMap (OSM), using and analysing data from social networks, and through gamification [6–9].

Among the aforementioned data collection methods, crowdsourcing deserves further description, as some of the most recent interactive maps related to the accessibility of places and buildings use the potential provided by VGI. Crowdsourcing can be defined as a participatory activity, mostly performed online, in which a subject (individual or group) proposes, in a mutually beneficial relationship, to share and exploit a set of resources [10]. In the context of VGI, the volunteers can contribute in different ways, they can actively map obstacles on a specific dedicated app or website, or they can simply passively collect data just by sending information collected by sensors (e.g., gyroscopes, accelerometers) that are then used for mapping places. It is worth mentioning that this kind of project can serve a double purpose: on the one hand, they are a useful data collection method (some of them implement also a validation phase of the inserted data), and on the other hand, they are a useful interactive map that can be used to retrieve data. One of the most popular projects is WheelMap, which is based on OSM architecture and uses a green, orange, and red scale to evaluate the accessibility of specific buildings, defined as Point of Interest (PoI) in OSM [11]. A similar project is "Project Sidewalk" which also implements a validation step and is based on images from Google Street View architecture [12].

In the same context, considering VGI and data coming from OSM, there are also some European projects which are related to the study of accessibility and sidewalks: i-Scope, which is related to Smart City services, and one of them was the creation of paths for visually impaired people and wheelchair users [13]; and Cap4Access, whose purpose was to improve the accessibility of some European cities, defining optimal ways for data collection [14]. In the Italian context, there are some interesting projects which try to consider sidewalks and accessibility [7]: "Milano Facile", promoted by Milano Transportation Agency; "Padova+Accessibile", which maps the architectural barriers of the city, connected with the PEBA (plan for the mitigation of architectural barriers) of the city; "ViaLibera!?", which tries to identify optimal ways to map accessible routes in Milan. It is also interesting the case of "Kimap" (www.kimap.it), which originates from the European pilot project "synchronicity", which focuses on testing technological solutions in European Smart Cities. This web portal, which provides information related to the accessibility of routes, public transport stops and points of interest, was tested on Firenze.

The presented projects not only implemented ways to gather useful accessibility data but also provided to the community interactive maps. In the same ways Geographic Information Systems (GIS) can make an important contribution to the field of disability issues, especially for navigation and orientation purposes [6].

The importance of accessibility and the evaluation of possible architectural barriers is visible in the UE Regulations, in particular through several key actions aimed at improving the social and economic situation of persons with disabilities [15]. This topic is also addressed by the United Nations through the Department of Economic and Social Affairs [16]. On the Italian framework, there are some reference laws (DM 236/1989, DPR 503/1996) [17,18], which define indoor and outdoor accessibility, providing also reference values for the geometry of elements (for example, regarding outdoor, it is referred to sidewalks, urban furniture, and parking lots). Italian regions have specific regulations about accessibility, for example, Region Lombardy provides an interesting regulation about the removal of architectural barriers in urban environments [19], its requirements are the same as Italian laws, and in some cases more restrictive.

In the Italian legislative context, there is an instrument called PEBA (*Piano di Eliminazione delle Barriere Architettoniche* - Plan for the elimination of architectural barriers), defined as a document required by law when planning the physical accessibility of a city. It was introduced by Law 41/1986 to assess the accessibility of public buildings. With an integration (Law 104/1992), PEBA was extended to urban spaces, introducing the PAU (*Piani di Accessibilità Urbana* - Urban Accessibility Plans). Today this tool is still not implemented by many Italian municipalities. There are several reasons behind this situation: cultural, economic, regulatory, and methodological [20].

Given the legislative framework, on the Italian and European levels, it is important to define a set of quality parameters that allows an evaluation of accessibility more comprehensively. The criticalities of environments should not be related only to "present" urban architecture barriers, but also to "absent" qualities [21], and to the relation between the space and the equipment, services and facilities that it connects. Towards the definition of accessibility evaluation criteria, it is important to define some main concepts: the continuity of horizontal surface, safety and comfort of the person that uses the space [22].

Based on these notions, relying on literature and empirical investigations, and studying national and international regulations and reports, it is possible to list a set of parameters that should be assessed when evaluating the urban physical accessibility (from [8]):

- Dimensional features of urban space: minimum dimensional performances, continuity of the path, no obstacles. Standards set the minimum path width at 90 cm, however, a higher value allows greater flexibility in the use of space and would, for example, allow a wheelchair user and a pedestrian to walk side by side; similarly, this could be convenient for a person with hearing impairments who could walk alongside his or her companion, thus being able to read lips. Table 1.1 provide some geometric reference values for physical accessibility in urban paths.
- Flooring features: paving materials and their processing techniques should not be a source of fatigue or danger for the users, it is also important to assess the state of maintenance of the paving surface, because the degraded state may make the path impracticable. For this purpose, assessment parameters are anti-slip properties, coplanarity of the elements, joint width (≤ 5 mm) and height (≤ 2 mm), durability, and lack of or damage to the material.
- Presence of wayfinding solutions: particularly useful for people with visual and cognitive impairments, who need support for their orientation in space and along paths. In this way, the presence of natural guides and/or integrated solutions, together with the removal of elements that may be a source of danger, should be assessed.
- Crosswalks: for the presence of both vehicular and pedestrian flows, it is considered a crucial point for urban paths. The physical features of the crosswalk and the presence of proper guidance tools should be assessed.
- Urban furniture: to enable anyone to undertake a path, irrespective of the person's physical capabilities, it is necessary to provide specific resting places along the route. Although not required by regulations, practice dictates that the distance between two resting places should be between 50 and 200 m.
- Public transport stops: considering that it is the connection between vehicles and the urban environment, it should be verified that all elements interface correctly.
- Reserved parking lots for people with a disability: their presence, maintenance condition, and their connection with the paths should be assessed.

The presented parameters were extracted from a paper [8] that tried to define evaluation criteria for an urban historical space. The main purpose of the presented parameters was to analyse and provide evaluation criteria for an urban space within the domain of Cultural Heritage (CH). It is clear, therefore, that these parameters have been selected with a very specific purpose in mind, and therefore do not pretend to represent all the reference parameters that can be taken into account when assessing accessibility in its broadest sense.

1.2. Research problem: 3D point clouds and accessibility

PARAMETER DEFINITION	REFERENCE VALUE
Level difference between pavings	≤ 2.5 cm
Minimum path width	≥ 90 cm AND every 10 m a resting area of 150 x 150 cm
Longitudinal slope	$\leq 5\%$
Transverse slope	$\leq 1\%$

Table 1.1: *Some reference values for physical accessibility in urban environments, parameters are derived from Italian legislation and common practice. Adapted from [8]*

1.2 Research problem: 3D point clouds and accessibility

By conducting a bibliometric search in the scientific literature and searching through citation and abstract databases (e.g., Scopus and Web of Science), as usually done in bibliographic searches, it can be seen that the search term "point cloud" alone returns a very large volume of publications (more than 70,000 in Scopus, as of June 2021). Certainly, such a term is very general and encompasses articles that deal with point clouds for the most diverse purposes. For example, papers can be found dealing with such topics as conducting surveys and discussing the results, comparing different measurement methods, processing the survey data, producing deliverables generated thanks to point clouds, and discussing point cloud representation methods.

It then becomes necessary to focus the attention of the literature search on a specific category of articles. In this research work, the point cloud was treated as input data and was processed to obtain output data. To investigate the interest of the scientific world in such types of procedures, we wanted to include the term "processing" in the search, whether it was present in the title, abstract or keyword, regardless of the type of processing. In fact, what was chosen to do was to quantify the extent of interest in methods for processing the point cloud, but without any distinction regarding the extent of processing, the type of processing adopted, and the type of final product realized. Finally, the research topic of interest is indeed point cloud processing, but with a specific purpose: the evaluation of physical accessibility. Consequently, it was decided to add an additional search field containing the keyword "accessibility."

Therefore, searching through abstract and citation databases using the terms "point cloud" and "processing" as a search key (excluding articles from the medical and economic area) and comparing the results with those obtained by adding the search key "accessibility" to the previous ones, it can be seen that the research trend related to point cloud processing for accessibility represents a small portion of the total. Specifically, according to a Scopus search in June 2021, this percentage is 5% (396 articles, out of a total of 7664). Different percentages can be found by performing the same search in other services such as Web of Science, which shows that this percentage represents 0.7% of the total (44 articles out of 6363).

The interesting element that emerges, and is the same analysing the articles of both databases, is that about 80% of the articles are in the period 2014-2021 (the exact percentages are 88% for Scopus and 84% for Web of Science). Charting the articles by date (Fig. 1.1) shows that since 2014 there has been a continuous growth of articles on point cloud processing related to accessibility.

Documents by year

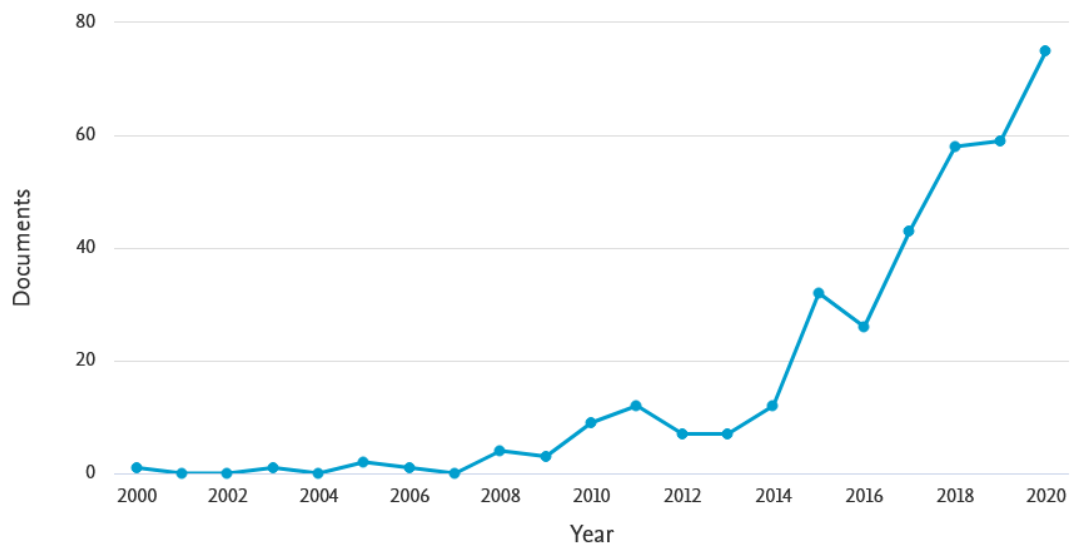


Figure 1.1: *Published papers by year on the topics: Point Cloud processing AND accessibility. From a literature review made in 2021 on the Scopus database.*

Documents by year

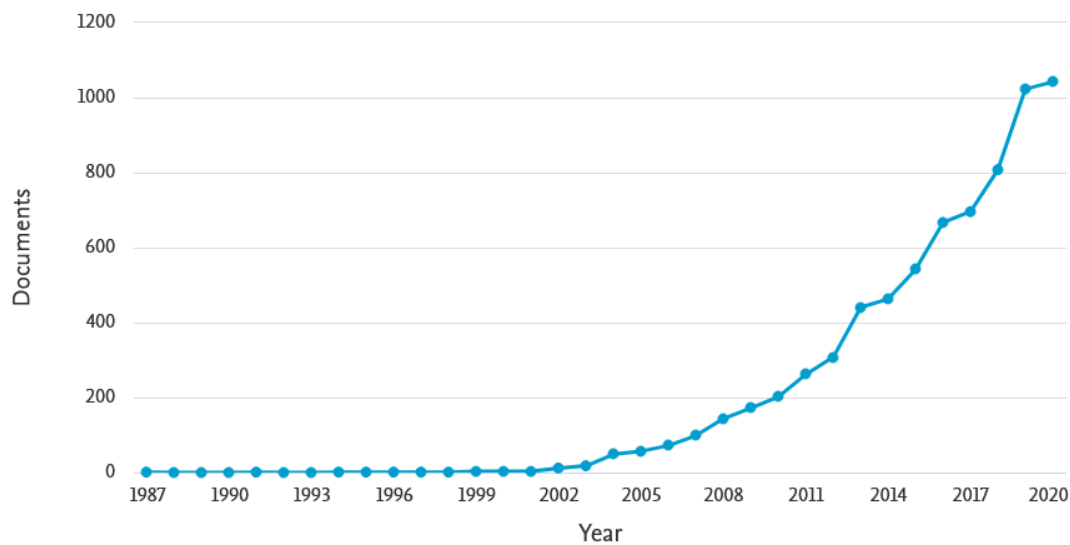


Figure 1.2: *Published papers by year on the topic: Point Cloud processing. From a literature review made in 2021 on the Scopus database.*

To understand if this growth is somehow related to the growth of articles related only to point cloud processing it is possible to compare it with the graph related only to the keywords "point cloud" and "processing" (see Fig. 1.2). This second graph shows how articles linked to point cloud processing have had continuous growth since 2002, long before the starting point of the growth of accessibility-related papers.

In light of what has been said, it can therefore be considered that the growth of

interest on the part of the scientific community towards the topics of accessibility analysis through point cloud processing is currently growing with increasing interest since 2014. The top 5 authors are Arias, P., Balado, J., Diaz-Vilariño, L., Mallet, C., Weinmann, M. and the top 5 affiliations are Universidade de Vigo, IGN Institut Geographique National, Université Paris-Est, Delft University of Technology, Zhejiang University. Two of the most cited papers are "Urban accessibility diagnosis from mobile laser scanning data" (2013) [23] and "Automatic classification of urban ground elements from mobile laser scanning data" (2018) [24].

In the scientific papers analysed, the role of point clouds was manifold: they could be used as a tool in which to search for specific elements of the urban ground and assess their accessibility (e.g., sidewalks, curbs, obstacles, crossings) [23–25], they could be used to make an inventory of sidewalks and assess their conditions [26–29], they could be used to make maps and take measurements [30, 31], they could be used to assess the safety of urban environment [32, 33], they also supported the calculation of accessibility between the interior and exterior of buildings and the calculation of indoor pathfindings [34–37], and finally, they could be a support for the calculation of navigable routes that take physical accessibility into account [9, 38–41].

The analysed papers showed that processed point clouds were mainly acquired with Mobile Laser Scanning (MLS), the main final goal was the definition of accessible routes and most of them were tested on real case studies. Typically, the purpose of the segmentation was to characterize sidewalks and curbs accessibility, relating to specific national standards of the authors' countries. In all the presented papers, the detected elements and the criteria used to assess them were referred to some national standards or best practices.

1.3 The research project

The purpose of the work presented in this thesis was to develop a workflow for the processing of a point cloud of a historical urban environment with the final goal of extracting and delivering useful pieces of information related to accessibility management. In light of this, the basis of the work was a point cloud of an urban environment and the core of the thesis was the framework implemented to automatically process it and to deliver the extracted data in a specific way.

As is evident from previous paragraphs, the processing of point cloud data was already used to extract accessibility features, but the only accessibility features investigated and extracted from the point cloud were the geospatial position of ramps and the presence or absence of sidewalks. Plus, the urban environments analysed were located in cities where the urban elements were standard: the sidewalk was at a higher level from the road and it was separated from it by the presence of a curb. From the context previously discussed emerges that the ramps on sidewalks are not the main issues for accessibility evaluation of an urban environment. The first element is for sure the detection of sidewalks, but then there are several attributes, of the sidewalk itself, that should be addressed. For example the sidewalk paving material, its maintenance conditions, the width of the sidewalk, its longitudinal and transversal inclination, and the presence or absence of a resting area.

Among the many data related to accessibility that could be extracted from the

3D survey, in this thesis work the focus was the detection of sidewalks area, and the characterization of sidewalks in terms of paving material, width and slope. The original aspect of this work was related to the fact that the urban environment investigated was a historical site, which means that typical processing methodologies described in the literature could fail; as an example, several methods rely on the presence of curbs and the different Z level of road and sidewalks, in historical sites the two elements are often at the same level. Dealing with all the peculiarities of a historical site was challenging, and it was even more interesting trying to automatize the process as much as possible, reducing at minimum the user intervention. The purpose of the thesis was also to produce a replicable method, for this reason, the main case study was the historical city of Sabbioneta, but the method was also tested on other datasets produced with different acquisition systems and in other different cities: Mantova (Italy), Sacromonte di Domodossola (Italy), and Porto (Portugal).

The processed point cloud was a 3D dataset with some extracted accessibility features (sidewalks points and their attributes), but it was not in a format that could be widely used by the final users defined in this work. The final users (planners, administrators, citizens, tourists) were for sure much more familiar with maps rather than with point clouds. For this reason, the sidewalk geospatial position and their extracted attributes were vectorized and stored in a digital vector storage format that could be easily read by a Geographical Information System (GIS), i.e. in a shapefile format.

The first step of the research was a geometric survey, the selection of the instrument depended on the required level of detail of the geometric data and also on the purpose, in this work the selected instrument was a mobile mapping system: Leica Pegasus:Two, mounted on top of a car. The point cloud segmentation method used in this research followed, implemented, and upgraded existing methodologies proposed in the literature. The segmented points then were labelled and characterized according to their specific attributes. The results were vectorized in a shapefile. A scheme of the thesis workflow is shown in Figure 1.3; the whole thesis work was developed and implemented to be as much automated as possible.

Both the labelled point cloud and the vectorized data could be used at different levels and for several purposes, dealing with the accessibility of historical urban environments. For example, they could support public administrations to develop Safety and Security analysis, they could be a starting point to detect critical issues and to plan interventions. Furthermore, the segmented point clouds could be used to take measures directly on them, could be used to help the modelling stage in 3D city modelling, they could be a basis for making pathfinding simulations, or they could be used directly as a 3D informative model by linking and retrieving information to points. The shapefile could be used to produce interactive maps with accessibility issues and paths to help the visit of the historical city. It is also worth mentioning that the results of this work could be helpful also for autonomous driving, for the robotic community, and smart cities.

1.3.1 The workflow

This dissertation describes applied research that could be schematized by some interconnected working packages (WP), which are a set of functions and methodologies

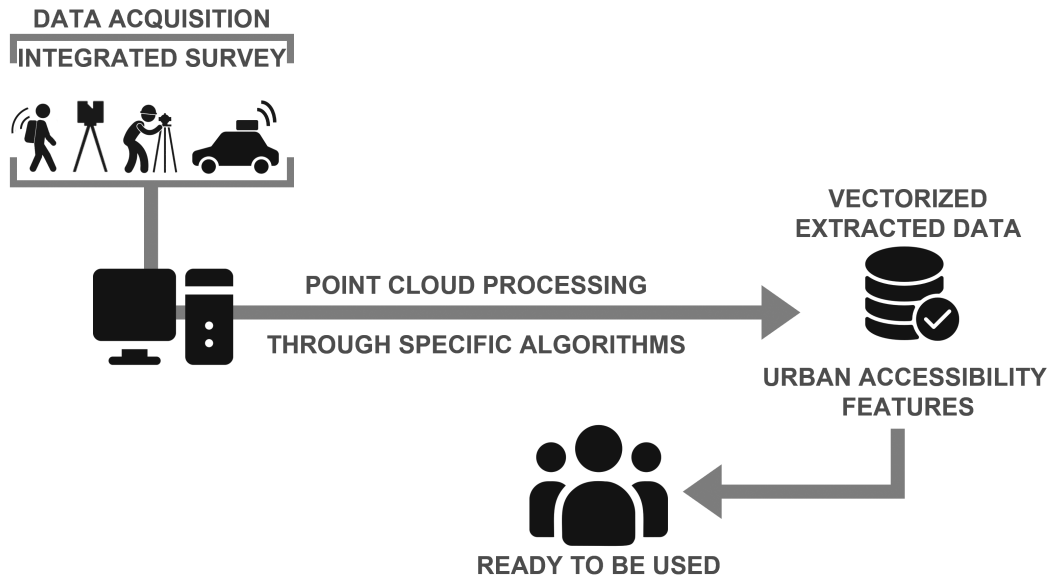


Figure 1.3: The main workflow of the thesis, from the data acquisition with specific instruments, to the development of appropriate algorithms, the extraction of useful accessibility features, and to the production of final deliverable for several users: administrators, planners, citizens and tourists.

with a specific purpose, organised in inputs, procedures, and outputs. The first WP provides the base element for all the further developments: the geometric survey. Then, the results of the subsequent two WPs (segmentation and characterization) were used to implement the last WP (vectorization and production of deliverables). The idea was to extract the accessibility data by processing the point cloud at some different levels, in the first WPs, and to use the data to produce a digital vector layer for GIS (last WP). Each WP had specific inputs and outputs and also used custom coding and specific software. Figure 1.4 shows the main method with the four WPs, which are:

- WP-0: geometric survey, selection of the most suitable instrument, execution of the survey, data processing and validation of the measures.
- WP-1: point cloud processing for the detection and segmentation of sidewalks; starting from the raw point cloud, and by implementing algorithms based on geometrical features, topological relations, and proximity to buildings, this WP extracted points of sidewalks area. Also, an alternative approach exploiting Machine Learning (ML) was tested.
- WP-2: sidewalks characterization; by analysing more in deep the classified sidewalks points, some attributes were extracted: paving material, and some geometric attributes of the pedestrian area.
- WP-3: vectorization of the previous data; the data extracted from blocks 1 and 2 were merged and prepared to be used on a GIS platform, so that they could be helpful to the final users previously defined.

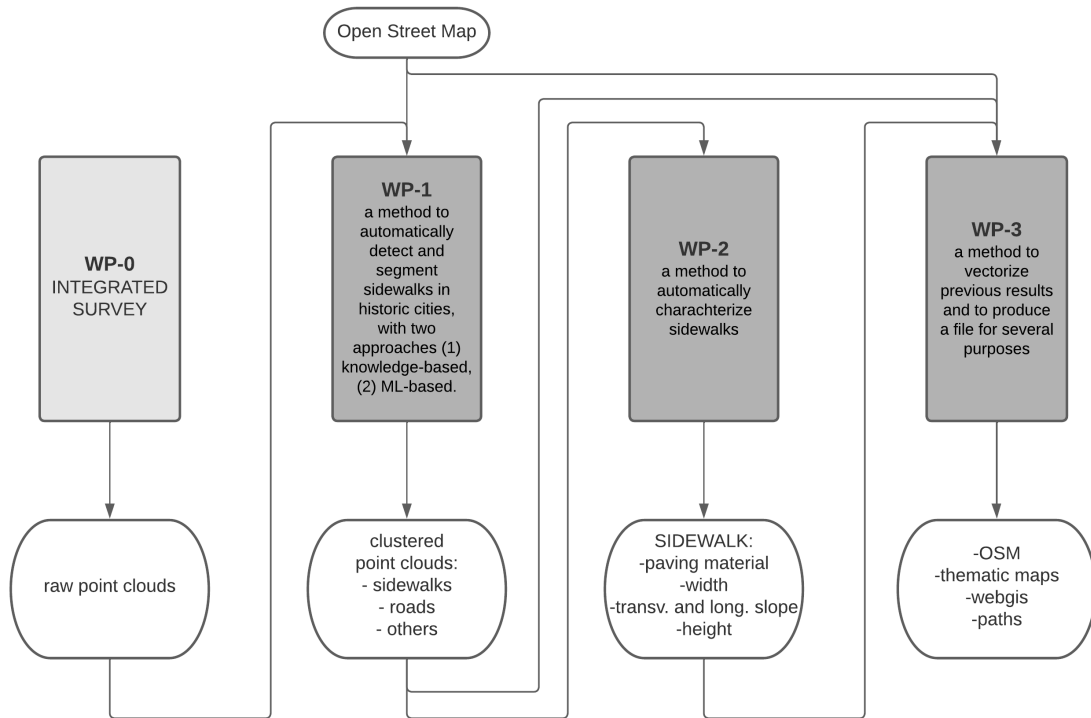


Figure 1.4: Scheme of the thesis in terms of Working Packages of processes, with related inputs and outputs.

Each WP takes some inputs and provides outputs; inside each block, there are a series of steps that involve the use of existing commercial and open-source software, but also, and for the most part, the writing of specific programming code. Although at the moment the code that was written for this thesis is not made public, it is planned to make it available in an open format through the GitHub page³ of the HeSuTech group of MantovaLab research laboratory at Mantova Campus of Politecnico di Milano, where the research was carried out.

The software used to perform data visualisation and manipulation was CloudCompare⁴ (version 2.12). To work with cartographic datasets and manipulate shapefiles, the open-source software QGIS⁵ (release 3.4) was used. All the pieces of code were written in python programming language (version 3.6), with the use of specific libraries. To deal with point clouds "open3d" was used [42], it is an open-source library that supports the rapid development of software that deals with 3D data. In the work presented, it was used mainly to import, export, crop point clouds, and also to create bounding boxes. To deal with the big amount of data, "NumPy" [43] and "pandas" [44] were used; "pandas" is capable of managing a large set of data with several columns and rows. Clustering task was performed implementing "scikit-learn" library [45], in particular the modules referred to k-means clustering and DBSCAN. To import and manipulate shapefiles, "pyshp" [46] was used. To perform computations

³<https://github.com/hesutech>

⁴downloaded from <http://www.cloudcompare.org/>

⁵downloaded from <https://www.qgis.org/it/site/>

on polygons, "*shapely*" [47] was used. To plot graphs "*matplotlib*" [48] was implemented. For the ML workflow, classifiers from "*scikit-learn*" were used, and to deal with large files "*Dask*" [49, 50] was implemented.

1.3.2 General contributions and advancements respect the state of the art

As already mentioned, the innovative character of this thesis work was, above all, in wanting to process a point cloud of a historical urban environment, thus taking into account its non-standard aspects. Furthermore, it can be considered an advancement with respect to the current state of research:

- the willingness to use data from state-of-the-art surveying systems (MLS), also assessing the quality of the resulting data and its suitability for subsequent calculations;
- the development of algorithms targeted for the identification of sidewalks in a generic urban and historical point cloud, taking into account some of the peculiar characteristics of the urban framework of the city;
- the development and testing of procedures implementing Machine Learning approaches on point clouds for the processing and classification of urban elements (streets and sidewalks), but also for the classification of urban paving materials;
- the realisation of a complete and scalable procedure that takes an input (the point cloud), as automatically as possible extracts specific information from it, and provides an output (a shapefile), in a format which is easily accessible, usable and exploitable for various end users.

1.4 Organization of the contents

The thesis work is developed as the intersection of several interdisciplinary fields. For this reason, the work will be presented referring to several specific research fields: accessibility, Geomatics, and Information Technology.

In the second chapter, **State of the art**, after a brief introduction about point clouds and their geometric features, the state of the art of point cloud semantic segmentation methods are presented, focusing on segmentation, classification and ML methods. Then the existing application and strategies for the semantic segmentation of urban areas are presented. Lastly, a literature review of researches addressing semantic segmentation of urban areas, dealing with accessibility issues is described.

The third chapter, **Data acquisition with a Mobile Mapping System**, describes briefly the case study selected for the tests of this thesis work (the historical city of Sabbioneta), then presents an overview of mobile mapping technologies, and describes all the steps followed during the on-site survey. The survey data processing and validation are also discussed.

The fourth chapter, **Knowledge-based approach for segmentation**, presents the method for the semantic segmentation used in this thesis work. It describes the pre-processing of the point cloud data, the segmentation strategy implemented (which exploits unsupervised ML techniques), the voting system developed for the labelling of the segmented data, and the evaluation of the performances. The results for the tests on the whole Sabbioneta dataset and also on other Italian cities and a Portuguese city, are presented and discussed.

In the fifth chapter, **Segmentation exploiting Machine Learning**, some tests are described, to define and analyse the possibility of using an alternative method for semantic segmentation, exploiting ML techniques. In this chapter, the pre-processing of the dataset, the classifier selection, the feature selection, and some tests are presented and discussed. Then the ML method is tested on the dataset, and the results are discussed.

The sixth chapter, **Sidewalk characterization: attributes computation**, is devoted to the computation of attributes for sidewalks previously identified in the point clouds. The attributes computed are the paving of the surface of the sidewalk, its width, the relative elevation respect of the road, and the slopes in its two main directions.

The seventh chapter, **Sidewalk network vectorization and some possible implementations**, presents a method for the vectorization of sidewalk spatial position together with the attributes previously computed; then some possible applications for the generated vector file are presented. Several final products are discussed: the generation of thematic maps, the computation of routes, and the exploitation of webGIS platforms for the dissemination of the information. Lastly, the possibility of using the vector file for updating the existing OSM database is discussed.

The last chapter, **Conclusions**, presents a summary of the entire work, subdivided into several steps, that are discussed in detail. Lastly, some future perspectives are outlined and some final remarks are stated.

2

State of the art

The purpose of this chapter is to frame the topics covered in this thesis and place them in their relevant scientific context, by conducting a serious review of the scientific literature. Therefore, an overview of point cloud processing methodologies is presented, with a focus on their application in urban areas and for urban accessibility.

The chapter is organized as follows: since the object of interest of point cloud semantic segmentation is the point cloud, the first section presents a brief recall of the main features of a point cloud. Then, since the core of the research is the semantic segmentation of the point cloud, the following sections present a literature review of point cloud semantic segmentation methods, with a focus on applications for the management of physical accessibility over urban areas.

Analyzing the main workflow of the thesis and the various Working Packages, it can be seen that in addition to point cloud processing, other topics of interest covered in this research are Mobile Mapping Systems, Machine Learning techniques, and point cloud representation. These topics have not been covered in this chapter, (as they are not considered the core of the thesis) but a brief theoretical introduction of the topics can be found in the relative chapters (3, 5, and 6).

2.1 Theoretical background on Point Clouds

2.1.1 Point cloud data

The term point cloud, in Geomatics, is used to describe a data structure that represents a collection of 3-dimensional points in a given 3D reference system. The point cloud represents the measured or generated counterpart of physical surfaces in a scene; each point is characterized by spatial XYZ coordinates and may optionally be assigned additional attributes such as intensity information, thermal information, specific properties, or any abstract information [51].

Broadly speaking, in computer vision and remote sensing, four techniques can be found for the acquisition of point clouds [52] which exploit four different systems:

- Image-derived, Light Detection and Ranging (LiDAR), Red Green Blue -Depth (RGB-D) cameras, Synthetic Aperture Radar (SAR). Image-derived methods are capable of generating point clouds from spectral images according to principles from photogrammetry or computer vision theory, automatically or semi-automatically [53, 54];
- LiDAR is a surveying and remote sensing technique, which uses laser energy to measure the distance between the sensor and the surveyed object, the result of

such technique is a point cloud whose density can vary from less than 10 points per square meter (e.g., in the case of aerial laser scanning) to thousands of points per square meter (e.g., in the case of TLS) [55, 56];

- RGB-D cameras can acquire RGB colours and depth information, similar to a LiDAR, those cameras can measure the distance between them and the object, since the position of camera's centre is known, the 3D space position of pixels in a depth map can be obtained and used to generate a point cloud [52];
- Interferometric SAR (In-SAR) is a radar technique crucial to remote sensing, which generates maps of surface deformation or digital elevation based on the comparison of multiple SAR image pairs, SAR tomography (TomoSAR) and Persistent Scatterer Interferometry (PSI) are two major techniques that generate point clouds with InSAR, extending the principle of SAR into 3D [52, 57].

The different techniques that allow the generation of a point cloud can also be grouped according to the measurement principle they are based on. This categorization can be focused on the distinction between passive and active techniques. Passive techniques for point cloud generation simply record radiometric information that is emitted by a light source present in the scene (e.g., ambient light). A typical example of a passive technique is Photogrammetry, which allows the reconstruction of the position, orientation, shape and size of objects from pictures [58]. Active techniques involve the measure of a signal (e.g., structured light) which is emitted by the measurement system itself. An example of an active technique is Terrestrial Laser Scanning (TLS). A laser scanner system emits an electromagnetic signal and performs the measure by analysing the returning signal.

Focusing on Photogrammetry and Laser Scanning and discussing the colour attribute of the point clouds, it is possible to state a major distinction: while for Photogrammetry, the colour information is straightforward and closely linked to the working principle of the technique itself, for Laser Scanning, there are techniques which permit to apply colours to points by orienting photos taken from the laser scanner position, and there are laser scanners which embed cameras to take photos with this purpose. An important feature, which is measured by the laser scanner, is the Intensity. It is a value representing the intensity of the backscattered signal and depends on the combination of the type of material and scanning angle [59]. The intensity values range may vary depending on the laser scanner and typically it is normalized and represented by an 8-bit greyscale.

2.1.2 Point cloud Geometric features

The characterization and the analysis of a point cloud are based on some properties of its points. A feature is a representation of the global or local properties of the given data. In the case of 3D point cloud data, features typically result from specific geometric characteristics of the global or local 3D structure and therefore represent geometric features. A geometric feature may be defined in terms of point attributes, shape, size, orientation, roughness, and more [51]. Feature selection and extraction constitute the essential part of point cloud classification, and they play a decisive role in classification results [60].

The authors of [61] categorize point cloud features into intensity features, colour features and geometric features. While the colour and intensity features are mainly related to the mean and variance of Red (R), Green (G), Blue (B) and intensity values, the geometric features of a point P are related to the spatial arrangement of a set of points, called neighbourhood, surrounding the point P .

Neighbour definition

To calculate a geometric feature, the selection of the neighbours of a point is fundamental. There are several ways to define which are the neighbouring points and their selection will affect the result of geometric feature computation. An exemplification of the neighbours is presented in Figure 2.1. Around a point P , we can select the neighbour in different ways:

- a spherical neighbourhood: formed by all those 3D points within a sphere centred at P and with a fixed radius r (Figure 2.1a);
- a cylindrical neighbour: composed by all the points whose 2D projections onto a plane are within a circle with radius r and centred at the projection of P on the plane (Figure 2.1b);
- a fixed K Number of Neighbours (KNN): formed by all the " K " closest point to the point P , according to a specific distance metric, either on 2D or 3D (Figure 2.1c).

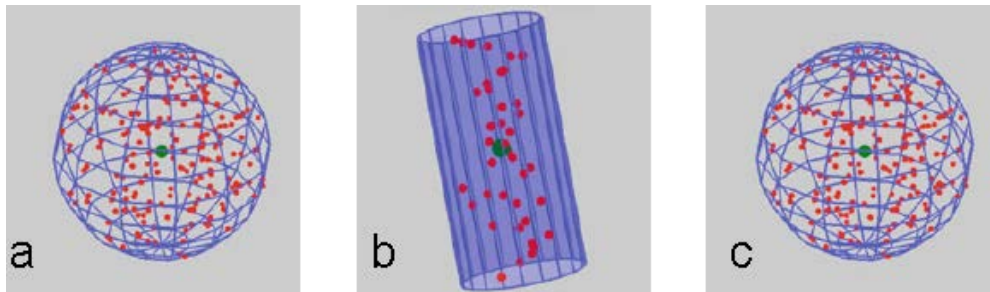


Figure 2.1: Visualization of different neighbourhoods (in red) of point P (in green). a: spherical neighbourhood. b: cylindrical neighbourhood. c: k number of neighbours (KNN). Adapted from [51].

Among the considered neighbourhood types, the cylindrical one reveals less suitability for classifying terrestrial or mobile laser scanning data, whereas the spherical neighbourhoods (parameterized by either a radius or the number of nearest neighbours) have proven to be favourable [62].

Local 3D shape features

The geometric features linked to the spatial distribution of points in the neighbour can be computed on the basis of the 3D coordinates of points in the neighbour itself.

"The 3D structure tensor of the neighbour (which is the 3D covariance matrix computed from the XYZ coordinates of the point P and its neighbouring points) represents a symmetric positive semidefinite matrix, its three eigenvalues exist, are

nonnegative, and correspond to an orthogonal system of eigenvectors. Furthermore, there may not necessarily be a preferred variation with respect to the eigenvectors so it is possible to consider the general case of a structure tensor with rank 3. As a consequence, the three eigenvalues λ_1 , λ_2 , and λ_3 with $\lambda_1, \lambda_2, \lambda_3 \in \mathbb{R}$ and $\lambda_3 \geq \lambda_2 \geq \lambda_1 \geq 0$ indicate the extent of a 3D ellipsoid along its three principal axes" [51].

As mathematically explained above, the local covariance matrix can be used to compute local features. In particular, by performing a Principal Component Analysis (PCA) on the matrix, it is possible to obtain the three eigenvalues. The eigenvector corresponding to the lower eigenvalue is the so-called normal vector and represent the inclination of the plane which best fits the neighbour.

The eigenvalues can be used to compute and determine some useful geometric features that well represent the neighbour distribution in space. Yet it is especially important for these features to be derived from a suitably chosen neighbour size. It is the nature of second-order moments that the distance of one element from the mean contributes quadratically and therefore elements in the vicinity are far less important than those further away [63]. The equation for the computation of the geometric features are here reported:

$$C_\lambda = \frac{\lambda_3}{\lambda_1 + \lambda_2 + \lambda_3} \quad (2.1)$$

$$L_\lambda = \frac{\lambda_1 - \lambda_2}{\lambda_1} \quad (2.2)$$

$$P_\lambda = \frac{\lambda_2 - \lambda_3}{\lambda_1} \quad (2.3)$$

$$S_\lambda = \frac{\lambda_3}{\lambda_1} \quad (2.4)$$

$$O_\lambda = \sqrt[3]{\lambda_1 \lambda_2 \lambda_3} \quad (2.5)$$

$$A_\lambda = \frac{\lambda_1 - \lambda_3}{\lambda_1} \quad (2.6)$$

$$\sum \lambda = \lambda_1 + \lambda_2 + \lambda_3 \quad (2.7)$$

$$E_\lambda = - \sum_{i=1}^3 \lambda_i \ln(\lambda_i) \quad (2.8)$$

Where λ_1 , λ_2 , and λ_3 are the three eigenvalues.

The local surface variation C_λ , also called change of curvature, gives information about the convexity of points in the neighbour. The so-called dimensionality features provide information respectively in 1D, 2D, 3D, and are the Linearity L_λ , Planarity P_λ , and Scattering, or Sphericity S_λ . Linearity, Planarity and Scattering $L_\lambda, P_\lambda, S_\lambda \in [0, 1]$ and $L_\lambda + P_\lambda + S_\lambda = 1$. The dimensionality features may be considered as the "probabilities" of a 3D point to be labelled as 1D, 2D or 3D structure [64]. The Omnivariance O_λ , provides information about the distribution of points; low values correspond to planar regions, higher values represent a more volumetric distribution of points. Other features are Anisotropy A_λ , Sum of eigenvalues \sum_λ , and Eigentropy E_λ .

Other geometric properties

Considering some basic properties of the selected neighbour, it is possible to derive useful information about the 3D scene, without computing a covariance matrix.

For example, it is possible to calculate the local density of points in the neighbour, it can be derived knowing the number of points and the radius of the neighbour selection method implemented. Another useful feature is the Verticality, which can be derived from the N_z component of the normal vector. Also the maximum and minimum Z values and their variance could be a helpful feature.

Another feature that can be derived from the neighbour is the roughness. The roughness can be calculated in several ways [65], for example, the roughness of point P can be computed as the distance between the point P and the best fitting plane computed on the neighbours.

2.2 Point cloud semantic segmentation

During the last years, following the increasing interest in the topic, several literature reviews were presented and were considered as a source for the State of Art section presented here. Grilli et al. [66] proposed in 2017 a review which, starting from [67], analysed the most popular methodologies and algorithms, at that time, in segmentation and classification of point clouds. They have also made a declination of these algorithms in the field of CH. They subdivided the segmentation methods into edge-based, region growing, model fitting and hybrid, with a short introduction to ML techniques. A similar subdivision of methodologies, with an extensive review and critical analysis and comparison of different methodologies, was proposed in 2020 by Xie et al. [52]. Respect the review of [66], here the ML section is extended comprehending unsupervised clustering and regular ML; and there is also a section specifically devoted to Deep Learning (DL) methods. Another interesting literature analysis is the one presented by Xu et al. in 2021 [68], which is oriented on building and civil reconstruction from point clouds and considers point cloud segmentation and classification key techniques in these fields. As in the previous cases, they also make a breakdown of segmentation methods which is in some ways similar to those already seen, but which also has some new elements. In this paper region growing methods, model-based segmentation, clustering, and energy optimization based were described as the segmentation categories, and ML and DL methods are mainly used for classification. Focusing specifically on DL for point clouds, Liu et al. [69] proposed in 2019 a review of DL methods and applications, considering also their advantages and disadvantages using point clouds. Similarly, but mainly focusing on DL application for raw point cloud data, Bello et al. [70] proposed in 2021 a review which also provides a general structure of Artificial Intelligence applications with raw point clouds, comparing various methods.

Point cloud semantic segmentation, point cloud classification, or point labelling, can be defined as the job of linking each point of the point cloud to a semantic label [52]. This process, ideally as automatic as possible, gives the possibility of adding significant attributes to 3D data. Semantic segmentation can be broken down into segmentation and classification. Segmentation refers to clustering and grouping points depending on their geometric and radiometric characteristics, through mathematical

models and geometric reasoning. The classification task is related to allocating specific groups to classes or labels [66].

Semantic segmentation can be achieved following two approaches [67]: one that only performs segmentation and one whose result is a classified point cloud. The first one implements mathematical models, geometric logic thinking, and statistical rules, in combination with robust estimators to fit linear and non-linear models to point cloud data. The main results of this procedure are regions and clusters that contain similar points, depending on custom-made geometric constraints. These results do not have semantic information and require further steps to apply a semantic label. The second approach extracts 3D features from point cloud data using a feature descriptor, uses ML techniques to learn different classes of object types, and then uses the resultant model to classify acquired data. Alternatively, to the second method, DL techniques directly implement high dimension features to train the data without the need for a feature descriptor.

2.2.1 Segmentation-only methods

The task of segmenting point clouds has the purpose of grouping points in clusters according to similar characteristics or features. Segmentation can be achieved relying on point attributes, like intensity value or points colour or other attributes that are acquired together with point coordinates, simply grouping points that share the same semantic information or the same attribute. The colour information can be influenced by environment light conditions, and sensor recording technology, making this attribute less useful under specific cases [68]. On the other hand, segmentation can be achieved according to geometry-based techniques, which rely on the surfaces or structure to which points belong to. Typically, no prior knowledge of the segmented surface is required, so the final result does not have strong semantic information [52]. The segmentation approaches can be categorized in:

- edge-based;
- region growing;
- model fitting;
- unsupervised clustering;
- energy optimization.

Edge-based approaches exploit the fact that real objects are defined by edges, so the segmentation can be achieved by finding points on the edge region. Rabbani et al. [71] describe an edge-based segmentation algorithm as composed of two steps: edge detection (extraction of boundaries of different regions), and grouping of points (all points inside the boundaries are clustered together). The edges can be defined as the points where changes in a local surface property (normal, gradient, principal curvature, or higher-order derivative) exceed the given threshold [66]. Edge-based algorithms ensure fast segmentation, but can only be used with good results when simple scenes with ideal points are provided (low noise and homogeneous density), for this reason, this approach is rarely applied for datasets with dense point clouds or in large areas [68].

Region growing methods involve the definition of one or more points, called seed points, featuring specific characteristics and then, following an iterative process, the algorithm analyses the neighbouring points to assess whether or not they belong to the same region of the seed point. This approach requires the selection of seed points as the origin of the growing process, and the definition of the growing criteria, which is used to assess if the growing process should continue or should be stopped. Some of the most used growing criteria are the consistency of orientations of normal vectors [72], the curvature of points [73], the smoothness of surface [71], and some PCA-based local features [74]. Often, with large point clouds, to reduce the data volume, the growth units used are not single points, but region units (e.g., voxel grids or octree structures) [52]. Many region growing algorithms aim at plane segmentation, so for the selection of seed points, the usual practice is to design a fitting plane for a certain point and its neighbour and select as seed the point with minimum residual to the fitting plane [71], other methods define the seed point by looking for the one with the least curvature [74]. The accuracy of this method depends on the growth criteria and the location of seed points [52], typically edges and borders of objects are well defined, but these approaches are computationally intensive and vulnerable to noise and outliers [68].

Model fitting approaches rely on the fact that many real objects can be decomposed in geometric primitives, like planes, cylinders and spheres. These methods try to fit primitive shapes into the point cloud and the points that meet the criteria for fitting the mathematical representation of the primitives are labelled to the same segment. If the geometric primitives have some semantic meaning, these approaches can also be used for classification [66]. Most of the widely used model fitting techniques are based on two popular algorithms: Hough Transform (HT) [75] and RANdom SAMple Consensus (RANSAC) [76]. Many improved algorithms based on RANSAC and HT have emerged over past decades to improve efficiency, accuracy and robustness [52]. Model fitting methods are resilient to noise and outliers, but they usually require a high computational cost and use of storage; there are also problems with artefacts and structures for which there are no mathematical expressions [68]. They have been used to detect lines, planes, cylinders, cones, spheres, cones, and torus in parametric space [66].

Unsupervised clustering methods analyse relations or the similarities between points in adjacent positions in a given region according to their geometric features or spatial coordinates. Points that meet the proximity or similarity thresholds are grouped. The clustering criteria (defines if two points should be connected or not) and the clustering method (defines the next candidate point to check) are of primary importance. Typical criteria are the euclidean distance, the angle between normal vectors, and the consistency of density between elements [68], while typical clustering methods are K-means, mean shift and fuzzy clustering [52]. K-means is a widely used unsupervised clustering algorithm that subdivides the dataset into K classes; it can be adapted to all kinds of feature attributes and can be used in multidimensional feature space. The main drawback is that sometimes is difficult to predefine the value of K clusters properly. Fuzzy clustering is similar to K-means and implements weights to dataset elements. Mean shift, in contrast with K-means, does not require to predefine the K value [52]. Typically clustering methods are computationally heavy, depending on

the complexity of the criteria and considering that in many cases multiple clustering criteria are implemented. Furthermore, the definition of optimal thresholds influences the granularity of the result and the presence of under-segmentation [68].

Energy optimization methods convert the segmentation task into the optimization of an energy function. Assigning a point into a cluster will create a cost, and only when the points are assigned to the optimum cluster, the sum of all costs will be minimized and the energy function will be optimized [68]. The most common approach is graph-based segmentation, which depicts points with a mathematical structure using context and topology to deduce hidden information. These approaches are often used to refine initial segmentation results, they are resilient to high noise and outliers but require a high computational cost.

2.2.2 Classification methods, including Machine Learning

The classification task consists of applying a semantic label to a previously segmented point cloud. Recent progress in ML and computer vision shows that a well-designed solution to point clouds classification is suitable for the labelling task [68]. The classification can be implemented by following rules that come from prior knowledge of the 3D scene (in this case the classifier is manually designed), or it can be implemented directly by learning from data. In this second approach, labelled training samples are used by the classifier to learn and optimize classification rules. This last is called supervised ML and requires a dataset of manually annotated data with semantic categories, it is used by the classifier to learn and then provide a semantic classification to the entire dataset. Essential steps for supervised ML classification are (see Fig. 2.2): neighbourhood selection, feature extraction, feature selection, supervised classification.

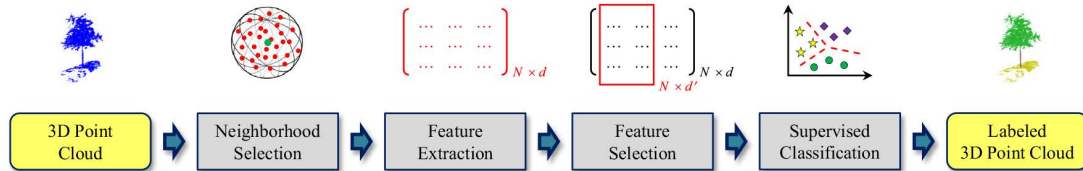


Figure 2.2: *Generic point cloud supervised classification workflow, from [64]*

The recovery of the neighbourhood is a crucial step. In fact, the description of various point features is dependent on the local context of all points within the neighbourhood. Considering that the scale and shape of objects may vary, also the selected neighbourhood should be capable of describing geometric information at various ranges and scales. As previously described in par. 2.1.2 the neighbourhood can be defined by a cylindrical, spherical or k-nearest points selection (see Fig. 2.1). The scale of the neighbourhood plays a vital role in the performance of the feature expression. Typically, to describe objects with significant scale differences, a multi-scale neighbourhood selection can usually offer more effective solutions [68].

Feature extraction involves the description of local geometric information in a defined neighbourhood of the investigated point. A wide range of feature extraction algorithms was introduced in the last decades, with various description methods of local geometry developed. They can be grouped into two categories: low-level and

high-level descriptions [68]. The low-level description consists of fundamental geometric properties of the neighbourhood and its spatial arrangement, which can mainly be derived from the eigenvalues of the variance-covariance matrix (as described in par. 2.1.2). The low-level features are highly dependent on the neighbourhood size. High-level features can be derived by optimizing the neighbourhood dimension, and by combining low-level features; the result is an abstracted or compact depiction of points characteristics based upon their supporting area [68].

The extracted features should be appropriately selected, avoiding redundancy, and then must be loaded into a classifier to deduce semantic labels. Supervised classification means learning a classifier through training data, including its associated vectors and labels. Two methods of classification can be identified: point-based and segment-based [68]. In point-based classification, each point is labelled during the inference process, while in segment-based classification individual objects in the scene can be distinguished. Commonly used classifiers for point cloud classification include almost any supervised learning classifiers like Support Vector Machines (SVM), AdaBoost, Random Forest (RF) and Conditional Random Field (CRF) [68]. The authors of [64] described several methods and tested them on different datasets, according to their experiments the RF classifier has a good trade-off between accuracy and efficiency.

2.2.3 Deep Learning methods

DL is an ML approach based on artificial neural networks designed to mimic the human brain [77]. Unlike traditional ML classification approaches, where features are manually selected and then given to the classifier, DL techniques are usually implemented in an end-to-end way, combining feature extraction with classifiers in a single network, so that usually these two parts cannot be separated in a DL based method [68].

DL can be implemented in point cloud processing with different purposes: 3D object classification, semantic segmentation, and 3D object detection [69]. 3D object classification involves the recognition of objects inside a point cloud and it has numerous applications in robotics, virtual reality and city planning. Semantic segmentation's purpose is to label each point with its corresponding semantic category. 3D object detection not only recognizes single objects inside the point cloud but also defines their spatial position with a 3D Bounding Box.

The application of DL on point clouds is not easy due to the nature of point clouds. The most significant challenges are the followings [70]: irregularity, as point cloud density typically varies a lot in a scene; unstructured, as point clouds are often not placed in a regular grid (e.g., the pixel in an image are structured in a grid); unorderness, as points in a point cloud file are stored in an unsorted order. DL methods, like Convolutional Neural Networks (CNN), are based on operations that are performed on ordered, regular, and structured-on-grid data. Early approaches overcome these issues by converting the point cloud into a regular structure (e.g., voxel-based approaches, multi-view CNN), while new approaches apply DL directly on raw point clouds [70].

DL on raw point clouds has received increased attention since the presentation of PointNet [78] in 2017, which introduce a novel scheme that directly processes unstructured points. Since then, many other techniques that used PointNet as a basis were proposed. An extension of PointNet, PointNet++ [79], allows local region com-

putation by applying PointNet hierarchically in local regions. A further improvement was proposed with PointCNN [80] which explores the local correlation between points.

Compared with regular ML, DL has advantages: it handles large datasets more efficiently, the feature design and selection are not necessary (which is a difficult task in regular ML), and it has high accuracy results on public benchmark datasets [52]. On the other hand, some drawbacks may arise when processing large areas; in fact, they are usually processed by cutting them into smaller pieces. In such a case, DL method performance depends on the sampling of input data, since noise and errors can be introduced in the splitting of data, especially in the boundaries between objects, requiring a further post-processing step [68].

2.2.4 Applications

Point cloud semantic segmentation can be applied in a variety of fields and scenes: urban monitoring, vegetation monitoring, power line detection, 3D object reconstruction, forestry survey, disaster monitoring, object tracking, human pose recognition, and indoor reconstruction [52]. A wide research topic is also building reconstruction, buildings are usually constructed with regular planes, so plane segmentation is a fundamental topic. Many approaches were also tested in Scan-to-BIM reconstruction methodologies, in particular, exploiting ML and DL techniques [68]. The application of semantic segmentation was also investigated in the CH field, exploiting ML and DL [81–83], and with the purpose of HBIM reconstruction [84].

2.3 Semantic segmentation of point clouds of urban areas

Focusing on urban areas, point clouds are typically acquired through Mobile Laser Scanning (MLS) systems, or Mobile Mapping Systems (MMS). An MLS is an acquisition system, mounted on a moving platform, composed of a set of sensors (in general LiDAR, Global Navigation Satellite System (GNSS), Inertial Measurement Unit (IMU), cameras), and capable of acquiring point clouds datasets while moving in the area of acquisition. Generally, to survey urban areas, an MLS system is mounted on top of a vehicle or on a backpack and can capture detailed geometric information about the roadway and surrounding area. The result of processing data from LiDAR, GNSS, and IMU are precise 3D coordinates for each point of the point cloud; if the MLS system mounted also cameras, the point cloud can also have colour attributes. In section 3.2, a further description of MLS is provided.

2.3.1 Applications

An MLS survey can have many purposes and can be exploited by several research fields: architectural purposes (visualization and management of built environment), civil purposes (assessment of conservation state of infrastructures such as railways, tunnels, streets), autonomous vehicles driving, and vegetation mapping and forest inventory. MLS datasets provide highly detailed geometric data which are naturally prepared for a subsequent processing phase. Regarding urban areas, the main processing focus is the extraction of objects with fine-scale and detailed information.

The authors of [85] organize those applications in categories: transportation infrastructure mapping (e.g., the extraction of railroad central line, road edges, street curbs, road markings, road cross-sections, curved facades, cars, and traffic signs), building information modelling (e.g., to identify building shape, structure, boundary, facade, roof, and outdoor/indoor environments), utility surveying and mapping (e.g., extraction of street-scene objects, like urban furniture, traffic signs, and trees), vegetation mapping and inventory (e.g., detect and classify tree species), autonomous vehicle driving (e.g., to perceive street environments and detect the precise position, orientation and geometric features of vehicles).

Another research field that can benefit from MLS and semantic segmentation is 3d city modelling. In fact, from the high-density point cloud acquired with MLS, a detailed city model can be regularized and reconstructed geometrically [85]. The data acquired (images and point clouds) can be combined for texture mapping, and together with semantic labels can be exploited to construct 3D city mesh [86,87], or can be used for automatic geometry generation and shape detection for BIM and city models can be implemented [88,89].

2.3.2 Strategies

The processing workflows presented in the previous section apply also to urban areas and MLS. Processes like feature extraction, segmentation, object recognition and classification are implemented, through specific algorithms, for different purposes related to urban areas inventory and description.

The authors of [90] stated that general approaches used for feature extraction and segmentation on urban areas includes: Hough Transform (HT), that can be used to detect 3D shapes, or to model a 3D scene with geometric primitives; Random Sample Consensus (RANSAC), which can be used for extraction, segmentation and modeling of pre-defined geometric shapes; PCA, that can be used to extract geometric information by analysing local point distribution; Fast Point Feature Histograms (FPFH), which can be used as a feature descriptor; Region Growing, that is a data-driven approach and used to segment point cloud data; Connected Components, which is similar to region growing and often used for segmentation; Graph-cut, that looks for breaks between nearby point, is used to segment a point cloud but can also be used as a refinement approach; Supervoxelization, which is a common segmentation approach that relies on voxels, defined by grouping points into a 3D grid; Mobile Normal Variation Analyses (Mo-norvana), that is a feature extraction and segmentation method implements edge-detection and region growing.

It can be observed that algorithms and workflow specifically developed for one scene type (e.g., road, railway, tunnel, forest, or urban areas) have difficulties in being applied with satisfactory results on another scene type [90]. This happens for several reasons: different assumptions and constraints are built-in algorithms based on the objects present in the scene, data acquisition strategy can vary a lot depending on scene type, noise levels may vary between scenes.

Currently, learning-based point cloud processing algorithms are the mainstream trend [91]. Even in urban objects extraction and road inventory, ML methods are applied. DL approaches are more used to object detection, cause the processing of large scenes is challenging for them.

As depicted by the authors of [90], it is important to note that most of the existing methods for object recognition and point cloud classification focus the attention on extracting objects in an urban street scene. Data collected in a suburb or rural roads are rarely tested and can have different challenges compared with objects in typically investigated urban scenes.

2.3.3 Detected objects

Data extraction purposes can be grouped depending on the position of detected objects with respect to the road. The authors of [91] subdivide potential urban applications in "on-road" and "off-road", explaining also some possible methods implemented in literature. On-road information extraction comprehends:

- road surface (implementing voxel-based methods, trajectory-based filters, ground segmentation, terrain filters, RANSAC-based filters);
- road markings extraction and classifications (using image-driven extraction and point-driven extraction);
- driving line generation (implementing segmentation, clustering, voxelization, region growing algorithms);
- road crack detection (morphological mathematics based algorithms, edge detection, advanced data mining and ML);
- manholes detection (using 2D images and Hough transform).

Off-road information extraction is usually conducted by exploiting the geometric and spatial attributes of the off-road objects. Geometric attributes can be the geometric dimension, local coordinates, geographic coordinates, geometric relations, and shape. Spatial attributes include topology, surrounding attributes, and point cloud attributes. Two main density-based clustering methods are Euclidean clustering and DBSCAN; there are also implementations of ML and DL methods, and rule-based extraction. Typical detected elements are essentially pole-like objects (traffic signs, light poles, roadside trees, power lines), and all the other road objects and urban furniture.

Road surface extraction

An accurate definition of the road surface and its structure plays a crucial role in many applications, from urban planning to roadmap generation. Plus, accurate extraction of the road surface is a pre-requisite for further extraction and classification of other on-road objects such as road markings, driving lanes, cracks, and manholes.

A common approach relies on the definition and extraction of road edges (e.g., barriers, curbs), used to define the road boundaries and perform a separation between road surface and non-road surface. In the literature, it is commonly assumed that the road surface has a lower height compared to non-road surfaces to extract the road boundaries by detecting height jump [90]. Many of the existing works exploiting boundary definition, rely on curbs to define road edges, so the extraction of road surface will not be robust when it is not delimited by curbs, as in the case of most non-urban roads [92].

Another approach is based on prior knowledge of geometric and contextual features of the road environment, which can be computed from the point cloud and used to drive segmentation and classification. For this purpose, the elevation coordinate of the points was used as a key feature to analyse the local height difference [23]; another used feature is the roughness of road surface [65], together with radiometric features and point density [93]; in this case, the different types of road pavements are supposed to have different values of surface roughness.

On-road elements

In general, the extraction of ground objects can be exploited through three processes: 2D rasterization, point-based, and scanline methods. 2D rasterization consists of the conversion of point cloud into a 2D image, in this way common image processing methods can be implemented and computational complexity can be decreased. However, in rasterization approaches the accurate selection of pixel size and point cloud resolution are two important features influencing the final result. 3D point-based methods directly work on point cloud data, resulting in a higher detailed extracted information, but increasing the computational effort. Typically, in this method, point clouds are organized in 3D data grids using octree or voxelization. The scanline method exploits the fact that MLS collect data by multiple scan lines, in this way the processing approach can be implemented on each scanline; furthermore, parallel processing can be used to improve efficiency. Additionally, the trajectory of the MLS system is often used and implemented in segmentation methods.

Road markings are typically made of high reflection material, which provides high-intensity values; for this reason, typical approaches focus on radiometric attributes and intensity information. Rasterization methods are widely employed [94], but also ML [95] and DL [96] approaches are exploited.

Driving lines, intended as the driving routes for motorized vehicles or autonomous vehicles, play a crucial role in generating the definition of a roadmap for autonomous driving systems. The generation of road lines with centimetric accuracy can boost the development of high definition roadmaps and autonomous driving [91]. To reach the purpose, a processing method that includes trajectory information, curb detection, road surface and road markings extraction can be implemented [97, 98].

Road crack detection is of primary importance for the assessment of potential road distress and traffic risk. Furthermore, manhole cover detection is of primary importance both for infrastructure inventory and to assess the safety of road users, preventing them to fall into wells. Both crack detection and manhole cover extraction are handled in similar ways. To reach the purpose, the most used are morphological mathematics-based algorithms [99, 100], or rasterization-based methods converting the point cloud into a 2D geo-referenced feature (GRF) image [101, 102]. Also, ML and DL methods were investigated [103, 104], although the training process needs a lot of labelled data.

Pole-like objects

Traffic signs play a crucial role in the transportation network and autonomous driving; their standardized geometry and their reflective properties were used by authors to develop several approaches with two main purposes: traffic sign detection and traffic

sign recognition. Typically localization of traffic signs is done using the point cloud, and traffic sign recognition is performed through digital image processing. Traffic sign detection was implemented applying geometric and spatial features (position, height, orientation, intensity) driven algorithms [105], as well as ML [106]. Traffic sign recognition was achieved by integrating image data and point clouds together [91] and exploiting ML or DL on images.

Street lights and power line poles can be detected in similar ways, relying on the fact that their geometry is well defined and easily recognizable in an MLS point cloud. Segmentation approaches rely on a prior separation between the ground and non-ground objects, then apply some refinement and further segmentation to non-ground objects using geometric features and shape features. The segmentation approaches that do not rely on ground non-ground segmentation, usually implement voxelization and detect pole objects by analysing the voxelization structure [92].

Other elements

Roadside tree inventory is a process of prior importance for road management also considering the correlation between vegetation and fire risk [92]. Existing classification methodologies can be applied to extract single trees and to detect tree species. Algorithms implement the use of euclidean clustering algorithms, region growing-based methods applied on voxels [107]. Also, ML and DL methods are implemented, in particular for the definition of tree species a neural network can be trained to learn tree species from images [108].

MLS data segmentation can be exploited also for building reconstruction, the existing methods are based on the assumption of high verticality of building elements. Taking advantage of this assumption, proposed algorithms used surface growing, plane fitting and plane segmentation through RANSAC, 2D GRF images. After building facade extraction, some methods are extended to extract windows and doors by observing opening areas [90].

Parked vehicles on the roadside can be detected in several ways and can be useful for street parking management. Typically the non-ground objects are separated from the ground, and then a classification of vehicles on non-ground objects can be performed using supervised learning techniques [90].

2.3.4 Benchmark Datasets

For the sake of completeness, it is interesting to mention several benchmark datasets that were proposed to the scientific community over the years. The purpose of such benchmarks is to train, test, evaluate, validate, and compare the developed algorithms. Some of these datasets are related to urban scenes, for example Oakland [109], Paris-rue-Madame [110], IQmulus Terramobilita Contest MLS [111], Paris-Lille 3d [112].

After a closer analysis of those Benchmark datasets, they were not considered adequate for this work, because either the objects of the surveys were not historical areas, or the characteristics of the surveyed areas did not meet the purposes of this thesis. For this reasons, a new survey was conducted, and the case study selected was a historical city: Sabbioneta (see chapter 3).

The main characteristics of the benchmark datasets are reported here:

- Oakland dataset is composed of 1.6 million points, whose attributes are the XYZ coordinates and 5 classes: 'vegetation', 'wire', 'utility pole', 'ground', 'facade'.
- Paris-rue-Madame was acquired with a Velodyne HDL32 and is composed of 20 million points. The point attributes comprehend the coordinates, intensity, object Id and class; it is segmented into 17 classes including 'facade', 'ground', 'cars', 'light poles', 'pedestrians', 'motorcycles', 'traffic signs'.
- iQumulus dataset was acquired with a Riegl LMS-Q120i, it is composed of 12 million points and point attributes including coordinates, intensity, GNSS time, scan origin, number of echoes, object Id and class. It is segmented into 22 classes, including 'road', 'sidewalk', 'curb', 'building', 'post', 'street light', 'traffic sign'.
- Paris-Lille-3D was acquired with a Velodyne HDL32, for a total of 143.1 million points, points attributes include point coordinates, intensity, and class. The classes are 50, including 'ground', 'building', 'pole', 'trash can', 'barrier', 'pedestrian', 'car', 'vegetation'.

2.4 Semantic segmentation of point clouds for urban accessibility

Among the many papers related to point cloud processing, directly referring to urban areas, only a few of them deal with topics related to accessibility. In literature, the role of point cloud processing for urban accessibility management is manifold:

- they can be used as a tool in which to search for specific elements of the urban ground and assess their accessibility (e.g., sidewalks, curbs, obstacles, crossings);
- they can be used to make an inventory of sidewalks and assess their conditions;
- they can be used to make maps and take measurements;
- they can be used to assess the safety of the urban environment;
- they also support the calculation of accessibility between the interior and exterior of a building and the calculation of indoor pathfindings;
- they can be a support for the calculation of navigable routes that take physical accessibility into account.

Concerning the assessment of urban ground elements, the authors of [23] focus on urban objects segmentation and curbs detection, starting from an MLS point cloud. Basing on range images, height, and geodesic features they segmented urban objects and detected curbs. Accessibility analysis was later defined using geometrical features and accessibility standards. With those data, they also built an obstacle map to use for the creation of adaptive itineraries that take into consideration wheelchair width. Curbs detection and classification was also performed by the authors of [25], in fact, from MLS they extracted curbstones and classified whether they allow or not access to off-road facilities. By classifying the curb types they assessed also the accessibility.

The method is based on analysis of the angles of adjacent points on a scan line, then a voting process is implemented using surrounding classification results. A method to automatically classify urban ground elements from MLS data was proposed by [24]. Their method is based on a combination of topological and geometrical analysis. Element classification is based on adjacency analysis and graph comparison. Road, tread, riser, curb and sidewalks are detected to provide useful data from an accessibility point of view.

Regarding sidewalk inventory and assessment, in reference [26], the authors extracted sidewalks and curb ramps from a combination of images and mobile lidar, basing on some specific dimensional characteristics of curbs. The sidewalks features (width and slope) were measured and compared with the "American with Disabilities Act" (ADA), resulting data are stored in a GIS layer, which is considered more useful for transportation agencies. Authors of [27] propose a deep neural network approach to extract and characterize sidewalks from LiDAR data. The stripe-based sidewalk extraction is also able to detect sidewalks geometry features like width, grade, and cross slope, and compare them with ADA requirements. To support administrations in assessing existing conditions of sidewalk networks and their compliance with accessibility requirements, authors of [29] used ML, photogrammetry and point cloud processing to extract sidewalk dimensions and conditions. A different approach is presented by [28], where a portable 3D point cloud data acquisition system (called Walkbot) is used to scan and map sidewalk information. Key physical measures are then extracted for the accessibility assessment of the sidewalk. Furthermore, in the literature, there are several approaches to sidewalk detection, and they mainly refer to road boundaries detection and curbs detection methods to separate road pavement from roadside [91]. Curb detection can be based on basic rules related to elevation values, topological relations, and ML and DL techniques. The purposes are various, from urban inventory to path planning. The instruments used to collect point clouds are MMSs and TLS.

With respect to the generation of maps, in reference [30] authors propose the use of smartphone-based photogrammetric point clouds to manually take measures (dimensions of sidewalks, steps and ramps) for the further creation of detailed accessibility maps. Also in [31], authors present an accessibility evaluation model to map urban accessibility. It was then implemented into a GIS for the definition of key destinations and their accessibility.

About safety assessment, [32] proposes a method for safety evaluation of pedestrian crossings basing on geometric and radiometric information from an MLS; they perform an accessibility evaluation by relying on curbs detection to evaluate road entrances accessibility, and they extract longitudinal and transverse slope of the crossing area relying on the coordinates of specific points. Then they check the presence of traffic lights and traffic signs, and they perform a visibility analysis. The results of the analysis are exported into a GIS platform. Other authors [33] evaluate the safety of urban environments focusing on the tripping risk along specific routes; they automatically detect tripping hazard regions by combining Monte Carlo simulations and human behaviour. Their simulations took place on a 3D model reconstructed from a laser scanner point cloud in either indoor or outdoor environments.

Concerning indoor-outdoor accessibility evaluation, the approach presented in [35]

can detect specific floor elements that are in the immediate environment of buildings and that can affect their accessibility, such elements are steps, ramps, curbs, and curb-ramps. The method exploits MLS point cloud and trajectory data, information is extracted using a raster process, connected components and adjacency. Extracted data are exported to GIS to enrich buildings data in Open Street Map. Similarly, in [34], authors focused on the detection of inaccessible steps in building entrances from MLS. Steps are extracted by projecting points into a 2D image, and classified as accessible or inaccessible depending on their height, proximity to the ground, and width. A different approach is presented by [36], where the focus of the paper is not the accessibility of the building entrance, but the development of indoor pathfindings, while taking into consideration accessible routes. In the paper, the authors used 3D models of building elements; obstacles to navigation are detected in the point cloud, and the path is then computed based on surfaces representing floors and doors. Another method for the simulation of wayfinding and indoor accessibility evaluation is presented by [37], where laser scanner point clouds are used to create a 3D model of the environment, later texturized with images; on the basis of the 3D model, a dynamic walking trajectory is generated depending on the presence or absence of useful visible and legible signs. The paths are generated also using a visually impaired simulation model, to detect disorienting places and plan interventions.

The topic of computation of navigable routes in outdoor environments typically includes a first part concerning the classification of urban elements and their accessibility evaluation, followed by the computation of routes through a pathfinding algorithm. An interesting project is the one presented in [39] and [38], where a method for the direct use of MLS point clouds for the development of paths for pedestrians with different motor skills and also taking into consideration possible barriers for people with reduced mobility. The method involves the use of an already classified point cloud, obstacle refinement, graph modelling and the creation of paths considering people with different motor skills. Similarly, the project presented by [113] and [40] has the main goal of developing a tool to assess the accessibility of public space and compute optimal routes. The tool was developed both on web and mobile phone platforms. The starting point of the work are TLS point clouds analysed through specific algorithms, the results were stored in specific GIS raster layers and applied for further accessibility studies. A very recent project, [9], is comprehensive of several previous works, with the aim of computing accessible routes integrating data from multiple sources. In this work, the starting dataset is OSM, which is improved with information extracted from MLS point clouds (ramps, steps, pedestrian crossings). Then obstacles and accessibility problems are detected through VGI, using citizens' mobile devices, and also analysing social network interactions. The computed routes take into consideration all the needs and limited mobility of each individual and are provided to the final users employing a specifically developed mobile application. A different point of view is provided by a last interesting paper [41], where authors propose a method to model urban contours and generate pedestrian maps taking into consideration the location of shaded areas and accessibility barriers. The input data is an MLS point cloud, segmented and rasterized, based on which the navigation map is generated.

The presented papers are summarized in Table 2.1. On the basis of the literature review carried out, some final considerations can be drawn. The first one is based

on the methods. Processed point clouds were acquired mainly by MLS, less often by TLS and rarely using photogrammetry. The processing methods proposed mainly rely on road boundary detection, with the definition of curbs position, curb ramps, and sidewalks area. Curbs presence is considered an important starting point in many of the proposed methods, however, some methods rely on different features, like height difference, or trajectory information. Although most of the papers are based on specific algorithms, developed directly by the authors or adapted from previous papers, there are also some of them, mainly the more recent ones, which count on ML and DL to identify urban elements.

The second one is related to the purposes. The objectives of the analysed works are not homogeneous, but it can be noticed that one of the main aims is the definition of accessible routes, which requires the previous detection and assessment of urban elements. Another frequent goal is the definition of accessibility maps, it is here that all the detected elements and computed paths can be stored, typically in a GIS layer.

The third refers to the case study on which methods are tested. The proposed test dataset varies from portions of roads to portions of cities, but in any case, the analysed areas have a very standard configuration typical of a modern city, where the sidewalks are at a higher level than the road and are separated from it by a curb.

Purpose	Data acquisition method	Case study	Processing strategy	Related papers
Detection of specific objects and assess their accessibility	MLS.	6th district of Paris (France); three sites on Enschede (The Netherlands); Avila, Malaga, and Lugo (Spain).	process range images and project results onto the point cloud; planar segmentation, topology classification, graph libraries; scan line analysis and accessibility evaluation through a voting system.	[23–25]
Sidewalk inventory	Photogrammetry; ad hoc portable scanning device; MLS.	700m of sidewalks in Lisbon, State Route 9 Massachusetts.	ML; scene segmentation using RGB-D camera; comparison with American with Disabilities Act; deep neural network (PointNet++), strip based extraction.	[26–29]
Generate maps and take measures	Smartphone-based photogrammetry.	some specific urban elements (ramps, stairs) in areas of a city in Catalunya.	directly take measures on the 3D point clouds generated by the survey and using RANSAC.	[30, 31]
Safety of urban environment	TLS; MLS.	some areas in Lugo and Vigo (Spain).	ground segmentation, curb detection, pole-like object segmentation.	[32, 33]
Accessibility interior/exterior, indoor pathfinding	Photogrammetry; TLS; indoor MMS; MLS.	rooms inside an academic building; ISPRS benchmark dataset on Indoor modelling; urban road in Vigo (Spain).	wayfinding simulation; obstacle classification, planar regions segmentation, indoor pathfinding; rasterization of point cloud, connected components, topology relations.	[34–37]
Navigable accessible routes	MLS, Open Street Map.	some areas of the city of Avila (Spain); city of Vigo (Spain).	graph modelling, ML, Dijkstra algorithm; Network and obstacle detection, ML, route computation.	[9, 38–41]

Table 2.1: Summary of scientific articles related to the topic of semantic segmentation of point clouds for urban accessibility, subdivided in purpose, data acquisition method, case study, processing strategy.

3

Data acquisition with a Mobile Mapping System

The purpose of this chapter is the presentation of the case study, the description of the type of instrumentation used for the survey, the presentation of the execution of the survey, and the validation of the resulting data. This chapter is preparatory to the development of the entire research work, as its main result is the dataset which will then be subsequently processed.

The chapter is organized as follows: firstly, the city of Sabbioneta (the case study) is described with particular consideration to its urban layout and its street network, which will then be the object of attention in this work. In the following section, the category of instrument chosen for the data collection, Mobile Mapping Systems (MMS), is described in detail. In the last section takes place a description of the exploitation of the geometric survey of Sabbioneta with the instrument Leica pegasus:Two, the raw data processing, and the validation of the resulting point cloud.

3.1 Case study: UNESCO site of Sabbioneta

The case study chosen for the research is the city of Sabbioneta (Figures 3.1 and 3.2) located in Mantova province, northern Italy. On 7th July 2008, Sabbioneta, together with Mantova, were inserted in the UNESCO World Heritage List. At the 32nd session of the World Heritage Committee¹, the two cities have been included in the list because they offer an exceptional testimony to the urban, architectural and artistic achievements of the Renaissance, linked together through the ideas and ambitions of the ruling family, the Gonzaga. From the committee declaration, it is possible to read in detail the reasons that led to the inscription of Mantova and Sabbioneta; which are described below.

Mantova, a city whose traces date back to Roman times, was renewed in the 15th and 16th centuries through urban, architectural and hydraulic engineering works. Sabbioneta represents the construction of an entirely new city, according to the modern and functional vision of the Renaissance. The defensive walls, the chessboard layout of the streets and the role of public spaces and monuments make Sabbioneta one of the best examples of an ideal city built in Europe, capable of exerting an influence on town planning and architecture inside and outside the Old Continent. The two cities represent two significant stages in the territorial planning and urban interventions undertaken by the Gonzaga in their dominions.

¹Decision 32 COM 8B.35 Québec City, Canada, 7 July 2008



Figure 3.1: Some photos of the case study: the historical city of Sabbioneta, which is a Unesco site together with Mantova. a: a bird's eye view of the city, the city centre and the fortified walls with the peculiar shape. b: the Galleria degli Antichi facade on Piazza d'Armi. c: the main facade of Palazzo del Giardino. d: Piazza Ducale, with on the background Palazzo Ducale. e: one of the fortified walls, seen from outside the city. f: Porta Imperiale, one of the entrances of the city. g: the ground floor of the Galleria degli Antichi, which is a porch with cross-vaults and pillars.

Furthermore, Mantova and Sabbioneta represent two exemplary and original urban entities in the panorama of European Renaissance courts, whose characteristics are linked to the personality of masters such as Vittorino da Feltre, humanist artists such as Leon Battista Alberti, Andrea Mantegna and Giulio Romano, and extraordinary prince architects, from Ludovico II to Guglielmo and Vespasiano Gonzaga. However, more than the individual works, it is the organic whole of the various constituent elements that provide the key to understanding the exceptional nature of the site. Mantova and Sabbioneta emerge as two perfect bodies where every single element finds its meaning in relation to the others. Moreover, the two different urban planning processes make the site an exemplary model capable of encompassing all Renaissance urban forms.

The choice of Sabbioneta as a case study for this thesis was driven by several facts. First of all, the strong link between the territory of Mantova and Sabbioneta and the Mantova Campus of the Politecnico di Milano, which since 2012 has been home to the UNESCO Chair for Architectural Planning and Protection in World Heritage Cities. The Mantova campus has a long history of active collaboration with the city of Sabbioneta. In Figure 3.3 there is a map of Sabbioneta with the works carried out by the HeSuTech group of the MantovaLab, which deals with survey technologies for CH.

3.1. Case study: UNESCO site of Sabbioneta

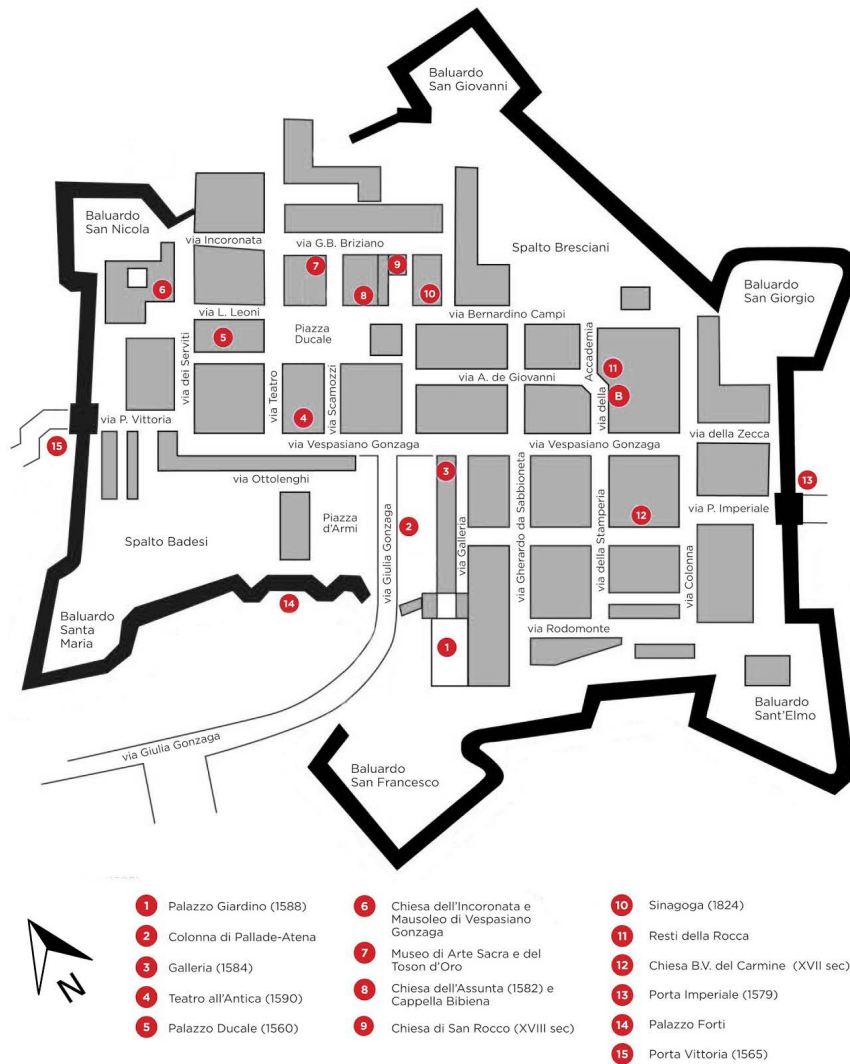


Figure 3.2: A schematic representation of Sabbioneta, its roads (and their names) and the historical buildings defined as POIs for tourists. Modified from a tourists map found online.

A second reason, which is also of greater importance, is related to the structure of the city itself; it has a historical urban context that has undergone some changes over time, and it has the coexistence of interventions made in different time periods. In addition, the particularity of the road system and the street layout represented a set of borderline situations optimal for the definition of an all-inclusive algorithm. The road system presents several specific peculiarities: roads with different widths, several crossings with various shapes, and roads that start with a width and then increase in width; all those elements are better described in the next section.

A third reason is that being a perfect representation of the canons of the ideal city, Sabbioneta encompasses all the specific features of a Renaissance city, albeit reworked and well-conceived to harmonise with one another. In this way, it is possible to recognise Sabbioneta as a perfect base camp for carrying out tests that are then replicated in other historical cities with similar characteristics.

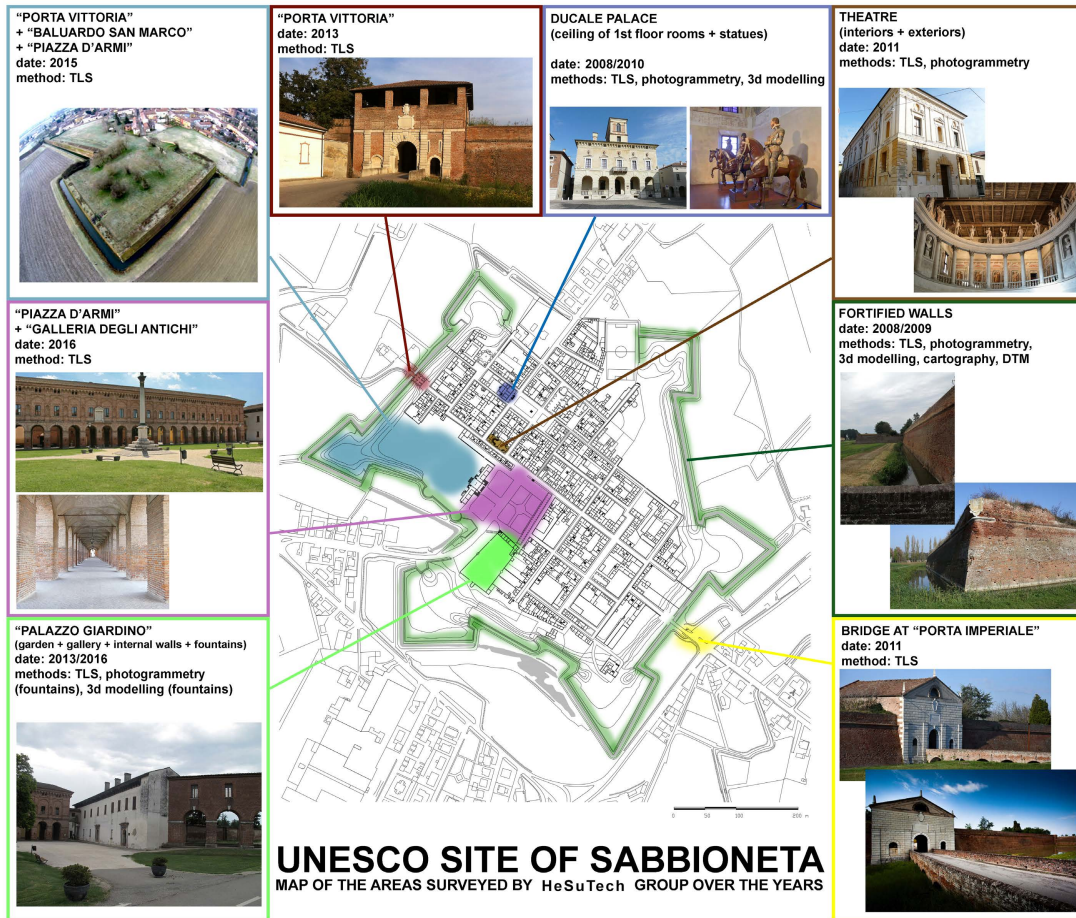


Figure 3.3: Map of the previous surveys and works developed in Sabbioneta by HeSuTech Group, MantovaLab, of Politecnico di Milano, Mantova Campus.

3.1.1 Urban structure of Sabbioneta and its historical evolution

A description of the city's historical evolution and its urban framework is provided in "Sabbioneta. Progetto di una città." [114]. The description that follows in this section is mainly taken from that book.

Sabbioneta was re-founded by Duke Vespasiano Gonzaga in the second half of the 16th century, based on a pre-existing medieval village. The town was enclosed in a hexagonal defensive wall with wedge-shaped corner bastions. This defensive system, whose construction began in 1554 and was then continued by Vespasiano Gonzaga, appeared controlled in its geometry, but irregular, given the presence of the pre-existing medieval fortress. Inside the walls, the city is structured in a pattern reminiscent of the Roman castrum, but also of the developments described in Renaissance architectural treatises on the ideal city. There are, however, exceptions to the regularity: at the points where the chessboard pattern meets the hexagonal fortified walls, the off-centre arrangement of the squares, the irregularity of the size of the streets, and the conformation of the crossroads.

The main road axis is represented by Via Vespasiano Gonzaga, oriented in an East-West direction, which joins the two entrance gates to the city: Porta Vittoria (to the

northwest) and Porta Imperiale, to the southeast. The two squares, located a short distance from each other, represent the centre of civil life (Piazza Ducale) and the area dedicated to the Duke's private factories (Piazza d'Armi). Piazza Ducale houses the Duke's palace and the church of Santa Maria Assunta. It is a rectangular square, enclosed and porticoed on three sides, with the fourth side enclosed by Palazzo Ducale. Piazza d'Armi, on the other hand, is a larger, discontinuous space that has undergone many transformations over time. This space was delimited by single buildings in the 1500s: to the west was the Rocca of medieval origin, separated from the square by a moat, to the south was Palazzo Giardino, used by the duke as a leisure residence and connected to the Rocca by tunnels that still partially exist today: the Corridore Piccolo and the Armeria. To the east, it was bounded by the Corridore Grande (Galleria degli Antichi). Towards the end of the 18th century, the Rocca and the Armeria were demolished, in 1920 a breach was made in the defensive wall and in 1931 the school building was constructed on the site of the Rocca. Two other monuments are located along the main streets: the *Incoronata* complex and the *Teatro all'Antica*. The construction of the main buildings took place between 1575 and 1591, around which rose the palaces of the nobility and the residences and houses.

The street structure defines a system of *cardi* and *decumani* oriented north-south and east-west [115], which, far from being simple, is very reminiscent of the shape of the labyrinth, a recurring figure in Gonzaga iconography. This chessboard shape, reminiscent of the Roman founding city, is the result of Renaissance discussions on the ideal city. In Sabbioneta, however, the chessboard shape clashes with the hexagonal form of the fortified walls. The city consists of 34 blocks, whose morphological structure has remained unchanged and stable over time. The blocks are fairly regular, rectangular or square in shape, but vary in size from 40 to 90 metres, with a predominantly east-west orientation. The blocks close to the walls break up and the streets are lost in a belt of irregular vegetation between the walls and the city.

The streets of Sabbioneta are wide compared to the proportions of the houses, which appear modest and simple while maintaining a relationship with the street. These elements give greater weight to the facades, changing the role of the houses and monumental buildings and changing how the people feel while living this space (*Walking along the streets of Sabbioneta we feel as if we are walking inside the scene of a theatre* [114]). The streets do not have a fixed width; in fact, there are different widths, from 5 metres wide stretches (via Bernardino Campi, a fragment of the pre-existing medieval village) to 14 metres wide stretches (via della Stamperia). Sometimes it happens that the change in width occurs within the same stretch of road, either slightly or more markedly. For example, Via Vespasiano Gonzaga, at Via Teatro, reduces from 9.50 metres to 6 metres. This size reduction is mediated by the presence of the Teatro all'Antica. Remarkable on this change in width, also the paving of this street is different: *sampietrini* for the wider portion and cobblestone for the narrower one.

As a sign of the intention to create an urban stage between the streets of the city, but also for defensive purposes, the crossing of streets rarely takes the form of a classical crossroads but is expressed in more complex forms. This choice also proved to be effective for defensive purposes, just think of the bayonet shape of the city's *decumanus Maximus* (Via Vespasiano Gonzaga) which joins the two entrance gates

within the fortified walls. The decumani and cardi Minori often have T- or L-shaped terminations, while other roads end in blind alleys, and others deviate slightly from the orthogonal conformation. It is here that the main character of Sabbioneta's streets returns: the streets form a closed perspective, *the street becomes like a room and the town like a palace* [114].

After the death of Vespasiano Gonzaga, the city slowly began to empty of activities, functions and people, until today it has only a few hundred inhabitants; *right here where history is important, intact and preserved, everything appears to have stopped in time, without a history that is instead made up of stratifications, transformations and events* [114].

3.1.2 Urban paving in Sabbioneta

Walking in the city, it is possible to note that in Sabbioneta there is a stratification of different materials which characterize the urban environment. Different paving materials are used to separate urban space, the paving assumes the role of highlighting the destination of use of elements. For this reason, there are areas where sidewalks and roads are not identified by different elevations, and can be found at the same level, but can be identified by the use of different materials. This condition is quite typical in Italian historical centres and, for example, can be found also in the city centre of Mantova, that together with Sabbioneta form the Unesco site.

Following this scheme, in Sabbioneta it is possible to note that cobblestone, sampietrini and asphalt are typically used for the road surface, while bricks and stone (mainly porphyry) are used for the sidewalk. The authors of [116], developed an interesting study on both the city of Mantova and the city of Sabbioneta, producing also several useful maps related to different topics about the two cities. One of these maps is referred to the paving materials of the city (see Fig. 3.4) and reflects exactly the characteristics described earlier regarding the material used for sidewalks and streets.

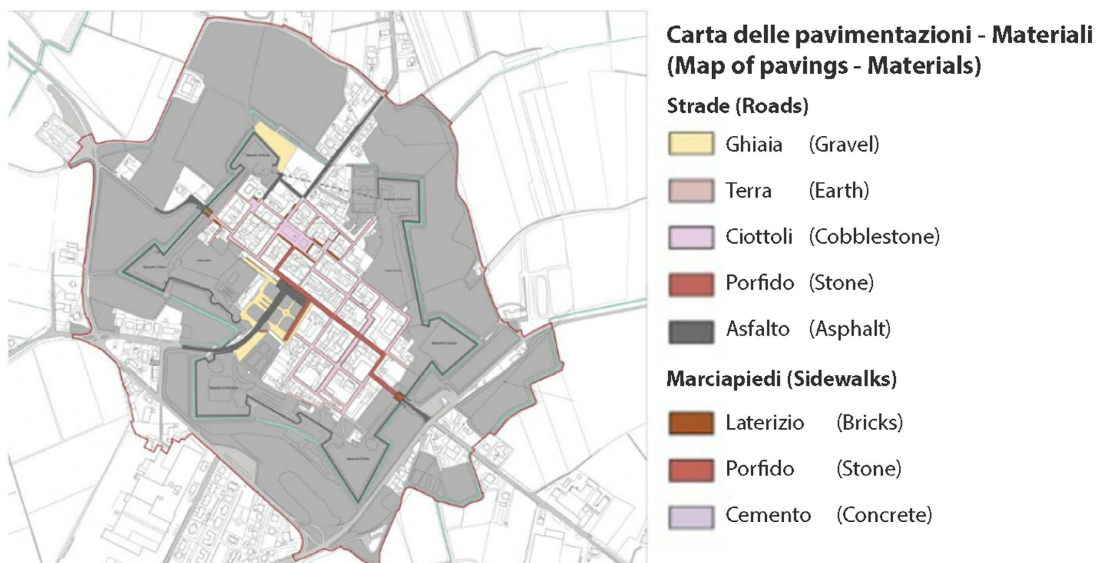


Figure 3.4: Map of the pavings in Sabbioneta, colours represent different materials used for the paving surface. From [116].

3.1. Case study: UNESCO site of Sabbioneta

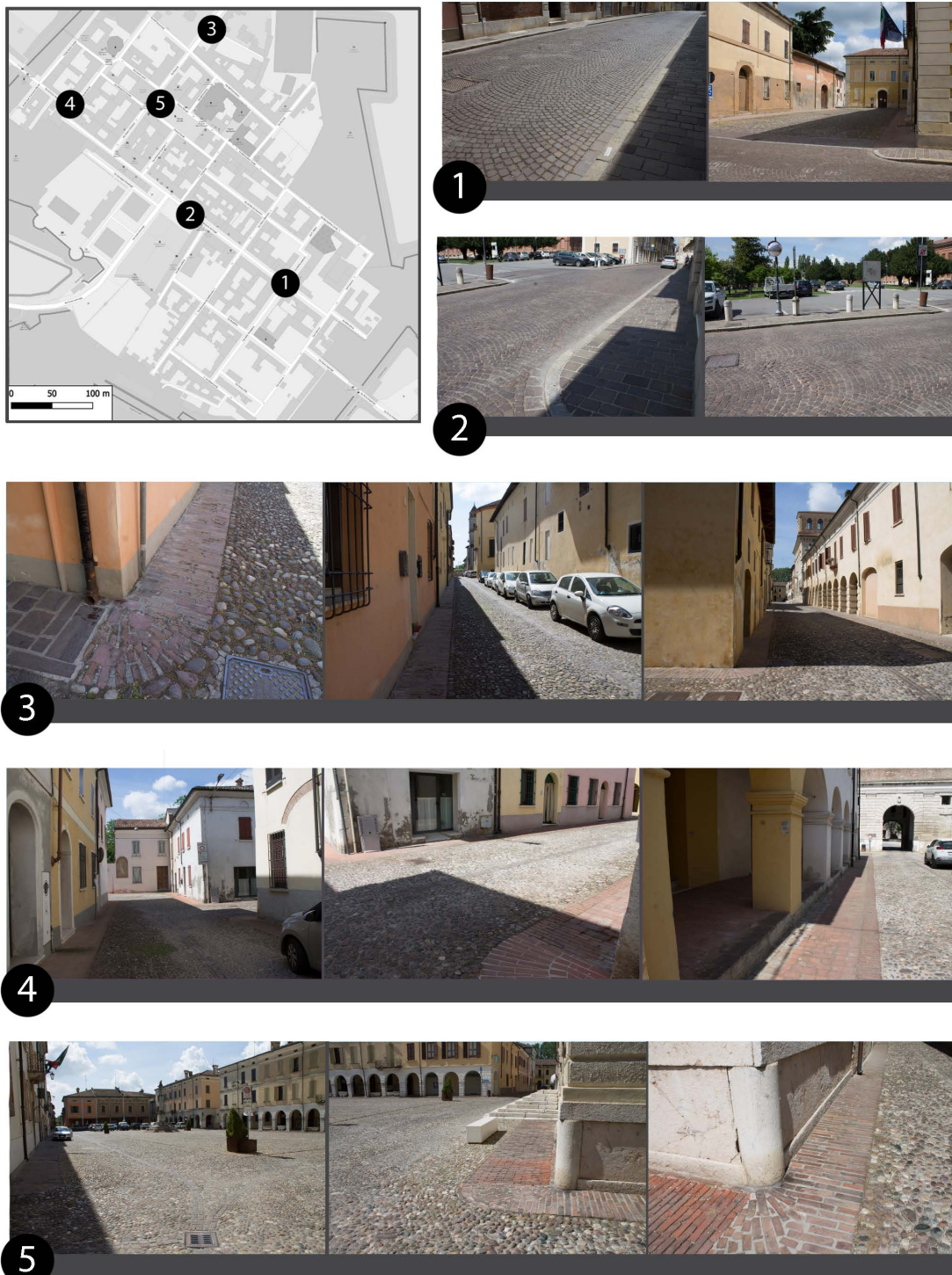


Figure 3.5: Photos taken in Sabbioneta in several areas around the city centre. Urban paving materials are mainly used consistently: sampietrini, cobblestone and asphalt are used for roads, while bricks and stone are used for sidewalks.

Furthermore, Figure 3.5 collects several photos taken during an on-site visit to Sabbioneta and shows exactly that the urban environment is characterized by the stratification of several different paving materials. There are also some exceptions, mainly in the piazzas, where cobblestone are used for the entire area and under the porches there are bricks. The maintenance condition of those paving materials varies a lot, there are elements recently posed and in a good state of conservation, and also elements in a very bad state, which need to be restored.

As an example, the decumanus maximum, Via Vespasiano Gonzaga, has a very particular conformation, in the westernmost part of the street, the width is smaller and the pavement is paved in cobblestones and bricks, while from the Teatro all'Antica and up to the end of the street, the width is almost doubled and the pavement is paved in cobblestones with porphyry on the sidewalks. The two squares also have different paving stones. The street leading to Piazza Ducale, Via Scamozzi, has a cobblestone pavement with porphyry on the sidewalks, while the square is completely paved with cobblestones, the sidewalks surrounding it are made of bricks, and different paving, in terracotta, is used under the porticoes. Piazza d'Armi, on the other hand, presents a very different situation, the road surface is asphalt, surrounded by green areas with trees and grass and gravel paths.

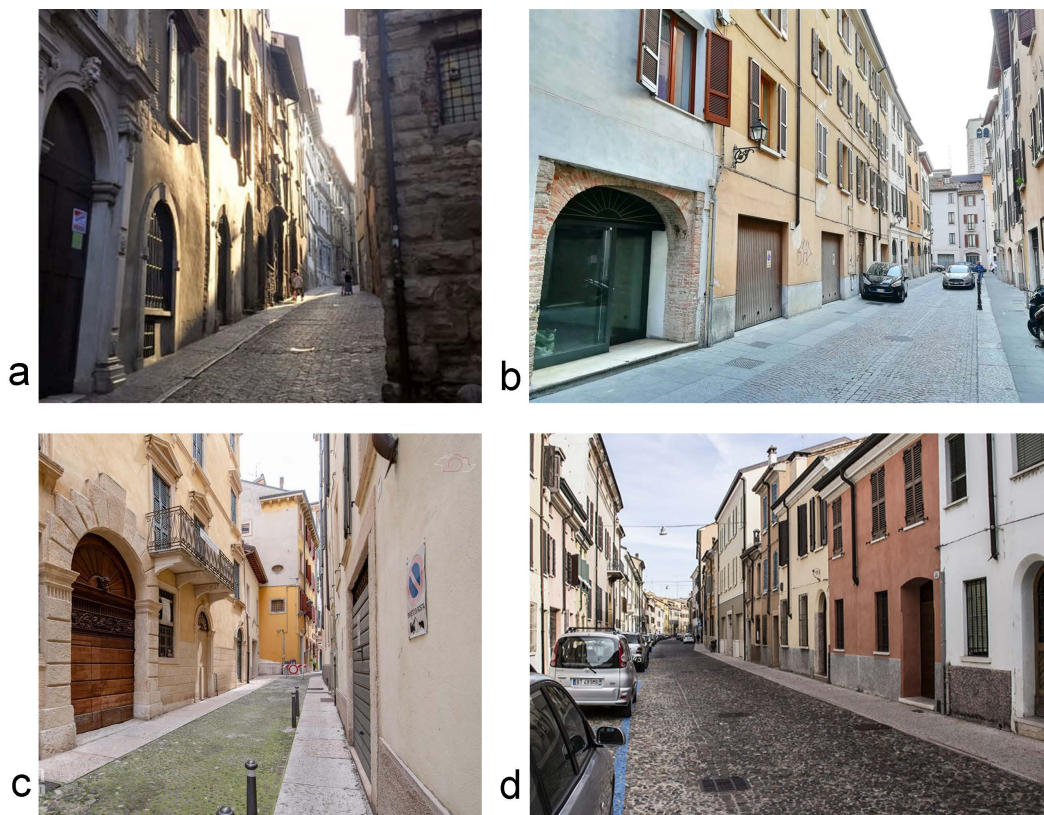


Figure 3.6: Photos taken in historical centres of some Italian cities, where it is possible to note that the peculiarities related to paving materials are used also here as a way to highlight different urban elements. a: Bergamo. b: Brescia. c: Verona. d: Mantova. Images from the web.

3.2. Theoretical background: Mobile mapping technologies

Although the selected case study is Sabbioneta, its urban paving peculiarities are not unique, the same concept (different paving materials for different urban elements) is implemented and replied in several other historical cities or even in historical districts of major Italian cities. In light of what has just been written, the method presented in this thesis becomes valuable and replicable as it can be applied to other datasets with similar characteristics. As an example, Figure 3.6 shows some photos of the urban paving in some historical centres of Italian cities (Mantova, Bergamo, Verona), where different paving materials are used for sidewalks and roads, including the fact that they can be at the same elevation. Similarly, it can be noticed that also in another historical site, in Spain, there is a similar behaviour. Figure 3.7, shows an image of a road in Santiago the Compostela. In this case, the road was the object of a recent renovation and can be noted that the designer decided to place sidewalks and road at the same level, using different pavings for the elements.



Figure 3.7: *Picture of the paving of a road in the historical centre of Santiago de Compostela, Spain. It is interesting to note that also in this case different pavings are used for different urban elements: sidewalks and roads, that are on the same elevation.*

3.2 Theoretical background: Mobile mapping technologies

A metric survey involves the record of several measures on the interesting object. Depending on the survey techniques used, the number of measures can be low: measures taken on specific points, chosen by the technician, using hand measurements, a total station or a GNSS. Or extremely high: millions of measures collected all over the object, implementing TLS, Unmanned Aerial Systems (UAV), photogrammetry, Airborne Laser Scanning (ALS). Depending on the distance from the object a specific acquisition technique can be close range or long range. Widening the scale of the survey object, the appropriate instruments and techniques change. Of course, together with a change in the approach and the scale of the survey, there is also a shift in the resolution and precision of the instruments used.

In dealing with CH and Urban Heritage, Geomatics can play a role in its recording, managing, and documenting. Informed decisions are the basis for good conservation actions [117]. A metric survey is the reference base for further analysis, diagnosis, interventions and monitoring actions. Geomatics approaches and methods can be suitable in CH field both for the formation of a metric basis and for further development of decision systems, like GIS and BIM. Typically for CH documentation, the most

appropriate methods are regarding close range acquisitions, using image data (from space, airborne, UAV, terrestrial platform and cameras) or using range data (pulsed, phase shift, triangulation-based) [118, 119]. The requirement of such works usually is a high point density on the surface of the object.

In addressing Urban Heritage environments, Geomatics can be helpful in all the stages of urban management: from the collection of data to the development of appropriate visualisation and management systems like GIS or 3D city models. Three-dimensional urban models are an integral part of numerous applications, such as urban planning and performance simulation, mapping and visualization, emergency response training and entertainment, among others [120]. An interesting survey approach, whose spatial resolution is more suitable for an urban environment, is enforced by MMS. In fact, they are capable of acquiring very wide objects with a high number of points on them. MMSs have been used in the latest years in several applications and they can be used either indoor or outdoor.

A MMS consists of a data acquisition system combined with the possibility of moving in the environment while acquiring data. Typically, it involves mounting one or more laser scanners and cameras on a moving platform, in combination with direct positioning and orientation sensors [91]. The movement needs to be recorded continuously for the location and orientation of the onboard instruments. Examples of moving platforms are a vehicle, a boat, a user wearing a backpack, a trolley, and a robot.

The accuracy and precision of the data acquired by an MMS depend to a lesser extent on the accuracy of the laser scanner, but for the greater part, they are highly dependent on the performance of the vehicle's self-localization [121]. The continuous collection of point clouds with high point density allows the capture of detailed road features such as curbs and surface condition [85]. Compared with TLS, ALS, and digital satellite imaging technologies, MMS systems represent a more flexible solution and have the ability to collect highly dense point cloud data with cost-saving and time-efficiency measurements [91].

Various applications related to urban management use MMSs as the main urban remote sensing platform. Such applications include 3D map reconstruction for intelligent vehicle navigation and control, 3D city modelling, city visualizations, road asset inventories, railway modelling, vegetation detection and urban forest inventories [121].

An useful table with a list of latest commercial MMS is provided by [122] and [123], a summary is provided in Figure 3.8.

3.2.1 On-board sensors

In general, an MMS system comprehends sensors that have two main purposes: the correct definition of the instrument position during the survey (i.e. the trajectory), and the collection of precise metric data. An accurate definition of the trajectory is mostly important for the correct deployment of the entire survey. The data acquired by onboard laser scanners are merged according to the trajectory of the system.

The typical combination of sensors is the following (see Fig. 3.9 as an example): one or more 3D laser scanners to collect the metric data, advanced digital cameras to provide additive information to the dataset, a GNSS (Global Navigation Satellite System) receiver to provide accurate positioning of the vehicle along the survey, an

3.2. Theoretical background: Mobile mapping technologies

Mobile mapping system		Laser scanner				IMU/GNSS	Absolute accuracy	Digital camera
Make	Model	Sensor(s)	Range	Accuracy (A)/ Precision (P)	PFR	Positioning accuracy (absolute)		Resolution
RIEGL	VMX-2HA	2, RIEGL VUX-1HA	Up to 420m @ p ≥80% and PFR =300kHz	A: 5mm @ 30m (1σ) P: 3mm @ 30m (1σ)	Up to 1MHz @ 235m range	Typ. 20-50mm (1σ)	NA	Options for up to 9 each 5,9,12MP CMOS, Nikon D810 (7360×4912px); FLIR Ladybug5+ (6×5MP)
Teledyne Optech	Lynx SG	2, Lidar sensors	Up to 250m @ p=10%	A: NA P: 5mm, 1σ	Up to 600kHz	NA	±5cm, 1σ @ 10m range and PDOP <4	Up to four 5MP cameras and one Ladybug camera
3D Laser Mapping	StreetMapperIV	1-2, RIEGL VUX 2D scanner	420m @ p =80%	A: 5mm P: 3mm	1000kHz	NA	NA	Ladybug5 6×5MP
Topcon	IP-S3 HD1	1, Velodyne scanner HDL-32E	Up to 100m @ p=100%	A: 2cm P: NA	700kHz	NA	50mm, 1σ @ 10m range and 10mm, 1σ on road surface	Six-lens digital camera system (8000×4000px)
Renishaw	Dynascan S250	1-2, Lidar sensor	250m	A: 1cm @ 50m (1σ) P: NA	36000	2 to 5cm	NA
Trimble	MX9	2, laser scanners	Up to 420m, @ p ≥80% and PFR =300kHz	A: 5mm P: 3mm	Up to 1MHz @ 235m range	0.02–0.05m	NA	One Spherical camera, 30MP (6×5MP); two 5MP side-looking cameras; one 5MP backward/downward-looking camera
Mitsubishi Electric	MMS-G220/ MMS-G220ZL (option)	2, laser scanners; 1 additional long-range and high-density laser scanner (option)	65m; 119m (option)	A: NA P: NA	27.1KHz; 1MHz (option)	Within 6cm (RMS)	Within 10cm (RMS) @ 7m.	Two cameras, 5MP
Leica Geosystems	Leica Pegasus:Two Ultimate	ZF 9012	119m	A: 9 mm @ 50m and p =80% (1σ) P: NA	1.1MHz	NA	0.020m RMS Horizontal; 0.015m RMS Vertical	4 built-in cameras 12MP, optional 1 or 2 additional adjustable external cameras; 2 dual fish-eye cameras 24MP
		Leica ScanStation P20	Up to 120 m; 18% reflectivity	A: 3 mm at 50 m P: NA	1 MHz			
Siteco Informatica	Road-Scanner 4	Faro Focus 130/330	130m/330m @ p=90%	A: 2mm/2mm P: NA	976kHz/976kHz	NA	NA	Spherical Ladybug5 camera 30MP.
		Z+F 9012	119m	A: 9 mm @ 50m and p =80% (1σ) P: NA	1.1MHz			
		RIEGL VQ250/ VQ450	300m/700m @ p ≥80% and PFR = 200kHz	A: 10mm/8mm, @ 50m (1σ) P: 5 mm/5mm @ 50m (1σ)	Up to 300kHz/550kHz			

Figure 3.8: A table, modified from [122] and [123], with a list of some MMS on the market and their characteristics.

Inertial Measurement Unit (IMU) coupled with an accurate odometer, which together provide orientation and position data to help locate the vehicle when the GNSS signals are weak or lost. The main components of a MMS are here described:

- A GNSS receiver, which provides the speed data and the geographical position (i.e. the location of the mobile mapping platform with respect to a global coordinate system) of the MMS along the survey time span [124, 125]. GNSS is intended as the collection of all satellite navigation systems and their augmentations, some of the most popular satellite navigation systems include the European Galileo System, the U.S. Global Positioning System (GPS), the Chinese BeiDou navigation Satellite System (BDS), the Japanese Quasi-Zenith Satellite System (QZSS), the Russian GLOBal Navigation Satellite System (GLONASS), the Indian Navigation with Indian Constellation (Navic) [126]. The GNSS receiver is capable of computing accurate, continuous, worldwide, three-dimensional position and velocity information from satellite navigation system signals. The



Figure 3.9: An exemplification of the instruments of a MMS mounted on a car.

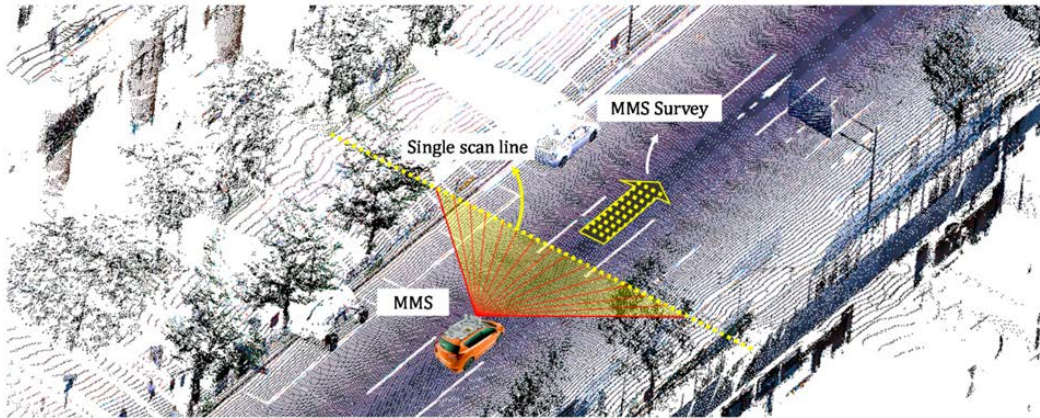


Figure 3.10: An exemplification of how it is conducted a survey with a MMS. In particular, this figure, from [121], shows the scan-lines acquired by the laser scanner while the vehicle moves.

position measure is computed exploiting the satellite signals. The position and velocity of the GNSS receiver can be determined in every moment if a minimum of 4 satellites are visible; the position accuracy is generally in the range of few metres. Some techniques could be used to enhance accuracy, robustness, and reliability of GNSS signals. The relevant technique for navigation is the dynamic Real Time Kinematic (RTK), which requires a minimum of 5 satellites, should be initialized on-the-fly, and need a base station. After the ambiguities have been resolved, centimetre accuracy can be reached. Obstructions such as buildings or trees that block the satellite signal, and multipath interference could result in unreliable navigation, and in such cases, the integration with IMU can be helpful. [125, 127]

- The IMU contains a cluster of sensors (two or more accelerometers, usually three, and three or more gyroscopes, usually three) that together with a Navigation computer can maintain and estimate the position, velocity, attitude, and attitude rates of the vehicle in which it is carried; these sensors are rigidly mounted to a common base to maintain the same relative orientations [128]. The IMU is

used in combination with the GNSS receiver, it can be helpful when the GNSS receiver loses the signal because of multipath effects and signal losses (e.g., due to tunnels, high buildings, road slopes, and tree canopies) [125].

- The laser scanner continuously emits laser pulses with a near-infrared wavelength to the surfaces of specified targets and digitizes the backscattered signals. Then, the scanning distances between the scanners and scanning targets can be calculated to obtain coordinates and intensity information. Typically, two main techniques are applied to range measurements in the plurality of MMS systems: time of flight and phase shift [91]. As a best practice, laser scanners are rotated horizontally or inclined vertically to increase the probability of contact between the laser scan plane and any surfaces that are perpendicular to the direction of travel [129]. Considering the operation of an MMS, each survey is made from thousands of scan lines where the relative accuracy between two consecutive scan lines is at a centimetre level. Figure 3.10 illustrates the definitions of the laser scan line and MMS survey. The absolute accuracy of the survey is defined by the continuous quality of the scan line registration, which is affected, as already said, by satellite visibility, the performance of the IMU, and driving conditions. [121]
- To provide visual coverage, most commercial MMS systems are integrated with high-end digital cameras, so that point clouds are realistically rendered using optical colour imagery data. However, digital cameras play a secondary role in that they are applied in visualization. [91]

3.2.2 Direct georeferencing

The onboard sensors of an MMS and their relative positions are carefully designed. The mechanism of direct-georeferencing (Fig. 3.11) to get the point P global coordinates exploits the positioning information of the GNSS antenna, the details of IMU and the laser scanner acquisitions [91].

In detail, the global coordinates of point P (X_P, Y_P, Z_P) are defined starting from the coordinates of the GNSS receiver ($X_{GNSS}, Y_{GNSS}, Z_{GNSS}$), using the relative position between IMU, GNSS antenna and laser scanner (L_x, L_y, L_z) and (l_x, l_y, l_z), corrected by the roll, pitch and yaw details of IMU in the mapping coordinate system, and considering the observation made by the laser scanner (in terms of shooting range d and scanning angle α). Other parameters are identified via system calibration.

3.2.3 Portable Mobile Mapping Systems

Specific types of MLS are the Portable MMSs. Portable Mobile Mapping Systems (PMMS), also called Wearable Mobile Laser Systems (WMLS), allow the generation of dense, geo-referenced, three-dimensional models while the operator is moving through the scene wearing a backpack containing the instrument, holding it in a hand, or pushing the instrument mounted on a trolley. The scope of application of PMMS is diverse: from CH buildings and civil structures to industrial sites, indoor spaces, and natural environments [130]. Their distinctive feature is the ability to acquire 3D information on the move, and, unlike vehicle-based MMS, the possibility to efficiently document narrow, indoor and confined environments [131]. A list of recent PMMS is provided by [131].

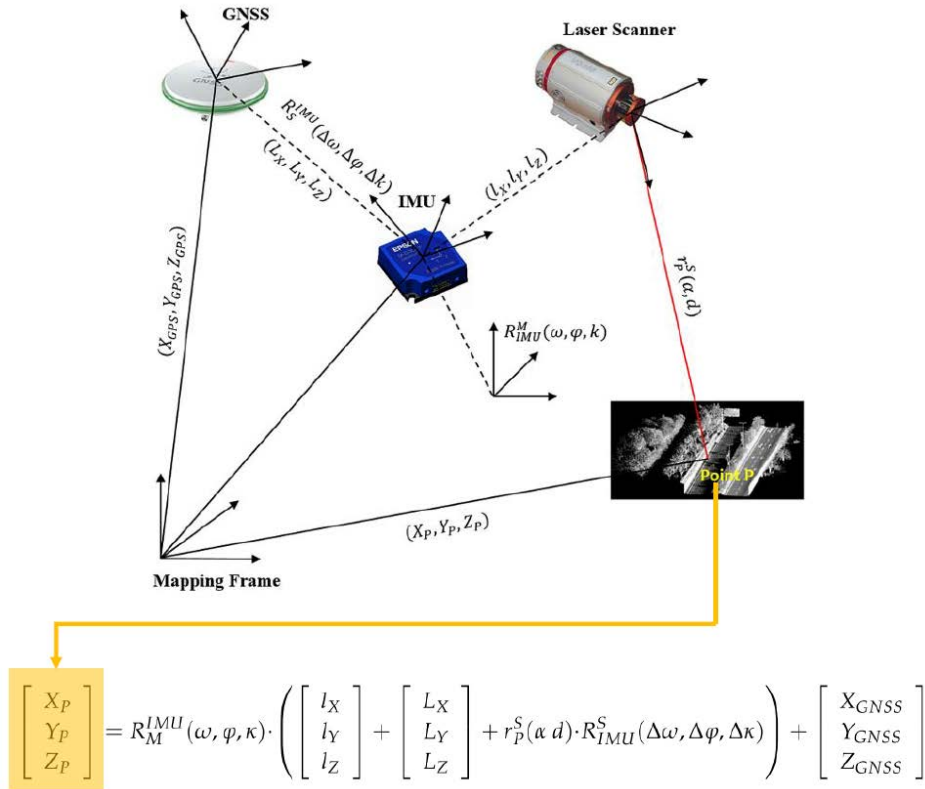


Figure 3.11: Scheme and equation of georeferencing approach for a MMS. Adapted from [91]

The situations in which GNSS data are not available urge a different approach to be used in solving the problem of instantaneous navigation. The analytical methods used to solve this problem often belong to the class of so-called "simultaneous localization and mapping" (SLAM) algorithms. With a SLAM algorithm, it is possible to simultaneously build a map and localize the sensor within the map: obviously, the essential aim is the mapping, but the localization problem is also solved, almost in real-time. [132]

MMSs are both suitable for indoor mobile mapping [133, 134] and outdoor mapping [132, 135]. While MMS are most suitable for 3D city modelling, PMMS can have a wider range of applications, they are suitable not only for city modelling, but also for Building Information Modeling, As-built mapping, industrial mapping, indoor navigation, underground mapping, post-disaster assessment, forest mapping, and forensic documentation [131].

Intending to survey an urban environment that has a vast extension, the most suitable technique is MMS, but when dealing with historical sites, mountain paths, and indoor environments, the PMMS can be a valid alternative. Even if MMS and PMMS are comparable techniques and share the same working principle, there are also some differences between the two.

The main differences are in the density of points, acquisition time, battery life, and accuracy of the mounted sensors. Usually, in a PMMS the main constraint is a limited volume and weight, which has an impact on his accuracy performances. Plus, PMMSs are mainly conceived for indoor use, so not all the instruments on market have the

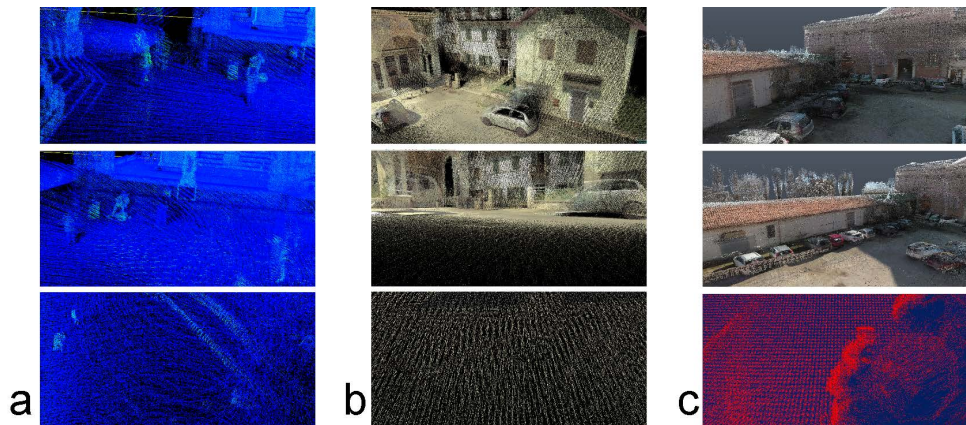


Figure 3.12: Screenshots of some point clouds acquired by mobile mapping technology. *a*: Kaarta Stencil. *b*: Leica Pegasus Backpack. *c*: Gexcel Heron. The three bottom images show the acquisition pattern of the instruments and also remark on the fact that there are some holes between superimposed single scan lines.

GNSS, but rely only on the IMU and SLAM to process instrument position.

Also, the density of points is an issue; in fact, even if the mounted laser scanner has comparable accuracy to an MMS, the acquisition path is not so homogeneous as the one for an MMS, resulting in possible "holes" in the acquired surfaces. This fact can be seen also looking at some datasets acquired using MMSs. We had the opportunity to observe a survey deployed by Gexcel Heron and Leica Pegasus, thanks to prof. Francesco Fassi, 3D Survey group of Politecnico di Milano and a survey made by Kaarta Stencil, thanks to Microgeo srl. The accuracy of each instrument can be found in their technical specification: Leica Pegasus Backpack: 2-3 cm, Gexcel Heron: 2-5 cm, Kaarta Stencil: 3 cm. The point cloud density depends on the acquisition speed (round per minute of the LiDAR), but also on the speed of the user while acquiring data. The resulting point cloud from each acquisition instrument has a different pattern. The acquisition of a generic surface is given by the superposition of several scan lines. Since the superposition does not cover exactly the whole surface, there are always some holes. Figure 3.12 shows some images from the available datasets, where it is possible to note (on the bottom images) the aforementioned holes in the acquisition.

3.3 Exploitation and validation of the geometric survey

The starting point of the thesis work is a point cloud of the city, that can be obtained by exploiting a metric survey. The elements of interest are streets, sidewalks, door entrances, porches and all other aspects of the urban environment that can relate to physical accessibility.

The requirements of the survey are several: the object of the survey is very wide and comprehends almost the entire city of Sabbioneta, the interesting elements are mostly located on the ground, the density of points on the object should be enough to perform processing, to identify materials, and characterize the elements.

Among the possible methods, it is possible to consider deploying several range

scans with a TLS and then merge them into a single point cloud, using a topographic network as support. This method will produce a point cloud with high density on the ground, but it will be highly time-consuming and will also produce a very huge amount of data, difficult to be further processed.

Another possibility is to use Airborne Laser Scanning (ALS) or Unmanned Aerial Systems (UAS) photogrammetry. In both cases, the quality and density of the final data depend on the flight conditions (e.g., the altitude at which the acquisition is performed) and the characteristics of the instrument used. Such approaches can be considered effective for the purposes discussed here, when the point cloud resulting from the survey possesses the appropriate characteristics, like appropriate point density and accuracy. ALS data on the Italian national territory, although available in open format upon request to the relevant office of regional and national administrations, cover only 63% of the Italian territory, and are characterized by flight altitudes between 500 and 3000 m, and with pulse density between 0.4 and 5 pulses per square meter [136]. It can be deduced that the low density of ALS data does not make it suitable for the purposes of this research. Speaking of UAS systems, on the other hand, point density is also a function of the flight altitude and the resolution of the on-board camera, it turns out that it is possible to calibrate these parameters when acquiring photographs in order to achieve a high point density on the ground; however, it should be considered that UAS surveys could incur on flight restrictions, which are common in urban areas and historic sites.

In light of that, one of the most appropriate methods appears to be MMS, because it allows a fast acquisition, with a high density of points, and appropriate accuracy. Considering that in historical environments some areas are not accessible by a car and can be accessed only by pedestrians, probably a PMMS should be preferred. It should also be mentioned that a survey carried out with an MMS mounted on a car, or carried out with PMMS, is the ideal solution when it is not necessary to survey roofs. In such a case UAS or ALS survey should be preferred.

The city of Sabbioneta was the subject of several survey campaigns in past years. They were done by the HeSuTech group of MantovaLab research laboratory, in Mantova Campus of Politecnico di Milano (for a detailed description of the surveys over time see Figure 3.3). These datasets were acquired through terrestrial laser scanner (TLS) technology. The laser scanner used was LeicaHDS 7000 and the range scan resolution was set to 6 mm at 10 metres. The team developed also a topographic network to georeference and connect the several range scans acquired in a complete point cloud. These existing works were done in different years, with different purposes, and in all cases the urban environment was not the purpose of the survey. For these reasons, it was decided to perform a new metric survey devoted to acquire the ground and the urban elements of the city.

Considering the areal extension of the city and the fact that all the roads, even the narrowest, are wide enough for a car, we decided to perform a geometric survey using an MMS mounted on a car. We decided to use Leica Pegasus:Two.

The workflow of this thesis work will be produced, tested and evaluated on the dataset acquired by this MMS. Considering the similarities between MMS and PMMS, the workflow parameters could be ideally adjusted to work also with dataset coming from other instruments, such as a PMMS.

3.3.1 The instrument

Leica Pegasus:Two² is a multi-sensors platform that mounts 8 cameras, 1 laser scanner, IMU and GNSS sensors. An image of the instrument and depicted onboard sensors is provided in Figure 3.13. Laser scanner determines the relative position and reflectance properties of points, cameras provide 360-degree coverage to complement laser colour data; images provide feature identification, rapid navigation through data, and stereo measurements. The density of the point cloud is dependent on the scan rate of the scanner, the number of points collected per second, and the speed that the vehicle is travelling. The density of the image coverage is based on an operator specified distance and is limited by the maximum frame rate of the video cameras.




Figure 3.13: Leica Pegasus:Two and typically mounted sensors. From instrument's datasheet.

According to the producers, Leica Pegasus:Two can be combined with sky cameras for city modelling, thermal imaging for energy loss detection, an additional camera for pavement assessments, pollution sensors to assess air quality, and ground-penetrating radar for underground asset management. This platform is vehicle and application-independent, in fact, it can be used for road, rail and water-based surveying.

The Leica Pegasus:Two can mount, as a laser scanner, either a Leica P20 or a Z+F profile 9012. The one used in Sabbioneta mounted the latter, whose main characteristics are: 200 Hz rotation speed, one profile every 5 cm at a speed of 36 km/h, an acquisition range from 0.3 to 119 metres, a 360 degree Field of View, a maximum scan rate of 200 profiles per second, and a scan rate of 1.016 million points per second. As provided in the technical data sheet of the instrument (see Fig. 3.14), the instrument has a declared typical accuracy of 0.020 m RMS in horizontal and 0.015 m RMS in vertical. The laser scanner can acquire up to 1 million points per second if the car moves at a speed of 40 Km/h. The acquisition range is 100 m. The IMU has a frequency of 200 Hz, so accidental bumps (e.g., due to the road surface) are not a problem. The data produced while working are 1.1 GB/Km, which

²https://leica-geosystems.com/it-it/products/mobile-sensor-platforms/capture-platforms/leica-pegasus_two

Chapter 3. Data acquisition with a Mobile Mapping System

CAMERA SENSOR		BATTERY	
Number of cameras	8	Weight	34.8 kg
CCD size	2000 x 2000	Size	65 x 32 x 37cm
Pixel size	5.5 x 5.5 microns	ENVIRONMENTAL	
Maximum frame rate	8 fps x camera, equal to 256 M pixels x second (collected, compressed, stored)	Operating temperature	0° C to + 40° C, non-condensing IP protection level IP52, excluding the scanner. Please refer to scanner documentation.
Lens	8.0 mm focal, ruggedised; 2.7 mm focal, top	Storage temperature	- 20° C to + 50° C, non-condensing
Coverage	360° x 270° excluding rear down facing camera	TYPICAL ACCURACY*	
SCANNER		Horizontal accuracy	0.020 m RMS
Please refer to scanner manufacturer datasheet.		Vertical accuracy	0.015 m RMS
CONTROL UNIT		Conditions	Without control points, open sky conditions
Multi-core industrial PC, low power consumption, 1 TB SSD hard disk with USB3 interface, USB, Ethernet, and wireless connections available through the battery system. Service support available through remote interface.		PRODUCTIVITY*	
BATTERY SYSTEM PERFORMANCE		Data produced per project (compressed)	43 GB/h or 1.1 GB/km
Typical operating time	9 hrs, profiler version; 13 hrs, scanner version	Data produced after post processing (images and point cloud)	60 GB/h or 1.5 GB/km
VAC input voltage	100 min to 240 max VAC autoranging	Post processing time	1 hr of data collection equals 1 hr post-processing without colourising, 1 hr of data collection equals 5 hrs of post-processing with colourising.
AC input power (charge cycle)	350 W Max	EXPORT OPTIONS*	
AC input frequency	50/60 Hz	Images	JPEG and ASCII for photogrammetric parameters
Time to full charge	11.0 max h starting 0 %	Point cloud	Binary LAS 1.2, X,Y,Z, intensity, RGB values Colourisation by camera pictures Hexagon Point Format, Recap
DC output	21 - 29 volts	ACCURACY TEST CONDITIONS*	
Watt/Amp hours	2685 Watts hours / 104 Amp hours	Scanner frequency	1,000,000 points per second
GNSS/IMU/SPAN SENSOR		Image distance	3 m
Includes triple band - L-Band, SBAS, and QZSS for GPS, GLONASS, Galileo, and BeiDou constellations, single and dual antenna support, wheel sensor input, tactical grade - no ITRAR restrictions, low noise FOG IMU.		Driving speed	40 km/h
Frequency	200 Hz	System configuration	No wheel sensor, no dual antenna
MTBF	35,000 hour	Laser scanner	ZF 9012
Gyro bias in-run stability (±deg/hr)	0.75	Max baseline length	3.2 km
Gyro bias offset (deg/hr)	0.75	REPEATABILITY*	
Gyro angular rand. walk (deg/√hr)	0.1	Based on open sky, GP5+GLONASS processing, and phase differential. Points were measured manually from within the point cloud. A ring with 26 check points were collected 4 times, for a total of 104 observations. Check points were measured with TPS and levelling.	
Gyro scale factor (ppm)	300	Resulting mean error for X,Y,Z was -0.004,-0.004,0.001 meters, and the resulting standard deviation for X,Y,Z was 0.011,0.012,0.008 meters.	
Gyro range (±deg/s)	450	From left to right: Optional wheel sensor, battery with power cable and rain cover, sensor system.	
Accelerometer bias (mg)	1		
Accelerometer scale factor (ppm)	300		
Accelerometer range (±g)	5		
Position accuracy after 10 sec of outage duration	0.020 m RMS horizontal, 0.020 m RMS vertical, 0.008 degrees RMS pitch/roll, 0.013 degrees RMS heading.		
OPTIONAL ACCESSORIES			
Wheel sensor 1,000 pulses per rotation, IP 67, integrated time stamping of wheel sensor data (handled by GNSS controller). Processing of wheel sensor data is integrated with the Kalman filtering based trajectory computational software. A variety of wheel sizes supported.			
Rotational platform Optional rotational platform is available to provide an alternative scanner or profiler position while maintaining the camera geometry.			
SENSOR PLATFORM			
Weight	51 kg (without case), 86 kg (with case)		
Size	60 x 76 x 68 cm, profiler version 60 x 79 x 76 cm, Leica ScanStation P20		
Size with case	68 x 68 x 65 cm		

* If not specified, data refers to a Leica Pegasus:Two with a ZF9012 profiler and an iMAR FSAS IMU. Datasheet is subject to change without notice.

Figure 3.14: Technical datasheet of Leica Pegasus:Two.

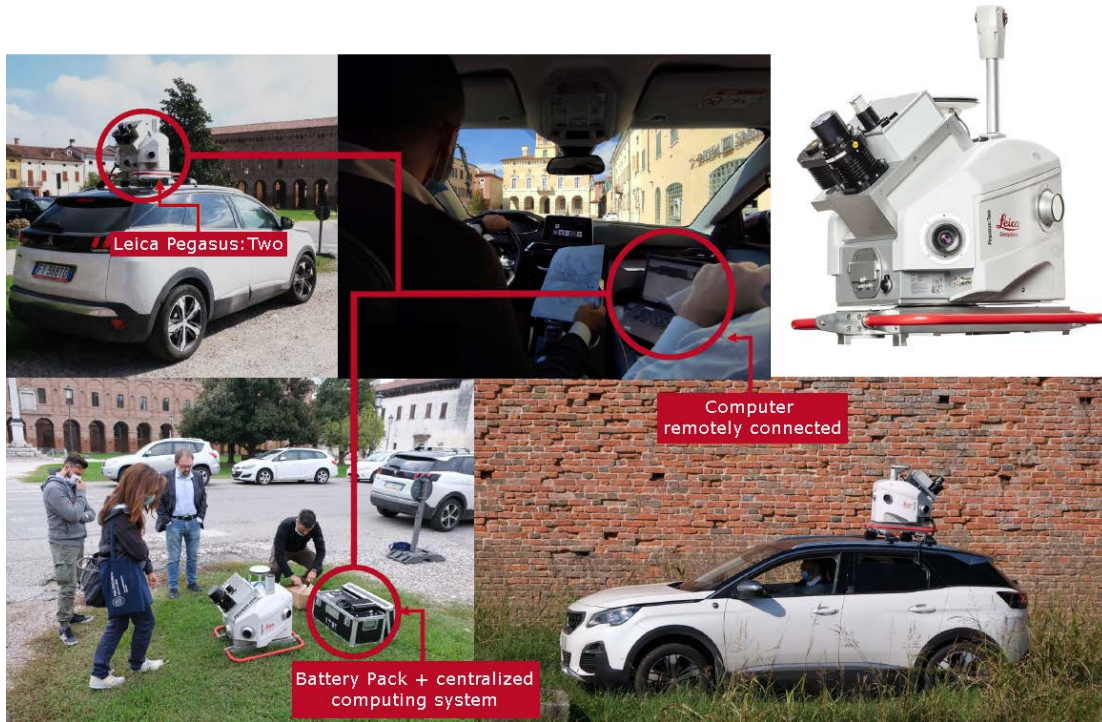


Figure 3.15: *Leica Pegasus:Two on-site during the survey. Top right: the instrument. The other photos were taken during the survey and represent the main components of the system.*

after processing becomes 1.5 GB/Km. The usual processing time with respect to the acquisition time has a proportion of 1:1 when processed without colourising the point cloud, which increases up to 5:1 when colourising. Battery operating time is 9 hrs in the profiler version and 13 hrs in the scanner version. Optionally, also a wheel sensor can be used to help the trajectory computation. Figure 3.15 shows the instrument during the survey on Sabbioneta. There is the instrument itself which contains all the sensors (mounted on top of the car), the battery and the central computing system (which are inside the car), then a laptop is used as a remote controller of the entire system. All data are stored inside the instrument.

3.3.2 On-field survey

In October 2020 Leica Geosystems provided the system for the metric survey. This section describes the work done on-site, during the acquisition of a wide portion of the city of Sabbioneta. The first important step is to carefully plan the survey, in fact, to reduce processing time it is better to avoid certain behaviours, some of which are listed here:

- do not use the reverse gear, because it creates problems with the IMU sensor;
- it is better to make several short missions instead of a single big mission, to avoid the acquisition of the same areas too many times; this phenomenon might occur because in the processing step the software tries to find common points as a constraint, too many constraints can be a source of too long and time

demanding processing or not convergence of the algorithm which results in an error of trajectory reconstruction;

- trajectories without too strict curves will be better processed later;
- the car should be driven such that there is a sort of symmetry of acquired volumes (i.e., the car should be in the middle of the road, if possible);
- if along the survey trajectory there are some points of known coordinates, plan the survey such that they are acquired and their coordinates can be used to help the processing phase.

The presented recommendations, coming from Leica technical experts, mainly help the post-processing step, where all the acquired points should be processed according to the trajectory. A good definition of the trajectory is of primary importance for a good result. It appears clear that a good plan of the missions (or tracks of acquisition) is very important. In Sabbioneta, before starting the acquisition phase, the survey technician planned the survey according to the previous considerations and based on the urban scheme of the city. Planning the path, they included in the survey the two piazzas, a portion of the walls and the two main doors of the city. Then they prepared a plan composed of several acquisition paths, that together can cover almost the whole city.

When the survey is planned, before starting it is necessary to calibrate the GNSS; to do so the system has to perform an acquisition in static mode for half an hour. Then it is necessary to calibrate the IMU, by driving while making rectangular trajectories with sudden braking and changes of direction. Before starting also the white balance of the cameras should be set. Once it is set at the beginning of the survey, it cannot be changed during a mission.

The complete survey was done in 10 missions, or tracks of acquisition, which all together compose the survey. Each track was acquired with images, to colourise the point clouds in the processing phase. The driving speed was, on average, 30 Km/h. The complete path, summing all the tracks, consists of 7714.13 metres and, disregarding the calibration phase, was completed in 40 minutes. The length of each track is shown in Table 3.1. Almost the entire city was surveyed, including the two main Piazzas: *Piazza d'Armi* and *Piazza Ducale*, practically all the roads, and also a part of the fortified walls.

Specifically (see Fig. 3.16), Track A, D, and I are related to the North-East part of the city centre, including *Piazza Ducale*. Track B starts outside the city centre, at North, passes outside the fortified walls on the South-East and concludes its path in *Piazza d'Armi*. Track C passes through the two doors of the city, *Porta Vittoria* on North-West and *Porta Imperiale*, on South-East, passing on the main axe of the city: *via Vespasiano Gonzaga*. Track E, F, G, and H are related to the South-East part of the city centre. Track J is entirely devoted to the acquisition of the South portion of the fortified walls.

3.3. Exploitation and validation of the geometric survey

Name of the track	Length (m)	N. of points in the p.c.
Track A	1070.10	163,978,878
Track B	1684.40	246,213,213
Track C	975.03	121,343,926
Track D	1028.40	167,054,560
Track E	707.48	110,907,544
Track F	181.82	35,589,428
Track G	77.85	23,610,095
Track H	165.93	43,903,368
Track I	151.22	32,262,130
Track J	1671.90	264,605,024
Total	7714.13	1,209,468,166

Table 3.1: Description of length (in metres) and number of points in the point cloud for each acquisition Track, from Track-A to Track-J.

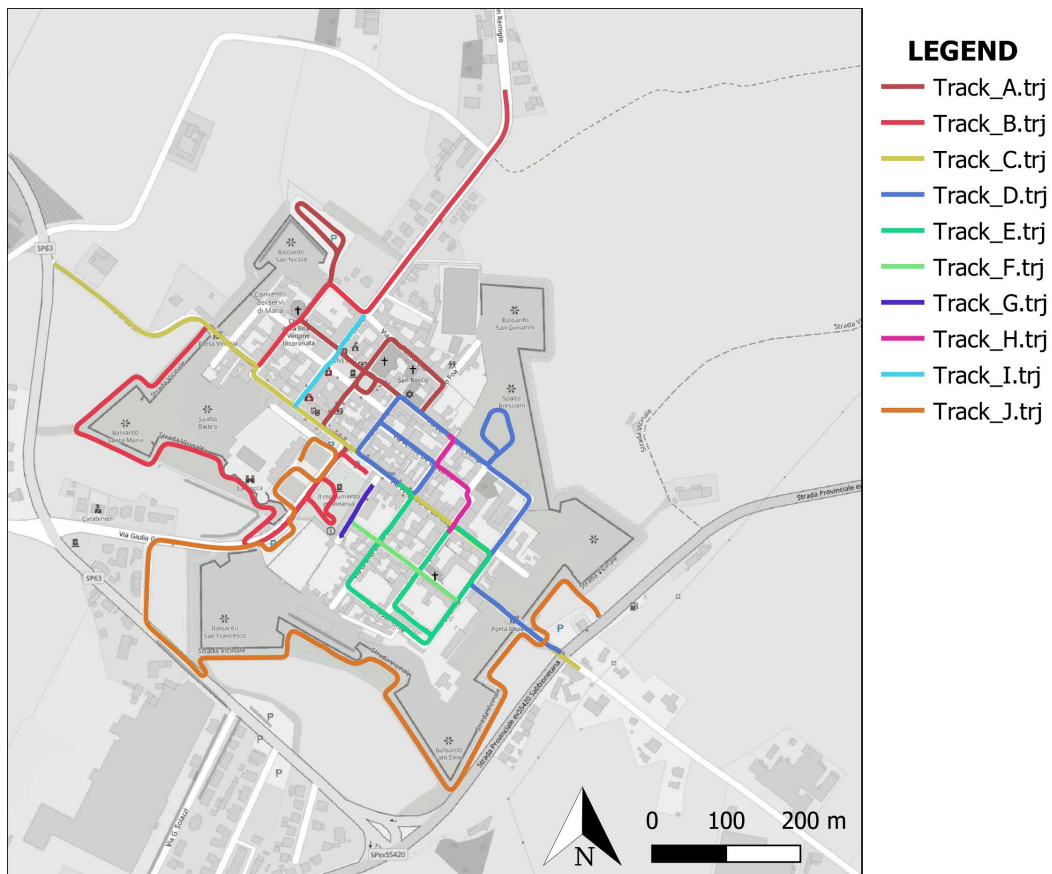


Figure 3.16: Map of the 10 tracks (or missions) of acquisition done while surveying Sabbioneta with Leica Pegasus: Two.

3.3.3 Survey data processing

After the on-site acquisition, all the acquired data are stored in a set of several folders, sub-folders and files which are not in a usable format. These data must be processed together to produce a final deliverable. In fact, during the on-site acquisition phase, all the sensors work alone, then in the processing stage, all the data are put together on the basis of the acquisition trajectory. The processing software for this purpose is Leica Pegasus Manager.

Pegasus Manager³ is subdivided into sub-modules that allow performing specific steps, each of them for a specific task: mission planning, processing, basic and advanced trajectory adjustment, feature extraction, visualisation. The Mission Planning module, based on the area of the survey, the building's position and height, and the GNSS signal, helps the user in the definition and planning of the missions. It can calculate the best time for data collection, predicts expected trajectory quality, forecasts GNSS availability, and estimates shadow effects. The Processing module involves the integration of data coming from different sensors (IMU, GNSS, cameras, and laser scanner), and the production of a unified complete 3D project, composed of a calibrated and georeferenced combination of points and images from all missions. The Trajectory Adjustment module is used to refine the trajectory, it can be done based on auto-detected tie points between multiple passes, combining overlapping trajectories, or using user-defined ground reference points. Feature Extraction is possible through some modules which are related to roads, railways, and image manipulations. This module can work on images to automatically detects cars and people, and blurs them completely to make them unrecognisable (e.g., for legal purposes); it can extract road elements acquired with the system (curbs, lane lines, street furniture) and create road profiles, DTM, and contour lines; it extracts rail elements (wires, cabling, platforms, catenary, sleepers, trackbed, vegetation and buildings). Finally, the viewer is a free tool that allows to quickly check the quality and accuracy of data.

The data processing involving Pegasus Manager was performed by Leica Geosystems technicians, following a workflow that involves the aforementioned software sub-modules. The final result was then provided to He.Su.Tech Laboratory to perform research tests, like the one presented in this thesis work. Leica technicians organised a specific webinar to show how they completed the work. For this reason, the main steps of the data processing workflow they followed are summarised in the next paragraphs.

The first and most important element of the entire process is the trajectory definition. The raw data coming from Pegasus:Two are firstly used to define an initial rough calculation of the trajectory. A statistical computation is performed on IMU and GNSS data, following the kinematic registration chain from start to finish, from finish to start or both ways. Several thresholds and parameters can be set to give weight to IMU and GNSS data according to the survey conditions.

When the trajectory is defined, scans and images are imported and connected to the trajectory, according to the time of acquisition. Each point acquired by the laser scanner has, as an attribute, the timestamp of when it was surveyed, the same goes for the images and the GNSS points of the trajectory. In this way points were mounted on the trajectory and photos were oriented and used to add colour information to points.

³<https://leica-geosystems.com/it-it/products/mobile-sensor-platforms/software/leica-pegasus-manager>

3.3. Exploitation and validation of the geometric survey

At this stage, a unique point cloud for each mission is generated. It is also possible to filter the point cloud, for example, according to a range filter, which removes too far points. In this project, a range filter was applied, and points more than 50 metres away from the scanner were removed. Another possibility is to apply masks to all the photos, to remove elements that are consistently in each image, for example, the area of the photo that covers the back of the driving car. In this survey, the back of the car was present in all photos and it was masked away.

Now, all the tracks can be visualised as attributes in a shapefile, they can be selected and the associated point clouds and images can be inspected through the Pegasus Visualizer module. This module allows to select a specific portion of a single track, visualise the point cloud connected to it, check the images and select which of them should not be used (e.g., the ones with bad exposure), make slices and check if there are misalignment errors between scans of the same object acquired by different tracks. It is also possible to perform features extraction to segment and classify urban objects. In this project, feature extraction was not performed. Even if the process seems to be completed, the workflow is not concluded, in fact, a deep examination of the result is necessary before exporting the data in a usable format. To help this check, Pegasus Manager can colour the point cloud according to a quality factor, established by the software.

If after the check, there are some visible errors (e.g., misalignment between scans), the user can use the Advanced Adjustment mode to correct them. There are three possibilities for advanced processing: adjustment in multi-pass, time adjustment, and target-based adjustment. The multi-pass recalculates the trajectory according to automatically detected homologous points (tie-points), in areas where two tracks overlap. Automatically detected tie-points can be integrated with user-defined points. Time adjustment redistributes the errors according to the acquisition timeline. Target-based adjustment implements user-defined coordinates of specific points to recalculate the trajectory. Leica Geosystems used some GNSS-measured points that were surveyed by HeSuTech laboratory the day after the Pegasus:Two survey (those points are discussed in paragraph 3.3.4). To correct the trajectories of each mission, Leica technicians used a combination of multi-pass and target-based adjustments. Figure 3.17 shows a screenshot of Pegasus Manager while making an advanced adjustment of the trajectories, green and blue triangles and squares represent tie-points and checkpoints defined by the user and by the multi-pass method. After the definition of the adjusted trajectories, the previous steps were recomputed (points and images were mounted again on the base of new trajectories and a new check was performed).

The process is iterative and theoretically can continue until the user is satisfied. The computational time is very huge, so the number of iterations and the quality of the final result is a matter of trade-off between survey requirements, available time and costs. After this final consideration, it is clear why it is very important to carefully plan the paths and trajectories of the missions.

The last step of the workflow is to export the data. The resulting point cloud of the final project can be exported in various formats (.las, .lcs, .e57, .rcp). Alternatively, there is the possibility to create a database in another Leica software: Leica Cyclone⁴, which is a specific software for managing point cloud databases. In this case, each

⁴<https://leica-geosystems.com/en-gb/products/laser-scanners/software/leica-cyclone>

Chapter 3. Data acquisition with a Mobile Mapping System

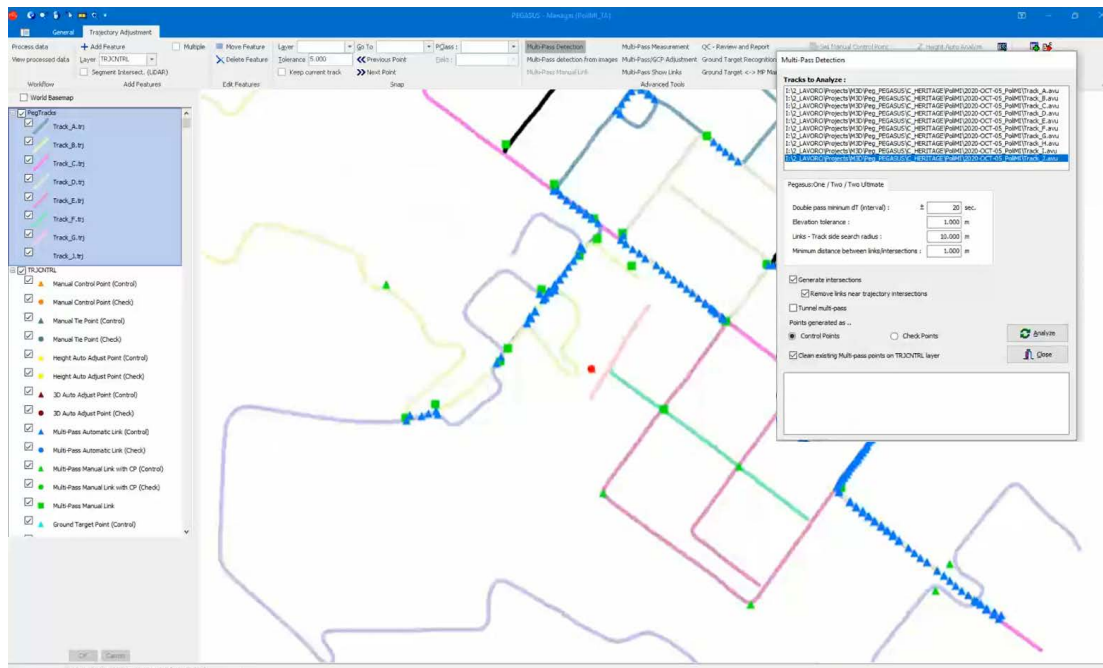


Figure 3.17: A screenshot from software Pegasus Manager. The user is performing an advanced trajectory adjustment. Green and blue squares and triangles are the tie-points.

track will be decomposed and seen by a set of several fictitious range scans, distributed with a regular span. We asked Leica technicians to provide a fictitious scan every 50 metres, so that, in Cyclone a single track is composed of several range scans. Each of them has a single panoramic image created by meshing and joining the 7 images acquired by the cameras.

The final project received by He.Su.Tech Laboratory was composed of a point cloud of the entire project in .e57 format (35 GB), a Cyclone database with all the tracks composed by one Scan World every 50 metres (168 GB), and a shapefile with the trajectories of each track. The initial folder with the raw data stored by Leica Pegasus:Two was 75 GB, the final project, in Cyclone, is almost doubled in storage size.

3.3.4 Data validation

Since the adjustment and processing operation was carried out by Leica experts, for the purposes of this thesis work, it is important to make some considerations on the reliability and accuracy of the final dataset.

This section describes the critical analysis performed on the final data received. The data validation workflow is summarized in Figure 3.18. The evaluation was done firstly by checking the dataset with Pegasus Manager Viewer. Then some considerations are made in Cyclone, comparing the dataset with a set of ground points acquired with a GNSS, and making some slices of the point cloud. Lastly, a navigation of the point cloud will lead to some final considerations and comments. Figure 3.26, at the end

3.3. Exploitation and validation of the geometric survey

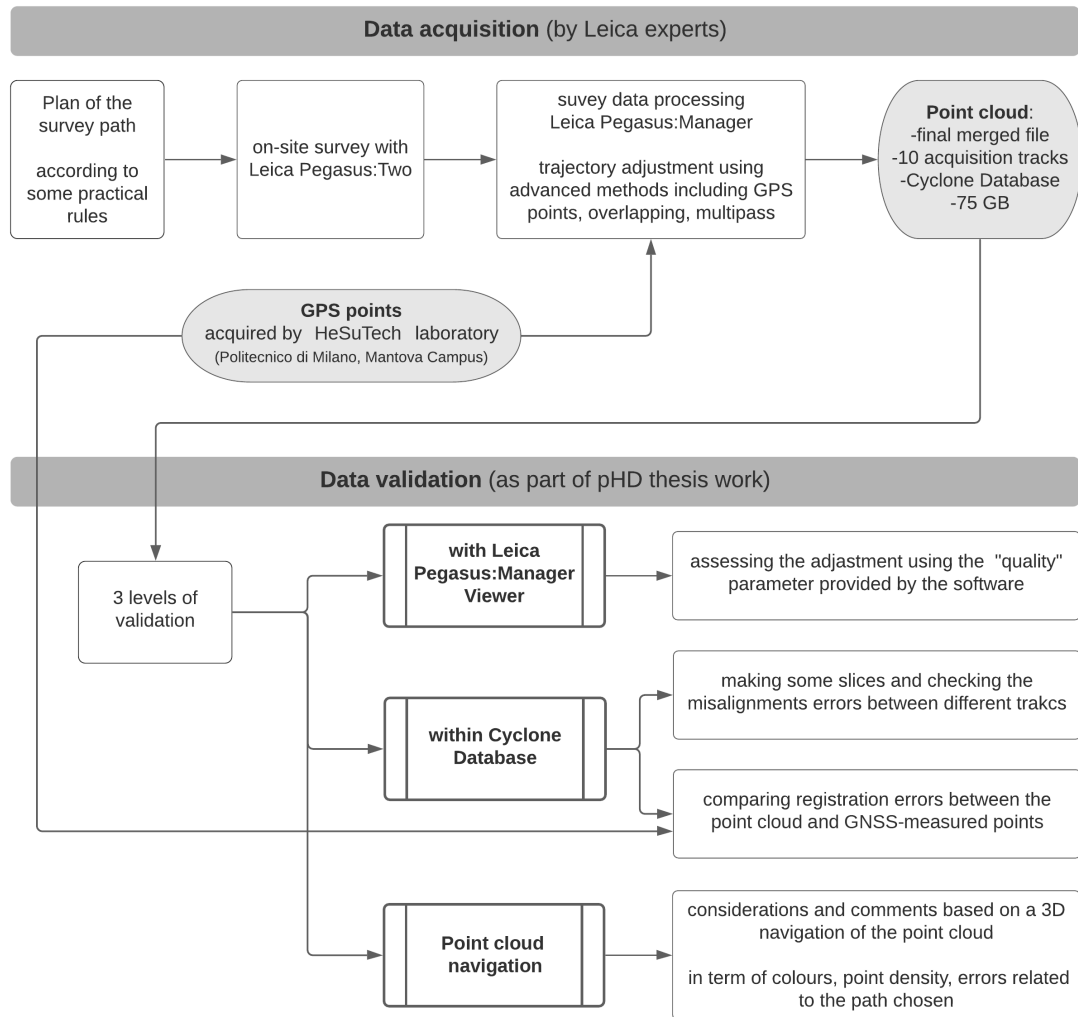


Figure 3.18: The data validation workflow followed in this work, from the acquisition of the dataset to the three levels of validation.

of this chapter, presents some extract of the point cloud both in colour scale and in intensity false colour scale.

Pegasus Manager Viewer is a free tool that allows visualising the Pegasus:Two project, combining several elements. It is possible to see, on-screen, the trajectory, to select one point of it and see the associated point cloud (either in 3D or in an orthographic view from the top and in vertical section) and images (Fig. 3.19). By looking at the point cloud menu, it is possible to visualise the point cloud coloured with a "quality" scale. According to the software manual, the "quality" attribute indicates the quality of the point cloud depending on the trajectory. The software manual does not describe how the quality attribute is computed. For the Sabbioneta point cloud, the quality values went from 0 to 6. Subdividing that range into three steps it was possible to define three quality evaluations and colour them accordingly. Then, red could be seen as "Low", yellow as "Medium" and green could be interpreted as "High" quality. In Figure 3.20 is possible to see the result of this colour map.

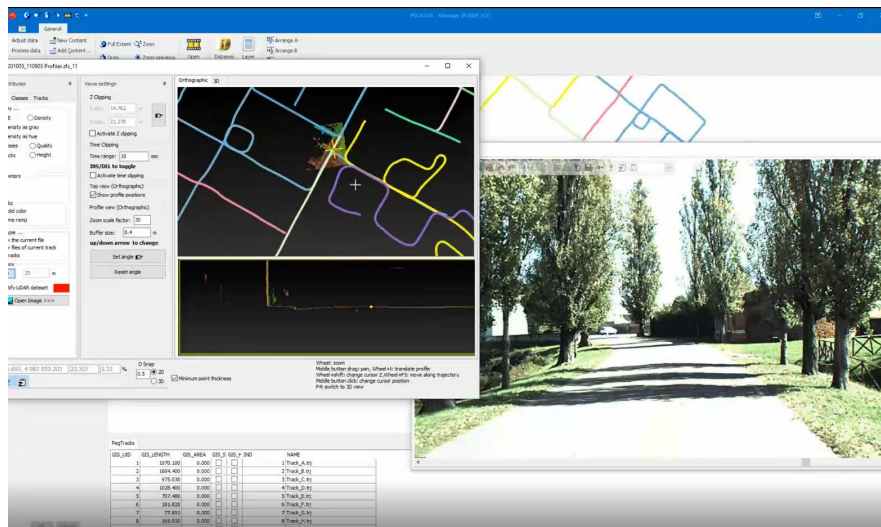


Figure 3.19: A screenshot of the software Pegasus Manager. The user can interact with all the trajectories, and see the associated point clouds and the images. The user can also check several settings parameters and perform all the processing steps.

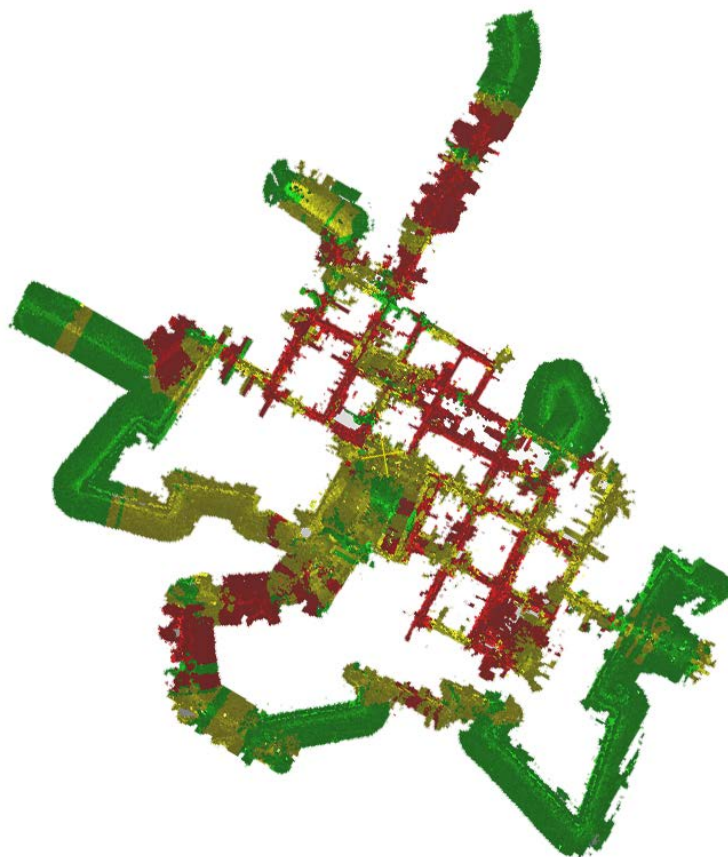


Figure 3.20: The point cloud of the project coloured by Pegasus Manager according to a quality parameter. View from the top. Green is for "high" quality, yellow stands for "medium" quality and red is for "low" quality.

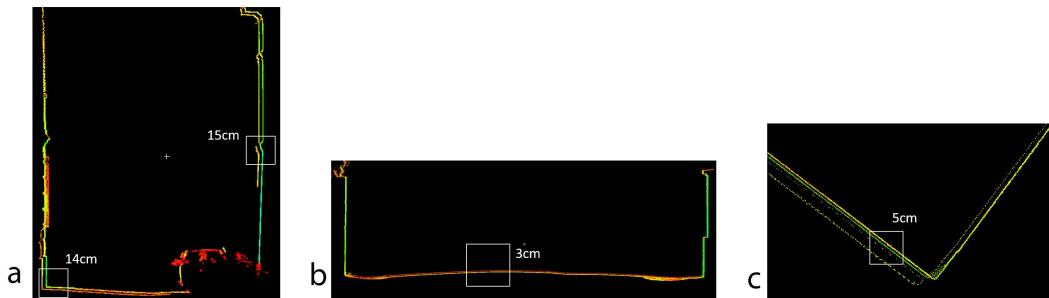


Figure 3.21: Some slices of the point cloud, made in Cyclone, to check the alignment between different tracks. Images a and b represent two vertical sections, while image c is a horizontal section.

Comparing these results with a map of the city, it is possible to notice that the "low" quality is mostly in narrow roads, where surrounding buildings create an urban canyon which is not good for GNSS receivers. The "high" quality areas are mainly along the walls of the city or on roads that have a wide clear area around them. In both the piazzas of the city, the quality is partially "high" and partially "medium". In the next check with Cyclone, more errors in the "low" quality area should be expected.

The Cyclone database is composed of one ScanWorld for each track from A to J, and another ScanWorld with all the tracks together, which is composed simply by merging all the previous ones. As previously discussed, the ScanWorld of each track is composed by creating a fictitious range scan every 50 metres along the track trajectory. The first check made in Cyclone was related to the correct alignment of different trajectories. To do that, some slices on the point cloud were made in areas where two tracks overlap.

The misalignment errors are higher in the areas where the quality was "low", as expected, in fact, an average misalignment error of 0.12 m in "low" quality areas, and a smaller error (0.03 m) in the "high" quality areas were noted (see Fig. 3.21). By making a slice on the Z-axis, it is possible to see that on average the error is 0.05 m. It is also interesting to report that the error shown in Figure 3.21-a, is not between two tracks, but it is on a single track, in fact, the last part of track-D overlaps a portion of its path.

The day after the survey with Pegasus:Two, some architectural points were identified on the road surface (manhole edges, notable points on road markings), taking care that they were visible in the point clouds of the survey. The spatial position of these notable points was then measured with a multiband GNSS receiver (Emlid Reach RS2⁵), used in Real Time Kinematic (RTK) mode connected to the SPIN3⁶ GNSS satellite positioning service of the Lombardia, Piemonte and Valle d'Aosta regions, providing centimetre accuracy. The coordinates of some of those points were used by Leica technicians during the advanced trajectory adjustment process. In order to assess the final quality of the point cloud provided by Leica at the end of the process, the point cloud was compared with the measured points. Using Leica Cyclone software,

⁵<https://docs.emlid.com/reachrs2/>

⁶<https://www.spingnss.it/>

the point cloud was registered based on the architectural points measured on-site with the GNSS receiver. With this procedure, the point cloud is rototranslated rigidly to be correctly oriented on the provided points and thus minimize registration errors, i.e., the distance between the point measured with GNSS and the corresponding point on the point cloud. Table 3.2 shows the GNSS-measured points with their registration errors. Looking at the registration error for each point, it is possible to note that 66% of the points have an error lower than 0.10 m, 24% of points have an error between 0.10 m and 0.16 m, and only 10% of points have an error higher than 0.16 m. Then, a map can be produced (Fig. 3.22), where the GNSS-measured points are coloured depending on the registration error. The same colour scale as Pegasus Manager was used, green for lower errors and red for higher errors. Specifically, red points have a registration error ≥ 0.160 m, yellow points have a registration error between 0.050 and 0.160 m and points in green has a registration error ≤ 0.050 m. By checking the map, the distribution of errors is again consistent with the "low" quality areas already discussed before.

Point	Error	Error vector (x,y,z)	Point	Error	Error vector (x,y,z)
51	0.221 m	(0.059,0.199,0.077)m	2	0.074 m	(-0.015,-0.060,-0.040)m
48	0.210 m	(0.050,0.196,0.055)m	18	0.069 m	(0.025,0.046,0.045)m
26	0.160 m	(0.052,0.029,0.148)m	28	0.066 m	(0.005,-0.065,0.008)m
41	0.155 m	(0.072,0.113,0.079)m	47	0.063 m	(-0.010,-0.061,0.013)m
42	0.153 m	(0.049,0.123,0.076)m	57	0.058 m	(-0.040,-0.037,0.018)m
4	0.149 m	(0.059,-0.088,0.104)m	21	0.058 m	(0.013,0.009,0.056)m
54	0.130 m	(-0.023,-0.128,0.002)m	3	0.057 m	(0.026,-0.050,-0.003)m
44	0.125 m	(-0.062,-0.108,0.010)m	39	0.055 m	(-0.031,-0.029,0.035)m
15	0.111 m	(-0.007,-0.014,-0.110)m	46	0.055 m	(0.018,-0.048,-0.020)m
52	0.108 m	(-0.064,0.042,0.076)m	27	0.048 m	(-0.008,-0.046,0.011)m
19	0.108 m	(0.062,0.056,0.067)m	6	0.043 m	(-0.004,-0.041,-0.012)m
23	0.089 m	(-0.052,-0.072,-0.007)m	53	0.041 m	(-0.032,-0.025,0.006)m
9	0.087 m	(-0.024,-0.027,0.079)m	11	0.037 m	(-0.009,-0.036,-0.002)m
45	0.085 m	(-0.036,-0.039,0.066)m	58	0.033 m	(-0.011,0.000,0.031)m
13	0.084 m	(-0.003,-0.084,0.005)m	5	0.023 m	(0.011,0.002,0.020)m
55	0.074 m	(-0.046,-0.058,0.009)m	43	0.017 m	(-0.004,-0.014,0.009)m

Table 3.2: Registration errors. The error is intended as the distance between the GNSS-measured point and the correspondent point on the point cloud. The values are in descending order; the error vector decomposed along the x,y,z axes is also reported.

The same points were also used to perform evaluations on every single track. For this evaluation one point cloud for each track was used. Registering one track at a time on the basis of GNSS-measured points, the global registration error was lower than before, but by making some slices in the same position as before, the misalignment error between different tracks increased. The final result after Leica Pegasus Manager processing seems to have the best trade-off between tracks alignments and registration errors on GNSS points. It follows that the adjustment made by Pegasus Manager provides the best result.

The last evaluation consists of a navigation of the point cloud, conducted again in Cyclone, with the purpose of observing the point cloud and deriving some considerations from it. The first observation is that the point density is higher on the ground

3.3. Exploitation and validation of the geometric survey

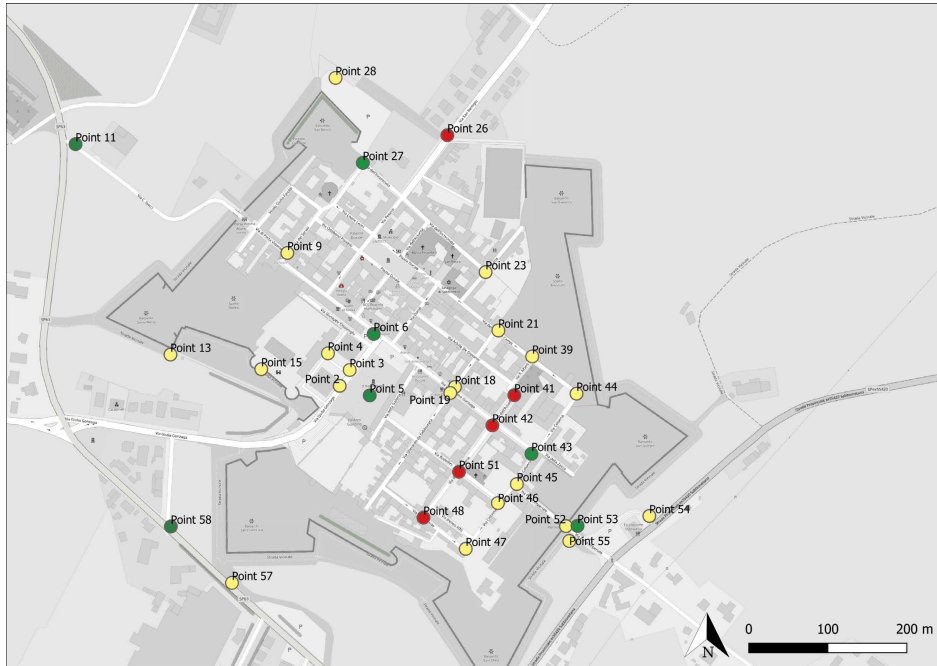


Figure 3.22: Map of the GNSS-measured points used in the quality check by HeSuTech Laboratory. The set of GNSS-measured points was registered together with the final point cloud produced by Leica Pegasus Manager. The points on the map are coloured according to the registration error of the GNSS-measured point with respect to the corresponding point on the point cloud. Red points have a registration error higher than 0.16 m, yellow points have a registration error between 0.05 m and 0.16 m and points in green has a registration error lower than 0.05 m.

with respect to the elements on the higher portion of the buildings' facades. This is related to the scanner position: on top of the car, so it is closer to the ground (Fig. 3.23).

A second observation, looking at the point cloud with colours, is that the colours from the 7 cameras are not good, apart from some reprojection errors (for example a "stop" signal which was projected onto a facade), the colours themselves are very much affected by the camera white balance (Fig. 3.25). The 7 cameras have 7 different exposure, and the meshing process combines all 7 cameras into a single panorama with an averaged exposure. The final exposure is mostly affected by the exposure of every single image. The new version of Pegasus:Two Ultimate mounts also a panoramic camera to avoid this issue.

The last consideration that arises from the navigation of the point cloud, is related to the areas where the path has sharp bends. Because the laser scanner is mounted on top of the car and it is inclined, when the car turns right or left and the radius of curvature is low, a portion of the facade of the building which is in the corner is not surveyed. By looking at the point cloud (see Fig. 3.24), it is possible to see a cone of shadow in the survey, because while the car turns, the laser misses a portion of the building's facade. In the Figure, this behaviour is highlighted both on buildings and on the fortified walls. If the facades are the purpose of the survey, this problem can be solved by acquiring additional range scans with a TLS in a corner position, or planning the acquisition path so that the corner is acquired twice.

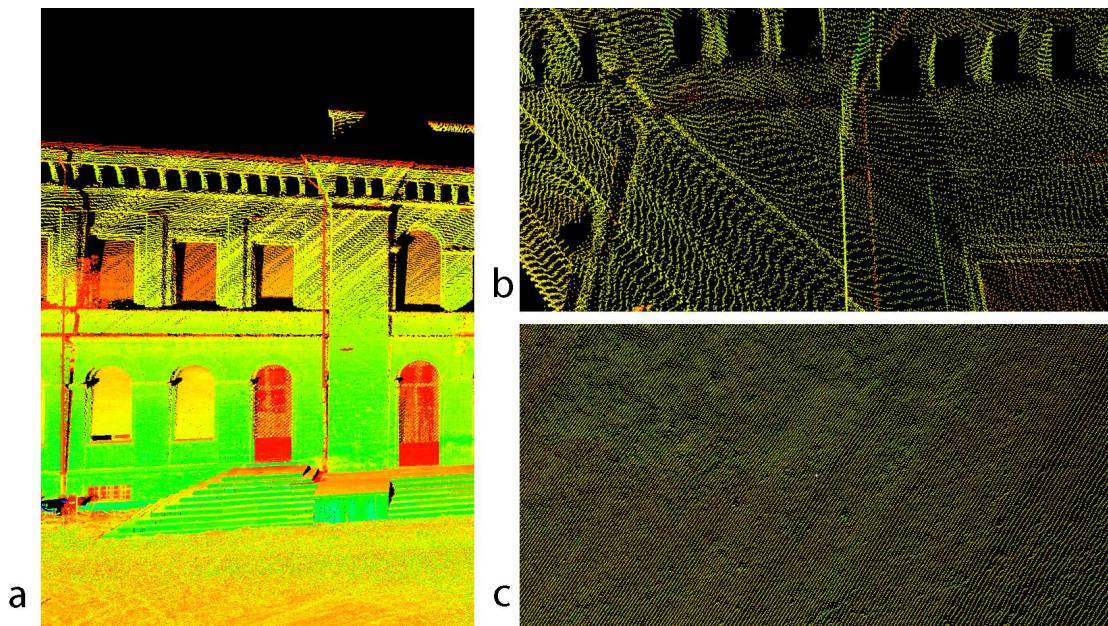


Figure 3.23: Snapshots of the point cloud. The point density is different according to the distance from the laser. a: the point cloud cropped near a building. Images b and c are two enlarged views of the portion of the point cloud in image a. in particular, b shows the top of the building (with lower point density), and c shows the ground (with higher point density).

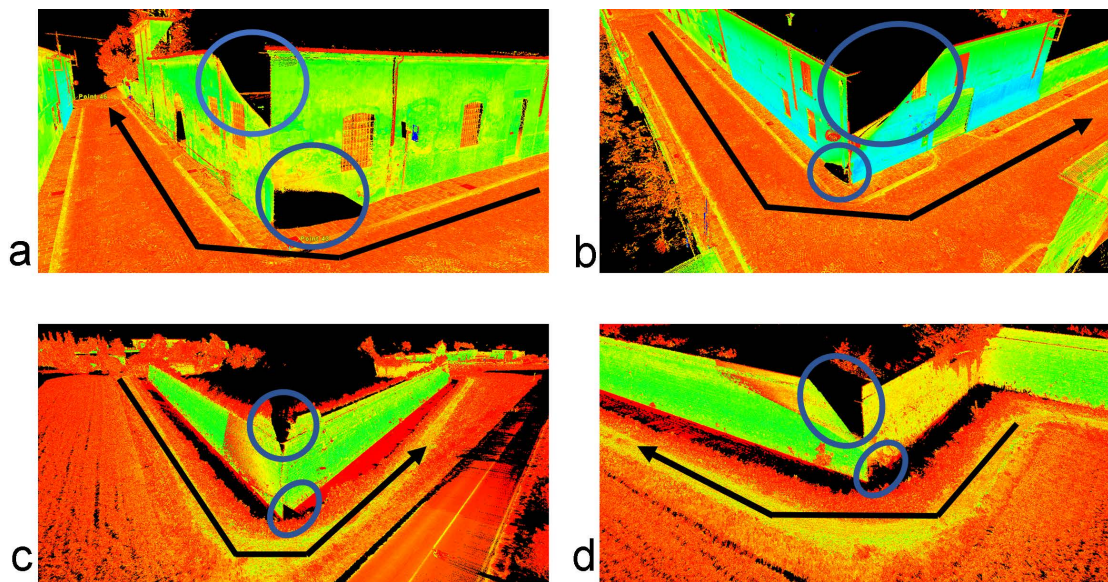


Figure 3.24: Four snapshots of the point cloud showing (in blue circles) the cone of shadow that happens when the acquisition path (black arrow) has sharp bends (i.e. when the car turn left or right). a,b: this phenomenon inside the built environment. c,d: the same situation can happen also outside the city while passing near the fortified walls.

3.3. Exploitation and validation of the geometric survey

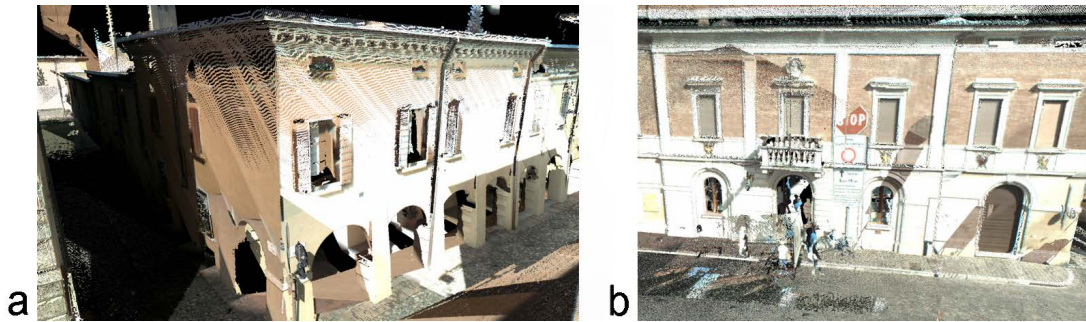


Figure 3.25: Two snapshots of the point cloud coloured with the images. a: we can note the exposure problem. b: we can note a projection error, in fact, a "stop" signal was projected onto a facade.

Concluding this evaluation and validation section, the analysis shows that the MMS allows fast acquisition of a wide survey area with high point density on the ground. On the other hand, there are real possibilities of shifts and drifts of the acquired data, with a general accuracy of 0.15-0.20 m, and there is a low point density on buildings' facades; the colours are sensible to under/over exposure, specifically in areas where there is one side exposed to sun and completely in shadow on the other side. The errors encountered in the coloured point cloud make the point colour not a suitable feature for urban elements detection; however, the Intensity value is not influenced by under and over exposure and is a suitable feature.

For the scope of this thesis, this dataset is suitable to perform a point cloud processing for the extraction and characterisation of accessibility features. The main interesting element is the ground, which has a high density of points which is enough for the thesis purposes. The processing will focus on one single track at a time, in this way also trajectory information can be used. To do so, every single track was exported from the Cyclone database. The number of points of each track was provided in this chapter at Table 3.1.

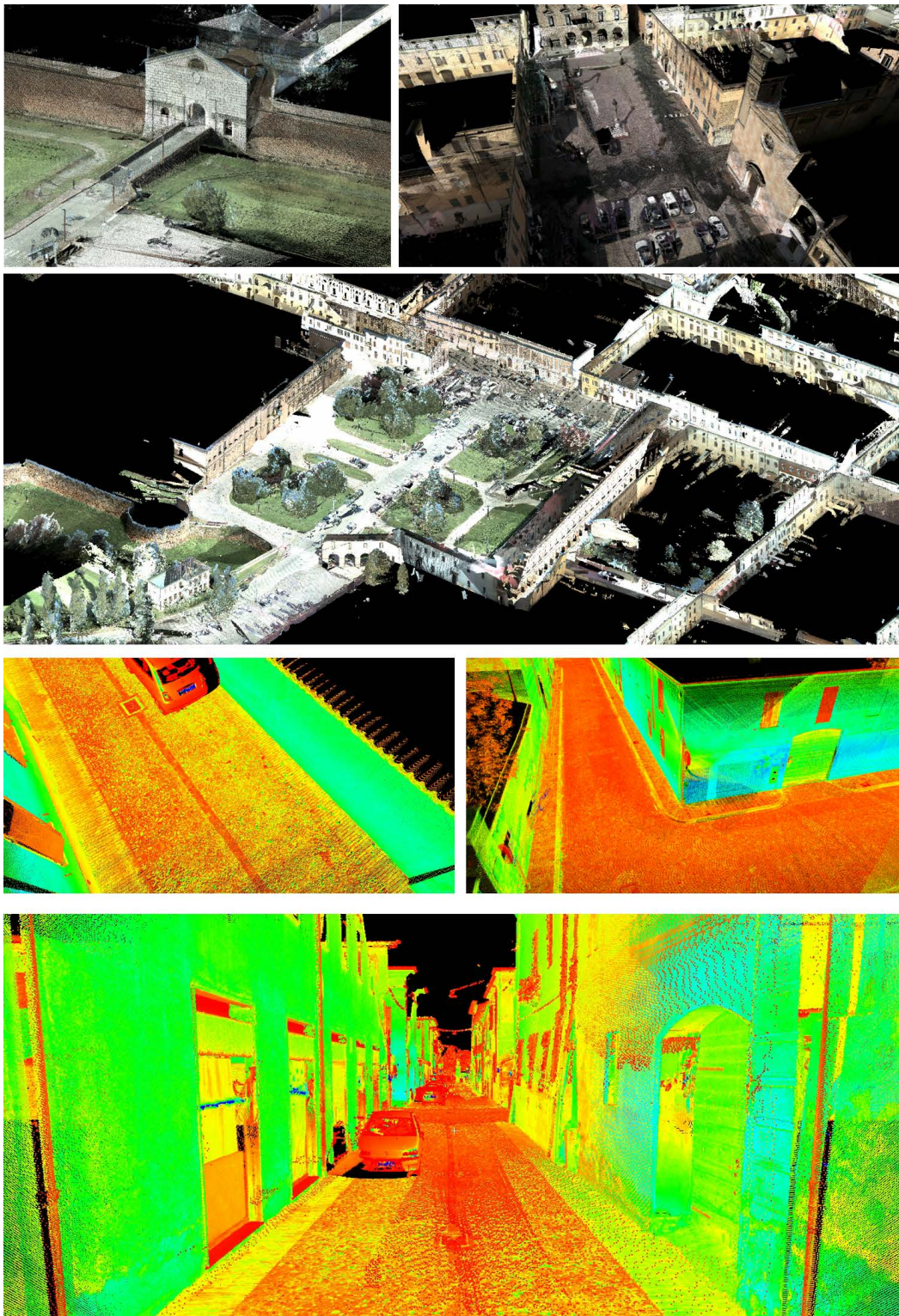


Figure 3.26: Some snapshots of the point cloud of the complete survey. The three top images show the point cloud coloured with the acquired images, the three on the bottom show it coloured by intensity values on a false-colour scale.

4

Knowledge-based approach for segmentation

The purpose of this chapter is to develop a method, based on the integration of various algorithms specifically made for the purpose, to analyze the point cloud and segment it, thus identifying the sidewalks and streets of the historic city. The method showed efficient results on the dataset from the city of Sabbioneta and similar results on datasets from other Italian historical cities, but low performances on datasets from more modern urban environments. The strength and innovation of this chapter are in the creation of a method that succeeds in taking into account the specific characteristics of a historic environment and exploiting them for the segmentation of the dataset, and has been shown to be applicable to datasets from other historic urban environments.

The chapter is organized as follows: the first section introduces the method; the following three sections describe the method, from the pre-processing, to the segmentation and to the semantic labelling. The last sections show the result and a critical discussion.

4.1 Introduction to the method

As discussed in chapter 2 (paragraphs 2.3 and 2.4), when it comes to performing a semantic segmentation of an urban point cloud to identify its characterising elements (e.g., road surface and sidewalks) the proposed methods in the literature relied on the detection of road edges and boundaries. To this end, a recurring hypothesis was a difference in height between the road surface and everything around it, and/or the presence of curbs. Many papers implemented and proposed methods for curbs detection. However, it must be said that there were, albeit few, methods that are based on the values of certain geometric characteristics for road surface recognition. As an example, following this second approach, authors of [93, 137] extracted road surface in rural roads, where the curb detection could not be performed because there was no curb in rural roads. They based their method on geometric parameters coming from PCA, point density, intensity and height differences. A similar approach was presented by [65]. They propose a classification of urban pavements (stone and asphalt) relying on roughness descriptors, applying clustering on scan profiles (i.e. scanlines) from the laser scanner mounted on the MLS system.

It is also worth mentioning that Pegasus:Manager, the software used to adjust and process the raw data from Leica Pegasus:Two, has a module that allows to automatically extract sidewalks. The approach of the module is the following: the user has

to provide to the software the width of the road, then the software will extract from the point cloud the area where there should be a curb, according to the width of the road and standard width of elements. Then basing on differences in elevation values, Pegasus:Manager detects the presence of curbs and everything which is further away than the curb is a sidewalk, everything else is the road; obviously, this approach works fine if there are curbs and if the road has fixed widths.

Typically in most cities, the sidewalks have curbs, but this does not happen in historical centres. For this reason, commercial solutions may fail in historical districts, where the curbs are not always present; urban ground elements could have the same height and the width of the road can change a lot along the survey path. In fact, in historical environments the urban configuration is not so regular and, very often, sidewalks and roads are at the same level. Furthermore, considering Sabbioneta as a paradigm of what a historical centre could be, its elements of diversity are manifold. The width of the road is not uniform, even in some cases the same road suddenly doubles in size. The streets of the city are intertwined with each other in a manner reminiscent of the Roman chessboard, so 90-degree bends and crossroads are very common. Squares, parks and buildings are interconnected in such a way that within a few metres one can go from a configuration similar to an urban canyon (bordered by buildings) to a large open space, with a park, garden areas or squares that can be either completely pedestrianised or used as a car parking. Moreover, sometimes uncommon solutions and peculiar spatial organization were implemented to cope with physical constraints due to the surrounding historical environment. In such a case, sidewalks detection becomes a challenging task and different strategies should be defined to solve the problem.

In this chapter, the segmentation exploits the fact that different paving materials were used for different urban elements; this characteristic of Sabbioneta, and historical sites in general, was previously described (par. 3.1.2). Generally, the method used as a fundamental hypothesis the fact that roads and sidewalks are paved with different flooring materials and used as a driver for segmentation the difference in values of geometric features of points on different paving materials.

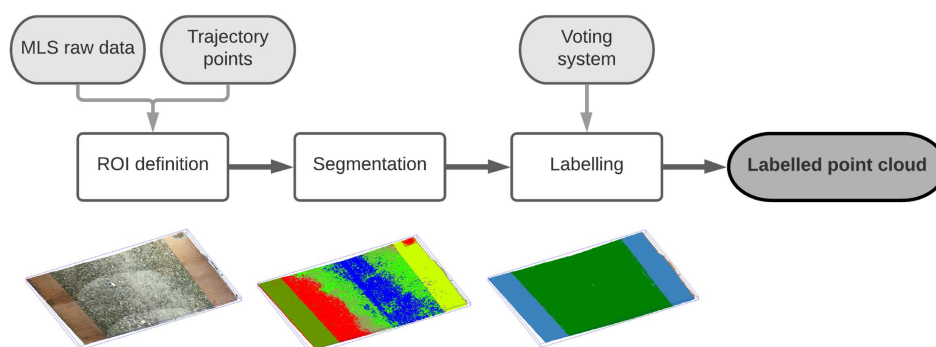


Figure 4.1: A brief scheme of the workflow presented in this chapter. From The survey raw data to the semantically segmented point cloud. In the lower part of the Figure, in order from left to right, a ROI in RGB colours from the survey, then moving to the right, the segmented ROI (subdivided into several clusters, each cluster represented by a different colour) and, in the end, the same ROI after the semantic classification process (in blue the points identified as 'sidewalk' and in green those identified as 'road').

The general aim of the method proposed in this chapter was to analyse a point cloud and segment the ground points in 'roads', 'sidewalks' and 'others'. An overview of the method is provided in Figure 4.1, while Figure 4.2 shows a more complete workflow. This method was presented, in its preliminary version, at an international conference [138]; in this chapter, it is discussed in a completed and improved manner. The segmentation approach was implemented through a well-known clustering algorithm (k-mean), that relies on some geometric features to subdivide datasets into macro-clusters. Then, candidate sidewalks were selected and the clusters were refined. Finally, each candidate sidewalk was confirmed or rejected using a voting system specifically developed that was based on topological relations, contextual and geometric information, like position respect existing buildings, trajectory line, and average Z values.

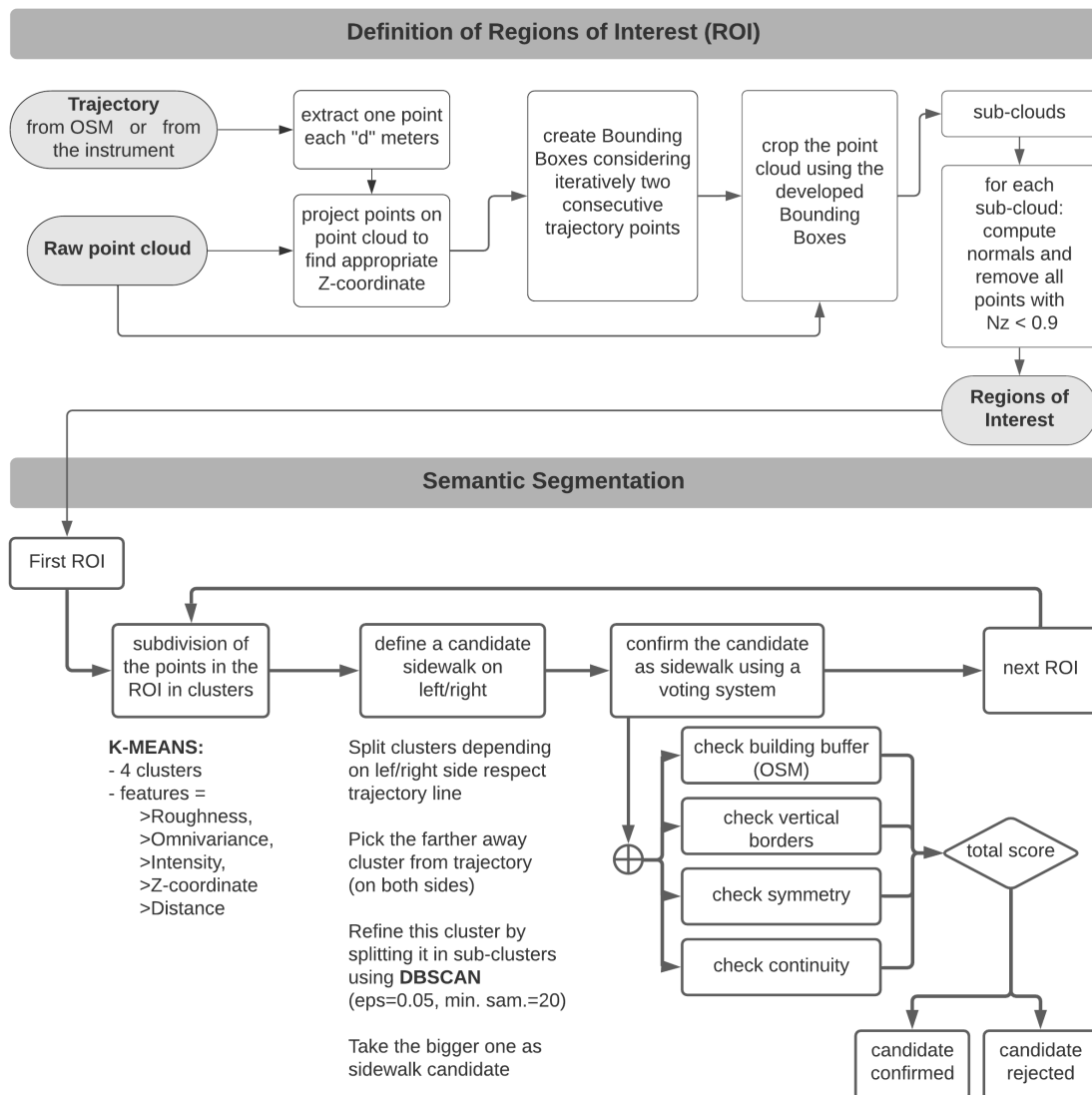


Figure 4.2: A complete scheme of the workflow presented in this chapter. From The survey raw data to the semantically segmented point cloud.

Considering that an urban geometric survey, deployed with an MLS, might potentially be composed of a huge amount of points (in the case of Sabbioneta, it was composed of 10 point clouds corresponding to the acquisition tracks, and the smallest was composed of 23 million points and the biggest was of 264 million points), an important preliminary step was the subdivision of the point cloud into sub-clouds. This was beneficial for the computational effort and processing time, but also for an easier definition and implementation of the processing algorithms, that worked locally on the point cloud by analysing one sub-cloud at a time instead of on all points together. For each sub-cloud, the Region of Interest (ROI) was defined by selecting only points on the ground surface. The segmentation approach was then applied to every ROI.

The method proposed in this chapter was defined as a "knowledge-based" method because it employed numerous techniques, procedures, and parameters that were optimized to achieve its goal, which was the segmentation of the point cloud into sidewalks and roads. Within the procedure, in one specific step, an algorithm (k-means) from the family of unsupervised ML methods was used to divide ROIs into clusters. This algorithm deals with the subdivision of a dataset into clusters, without having knowledge of the semantics of the clusters it is creating. For this reason, it was possible to define the main workflow of this chapter as a knowledge-based method, while the method presented in chapter 5 was designed as a Machine Learning method because it reached the same purpose basing on supervised ML techniques.

4.2 Pre-processing

4.2.1 Geometric features computation and selection

The method presented in this chapter included a subdivision of the point clouds in clusters which was guided by various geometric features that were computed by leveraging the neighbourhood distribution for each point. As already described (par. 2.1.2), geometric features of a point P represent the geometric distribution of points in the neighbourhood of the point P itself.

The geometric features were computed on all tracks (one at a time) without cropping them, and using the open-source software CloudCompare (CC), which allows a fast and reliable result even with big point clouds. The tool¹ provided in the software allows the computation of many geometric feature, here listed: Roughness, Mean Curvature, Gaussian Curvature, normal change rate, number of neighbours, surface density, volume density, 1st order moment, sum of eigenvalues, Omnivariance, Eigenentropy, Anisotropy, Planarity, Linearity, PCA 1, PCA 2, PCA 3, surface variation, Sphericity, Verticality, 1st eigenvalue, 2nd eigenvalue, 3rd eigenvalue.

Within this tool, many of the features were computed according to equations provided in [139]; the Roughness value of a point was computed as the distance between this point and the best fitting plane computed on its nearest neighbours; curvature values were computed from a local quadratic function provided by [140]. The only parameter that had to be set was the "local neighbourhood radius", which is the radius of the spherical selection for the points to be used by the feature computation tool. CC computed the selected features and created a scalar field for each.

¹https://www.cloudcompare.org/doc/wiki/index.php/Compute_geometric_features

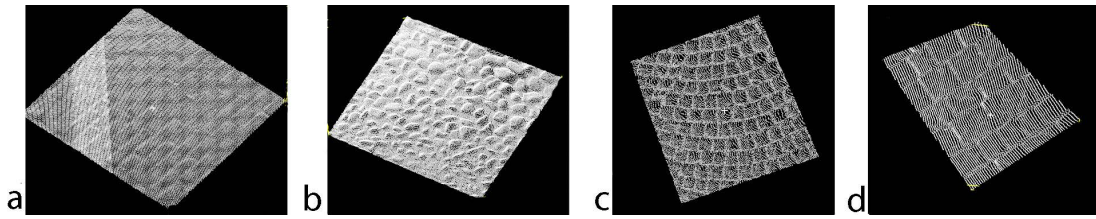


Figure 4.3: Sample of point clouds of some paving materials within the city of Sabbioneta: a. bricks, b. cobblestone, c. sampietrini, d. stone.

Additionally, with CC, also point normals were computed and stored in the point cloud file. To compute normals, the software estimates the local surface represented by a point and its neighbours through a best-fitting plane, a triangulation or a height function; the level of noise and the number and distance of neighbours will change how this surface looks like².

The choice of the most appropriate geometric features to compute was driven by the hypotheses at the base of the segmentation method: different elements of the urban ground surface are made of different materials. In this way, clustering worked better if the selected features showed very distinct values on different paving materials. So, the idea behind this method was to take advantage of the different characteristics of the paving materials used for the different urban elements.

As shown by Figure 4.3, which is an extraction of the different pavings from the point cloud, the points of each material had a specific spatial arrangement which may result in different values of some geometric features.

As an example, Figure 4.4 shows the same crop from the original point cloud, but with three different features used as a scalar field to colour the points in a false colour scale. The crop is a portion of the road on either side of which are two sidewalks, which can be recognised in the top image as the two greenest bands on the left and right. In the Figure, on the right of each crop, there is the distribution of values of that specific feature. The three features shown are Intensity (on top), Omnivariance (in the middle), and surface density (on bottom). The Intensity attribute and Omnivariance feature are different depending on the material, in such a way a stone pavement will have values that are in a very different range from the range of values of a cobblestone pavement. Other features, such as the surface density in the example, have a homogeneous distribution of values along the two materials and are ineffective for the segmentation. Geometric features value is also influenced by the number of neighbouring points selected and, hence, by the radius used for neighbours selection (in Figure 4.4, all features were computed with a 0.05 metres radius).

For the selection of the most appropriate geometric features, an empirical and evidence based method was implemented. Following this principle, a portion of the urban survey was selected, and its geometric features were computed (all those available in CloudCompare) with different radii (0.02, 0.03, 0.05, 0.10 m), in consideration that the point cloud average density on ground is one point each 2 cm. Then, starting from the fact that the higher is the difference between feature values on different urban elements, the more appropriate this feature is, observing the statistical distribution

²<https://www.cloudcompare.org/doc/wiki/index.php?title=Normals%5CCompute>

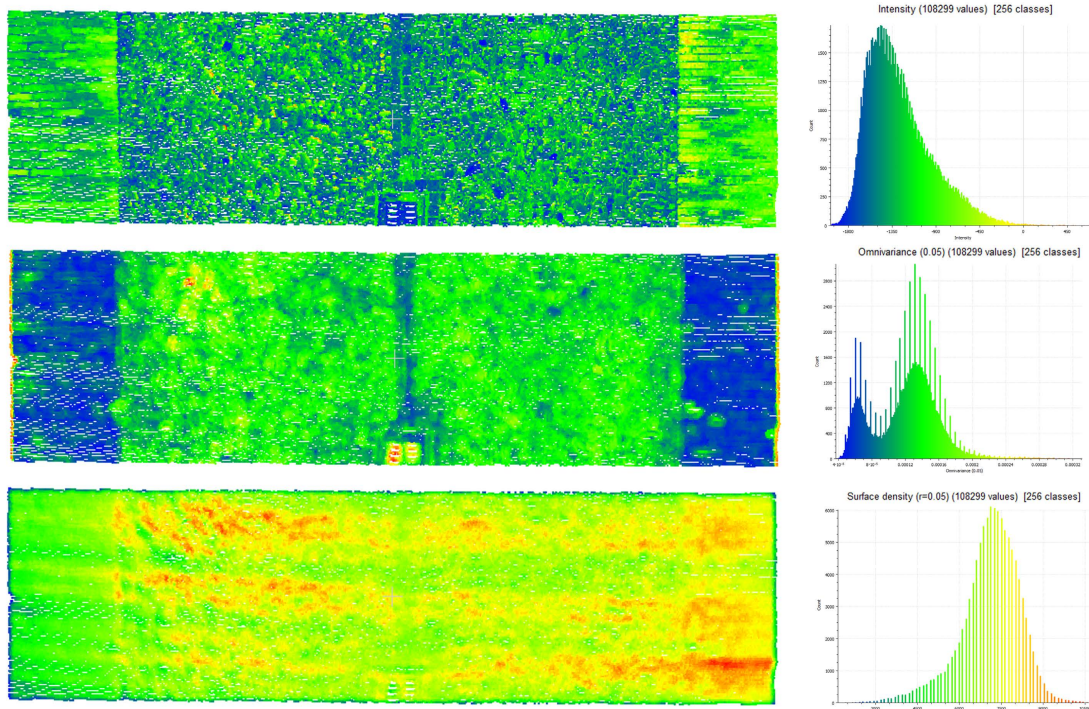


Figure 4.4: A crop of the point cloud, along the road trajectory, viewed from the top with three different features used to colour the points. Top: Intensity value, Middle: Omnivariance, Bottom: Surface density. From the Intensity, it is visible the roadway and the two sidewalks on the left and right. On the right, the distribution of values is provided. From a visual inspection, it is easy to see that some features are more suitable than others to be used for the segmentation. To segment sidewalks and roads, the Omnivariance and Intensity are more helpful than the surface density.

of the feature values (scalar field histogram), some of them were considered more effective and were selected. Plus, focusing on the meaning of each geometric feature, the most helpful could be the ones related to the distribution of points in space, like Omnivariance, Sphericity, and Roughness. All those three geometric features are related to the 3D arrangements of points and try to describe how much the considered neighbourhood is closer or farther to a hypothetical planar surface. Those geometric features can be reliable when materials have differences in point distribution in space, this means that, in theory, the approach can work even with other materials, as long as a difference in spatial distribution exist.

The geometric features were computed relying on a spherical neighbourhood computed in a radius of $r = 0.05$ metres. The choice of this radius was driven by the average density of points.

4.2.2 Corner identification

For the correct exploitation of the subsequent steps, an accurate definition of all road crossings and corners along the survey trajectory path is of primary importance. Both the definition of the ROI and the segmentation were defined slightly different depending on the nature of the ROI: in a straight portion of the road, or a crossing, or

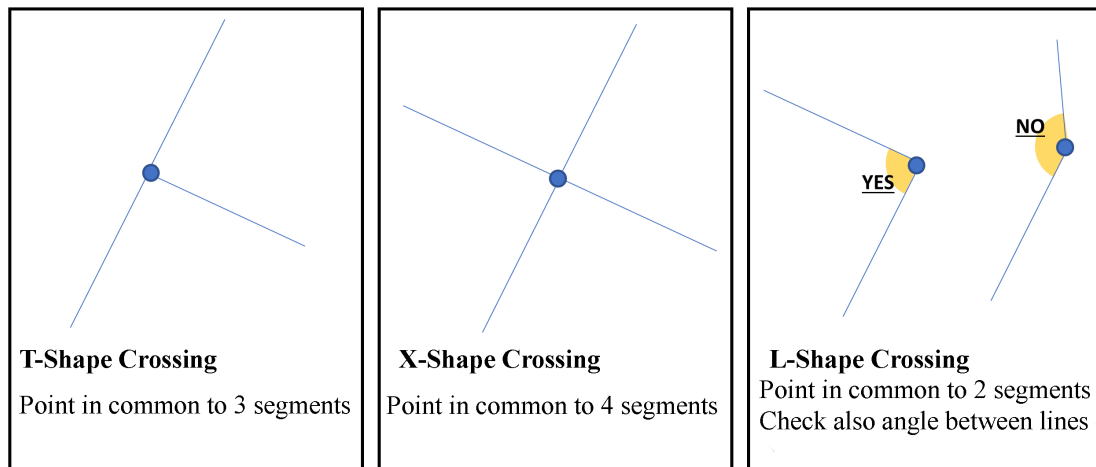


Figure 4.5: A scheme of the three types of crossings analysed in the method: T-shape, X-shape, L-shape. Under each crossing type there is an explanation about how they were recognized.

a corner. For the work presented here, it was necessary to identify the spatial position of crossings and their typology: L-crossing, when it was simply a curve; T-crossing, when one road entered another road; X-crossing, when two roads crossed.

To perform crossing identification, OpenStreetMap (OSM) was used. OSM is a free, editable map of the whole world that is being built by volunteers largely from scratch and released with an open-content license³. OSM data sets can be downloaded for free directly from the main website or using GIS software. For the open-source software QGIS⁴ there is a specific plug-in called "QuickOSM" that allows downloading OSM Tags. Specific features of map elements are represented by Tags which consist of two items, a key and a value. Examples of Tags as "key:value" are "building:apartments", "building:industrial", "highway:residential", "landuse:commercial". Each "key" may have several possible "values". The mentioned plug-in allows to download specific features as Key:value, but also allows to download all the values corresponding to a certain key. For example, we can download all the buildings, disregarding the value (apartments, industrial, etc.) or all the highways.

For crossing identification, all the highways, disregarding the value, were downloaded. In OSM, roads are represented by polylines that correspond to the centre of the street and are under the key "highway". By analysing the highways dataset in detail, and focusing on the crossings, it was possible to note that as different roads come together and create a crossroads in a city, in a similar way different polylines met in a single point on the dataset. The easiest way to identify crossroads spatial coordinates was to extract the point where polylines met. Then, to identify also the type of crossing, simple reasoning could be implemented and it is easily shown by Figure 4.5. By looking at the multiplicity of a point it was possible to identify the type of crossroads: if the point was common to 4 segments, it was an X-crossing, if it was in common to 3 lines it was a T-crossing, and if the multiplicity was 2, it might be a corner, but in this case, a further check was applied.

³https://wiki.openstreetmap.org/wiki/About_OpenStreetMap

⁴<https://www.qgis.org/it/site/>

A point in common to 2 lines could be every point of a polyline, made by several segments. To define if it was a real L-crossing the angle between the associated segments was checked. It was also possible to extract, together with the point coordinate, also the geometric data of the associated segments.

4.2.3 Region Of Interest selection

To compute the whole point cloud it was necessary to subdivide it into smaller sub-clouds and then compute them one at a time. To define an appropriate sub-cloud it was important to define which were the elements of interest of the survey. In this project the relevant elements were the ground pavings, so the focus was on the portion of the survey closer to the ground surfaces, along the trajectory followed by the instrument. First, the point cloud of each acquisition track was subdivided into sub-clouds, then in each sub-cloud, the ROI was extracted and then the workflow processed one ROI at a time.

To subdivide the entire survey into sub-clouds several approaches could be found in literature, these cases mainly relied on the development of vectors or polygons along the trajectory provided by the data acquisition instrument [34, 141]. In this research work, each sub-cloud was instead defined by an oriented Bounding Box (BB), the orientation was based on two consecutive points extracted from the trajectory, each BB was then used to crop the original point cloud.

A Bounding Box is represented by the *Open3D* python library as a rectangular cuboid, and can therefore be identified by specifying 8 points that correspond to the 8 vertices. A BB can be defined either by specifying the 8 vertices or it can be computed automatically, simply by providing a set of 3D points and computing the BB which encloses them. The resulting BB is an oriented rectangular cuboid that encloses the provided points and has the smaller possible volume. In this way it is possible to define a BB using less than 8 points, simply computing the oriented BB that encloses at least two points. In this research, the second method was implemented.

The trajectory of the instrument mainly corresponded to the centre of the road on which the car with the mounted instrument was driven. To make the method as versatile as possible, the possibility of having a dataset acquired with systems other than MLS was also considered (e.g. Terrestrial Laser Scanner, UAV photogrammetric acquisition). In such cases, the trajectory could not be provided. Plus, it might happen that the trajectory of the instrument did not correspond to the road or had a shape that was not useful for ROI development (e.g., a backpack MMS where the user walked on both sidewalks in two directions or followed a path which was not straight and does not correspondent to the road). For these reasons two approaches were considered to extract trajectory data: using directly the data from the acquisition instrument, or creating a fictitious trajectory using Open Street Map.

Trajectory from Open Street Map data

OSM data sets can be downloaded for free directly from the main website or using GIS software (par. 4.2.2).

In OSM, roads are represented by a polyline that corresponds to the centre of the street. To create a fictitious trajectory this road centerline can be a good solution. All the roads are under the Key "highway", and can be downloaded directly using

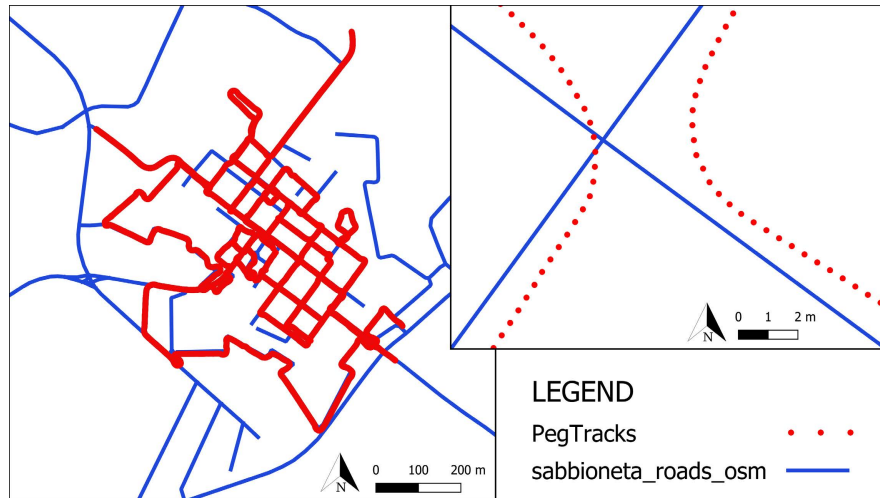


Figure 4.6: Comparison between a portion of the trajectory extracted from the instrument, composed by points (red), and the fictitious trajectory, computed from Open Street Map road centerline polylines (blue).

key="highway" value=" ". The plug-in downloads all the roads and stores them in a shapefile. To be consistent with the reference system, all the shapefiles should have the same cartographic projection used for the point clouds coming from Pegasus:Two. The Coordinate Reference System (CRS) used was WGS84/UTM zone 32N (EPSG:32632).

The road polylines were then converted into points and one point each "d" metres was extracted along each segment of the polyline. The extracted points were then used to define the Bounding Boxes.

It is worth mentioning that another way of creating a fictitious trajectory, exploiting GIS capabilities is to manually draw a polyline, with the correct coordinates in a shapefile. Then the polyline can be exported and treated as a trajectory.

Trajectory from the instrument

As already described in section 3.2, MMS collect data about trajectory; the trajectory is composed of a series of points collected along the survey tracks. Those points represent the position of the instrument (i.e., the trajectory), and are recorded each fixed amount of time t , the smaller is the time t , the closer are the points. During the processing with Pegasus:Manager, the trajectory points were adjusted and corrected. The trajectory points provided by the final production with Leica Pegasus:Manager were stored in a shapefile. Figure 4.6 shows a comparison between the road centerline from OSM and the trajectory points of the instrument.

In this case, the extraction of one point for each "d" metres was straightforward, and the extracted points were used to define the dimension and orientation of the Bounding Boxes.

Correction of the Z value of trajectory points

Either for the instrument trajectory or the OSM fictitious trajectory, the Z elevation of points did not correspond to the ground. For the instrument trajectory, the Z coordinate corresponds to the instrument position (on top of the car).

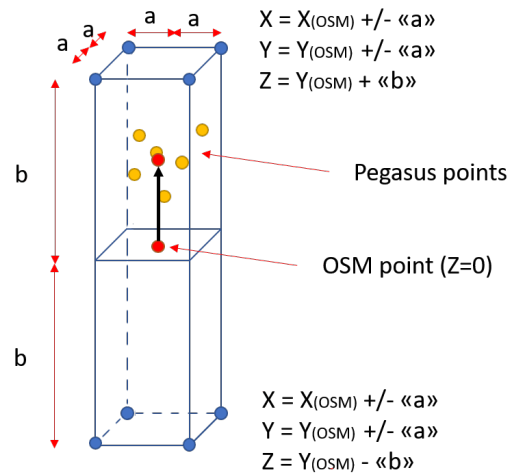


Figure 4.7: Exemplification of the method to project a trajectory point on the ground, valid either for trajectory extracted from the instrument or created from OSM.

For OSM, it did not have a Z value but it was only defined on the XY plane. For the BB creation, these points had to be projected on the ground surface of the road.

To proceed, the entire point cloud and the sequence of points derived from the trajectory were considered. To project a trajectory point A, a BB centred on A was created (see Fig. 4.7), the Bounding box had a shape on XY plane which is a square of side $2a$, whose centre was the point A; the height of the BB had to be enough to include both the point cloud and the trajectory point. Then, the average of Z_i values of the points inside the BB was computed and the resulting value was assigned as the Z_A value of the trajectory point A.

After this step, the trajectory was a list of points that were spanned one to the other with a certain distance "d" and which were projected onto the ground surface of the street.

Bounding Boxes creation

Before proceeding with the BB creations, a discussion on the value of "d" should take place. This consideration arises from the fact that the BBs are rectangular cuboids whose shape cannot be deformed. According to this, when the road is almost straight the BBs can collect all the interesting road points; some problems can arise when the trajectory has curves. In a historical city, straight portions of roads are typically not so long, and sharp bends are very common.

To solve this problem the solution is quite simple: the distance between two consecutive points of the trajectory ("d") should be reduced so that the BB is thinner on the XY plane. This is not enough, some overlapping between BBs should be applied in curves, so that they collect all the road points. There are several possible ways to tackle this problem, a first method is the one presented, which can be improved (e.g., minimizing the overlapping) or changed (e.g., defining polygons instead of BBs).

The area of interest was located where the trajectory curves. Those areas could be searched near the already identified crossroads (L, X, or T shaped) and confirmation was given by checking the angle in the XY plane formed between two consecutive

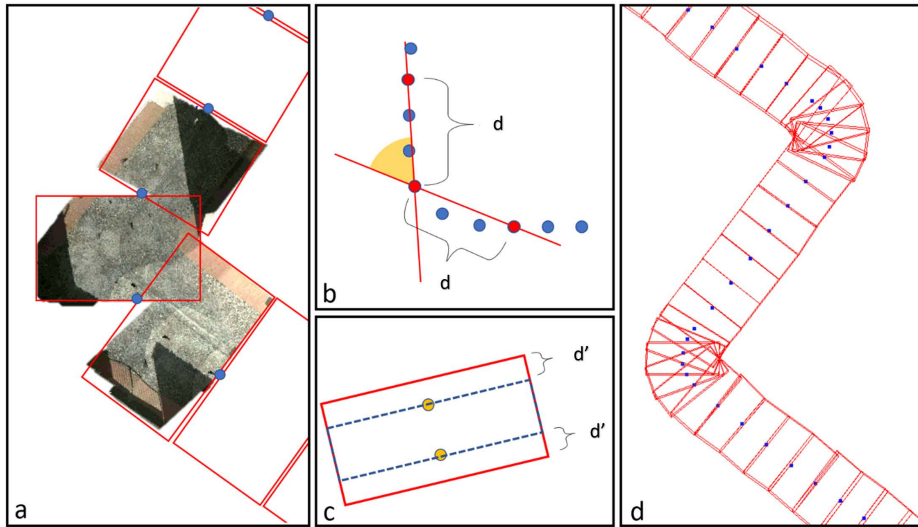


Figure 4.8: Method to detect a curve on the trajectory and adapt ROI shape. *a*: some ROIs in a curve before increasing the points selected. *b*: example of identification of angle between two ROIs using the trajectory points. *c*: increased width of ROI to add overlap. *d*: final result, several overlapping ROIs on curves.

points in the trajectory. Figure 4.8b shows the idea, two consecutive BBs could be simplified by two lines connecting the trajectory points AB and BC. If the acute angle between the two lines was lower than a threshold, the two BBs were on a straight road, if it was bigger, there was a curve. After empirical tests, the threshold was set as 15° .

As described in Figure 4.8, when trajectory portions involved in a curve were identified, the distance between two consecutive points of the trajectory was reduced (i.e., the value of parameter "d" was reduced) and then the width of the BB was enlarged on both sides by a value d' , to ensure overlapping and collect all the road points. After some empirical tests, the choice of $d=2$ metres was set for straight portions, while $d=1$ was set for curves with a dilation $d'=0.5$ metres on each side.

Now that the trajectory points were defined, they could be used as anchoring points for the development of BBs. An oriented Bounding Box is defined by eight vertexes and, as said, it must be a rectangular cuboid. Starting from two consecutive points from the filtered trajectory, it was possible to create a rectangle (CDEF in Figure 4.9) which was cut in half by the trajectory points (A, B). Then from the four vertices of the rectangle 8 points were generated by moving the vertices upward and downward by 1 meter. The solid generated by connecting those 8 vertices was a generic parallelepiped, because the angles between faces were not a right-angle. The BB must be a rectangular cuboid, so it could not be generated by those vertices; instead, it was created using a feature provided by the open3D python library, which computes the oriented BB that *enclose* all those 8 points. The final shape was very similar to the one desired. For example, in the case of the figure, it is possible to note that only points E, C bottom and D, F top exactly correspond to the BB vertexes, while E, C top and D, F bottom are enclosed inside the BB. The BB and the solid identified by the 8 vertices are very similar, so the points of the point cloud within the BB were the desired ones.

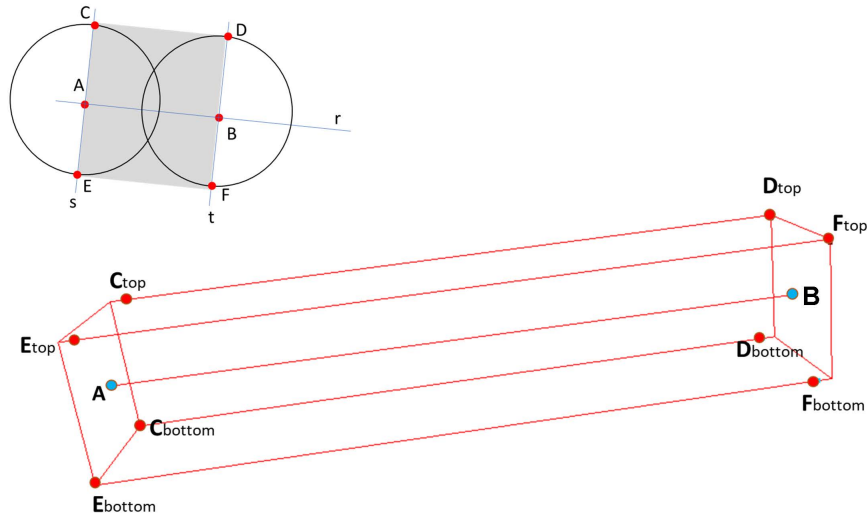


Figure 4.9: Exemplification of the generation of a Bounding Box using trajectory points. From two trajectory points A and B, the BB was generated as a rectangular cuboid enclosing the desired points.

Point cloud crops

When the process to create a BB was defined, the BB was then used to crop the point cloud. By intersecting the BB and the point cloud, all the points that were inside the BB were exported and the sub-cloud was created. This approach was iterative, starting from the first point of the trajectory and moving, two by two, to the last point. In this way, the point cloud was subdivided into smaller parts that were easier to be computed and analysed.

A final refinement on the sub-cloud involved reasoning based on the definition of ROI; the ROI of the work is the ground surface, so it was necessary to remove all points pertaining to vertical structures from each sub-cloud. In other words, since the research interest was only on flat and mostly horizontal surfaces, all the almost vertical surfaces were not interesting. To extract the ROI from the sub-cloud it was possible to exploit the normal values of the points. A perfectly vertical surface has the N_z component of the normal equal to 0. Respectively, a horizontal surface has the N_z component equal to 1. To remove points of vertical surfaces, in the ROI only the points whose N_z normal component was lower than 0.9 were considered.

From now on, in this text, sub-clouds are considered ROIs for all intents and purposes and therefore are called ROI-i, where i is an identifying number that is part of a continuous numbering that follows the trajectory of each acquisition track.

Method validation

To check the suitability of the ROIs creation in both the presented methods, the approaches were tested on two point clouds, one with trajectory data from the instrument and one using a fictitious trajectory from OSM.

The first test was made on a PMMS dataset, a path acquired by Leica Pegasus Backpack on a mountain road in Domodossola (Aosta, Italy), kindly provided by prof. Francesco Fassi, 3D survey group, Politecnico di Milano. As shown in Figure 4.10,

polylines were extracted from OSM, then BBs were developed and the point cloud was cropped using BBs. In the Figure is shown also one of the cropped ROIs.

The second test was made using a dataset acquired with an MMS mounted on a car, acquiring a road in Porto (Portugal), kindly provided by Lucía Diaz-Vilariño, Universidad de Vigo. As depicted by Fig. 4.11, the points of the trajectory (provided by the instrument) were used to create BBs and to crop the point cloud. The last image on the right shows one of the ROIs. The tests showed that the method was reliable either for OSM data or for trajectory from the instrument.

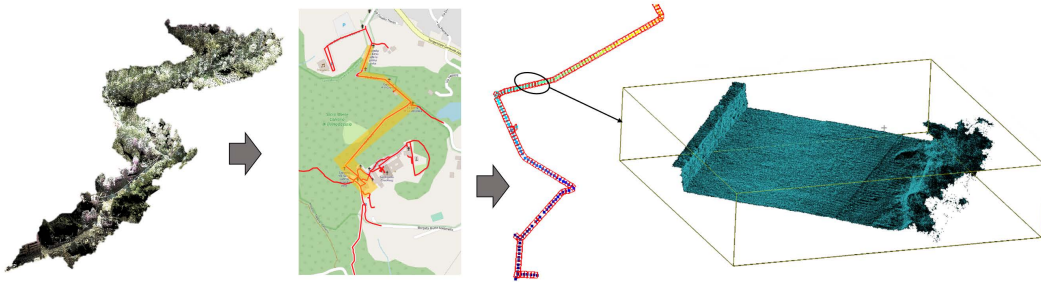


Figure 4.10: Tests on Bounding Boxes creation from a PMMS and using OSM data to extract the trajectory. The dataset was acquired in Domodossola (Italy) with a Leica Pegasus Backpack.

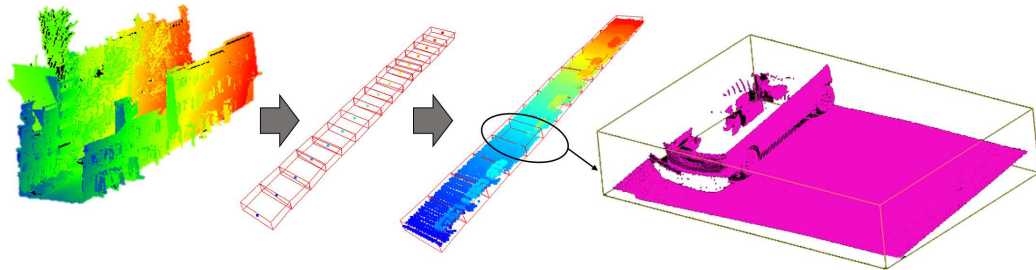


Figure 4.11: Tests on Bounding Boxes creation from a MMS and using instrument data to extract the trajectory. The dataset was acquired in Porto (Portugal) with a LYNX Mobile Mapper of Optech.

4.3 Segmentation

The final aim of the method is to subdivide the ROI into 3 classes: 'road', 'sidewalks', and 'other'. The 'other' class is necessary because it will act as a container of all points that belong neither to the street nor to the sidewalk. From now on, all steps were performed considering one ROI at a time.

The method was broken down into two fundamental steps: segmentation and classification. The segmentation consisted of the subdivision of each ROI in clusters

and the definition of a candidate cluster as a possible sidewalk. Considering that trajectory data were available, and considering that typically a road has sidewalks on both sides, one candidate on left and one on right were identified. The following classification step evaluated the probability that the selected candidates were sidewalks or not.

To carry out a cluster segmentation, several clustering algorithms⁵ were investigated and tested. The selected one was the most suitable for the purposes of this work, and with a good balance in terms of results and computation time. The algorithm chosen was *k-means*. This algorithm, originally proposed by Lloyd [142], tries to aggregate data points depending on their similarities. The similarity is evaluated basing on the features of points. The basic idea is [143]: given an initial but not optimal clustering, relocate each point to its new nearest centre, update the clustering centres by calculating the mean of the member points, and repeat the relocating-and-updating process until convergence criteria are satisfied. In other words, the algorithm starts with an initial clusters organization, then each cluster is iteratively refined by continuously moving the centres of the clusters, when convergence criteria (which are specific of the algorithm itself and should not be set by the user) are met the clustering result is considered optimal.

The *k-means* algorithm used in the workflow was implemented from the python library *scikit-learn*⁶. The method requires as input the points to be clustered, together with the features that should be used to compute clusters centres, and the number of clusters, which defines the numbers of centres. It is also required to specify a way to perform the initial clustering, among the available ones, *k-means++* was selected because it selects initial cluster centres for *k-mean* clustering in a smart way to speed up convergence. Additionally also the maximum number of iterations can be chosen (the default value is 300, and was left as is).

4.3.1 Feature selection

For clustering purposes, point features play a primary role; to have a good trade-off between performances and computation time, a selection of which are the most appropriate features to be used should be implemented. As already discussed in paragraph 4.2.1, the clustering should be carried out following the basic idea of the whole method: different paving materials represent different elements of the urban scene. For this purpose, features which take on different values on different floorings are useful.

Firstly, the features that should be considered are the ones that come directly from the survey dataset: XYZ coordinates, Intensity, RGB colours. XY coordinates are related to the spatial arrangement of points, but considering that roads are rarely parallel to the XY axis, and in most of the cases roads are inclined respect those two axes, points of the same element could have very different values of XY coordinates, making these two as not useful features for the clustering process.

The Z value is more important than the XY values of points, in fact, XY values can result in clustering errors, while the Z value could be helpful in the areas where there

⁵<https://scikit-learn.org/stable/modules/clustering.html>

⁶<https://scikit-learn.org/stable/modules/generated/sklearn.cluster.KMeans.html>

is a difference in elevation between materials. Even if in historical contexts the urban elements usually are not arranged with different elevations, it is also true that there can be a co-presence of elements that are at the same elevation and also elements that have a different elevation. Adding the Z value as a feature, the clustering can tackle those cases.

The Intensity is a function of several variables, including the distance from the laser, the angle of incidence of the laser beam on the surface and the specific material reflectance [59]. Since the method relies on the identification of different paving materials, which have different reflectance under laser light, the intensity could be an appropriate feature. The presence of the Intensity attribute is strongly linked to the survey instrument used, in fact it can only be measured if the survey instrument is based on a LiDAR, as it measures the intensity of the laser signal return.

A consideration should be made for the RGB-colour attribute. It is not suitable to be used as a feature because observing the survey data (in paragraph 3.3.4) it was noted that over- and under- exposure problems negatively affected the RGB-colour information of the points.

In addition to some intrinsic characteristics of the dataset, it is necessary to calculate some geometric features that, representing in different ways the spatial organisation of the points of the point cloud, may be useful for subdivision into groups. After some tests, the features selected were Omnivariance, Sphericity, and Roughness (par. 4.2.1). All these three geometric features are related to the 3D arrangements of points and try to describe how much the considered neighbourhood is closer or further to a hypothetical planar surface. The radius used for neighbourhood selection was set to 5 centimetres.

The last feature considered is related to a topological relation between urban elements, and it is based on the hypothesis that the road can be subdivided into three stripes: a carriageway, in the middle, and two sidewalks, one on each side. To help the segmentation method, the position of the points with respect the trajectory line is considered and enforced by computing the distance between the road centerline and each point. In such a way, points on sidewalks should have a similar distance and the clusters could be developed in a sort of stripe-based shape parallel to the road centerline. The road centerline can be assumed to be the trajectory segment passing within the ROI considered.

Distance-feature computation in crossings

Where two or more roads cross, the instrument trajectory could be a source of errors in the distance computation, in fact, the trajectory may be curved and not parallel to sidewalks as in straight road portions, hence it cannot be useful to the clustering method and it can be a source of clustering errors. To cope with that, a slightly different method should be used to compute the distance in crossings.

Once a crossing was identified and classified as an L, T, X-shape (par. 4.2.2), it could be used to detect which are the ROIs closer to it and subsequently compute differently the distance feature for them. Involved ROIs were detected as the ones that were inside a circle of 5 meter radius from the crossing point position.

Then, two different ways were implemented for computing distances: one for L and T shaped crossings, and a second one for X-shaped crossings. In both cases,

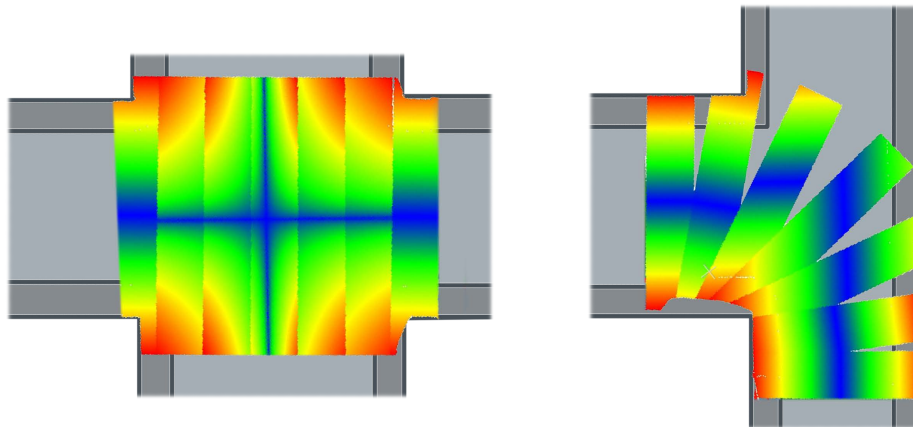


Figure 4.12: *Two examples of crossings, with several ROIs, and where each ROI is coloured depending on the computed distance value in a colour scale that goes from blue from the lower values, passing to green and yellow, and with red for the higher values.*

road polylines from the OSM dataset were used as alternative trajectory lines because here they were a better representation of the road centerline. For L and T shaped crossings, the OSM line selected was the one inside the ROI and more similar to the trajectory line; the similarity was checked by comparing the angles between the trajectory and OSM line. A different approach was used for X-shaped crossings, in this case, since two OSM polylines cross, the previous method could fail, so the distance was computed as the geometric mean of distances from all the OSM lines involved in the crossing. Figure 4.12 shows two examples of crossings, with several ROIs; each ROI was coloured depending on the computed distance value in a colour scale that goes from blue for the lower values, passing to green and yellow, and with red for the higher values. From the provided examples, it is clear that the method is much helpful for a better definition of stripes for clustering.

Dimensionality reduction

As the number of features increases, the feature space becomes sparse. This sparsity makes it difficult for algorithms to find data objects near one another in higher dimensional space⁷. PCA is one of many dimensionality reduction techniques. PCA transforms the input data by projecting it into a lower number of dimensions called components. The components capture the variability of the input data through a linear combination of the input data's features. In this workflow, the dimension of the variables was reduced to 3.

In addition to dimensionality reduction, it was also decided to re-scale the values of the various features to make them more homogeneous. If for Omnivariance the range is in the order of magnitude of 1, or even less, for the distance or Z value we reach larger values and considering Intensity we reach values in the order of thousands. After dimensionality reduction and features re-scale, the dataset is ready for segmentation.

⁷<https://realpython.com/k-means-clustering-python>

4.3.2 Number of clusters

The K-means algorithm requires that the number of clusters ("k") is specified in advance. To define the optimum number of clusters for the segmentation approach presented here, two parallel reasonings were used: a first one based on a statistical analysis of the features of the data to be segmented, and a second one based on a priori knowledge of the typical structure of the urban environment.

A fundamental step for any unsupervised algorithm is to determine the optimal number of clusters into which the data may be clustered. The Elbow Method is one of the most popular methods to determine this optimal value of k. It is a visual method. The idea is to start with k=2, and keep increasing it in each step by 1, calculating the clusters and the cost that comes with the training. At a certain k value (the optimal k), the cost drops dramatically, and after that, it reaches a plateau when k is further increased [144].

Several python libraries can exploit the Elbow method automatically, in this case, *yellowbricks* library⁸ was used to graphically see the optimal value for k clusters. The Elbow method was computed on several ROIs of different kinds, for a total of 20 ROIs including straight portions, corners, and crossings. Figure 4.13 shows one of the computed diagrams, in this case, associated with a straight road portion ROI; as it can be seen in the diagram, the suggested cluster value is 3. In general, on the tested ROIs, the k value was 3 or 4, depending on the ROI considered.

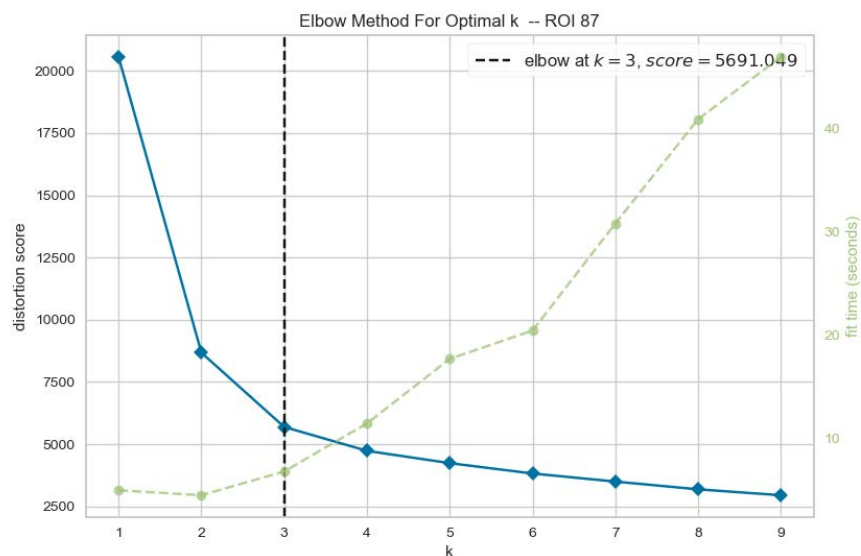


Figure 4.13: Graph of the Elbow method applied to a ROI. The blue line represents the distortion score, the green dashed line the time needed for computation (in seconds), and on the x-axis the corresponding number of k-clusters. In this case, the optimal k number of cluster was identified as 3, highlighted also by the vertical black dashed line.

⁸<https://www.scikit-yb.org/en/latest/api/cluster/elbow.html>

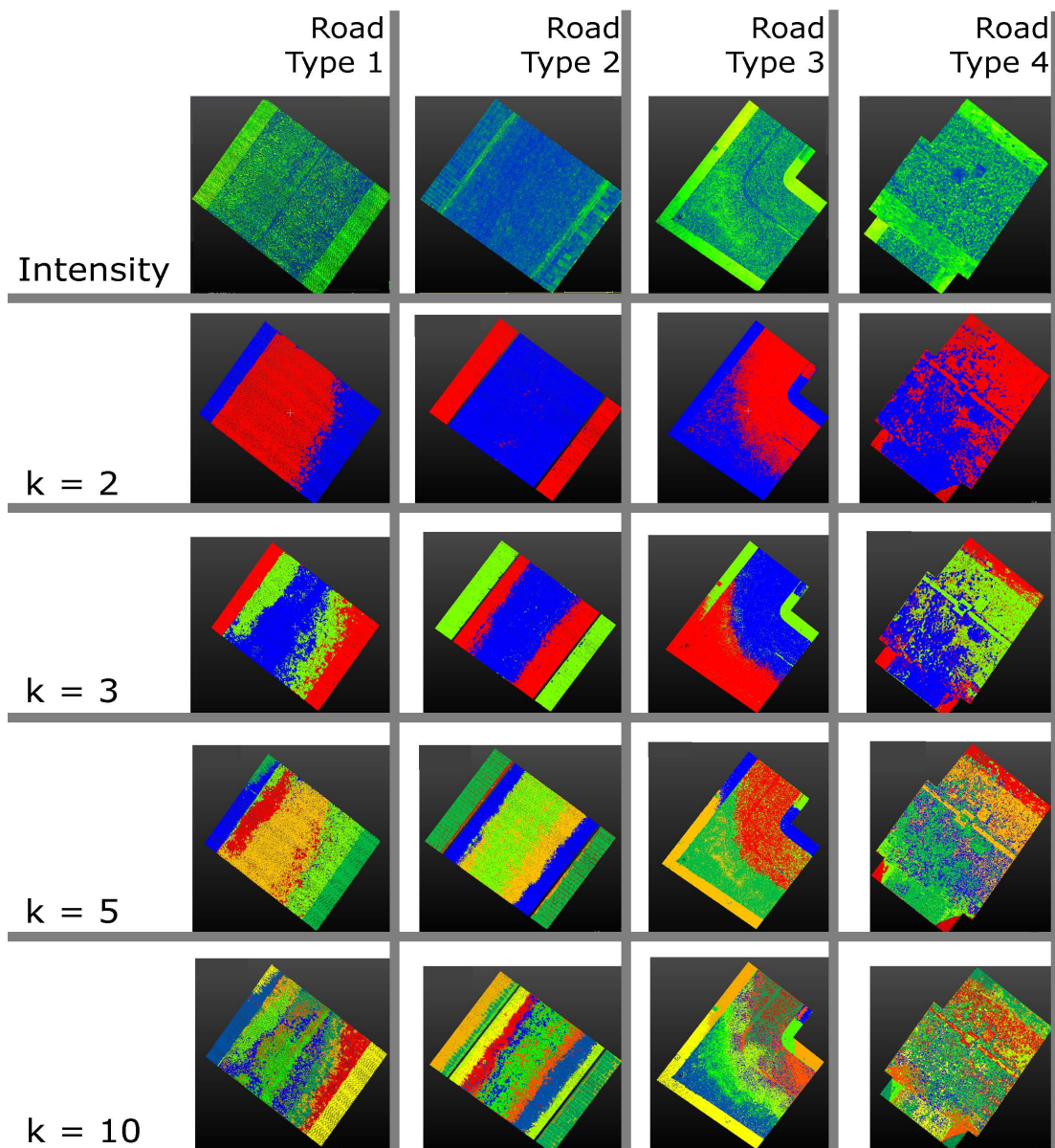


Figure 4.14: Reasoning put in place for the definition of the best value for the optimal k -number of clusters in the k -means algorithm, by making some visual comparisons on 4 types of ROI and with variable number of clusters.

The second approach used to investigate the optimal number of clusters relates to some knowledge of typical elements present in a ROI. The best situation is the one where there are 3 clusters: one cluster in the middle, for the roadway, and two other clusters on the side, for the sidewalks. Considering a case where on one side, or on both sides, there is no sidewalk, the number of clusters goes down to 2 or even 1. Another situation is when there are also some door entrances (maybe with a different elevation or material with respect the road and the sidewalks) or other paved surfaces; in this case, there are more than 3 clusters. To take into consideration those cases, the number of clusters should be greater than 3.

To understand which is the most suitable number of clusters to cover all the possible cases, 4 types of representative ROIs were extracted (see Fig. 4.14): one road with different paving materials at the same Z level (type 1 in the Figure), one with sidewalks at a higher elevation respect roadway (type 2 in the Figure), one containing a corner (type 3 in the Figure), and one in a crossing between two roads (type 4 in the Figure). Then, the k-means algorithm was applied to find which number of clusters was the most appropriate to split correctly each element. The first test with $k=2$ clusters worked well only in the road with different Z values, and probably for the presence of Z as a feature. Increasing the clusters to $k=3$, there were some improvements, but not enough for clustering purposes. With $k=4$ clusters, the results were much better. Each element of the road was well separated by the others, even if it was not in a single cluster. Increasing the clusters to $k=10$, the elements were split into too many regions.

Concluding, in consideration of the two reasonings presented, it was decided to use a number of clusters which was a trade-off between the correct definition of elements and forecasting of how many elements there should be in the ROI; the selected number of clusters was $k=4$. This number was also comparable to the presence of road, sidewalk and door entrances on both sides.

4.3.3 Candidate selection and refinement

The parameters for the algorithm were defined: features were Omnivariance (computed with a radius = 0.05m), Sphericity (radius = 0.05m), Roughness (radius = 0.05m), Intensity, Z-coordinate, and distance from centerline; the number of clusters was $k=4$. Now k-mean clustering could be applied to the ROI, as a result, each point was assigned to a label that identifies the cluster to which the point belongs.

Before continuing, a further feature could be computed, and it was helpful to split clusters on right and left sides. Simply by checking the spatial position with respect the road centerline, it was possible to split clusters depending on their side. After this step the number of clusters in the ROI increased; for example, if a cluster included points on both sides, now it was split into two sub-clusters: one on left and one on right. At this stage, the ROI points were subdivided into several clusters. To speed up the process, only the centre of each cluster was used as a representative of the cluster itself.

Considering the typical arrangements of elements in the ROI, the cluster that is most likely to be a sidewalk is the one furthest from the centre of the road, on both sides. The candidate was selected by computing the distance of each cluster centre from the road centerline; the candidate sidewalk was the one with a higher distance.

Once the candidate was defined, it was necessary to refine the cluster, in fact, some noisy points could be considered in the cluster and should be removed. Plus, sometimes happened that the cluster of points included the sidewalk but also part of the road, with this refinement, points of the road were excluded from the selection.

To perform the refinement, an algorithm that can cluster data according to their density was implemented: DBSCAN. Density-based clustering methods identify distinctive groups/clusters in the data, based on the idea that a cluster in data space is a contiguous region of high point density, separated from other clusters by contiguous

regions of low point density [145]. In other terms, this kind of method can identify clusters according to high-density regions separated by low-density regions. Firstly proposed by Ester et al. [146], it is designed to discover clusters of arbitrary shape, depending on point density. It is robust to outliers, and it does not require the number of clusters to be told beforehand, unlike K-Means, where it is mandatory to specify the number of centroids.

In this work, DBSCAN was implemented through the scikit-learn library⁹. The algorithm requires the dataset with features, and two parameters: epsilon value and min sample. Epsilon (ε) value is the maximum distance between two samples for one to be considered as in the neighbourhood of the other; in other words, it is the radius of the circle to be created around each data point to check the density. 'Min samples' is the number of samples in a neighbourhood for a point to be considered as a core point, including the point itself; in other words the minimum number of data points required inside that circle for that data point to be classified as a Core point.

When a certain number of points *min_sample* are closed together less than a specified threshold ε , they are considered as a cluster, then points in the neighbourhood are tested, and if they are closer to the first cluster less than ε , they are in the first cluster, otherwise, they are in a new cluster. If there are not enough points to create a cluster (the minimum is *min_sample*), they are considered outliers by the algorithm.

DBSCAN can cluster points according to features, considering that the purpose is to refine the candidate cluster formed by the previous step, this could be done according to the spatial distribution of points (i.e. their density). To do so only the XYZ coordinates of points were provided to the algorithm as features.

The other two parameters that have to be defined are the minimum number of points to create a cluster *min_sample* and the minimum distance ε to consider two points as of the same cluster. This choice is most important because the algorithm produces a list of outliers, which are points not assigned to a cluster. If the two parameters are too strict, the number of outliers will be very high, on the other hand, if they are too large, the refinement operation could be ineffective.

To keep under control outliers obtaining a good refinement result, the value of *min_sample* was set to 20. In such a way, smaller groups of points, which did not pertain to the candidate, were treated as outliers and removed from the main cluster. The minimum distance ε was set to 0.05 metres, a value comparable to the density of the point cloud (one point each 2 cm). In fact, in the cluster the areas with a homogeneous density were all pertaining to the same urban elements, and the isolated groups (that should be removed), had a lower density.

In conclusion, the results of the segmentation approach were two candidate clusters for each ROI, refined such that the candidate should include only the sidewalk points. A summary of the segmentation method is provided in Figure 4.15, where a ROI was shown after each step of the method. From top to bottom, the ROI is subdivided into 4 clusters using the k-means algorithm, then the clusters were subdivided depending on their position (left or right) with respect the road centerline, then the candidate was selected as the further away clusters from the road centre, then the candidate was refined with DBSCAN to remove outliers.

⁹<https://scikit-learn.org/stable/modules/generated/sklearn.cluster.DBSCAN.html>

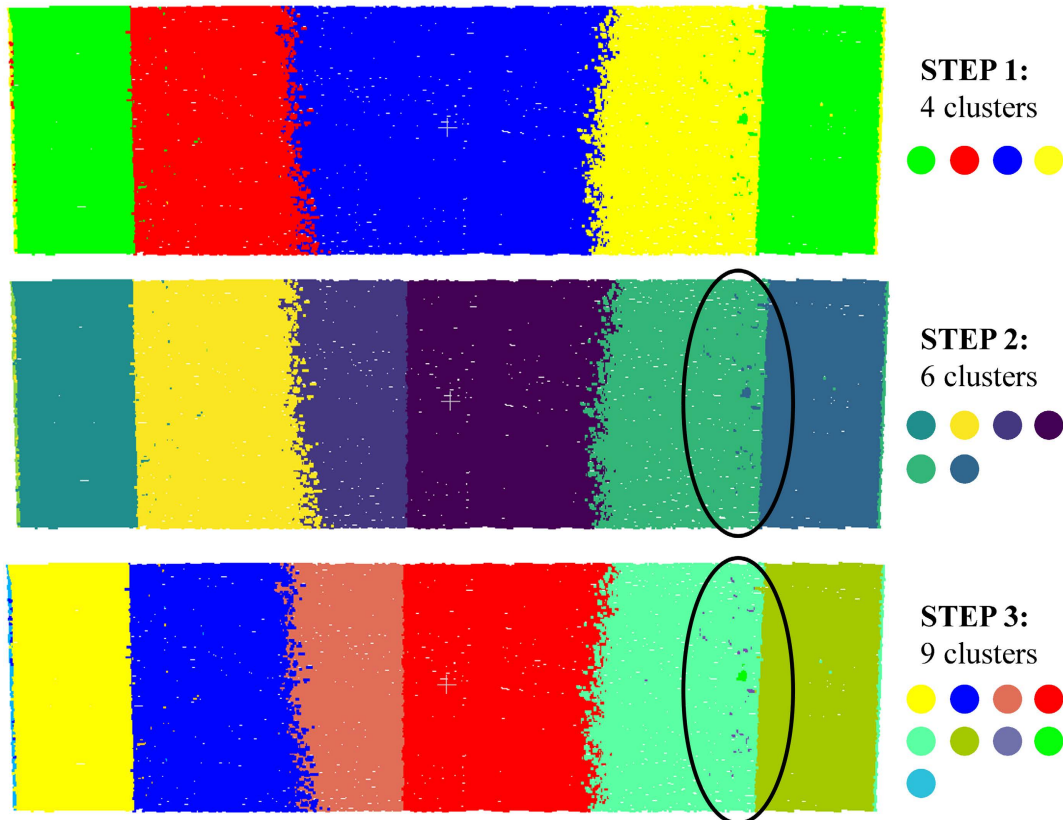


Figure 4.15: The subsequent result of the segmentation method on the same ROI. Top: after k -means (4 clusters). Middle: subdivision according to position (left/right) of points respect centerline of the road (6 clusters). Bottom: candidates are refined using DBSCAN (9 clusters). Note that the candidate sidewalk cluster on right has sharper borders after the DBSCAN step.

4.4 Semantic labelling

This section aims to classify the segmented ROI. For the labelling, 3 classes were considered: class 0 for "sidewalk", class 1 for 'road', and class 2, 'other'. Class 2 was an additional class considered for all the other areas which were neither road nor sidewalk. For example doorsteps, outdoor stairs and porches were considered as part of class 2.

The results of the previous step were two candidates sidewalks, one on left and one on right. At this point, what needs to be done is to find out whether a candidate is a sidewalk or not. To make this choice, it was decided to develop a voting system that, by analysing the candidate cluster and information related to its context, could define whether to confirm that candidate or reject him. Then after the definition of sidewalks, the other two labels were applied to the residual points, according to some contextual and spatial criteria.

4.4.1 "Sidewalks" class candidate check

To confirm a candidate sidewalk a voting system was implemented. To define this system, some considerations were done according to prior knowledge related to how

sidewalks could be recognized in urban environment. Thinking about a user walking in an urban environment and trying to figure out if there is a sidewalk and, if so, what part of the road surface is a sidewalk, it is reasonable to say that there are some unwritten rules that can be used for this purpose.

For example, a sidewalk is usually located on either side of a road, so there will be the roadway in the middle and two strips, which are the sidewalks, on either side. In addition, a sidewalk is generally located in the immediate vicinity of buildings and is in front of (and adjacent to) the building facade facing the street. In addition, the pavement is generally visually separated from the roadway: there are different methods, it can be identified by posts that separate the roadway from the pavement, or it can be identified by a different paving material, or it can be at a higher elevation than the road, or, if the paving is totally homogeneous and there is no difference in elevation, it can be identified by road marking that separates roadway from sidewalks. The last thing that can be noted is the symmetric organization of the road elements, such that if there is a sidewalk on left, there should be one also on right.

Some of the hypotheses described are not true for a historical site. Thinking about the change of elevation between sidewalk and roadway, it can be noted that it often does not happen, instead, it is much more frequent to notice a change of material between the two elements. Another element that cannot be considered frequent is the symmetry between elements, although often present, but not sufficient to be considered as a useful rule; considering a street passing in front of a square (a very common situation for a historical centre), symmetry is no longer present. On the other hand, it is noted that precisely being the sidewalk a way to promote a natural connection between buildings in the city, the presence of a building facade can be considered a rule to confirm the presence of a sidewalk.

Based on what has just been described, the voting system was developed considering four criteria, one intrinsic and three extrinsic. The first -intrinsic- criterion concerned the material of which the pavement of the sidewalk was made, if it was different from that of the roadway, then there was a probability that it was indeed a sidewalk.

The extrinsic criteria were based on information related to the context around the candidate sidewalk. An initial check was made for the presence of a building facade near the sidewalk. Then, for a similar purpose, the presence of vertical surfaces on the same side of the street as the candidate sidewalk was searched for. This was because the pavement might be in front of a building, but it might also be in front of a railing or a boundary wall which were not identified as buildings. The third extrinsic criterion concerned continuity between different ROIs. Since the segmentation and classification method analysed one ROI at a time, which has a limited width, it could be assumed that a sidewalk identified in the previous ROI must have been in continuity with the one identified in the next ROI. If such continuity was present, the candidate was likely to be a sidewalk, otherwise not.

The method involved testing the 4 criteria and scoring the condition "candidate *confirmed* as a sidewalk" and the opposite condition "candidate *rejected* as a sidewalk" respectively. Depending on the condition with a higher score the candidate could finally be confirmed or rejected.

Evaluation function

To define if a candidate was confirmed or rejected, a simple function was developed, that took into consideration two hypotheses: candidate confirmed and candidate rejected. The hypothesis which got the higher score was the result of this voting system. The functions that took into consideration the two hypotheses were the following:

$$H_1 = M_1 + B_1 + V_1 + C_1 \quad (4.1)$$

$$H_2 = M_2 + B_2 + V_2 + C_2 \quad (4.2)$$

where M is the material check, B the building footprint buffer check, V the presence of vertical structures check, C the connectivity check, and were referred to the specific case of hypotheses $H1$ or $H2$. The values of the 4 parameters change according to the concepts already described, they are summarized in Table 4.1. For example, considering the first criterion, if the two materials are different the candidate could be confirmed, so in this case, the $H1$ (confirmed) should have a greater mark than $H2$ (rejected), and the choice was to give respectively +1 and -1. On the other hand, if the two materials are similar, the candidate should be rejected, so the $H1$ should now have a lower value with respect $H2$, the choice was to give -1 and +1. Also, the marks for the other criteria were defined similarly.

Criteria		Hypoteses	
		H1: Candidate confirmed	H2: Candidate rejected
M	Is there material difference?	Yes	+1
		No	-1
V	Are there vertical structures?	Yes	+1
		No	-1
B	Is the candidate near and in front of a building?	Yes	+1
		No	-1
C	Is there connectivity between candidates of previous and following ROIs?	Yes	+1
		No	-1

Table 4.1: Scheme of application of the voting system. Each criterion has a specific mark according to the Hypotheses.

Criterion 1: paving material

One of the main assumptions also used for the clustering phase is the difference in paving for streets and sidewalks. Therefore it was almost a direct consequence to include this difference as a criterion to confirm a candidate. In this approach, the input was a sample of the cluster of points in which the paving material was to be identified. The sample was extracted by selecting a cylindrical neighbourhood of points around the central point of the cluster. The radius for the neighbourhood selection was 0.15 m, enough to ensure that there was a sufficient number of points in the sample (the average point spacing on the point cloud was 0.03 m).

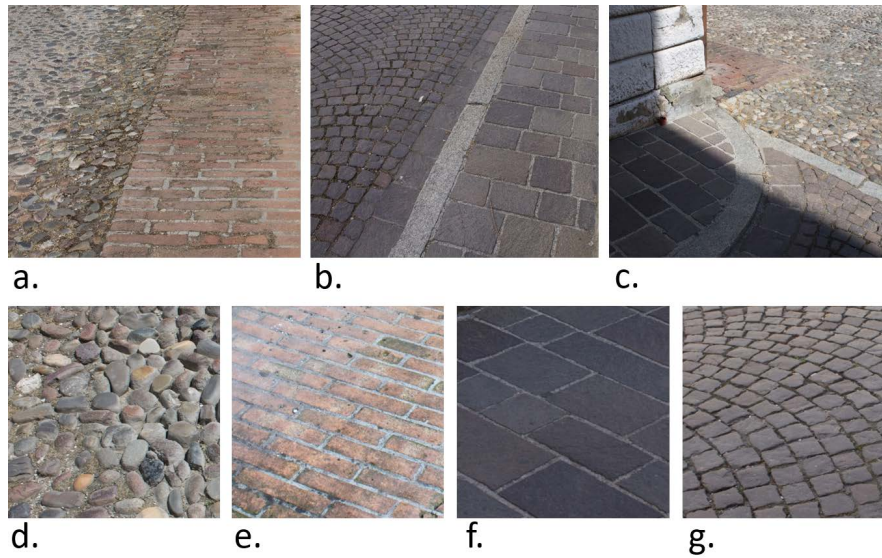


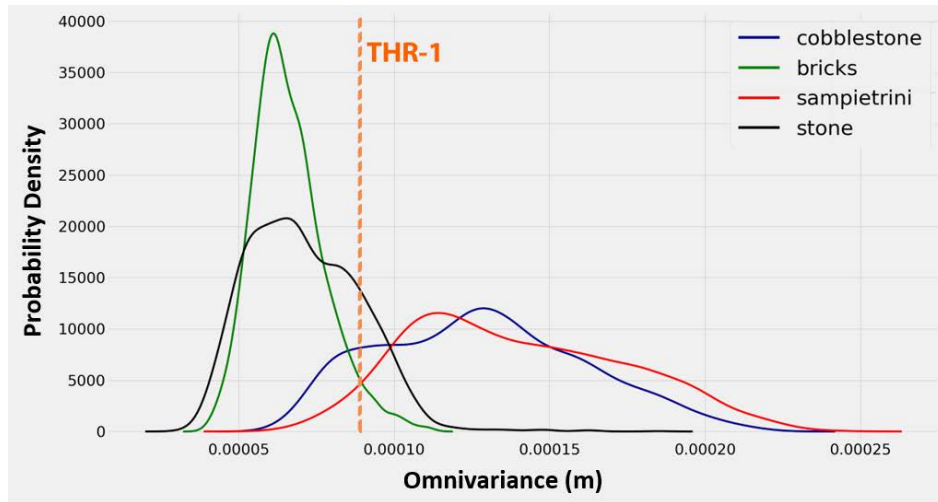
Figure 4.16: Pictures of some paving materials in the case study. a: combination of 'cobblestone' and 'bricks', b: combination of 'sampietrini' and 'stone', c: combination of 'sampietrini', 'stone', and 'cobblestone', d: 'cobblestone', e: 'bricks', f: 'stone', g: 'sampietrini'.

PAVING MATERIAL NAME	DESCRIPTION
Sampietrini	Squared stone blocks, aligned in consecutive radial grids (they can also be aligned on rectangular grids, but not in Sabbioneta). In Sabbioneta they are typically used for the road surface.
Bricks	Rectangular bricks arranged in a stretcher bond pattern. Typically in Sabbioneta, they are used for sidewalks, the largest dimension of the rectangle is orthogonal to the main direction of sidewalks.
Cobblestone	River stones lined up next to each other without any apparent regular arrangement. The individual elements are rounded and protrude from the surface. In Sabbioneta are typically used for roadways surface
Stone	Rectangular stone blocks, aligned on a regular grid similar to the English cross bond pattern. Typically used for sidewalks in Sabbioneta.

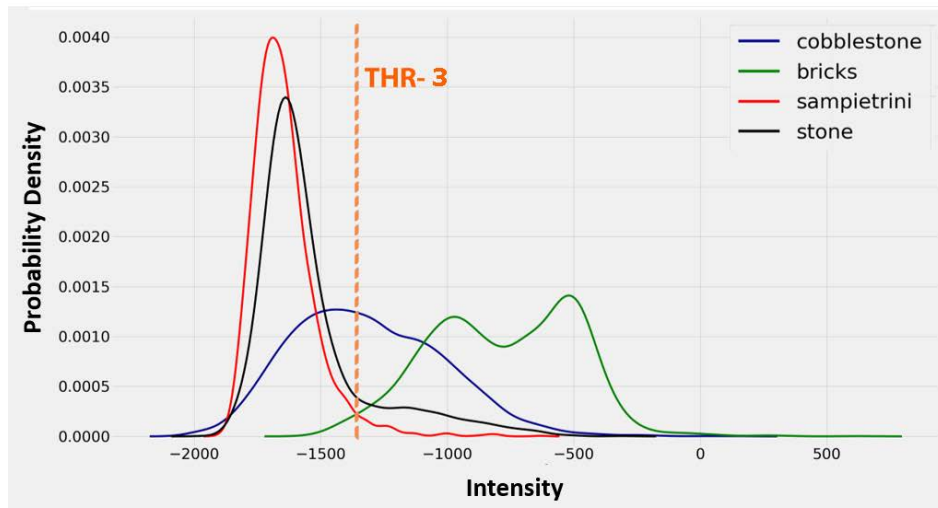
Table 4.2: Description of the 4 classes for paving material criterion identified for the knowledge-based method.

By analysing the geometrical features of points of the clusters, they were classified into one type of pavements, according to the paving material mainly present in the historical city: 'sampietrini', 'cobblestone', 'stone' and 'bricks' (see Fig. 4.16). A comprehensive description of these paving materials is also provided in Table 4.2. Several samples of paving materials were extracted and the statistical distributions of their features were analysed.

Among the features analysed, the two considered most interesting are reported in Figure 4.17. The presented graphs show the probability density functions of Omnivariance and Intensity values, for the 4 different paving materials. The Intensity values were managed using point cloud originally exported in ".pts" format from Leica Cy-



a.



b.

Figure 4.17: Cumulative probability density functions of several samples of the 4 materials analysed.
 a: Omnivariance, computed with a cylindrical neighbour selection with a radius of 5 centimetres.
 b: measured Intensity of the returning signal of the laser scanner.

clone software, which scales the Intensity values in a range from -2047 to 2048 for ".pts" files [147]. The Omnivariance was computed relying on a cylindrical neighbour of 0.05 metres.

Basing on the resulting graphs (Fig. 4.17) it was possible to develop a strategy that identified the paving material based on some thresholds. The proposed method, which is presented in Figure 4.18, firstly checks the Omnivariance mean value of the analysed sample, if it is lower than a specific threshold (THR-1) the sample is considered as 'sampietrini' or 'cobblestone'.

Then, to define which is of the two pavings, the shape of the Intensity distribution curve is analysed, a narrow curve is for 'sampietrini' and a wider curve is for 'cobblestone'. The shape of the curve is retrieved by looking at the standard deviation

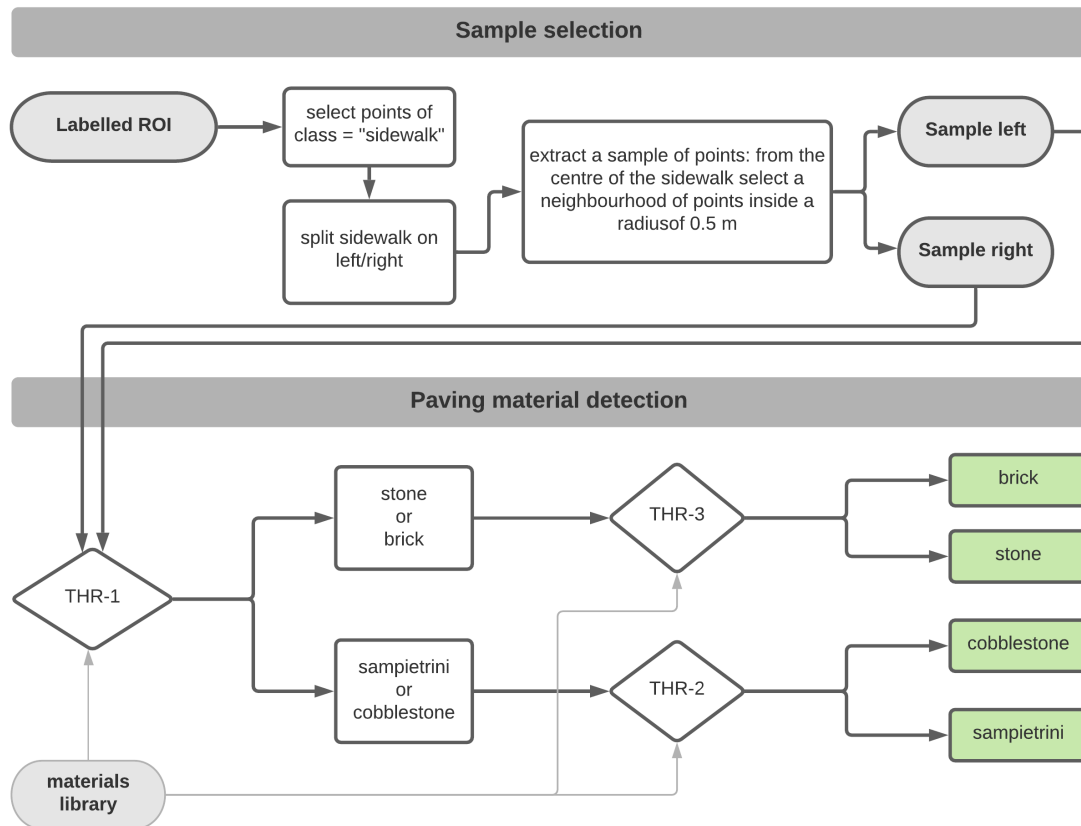


Figure 4.18: Workflow of the paving material characterisation methodology.

of the Intensity value of the sample and checked with a second threshold (THR-2). If a sample has the mean Omnivariance higher than the first threshold, the sample could be 'stone' or 'bricks', to identify the correct paving material the mean value of Intensity is checked and compared with the last threshold (THR-3).

To determine the three thresholds, their values were computed starting from the graphs in Figure 4.17. The threshold on the mean value of Omnivariance (THR-1) was determined as the intersection between the 'bricks' and 'sampietrini' curves, its value was 0.00009 metres. The threshold on the standard deviation of the Intensity distribution curve (THR-2) was determined as the average between the average standard deviation of 'sampietrini' and 'stone', and the average standard deviation of 'bricks' and 'cobblestone'. The threshold value was set as 125 metres. The last threshold was for the mean value of Intensity (THR-3), it was defined as the intersection between the 'stone' and 'bricks' curves. Its value was -1300.

In the voting system, for this first criterion, the identification of materials was conducted both on the sidewalk candidate and on the road points. To do so, a small amount of points of the candidate were selected, precisely a cylindrical neighbour with a fixed radius centred on the central point of the candidate. From the selected points the average value of Intensity and Omnivariance, and the standard deviation of Omnivariance were used to detect the paving material, following the threshold-based method. Following the same approach also the paving material of the road

was defined, in this case, the neighbourhood was selected centred on the point in the middle of the trajectory line in the analysed ROI.

As a result, the two pavings were detected and could be compared. Following the assumption, if the two pavings are similar, the candidate should be rejected, otherwise (i.e. if they are different) it can be confirmed as a sidewalk.

Criterion 2: building presence

Another important assumption is that a sidewalk is placed in front of and very near to the facade of a building. To perform this check there is the need to retrieve the building footprint and compare the candidates' position with respect to the nearest building.

Buildings' footprint can be exploited through the use of OSM. As described for the trajectory (section 4.2.3), it is possible to download layers from Open Street Map for free. One of these layers is buildings. Trying to download all the tags with key = "buildings", there is a set of polygons representing the building's perimeter. Figure 4.19 shows the polygons tagged as "building" and downloaded from OSM. Unluckily, the buildings' shapefile had some missing data, for example, the railings and boundary walls were not provided in OSM, and also some buildings seemed to be missing. It is worth mentioning that building perimeter is an attribute that can also be derived from other digital cartographic tools, if available: for example, the Topographic Database of the Lombardy region, or from other similar tools. Of course, implementing building footprint polygons from other sources could be a solution for missing data, or for geographic areas where the OSM dataset is not completely covered.

After the retrieval of buildings' footprints, to define if a candidate should be confirmed as a sidewalk, it is important to check its proximity with a building. In other words, if the centre of the candidate cluster is closer than 1.5 metres to the building, it is confirmed as a sidewalk.

To do so, building polygons were buffered by 1.5 metres and candidates inside the buffer zone should be confirmed; candidates outside the buffer should be rejected (see Figure 4.20). The value of 1.5 metres was selected because, on average, a sidewalk in Sabbioneta is not wider than 1.5 metres. This value can also be changed, making it somehow related to the ROI, by selecting for example a fraction of the width of the ROI itself.

Criterion 3: vertical boundary surfaces

The presence, or absence, of vertical structure on the same side of the road of the candidate, plays a double role. It can be used to confirm a candidate, when present, but also to reject a candidate, when absent. It is helpful to confirm a candidate when the previous criterion fails for data missing, or also for cases where even if the OSM dataset is complete, there are no buildings (so there is no building footprint), but still there is a boundary wall or a railing in front of which there is a sidewalk (see the example in Figure 4.21). On the other hand, the absence of a vertical structure can be helpful to reject a candidate, for example in the case of a crossing between roads, if a candidate is a portion of the crossing road, ideally, there should not be vertical structures and this criterion could be crucial for the decision of rejecting that candidate.



Figure 4.19: Map of the buildings' footprint in Sabbioneta. The building's polygons are in a shapefile downloaded from Open Street Map, and are coloured in purple.

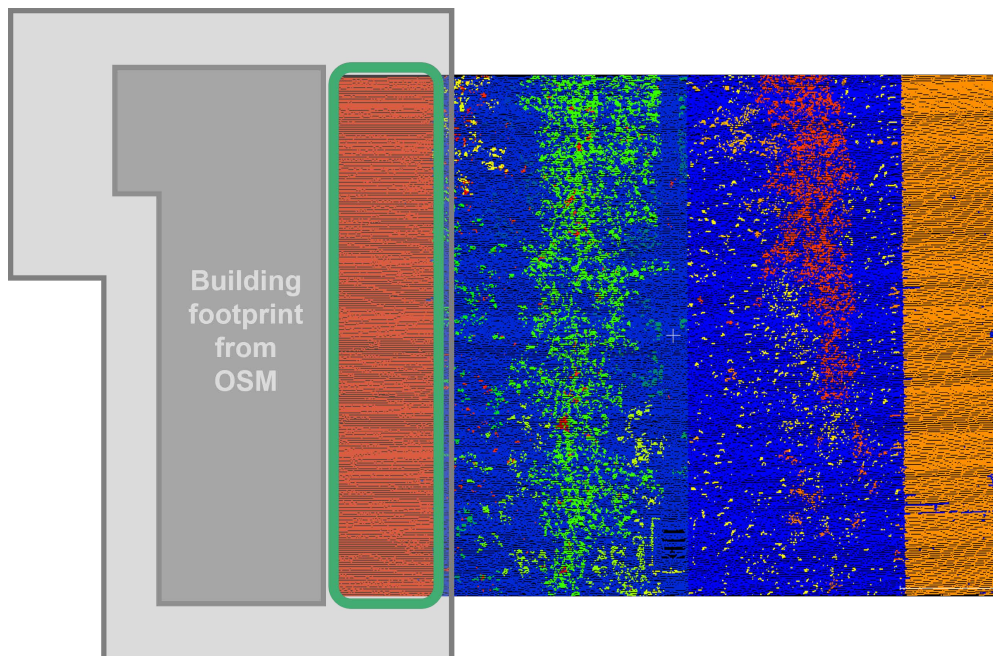


Figure 4.20: Scheme of the application of Criterion 2: building presence. A buffer zone is generated from the buildings' polygons. If the candidate is within the buffer zone it might be a sidewalk.

To detect vertical surfaces, the Nz component of the normal vector is used. It was already used in ROI selection to remove points on vertical surfaces, so what has to be done in this case is simply to retrieve the sub-cloud (as the one in Fig. 4.21 bottom) before removing these points, and check if they are sufficient in number to assume the presence of vertical structures on the side of the road where the candidate is located. In other words, if the points aligned on an almost vertical surface are greater than a certain value, it can be assumed that there are vertical elements. This threshold has been set as 1% of the points of the analysed ROI.

After this check, if there are vertical structures the candidate should be confirmed, otherwise, it should be rejected.

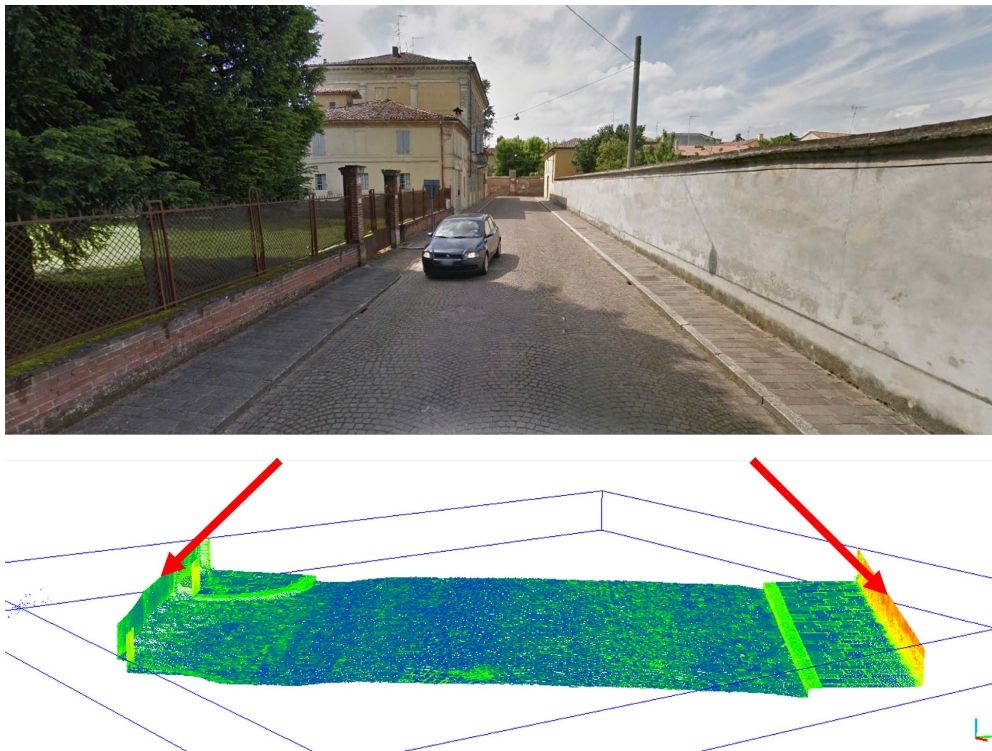


Figure 4.21: Scheme of the application of Criterion 3: vertical boundary surfaces. The presence of vertical surfaces near the candidate is a proof of the presence of walls near the sidewalk.

Criterion 4: continuity between ROIs

Commonly, two adjacent ROIs should have sidewalks clusters that are in continuity from one to the other, considering that they are two consecutive portions of the same sidewalk. So this criterion simply defines if three candidates (the one in the analysed ROI and the ones in the previous and following ROIs) are connected one to the other, if all three are connected, the candidate should be considered a sidewalk, otherwise rejected. The idea is graphically described in Figure 4.22, where 4 ROIs are near one to the other (in vertical), each candidate has a different colour, and the connectivity check is performed for the top candidate of the ROI in the red rectangle. The black rectangle serves to frame the 3 candidates whose connectivity is to be defined.

To define the continuity of candidates, the *connected components*¹⁰ can be used. Connected components are part of graph theory, the points belonging to the 3 candidates are considered as belonging to a single cloud and, depending on the distance between them, they are considered connected or not connected in a network formed by nodes and edges. A connected component inside a graph is formed by all the points that are connected. If the candidates are sufficiently adjacent, they will be in the same connected component, otherwise, they will be in two different components.

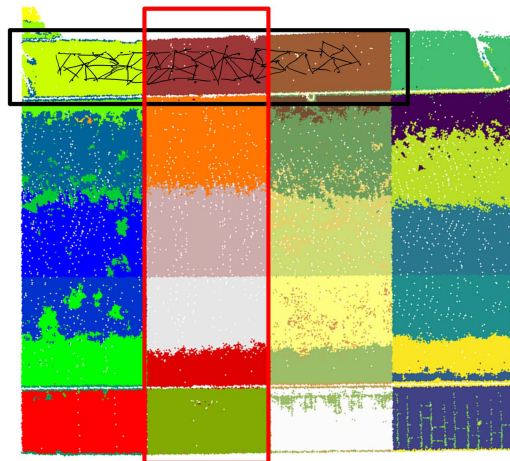


Figure 4.22: Scheme of the application of Criterion 4: continuity between ROIs. The continuity between sidewalks of closer ROIs is checked by exploiting the "connected components" of points within different ROIs. In the Image, on top, some black lines representing in a schematic way the idea of the network connection between connected components are drawn.

4.4.2 Labelling 'road' and 'other' class points

When the points of the ROI identifying a sidewalk have been defined, it is necessary to identify which of the remaining points are related to the road surface and which are to be considered external points and therefore belong neither to sidewalks nor to roads.

The first points that could be labelled are the ones in class 2 'others', which could be identified taking advantage of building polygons already used in the voting system. Disregarding the method used to derive buildings polygons, the assumption is that if a point is inside a building polygon, it should be a doorstep and so it should be assigned to class 2. In this way a first skimming of clusters was implemented, removing the points which were neither 'road' nor 'sidewalk'.

Concluding, sidewalks and other non-road elements were already identified, the remaining points, according to the method hypotheses, were assumed to be pertaining to the road surface and were straightforwardly labelled as 'road' class points. At this point, therefore, each ROI has been segmented and classified.

¹⁰https://docs.scipy.org/doc/scipy/reference/generated/scipy.sparse.csgraph.connected_components.html

4.5 Performances evaluation

At the end of the classification phase, it is necessary to establish how well this procedure has been carried out so that it can be considered reliable and the results can be considered acceptable and truthful. In other words, it is important to understand whether the points identified as sidewalks are actually sidewalks and thus give an estimate of the probability that the label assigned to a specific point corresponds to reality. This can certainly be done by a visual assessment, by observing the labelled point clouds and then giving a qualitative assessment based on that observation. However, this method does not provide objective information on the actual accuracy of the classification procedure presented. To give an objective assessment, it is common practice to rely on methods typically used in the field of data analysis (ML and DL) and which make use of performance metrics.

These metrics provide a comparison between the predicted labels and the real label. To do that, a prerequisite is a classified point cloud, whose labels are considered the real ones, this reference point cloud is called Ground Truth (GT). Practically speaking, the same point cloud that was classified by the method presented in this chapter, was also manually classified with the same semantic labels. Then, for each point, the resulting label from computations (predicted label) was compared with the label from GT; from the comparison it was possible to provide an evaluation of the algorithm performances.

Basing on the comparison between the predicted label and GT, it is possible to fill in the confusion matrix. The confusion matrix is a two-dimensional matrix, indexed in one dimension by the true class of an object and in the other by the predicted class by the algorithm [148]. Looking at that table, it is possible to see, for each class, how many points were correctly classified and how many points were misclassified and, for the second case, which are the wrongly assigned classes.

From the confusion matrix, three useful measures can be derived and used as an evaluation of algorithm performances: Precision, Recall and F1 Score.

Precision is a measure of exactness, in the information retrieval domain it can be seen as the ratio between the total number of documents retrieved that are relevant and the total number of documents that are retrieved. Recall is a measure of completeness, in the information retrieval domain it can be seen as the ratio between the total number of documents retrieved that are relevant and the total number of relevant documents in the database [149]. F1 score can be described as the harmonic mean or weighted average of precision and recall. The mathematical expression of these parameters are provided by equations 4.3, 4.4, 4.5:

$$Precision = \frac{TP}{TP + FP} \quad (4.3)$$

$$Recall = \frac{TP}{TP + FN} \quad (4.4)$$

$$F_1 = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (4.5)$$

where the value of terms TP , FP , FN , which stands for *True Positive*, *False Positive* and *False Negative*, can be retrieved from the confusion matrix.

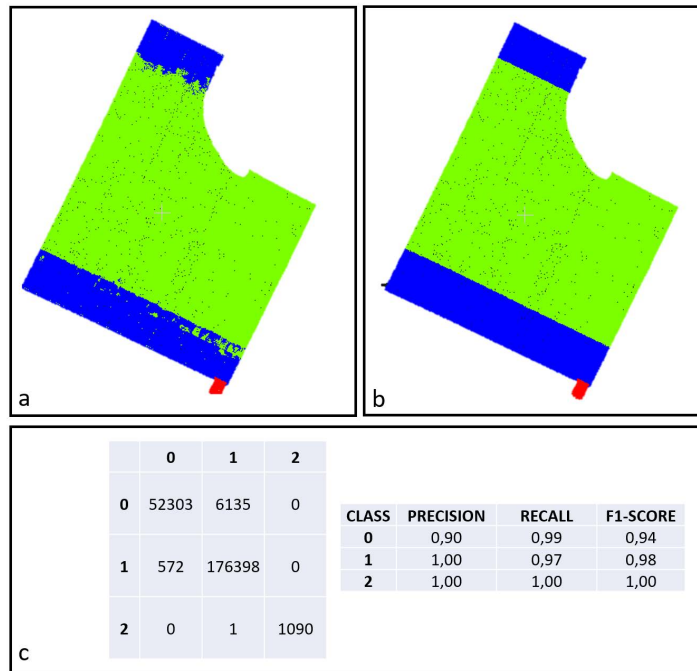


Figure 4.23: Example of performance evaluation on a ROI. Left: the GT. Right: the labelled ROI after the segmentation approach. Bottom: confusion matrix and precision scores.

For a generic class "A", its TPs are the points correctly classified, the FPs are the points which should have been classified as "A", but misclassified in other classes, and the FN are points that should have been in other classes, but misclassified as in class "A".

As an example, Figure 4.23 shows one ROI, on right the GT and on left the predicted labelled points, on the bottom of the image there are the confusion matrix and the precision parameters, being class 0 'sidewalks', 1 'road', 2 'other'. Looking at the first row of the confusion matrix, it is possible to notice that for class 0, 52303 points were correctly classified as class 0, 6135 points were misclassified as label 1, and 0 points were misclassified as label 2. Looking at the first column, it is possible to note again that 52303 points were correctly classified as class 0, but 572 points were misclassified as class 0, while they were of class 1, and 0 points were misclassified as class 0 while they were of class 2. An algorithm that produces predicted labels 100% equal to the GT would have a confusion matrix with values on the diagonal and zeros elsewhere.

4.6 Results

The presented method was tested on the Sabbioneta dataset. For all the computed point clouds, the performance metrics are presented and commented. To test the replicability of the method, it was tested also on other datasets acquired with different instruments in other cities: Mantova, Sacro Monte di Domodossola, and Porto. In the next paragraphs, the results of applying this method in different datasets are presented; for some of them the method was more successful than others. At the end of this

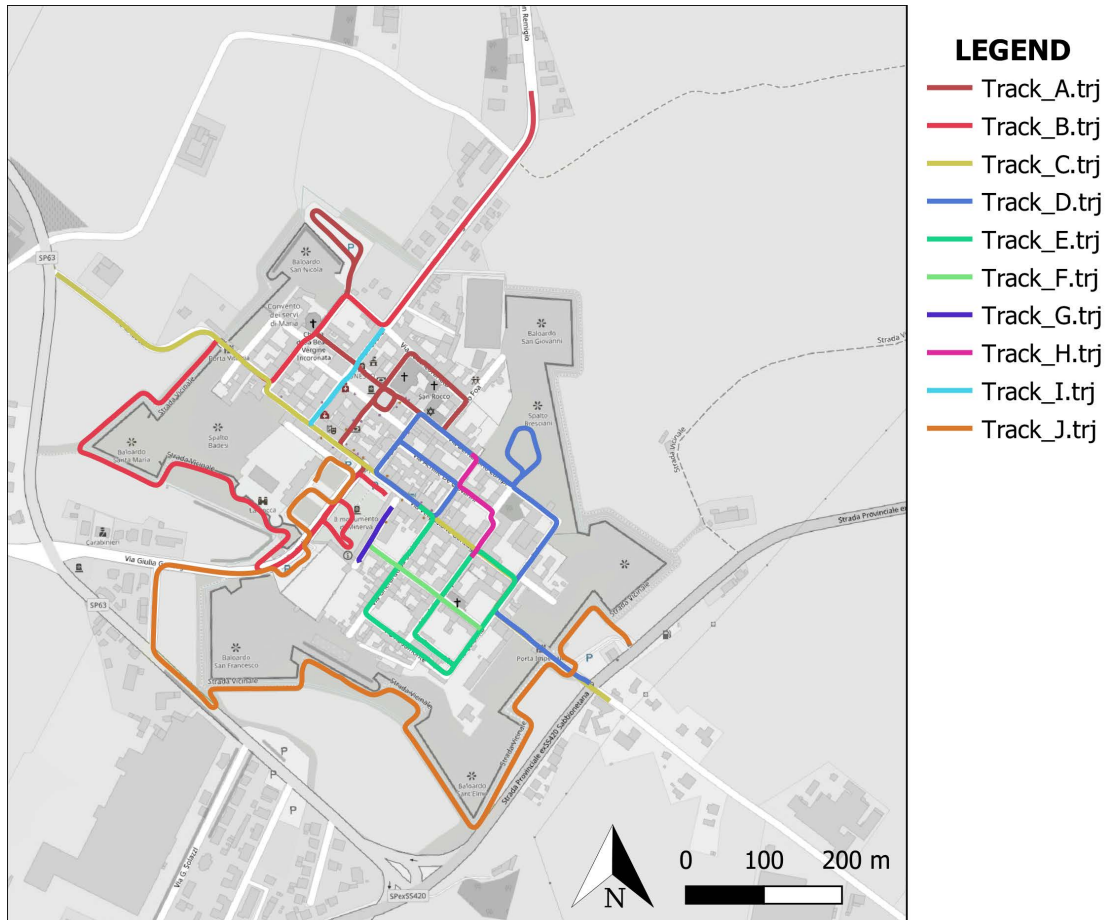


Figure 4.24: Map of the 10 tracks (or missions) of acquisition done while surveying Sabbioneta with Leica Pegasus:Two.

chapter a section discuss about that (sec. 4.7), and also provide some tables (Tab. 4.3, 4.4, 4.5) summarising the results obtained.

4.6.1 Sabbioneta

The survey of Sabbioneta, conducted with Leica Pegasus:Two, and previously described in Section 3.3, was subdivided into several acquisition tracks named in letters from A to J (in Fig.4.24 there is a recall of the Tracks' names and their paths' position). For requirements related to the execution of the survey the acquisition of each track started outside the city, so, after the subdivision in ROIs, only the ROIs inside the historical district were selected, as the roads outside the city were not elements of interest. For the same reason, also two tracks were excluded: track G and J. Track G was a small road (less than 100 m, behind the *Galleria degli antichi*) without sidewalks, used as a service road by the restaurant that was facing on it. Track J was a long path following a rural road running outside the city near the fortified walls, in this case, the surveyed road is not an element of interest for the research purposes.

In Appendix A, at the end of this dissertation, a summary of the tracks peculiarities and the computation performances are provided in a series of Figures, from page 255.

For each track two Figures are provided, a first one showing the path and some pictures taken on-site, and a second Figure showing the performance metrics, and a render of the point cloud, seen from the top. In green the road points, correctly identified, in dark blue the sidewalk points, correctly identified, in light blue the points of class 'others' correctly identified. In red points the ones that were erroneously identified. In the same Figure, there is also a pie chart with the proportion of correctly identified points to wrongly classified ones.

Track-A

This track starts from where via Vespasiano Gonzaga faces Piazza d'Armi, follows the road towards Piazza Ducale, passing via Scamozzi, crosses the square and follows via Assunta, skirting the block of buildings behind the church, then returns to Piazza Ducale, follows via Leoni, turning right into via Incoronata.

This route includes several 90-degree bends, passing (totally or partially) through squares, streets of different widths, and passing in front of porticoed areas, for a total of 534 metres of road analysed and for a total of 32 million points on the ground. The sidewalks are different: in the first section, up to Piazza Ducale, the road is paved with cobblestones and the sidewalk raised and with a kerb, is made of stone; from Piazza Ducale onwards the road is paved with cobblestone, with sidewalks at the same level as the road, made of bricks; the final street of this route, Via Incoronata, is asphalted and without a sidewalk.

This last road can be, and it was, a source of problems. The hypotheses at the basis of the algorithm assume that there is a sidewalk; via Incoronata does not have a sidewalk, so the algorithm, at the moment, is not able to understand that situation, and detected anyway a candidate sidewalk and, if the voting system criteria were met, it confirmed such candidate as a sidewalk. The implications of this error are discussed in the last section of this chapter.

After the computation, the performance metrics were calculated, and in consideration of what was just said for via Incoronata, two different performance metrics were computed: one for track A without via Incoronata, and one for only via Incoronata. Looking at the results, it is obvious that in the case of a road without sidewalks, but with all the voting system criteria fulfilled, the algorithm fails. The processing time was, on average, 45 seconds for each linear meter of the path.

Track-B

This track starts from via Incoronata and moves in the opposite direction with respect the previous track. After via Incoronata, the path passes through Piazza Libreria Grande, via dei Serviti and concludes on via Porta Imperiale, just in front of the door in the fortified walls.

This route is a short path, 258 metres (13 million points), composed of straight portions, some curves and the passage to a small square. The paving surface is only asphalt without sidewalks for via Incoronata, cobblestone for the Piazza and the road, with brick sidewalks. As for the previous track, via Incoronata is a source of errors, because its shape is such that the algorithm assumptions are not met, and it can fail. Looking at the rendered image of the point cloud, it is very clear that the first half

of the track (via Incononata and piazza Libreria Grande) has high errors in sidewalks recognition, for the reasons mentioned before.

Similarly to the previous case, two performance metrics were computed, one without via Incononata and one with only this road. The processing time was, on average, 62 seconds for each linear meter of the path.

Track-C

This track rides along the decumanus maximum of the city. It starts from Porta Vittoria, moves along via Vespasiano Gonzaga, passes on via Colonna and concludes its path on Porta Imperiale.

Track C is one of the most representative of Sabbioneta, because it has several unusual and interesting conditions, and it was used for the first tests while developing the algorithm. It has a length of 576 metres, for a total of 31 million points. Along its path there are narrow roads, 90 degrees curves, crossings with other roads, it passes near porches and there is also an area where suddenly the road doubles its width. Plus, sidewalks are sometimes at the same height as the road and sometimes at a higher level. The paving surfaces are various; for the road, cobblestone is used in the first half of the path, and sampietrini for the second half; while sidewalks pavings are made of bricks in the first portion and stone in the second one.

Performance metrics were computed. The processing time was, on average, 20 seconds for each linear meter of the path.

Track-D

This track starts from Porta Imperiale, moves along via Colonna towards the north direction, then turns left on via Bernardino Campi, which is a road parallel to the Decumanus of the city (via Vespasiano Gonzaga), then, at the synagogue, turn left onto via Dondi, then turn left again and return to the decumanus Maximus, via Vespasiano Gonzaga, which follow for a short distance; after about ten metres, turn left again onto via Albertoni and encircle the district, ending on via de Giovanni.

The path is quite long and complex, composed of long straight portions and several 90 degrees curves. In the middle of via Bernardino Campi, the path entered, on the right, the spalto Bresciani, an area with grass and earth, which surrounded the patrol path of the fortified wall. Since there are no pavements in this area and it does not represent an urban area, at the entrance to the spalt, the calculation was interrupted and then resumed. For this reason, an interruption in that area will be seen in the Figure with the rendered point cloud. The roads of this track present the familiar combinations of paving materials: cobblestone or sampietrini for the road and stone or bricks for the sidewalks.

This track has a total length of 812 metres, with 43 million points. The processing time was, on average, 32 seconds per meter.

Track-E

This track surveyed the blocks in the southern part of the city. It starts in via Vespasiano Gonzaga, then after a few metres, it moves south, turning onto Via Gherardo da Sabbioneta, turns left onto Via Rodomonte, and returns to Via Colonna and then

again Via Vespasiano Gonzaga. It runs for about ten metres and then turns left onto via della Stamperia, ending in a minor, less wide street, via Affo.

The shape of this path is a sort of spiral, capable of acquiring almost all the blocks at south respect the decumanus maximum of the city. There is only one road missing (via Rodolfini), which is surveyed by the next track.

This track has a total length of 812 metres, with 40 million points. The processing time was, on average, 46 seconds per meter.

Track-F

This track acquired only one road: via Rodolfini, which together with the previous track composes the southern blocks of the city. This road is paved with cobblestone for the road and bricks for the sidewalks, which are placed at the same height as the road.

It is a short path, only 172 metres, for 10 million points. The processing time was, on average, 21 seconds per meter.

Track-H

This track is useful to complete the city blocks at north respect via Vespasiano Gonzaga. Starting from Via della Stamperia, at Via Vespasiano Gonzaga, heading north, it passes in front of the municipal library, following the road that curves slightly to follow the façade of the building, then turns left onto Via Achille de Giovanni, to end its course at Via Albertoni, heading north. The streets in this track are paved with cobblestone for the road and bricks for the sidewalks, which are placed at the same height as the road.

The particularity of this route is the presence of three very close elbow curves which, as can be seen from the Figure with the cloud render, have been computed fairly well.

This track has a total length of 162 metres, with 16 million points. The processing time was, on average, 150 seconds per meter. This track, even if it is a short path, has several curves very close one to the other; evidently, the presence of curves increases the processing time.

Track-I

This track serves to complete the city's northeastern districts. It starts from via Pesenti, with porticoes on the left-hand side, passes through piazza Ducale, in front of Palazzo Ducale, and ends on via Teatro, at the junction with via Vespasiano Gonzaga.

The path is completely straight. Even if it does not have curves, the source of error is the Palazzo Ducale. In front of the palace, there is a staircase that acts as an entrance to the building; this staircase interrupts the pavements and joins the street. This condition is against the algorithm assumptions, and, creates errors.

It is a short path, only 142 metres and 10 million points. The processing time was, on average, 43 seconds per meter.

4.6.2 Mantova

Mantova is located about thirty kilometres from Sabbioneta. The urban layout of the two cities presents some differences (e.g., in Mantova it is not possible to identify a chessboard pattern), however, the characteristics and assumptions defined for the algorithm are certainly respected in the historical city centre.

Given the above, it seemed obvious and appropriate to carry out some tests in the city of Mantova as well. The survey was carried out with different instrumentation from the previous case. The instrument used was a Terrestrial Laser Scanner: Leica RTC 360¹¹. A total of 14 scans were carried out, with a resolution set on the instrument of 12 mm at 10 metres. The selected resolution was the lower possible, and it was chosen to be more similar to the resolution of the MLS system used in Sabbioneta. To work with similar point density on ground, the point cloud was then subsampled so that the final spacing between points was 2 cm, as in the case of Sabbioneta.

The scans were not georeferenced, the reference system was a local one and the origin of the system was on the first scan position. For this reason, after the registration of the 14 scans, a fictitious trajectory was created by manually picking points on the point cloud. For the same reason, the building polygons from OSM could not be used in the computation phase. To cope with that, the mark of OSM building presence in the voting system was set to 0.

The surveyed area is a straight road in front of the Mantova Campus of Politecnico di Milano. It starts in Piazza Carlo d'Arco and concludes in via Angelo Scarsellini. There are some peculiar elements: it passes near a piazza and in front of a historical building (Palazzo d'Arco, facing Piazza d'Arco); there are some parking areas aligned with the road on one side of via Scarsellini that separate the road from the sidewalk. The road is all paved with cobblestone, the sidewalks are different: some of them are made of white stone (in front of Palazzo d'Arco) and at the same level of the road, some others are in porphyry stone and at a higher level with respect to the road, some others are made by a different type of stone and with a higher level. Some pictures of the road are provided in Figure A.44, at page 236 in the Appendix.

The path has a length of 142 metres, for 7 million points. The processing time was, on average, 14 seconds per meter. Figure A.75 (at page 259 in the Appendix) shows the performance metrics, and a render of the point cloud, seen from the top, with the same conventions used for the images previously presented for Sabbioneta.

Analysing the results, it is possible to note that the performances are in line with those obtained in Sabbioneta. It is appreciable that the algorithm works properly also on a dataset acquired with a different system, and in a different city albeit with some similarities with the city in which it was first developed.

4.6.3 Sacro Monte di Domodossola

A test was performed on a dataset acquired with a PMMS, as it is believed that this instrument can be very useful in the historical urban context and therefore the potentialities related to the processing proposed in this chapter are to be investigated.

The dataset selected is Sacro Monte of Domodossola, the path winds from the village of Domodossola up to the hills of Mattarella, a place of very ancient origins

¹¹leica-geosystems.com/products/laser-scanners/scanners/leica-rtc360

where Roman and Lombard building materials, ceramics and utensils were found along with a marble headstone dating back to 539 A.D.¹²; Sacro Monte of Domodossola is a Unesco site since 2003. Some images of the dataset are provided by Figure A.51, at page 241 in the Appendix. The instrument used was Leica Pegasus Backpack¹³. This dataset was kindly provided by prof. Francesco Fassi, 3D survey group, Politecnico di Milano. To work with similar point density on ground, the point cloud was then subsampled so that the final spacing between points was 2 cm, as in the case of Sabbioneta.

As already mentioned, the quality (in terms of point density and homogeneity) of the data and the presence of specific attributes (Intensity), are fundamental elements for the algorithm to act correctly.

In the case of the Sacri Monti dataset, most of the route is on a footpath going up the mountain and therefore does not have the necessary characteristics for this test. However, in the dataset, there is an initial portion (a few tens of metres) that was acquired along the road of the village located further down the valley. This portion of the dataset has the right characteristics for the test and was therefore selected.

The selected portion of the point cloud has a length of 30 metres, for a total of 6 million points on the ground. The point density is homogeneous and sufficiently high for the algorithm to perform correctly, the Intensity attribute is provided. Both the road and the sidewalks have asphalt pavement, the sidewalks are slightly raised above the road level.

This dataset was georeferenced in a CRS, so it is possible to exploit OSM for trajectory recovery and also for building polygons to be used in the voting system. The processing time was, on average, 80 seconds per meter. Figure A.52 (at page 242) shows the performance metrics, and a render of the point cloud, seen from the top, with the same conventions used for the images previously presented for other tests.

Analysing the results, it is possible to note that the performances are in line with those obtained in Sabbioneta. This is a good result because it shows that the quality (intended as homogeneity and density of points) of the dataset is a primary requirement for the success of the algorithm proposed. It is also appreciable that the algorithm works properly also on a dataset acquired with a PMMS, in a different city (so a different urban environment) with few similarities with the city in which it was first developed; in fact, the paving material is the same both on-road and on the sidewalk.

4.6.4 Porto

All the datasets used for the previous tests referred to historical urban environments, which therefore complied with specific characteristics identified beforehand (i.e. historical environments, streets of a modest width, different street paving material and sidewalks). The functioning of the method in historical environments has therefore already been extensively tested. To understand how flexible the algorithm is, it is now desired to proceed with tests on urban areas of different conformation and more modern construction.

¹²www.sacrimonti.org/en/sacro-monte-di-domodossola

¹³leica-geosystems.com/products/mobile-sensor-platforms/capture-platforms/leica-pegasus-backpack

It was therefore decided to test the method on some urban areas of the city of Porto (Portugal). These datasets were kindly provided by Professor Lucía Díaz Vilariño of the University of Vigo. The instrument used to acquire the data was a MLS system: LYNX Mobile Mapper of Optech¹⁴. In this case, it was not possible to work with a similar point density on ground, because the dataset had a point spacing of 8 cm, on average, much lower than the point spacing of previous datasets.

The roads selected for tests areas are the crossing between Rua de Fernandes Tomás, Rua de Santos Pousada, and Rua de Cohelo Neto; a portion of Campo 24 de Agosto; a segment of Rua de São Filipe de Nery.

At the end of this dissertation, a summary of their peculiarities and the computation performances are provided in a series of figures, from page 246. For each track two figures are provided, a first one showing the path and some pictures taken on-site, and a second figure showing the performance metrics, a render of the point cloud, seen from the top, with the same conventions used for the images previously presented for other tests.

Before describing each test on Porto, a common comment is that the roads are very wide and the MLS system was able to survey the road surface with high density (5000 points per square meter), but the sidewalks are in a far position from the instrument and consequently, the point density on their surface is very low (100-500 points per square meter). In consideration of that, the geometric features were computed with a larger radius with respect to previous cases: 0.1 metres instead of 0.05 metres.

Furthermore, another issue is the urban structure of those roads, they are very wide, there is often a row of parking spaces between the roadway and the sidewalk, and the average width of sidewalks is comparable with the roadway width. In short, the urban conformation of the environments surveyed in Porto is not only very different from the environments surveyed in the previously analysed case studies (historical centres) but also very much in contrast with some of the assumptions underlying the classification algorithm developed and tested here.

In all three tests, the performance metrics show a very low rate of success. The algorithm is not able and not useful to semantically segment these point clouds. The reasons for this, as mentioned before, can be found both in the low density of points on the pavements and in the different conformation of the urban environment.

Test Area 1

The first area investigated is a portion of Campo 24 de Agosto, a wide road that passes in front of a bus station on one side, and near a park on the other side. The length of the path is 122 metres, with 5 million points on the ground surface. The sidewalk paving is in sampietrini, while the road is in asphalt.

Test Area 2

The second area is a crossing between three roads: Rua de Fernandes Tomás, Rua de Santos Pousada, and Rua de Cohelo Neto. This crossing is very close to the previous test area; the three roads are very large, and in their surroundings there are buildings, but also an open space area. Again the urban configuration is very different from

¹⁴www.connectcec.com/sites/cec/uploads/Lynx_SpecSheet.pdf

the hypotheses. The length of the path is 216 metres, with 11 million points on the ground surface. The sidewalk paving is in concrete, while the road is in asphalt.

Test Area 3

The third area is a portion of Rua de São Filipe de Nery, in a segment which passes near the church *Igreja dos Clérigos*. The road is modestly large, with a large pedestrian area in the final portion. The length of the path is 104 metres, with 11 million points on the ground surface. The sidewalk paving is in stone, while the road is in sampietrini.

4.7 Discussion

The method proposed in this chapter aims to perform semantic segmentation of a point cloud of a historical urban area, the surfaces identified were the sidewalks and the street. The assumptions that underlie the operation of the algorithm developed here include the difference in paving material between the street and the sidewalks, the presence of sidewalks on both sides of the road and their positioning in front of a building. Tests have shown that these assumptions were correct and represent characteristics that are repeated in the historical areas of different cities.

The algorithm was developed using the Sabbioneta dataset as a test field, actually using one specific track, the one passing through the decumanus maximum of the city (track C), which was representative of several peculiarities (e.g., straight portions, corners, crossings, enlargements and reductions in road widths). Thus, the algorithm was developed and calibrated on this track, and then extended and applied to all Sabbioneta datasets. The purpose of this research is to characterize the sidewalks, and for the subsequent steps (described in next chapters) it is important to identify, with high precision, the sidewalks. In light of this, the average rate of success should be determined by looking at the precision, recall and F1-score of sidewalk class. The Sabbioneta dataset has an average precision which can be considered satisfactory for the next phase of the research. Table 4.3 provides a summary of the precision of each track and the average precision metrics for the whole city.

Moving to the next tests, the method was tested to the near city of Mantova, and on a road in Sacro Monte di Domodossola. Two main elements make these further tests interesting and worthwhile: these tests were carried out on different historical areas from Italian cities, and the acquisition instruments are various (a TLS and a PMMS). These two elements make it possible to test the repeatability of the algorithm and its scalability in different areas, but with characteristics sufficiently similar to the hypotheses made when the method was developed.

These tests show that an important feature for good clustering is the Intensity attribute and the density of points on the ground. On both the datasets, the algorithm performed well and the results from performance metrics are in-line with the results from Sabbioneta tests. A summary of the performance metrics is provided in table 4.4.

It can therefore be deduced that a good density of points on the ground and a good signal intensity are a prerequisite for the correct operation of the method. One can then reason and think about the possibilities of improving or pre-processing the Intensity

Track	'Sidewalk' class			'Road' class			'Other' class		
	Precision	Recall	F1	Precision	Recall	F1	Precis.	Recall	F1
A	0.78	0.80	0.79	0.95	0.95	0.95	0.80	0.72	0.76
B	0.74	0.70	0.72	0.93	0.94	0.93	0.77	0.85	0.81
C	0.85	0.88	0.87	0.97	0.96	0.97	0.69	0.70	0.69
D	0.70	0.81	0.75	0.95	0.91	0.93	0.57	0.59	0.58
E	0.80	0.82	0.81	0.96	0.95	0.95	0.51	0.42	0.46
F	0.85	0.90	0.87	0.98	0.96	0.97	0.32	0.34	0.33
H	0.73	0.74	0.73	0.94	0.94	0.94	0.32	0.25	0.28
I	0.73	0.67	0.70	0.93	0.94	0.93	0.61	0.62	0.61
Avg.	0.77	0.79	0.78	0.95	0.94	0.95	0.57	0.56	0.56

Table 4.3: Performance metrics (Precision, Recall, F1-score) for all the Tracks in Sabbioneta, subdivided into the three classes, with average values.

Dataset	'Sidewalk' class			'Road' class			'Other' class		
	Precision	Recall	F1	Prec.	Recall	F1	Prec.	Recall	F1
Mantova	0.76	0.92	0.83	0.98	0.95	0.96	0.31	0.07	0.11
Sacro Monte	0.85	0.62	0.72	0.95	0.99	0.97	0.49	0.42	0.46

Table 4.4: Performance metrics (Precision, Recall, F1-score) for the dataset in Mantova and Domodossola, subdivided into the three classes.

Dataset	'Sidewalk' class			'Road' class			'Other' class		
	Precision	Recall	F1	Prec.	Recall	F1	Prec.	Recall	F1
Porto test 1	0.24	0.43	0.31	0.95	0.84	0.89	0.15	0.31	0.20
Porto test 2	0.46	0.43	0.45	0.85	0.88	0.86	0.36	0.11	0.17
Porto test 3	0.30	0.30	0.30	0.86	0.88	0.87	0.27	0.01	0.02
Average	0.33	0.39	0.35	0.88	0.87	0.88	0.26	0.14	0.13

Table 4.5: Performance metrics (Precision, Recall, F1-score) for the three dataset in Porto, subdivided into the three classes, with average values.

value to make it even more significant, especially in low and non-homogeneous density areas. Plus, the density of points is an issue, in fact, it is related to several factors related to the surveyed environment and the acquisition instrument.

One last test was carried out to analyse the ability of the method to perform even in more modern areas, which clearly "break the rules" and where some of the algorithm assumptions are not met. These tests were made on three portions of Porto (Portugal). The results showed that the method fails, but the reasons are not only to be found in the different conformation of the urban environment, but also in the lower

density of points (one point each 8 cm) on ground surfaces respect to the previous cases (one point each 2 cm). A summary of the performance metrics is provided in the following Table 4.5; it emerges that the average F1-score of 'sidewalk' class is 0.35, much lower than the previously tested datasets.

After all the tests have been carried out and the results analysed, some specific cases can be identified and the results can be discussed:

- *straight portions of road* are the optimal condition, in fact in almost all the ROIs with this situation, the segmentation was successful;
- *corners and crossings* represent a challenge, the difference between trajectory and centerline of the road is a source of errors when computing the distance feature, for this reason a specific computation of distance in corners and crossings considering OSM was implemented, anyway sometimes in corners some errors in classification may occur;
- *parked cars* act as obstacles and create holes in the point cloud, hiding the sidewalks and thus breaking the continuity between neighbouring ROIs, possible errors created by parked cars are however prevented by the operation of the voting system;
- *squares* are peculiar, since tendentially there is a sidewalk only on one side when a road passes near the square, also in this case the voting system was designed to take this possibility into account;
- *roads without sidewalks, or with sidewalks identified only by road markings on asphalt* are a problem, since the method is developed such that it always finds a candidate and tries to confirm or reject it as a sidewalk, in this case, even if there is no sidewalk, but the other criteria in the voting system are met, the method confirms the candidates as sidewalks.

Concluding the chapter, the proposed method is proven to be efficient and useful for the semantic segmentation of the urban point cloud, to detect sidewalks surfaces. The detected surface (specifically, their points) are now ready to be further analysed to extract geometric and other specific data that can be used to perform a physical accessibility evaluation of the investigated area.

5

Segmentation exploiting Machine Learning

The purpose of this chapter is to perform semantic segmentation of the point cloud of the historic city using a ML approach. Two approaches were tested, one by segmenting the point cloud into urban elements and a second by attempting to classify the cloud based on pavement materials. Segmentation into urban elements was found to be effective, with even greater accuracy than the method seen in the previous chapter. This approach also proved to be applicable to datasets from other Italian cities and even to datasets of modern urban areas. Classification concerning sidewalk material, on the other hand, did not prove to be as effective, but it can be a starting point for the characterization of sidewalks in the next chapter. An innovative element of this chapter lies in the development of an effective and scalable method for segmentation using ML, with features calculated and thought out in relation to the historical urban context.

The chapter is organized as follows: the first section introduces the method and provides some insight into ML and DL techniques. The two following sections describe the preliminary phases, related to the preparation of the dataset and the selection of the best classifier and features. The last section presents the test results and a critical discussion of the method.

5.1 Introduction to the method

Given the growing interest of the scientific community towards the exploitation of AI, and consequently in approaches that use ML and DL, and given the increasing number of scientific contributions and experiments and tests developed, also in the field of Geomatics, aimed at the use of these technologies, it was decided to develop and test an alternative method to carry out the semantic segmentation of the point cloud, exploiting ML approaches. Considering that the subject of this work is a historical city, with quite unique and peculiar characteristics, the aim of this chapter is even more challenging, because Artificial Intelligence algorithms should also be able to work despite complications due to irregularities in the urban structure, such as localised degradations or non-standard situations.

Previously, in chapter 4, a knowledge-based method to automatically segment an urban historical point cloud into 'sidewalk' and 'road' was presented, with satisfactory results. This method involved a workflow composed of several steps: the subdivision of the point cloud into sub-clouds, the segmentation of the sub-cloud using k-means (an unsupervised ML method), and the definition of sidewalks using a voting system.

Even if that method involved the use of an unsupervised ML algorithm, it was defined as a knowledge-based approach, because the ML algorithm was used only to subdivide the sub-cloud into clusters. The purpose of giving semantic meaning to clusters was assigned to a specifically conceived voting system.

In this chapter, instead, the semantic segmentation was performed completely by supervised ML. In other words, both the segmentation and the assigning of a semantic meaning were reached entirely by the ML tool in a single step. The purpose was to understand if supervised ML could be an alternative to the knowledge-based method, and compare the two workflows.

The idea at the base of the knowledge-based method was that different paving materials were used for different urban elements of the ground surface of the city. Relying on the same fact, the ML approach here presented tried to classify the point cloud depending on the materials present on the ground surface.

Specifically, two separated ML models were tested: a first one with the aim of classifying only two urban ground elements: 'sidewalk' and 'road' (section 5.3.2), and a second one with the purpose of labelling the paving materials of the whole city (section 5.3.3). As will be seen below, the first model achieved even better results than the method presented in the previous chapter (chap. 4); the second model did not lead to satisfactory results and a change of approach was necessary.

5.1.1 Machine Learning, theoretical references

Even if already introduced in chapter 2, a brief recall of some AI principles will take place in this section. AI can be intended as "*the discipline that studies and develops computational artefacts that exhibit some facet(s) of intelligent behaviour. Such artefacts are often referred to as (artificial) agents. Intelligent agents are those that are capable of flexible action in order to meet their design objectives*" [150].

In the second half of the 20th century, ML evolved as a subfield of AI involving self-learning algorithms that derive knowledge from data to make predictions [151]. ML offers a more efficient alternative for capturing knowledge in data and making data-driven decisions. "*ML algorithms can solve a wide range of complex problems in a generic way. These algorithms don't require an explicitly detailed design. Instead, they essentially learn the detailed design from a set of labelled data (i.e., a set of examples illustrating the program's behaviour). In other words, they learn from data. The larger the dataset, the more accurate they become. The goal of an ML algorithm is to learn a model or a set of rules from a labelled dataset so that it can correctly predict the labels of data points (e.g., images) not in the dataset*" [152].

Some examples of ML problems [152] include:

- Classification: classify something into one or more categories (classes). For example, identifying if an email is spam or not. Another example is identifying an image of an animal as a cat, dog, or some other animal.
- Clustering: divide up a large set of data points into a few clusters, such that points within a cluster have some properties in common. Unlike classification, the number of clusters may not be known beforehand.

- Prediction: based on historical data, build models and use them to forecast a future value. For example, demand for a particular product around the holiday season.

ML can be subdivided into three types: supervised learning, unsupervised learning and reinforcement learning (see Fig. 5.1). Supervised Learning refers to any ML process that learns a function from an input type to an output type using data comprising examples that have both input and output values [153]. The main goal of supervised learning is to learn a model from labelled training data that allows making predictions about unseen or future data. The term "supervised" refers to a set of training examples where the output labels are already known.

In Reinforcement Learning, the goal is to develop a system (called agent) that improves its performance based on interactions with the environment. The agent has goals related to the state of the environment; each state can be associated with a positive or a negative reward signal. The agent learns a series of actions that maximize the reward via an explanatory trial-and-error approach [151].

Unsupervised learning refers to any ML process that seeks to learn structure in the absence of either an identified output (labels, in supervised learning) or feedback (reward, in reinforcement learning) [154]. Three typical examples of unsupervised learning are clustering, association rules, and self-organizing maps. Unsupervised learning deals with unlabeled data of unknown structure and is able to explore the structure of the data and extract meaningful information without the guidance of a known outcome.

A final description should be made for Deep Learning, which is a special approach in ML that covers all three branches (supervised, unsupervised, and reinforcement learning) and seeks also to extend them to address other problems in AI which are not usually included in ML [155]. DL is concerned with algorithms inspired by the structure and function of the brain, called artificial neural networks.

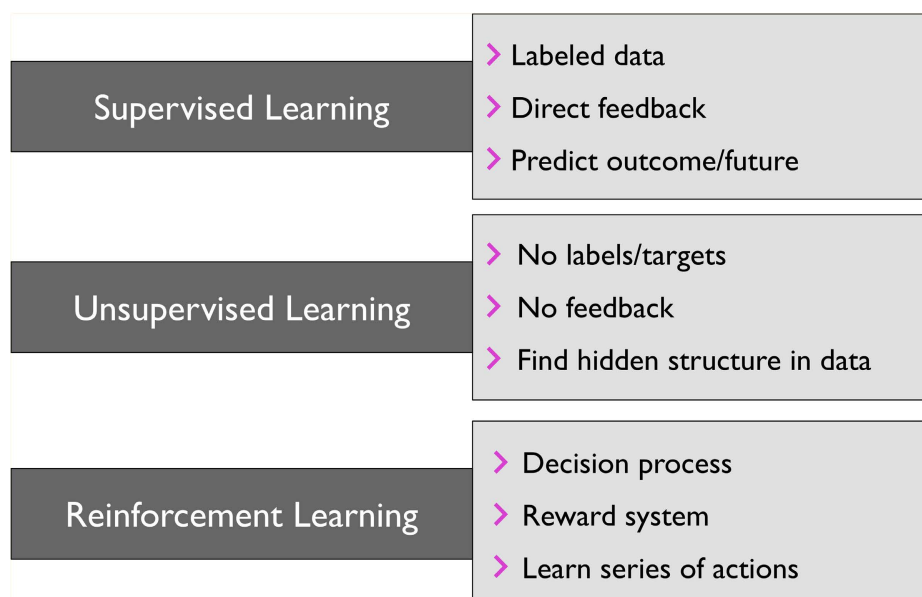


Figure 5.1: Types of ML and their characteristics, from [151].

Indeed, *Deep neural networks, in general, refer to neural networks with many layers and a large number of neurons, often layered in a way that is generally not domain-specific. Availability of computing power and a large amount of data has made these large structures very effective in learning hidden features along with data patterns* [156]. The particularity of DL is that tools have the ability to select themselves the features that are significant for the classification [157], while the rest of ML tools need to be fed with features human-selected.

5.2 Preliminary phases

To perform the classification of the Sabbioneta dataset, it was necessary to perform some preliminary steps. In particular, it was important to previously manually classify the whole dataset, these labels constituted the GT, and were useful both for the training of the ML model and for the validation, to assess the accuracy of the ML model. Once completed the GT, the dataset was subdivided into training and validation datasets.

In this section, the generation of the GT and the reasoning made for the definition of classes are described. Then, the section presents and discusses specific methods for the selection of training dataset, the peculiarities of the dataset and the actions undertaken to cope with them, some tests made to define the most suitable ML classifier, and a feature selection approach implemented to select the most appropriate features to use for training the ML model.

5.2.1 The Ground Truth for Sabbioneta

To train the ML model an already classified dataset is necessary. The ML model learns from it the features related to each class and then it is able to predict a new dataset. In this work, the whole dataset was manually classified, then a portion of it was used for training, and all the remaining portion was used for testing.

For the Sabbioneta dataset, a double level classification GT was performed. At first, all the dataset was subdivided considering the following urban elements: 'sidewalk', 'road' (including urban squares and rural roads), 'building' (intended as the building casing: facade and roof), 'urban furniture' (street poles, benches, waste baskets), 'vegetation' (trees and bushes), 'car', 'others' (mainly noisy points that should be removed, and people). Some images of the classified point cloud are presented in Figure 5.3. Then, as a subsequent step, a second level of manual classification, related to the paving materials, was performed only on those urban elements that compose the ground surface of the city (i.e. it was performed only on roads and sidewalks).

To define the classes for the second level of classification an on-site inspection was conducted. It was decided to add to the list all the materials present in the ground surfaces of the city. Following this approach, a total of 12 different types of material were identified (summarized in Fig. 5.2). As already stated in the previous chapter, the most common paving materials used for sidewalks and roads of the city of Sabbioneta were 'bricks' and 'stone' (for sidewalks), 'cobblestone' and 'sampietrini' (for roads). In fact, these 4 classes were used for the paving material criterion in the knowledge-based method presented in the previous chapter. The other materials included in the classes for this chapter, instead, were 'asphalt', and 'rural terrain' (used



Figure 5.2: Photos of the 12 classes of materials used for the ML predictions on paving surfaces in the city of Sabbioneta. The number of the class and the meaning is written below each photo.

mainly for roads but in a few areas). Other classes were related to specific features of the roads: 'manholes covers' (i.e. iron squared or rounded plates), 'stone curbs', and 'brick layers' (in a few roads some square bricks arranged in rows were used as a substitute for road markings). Concluding, some other classes were necessary for some different types of pavings used in small areas of the city, named: 'stone type 2', 'brick type 2', and 'gravel'. Figure 5.4 shows the GT for paving material classification.

Table 5.1 summarize the identified type of pavings (i.e. the classes) with a description of the characteristics of the surface. It is, again, worth noting that the first 4 classes correspond exactly to the paving material classes identified and used previously for the knowledge-based method, and already discussed in Table 4.2. Then a deeper description of the Sabbioneta dataset (in terms of paving materials) subdivided for each track of acquisition, is provided by Table 5.2. It is noticeable that in the entire dataset some paving materials had exclusive destinations of use: 'sampietrini', 'asphalt' and 'cobblestone' could be found on roads and squares; 'stone' and 'bricks' could be found on sidewalks and porches areas. Another interesting aspect to note is that the 'asphalt' and 'rural terrain' pavings were used mostly on track J, whose path was outside the historical centre and on the outer side of the fortified walls.

The manual classification was performed using the software CloudCompare, by manually cropping and merging points of the same class. To help this phase, after an on-site visit, two main sources of information were used: the photos taken during on-site visits to the city, and the publicly available images from Google Street View.

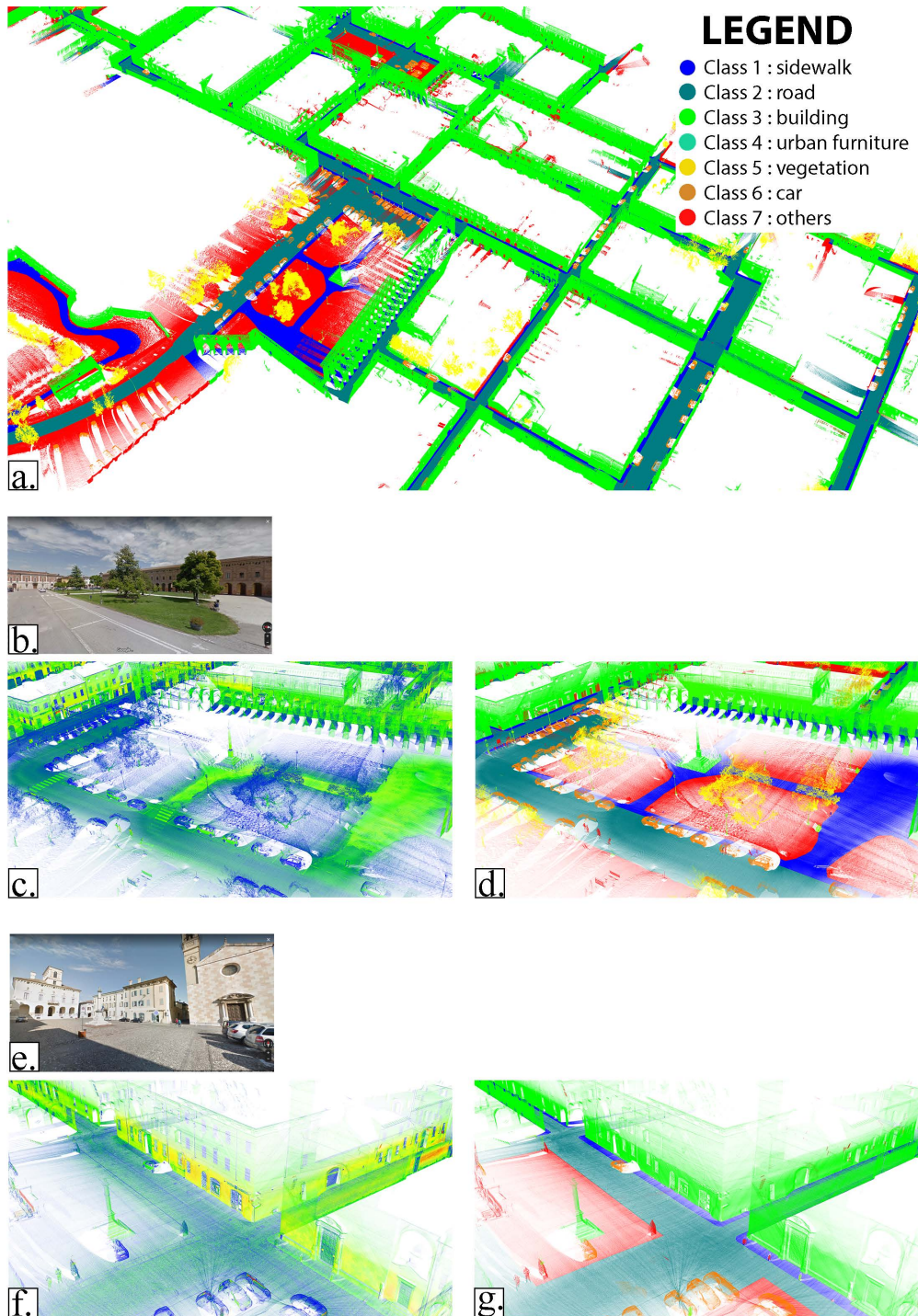


Figure 5.3: Several screenshots of the point cloud for the GT for urban element. a: general perspective view. points are coloured according to the GT for urban elements. b: photo of Piazza d'Armi. c: portion of point cloud of Piazza d'Armi, points rendered according to Intensity value. d: portion of point cloud of Piazza d'Armi, points rendered according to GT value for urban element. e: photo of Piazza Ducale. f: portion of point cloud of Piazza Ducale, points rendered according to Intensity value. g: portion of point cloud of Piazza Ducale, points rendered according to GT value for urban element.

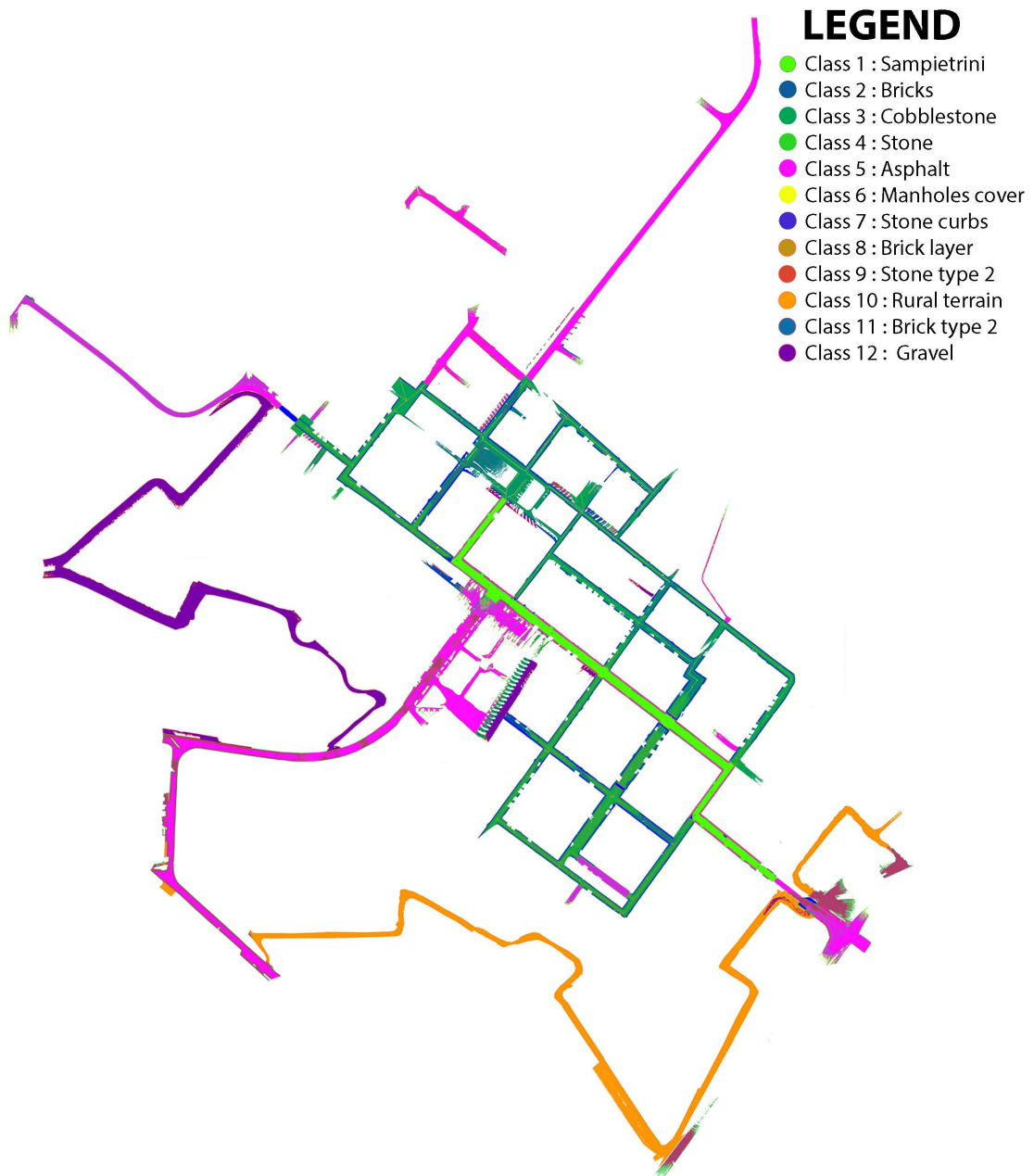


Figure 5.4: Point cloud of the whole city, points rendered according to the GT for paving materials.

CLASS NUMBER	PAVING MATERIAL NAME	DESCRIPTION
1	Sampietrini	Squared stone blocks, aligned in consecutive radial grids (they can also be aligned on rectangular grids, but not in Sabbioneta). In Sabbioneta they are typically used for the road surface.
2	Bricks	Rectangular bricks arranged in a stretcher bond pattern. Typically in Sabbioneta, they are used for sidewalks, the largest dimension of the rectangle is orthogonal to the main direction of sidewalks.
3	Cobblestone	River stones lined up next to each other without any apparent regular arrangement. The individual elements are rounded and protrude from the surface. In Sabbioneta are typically used for roadway surface
4	Stone	Rectangular stone blocks, aligned on a regular grid similar to the English cross bond pattern. Typically used for sidewalks in Sabbioneta.
5	Asphalt	Homogeneous, mostly flat surface.
6	Manholes	Regions typically with a square shape, made by iron/steel. They are both on the roadway and sidewalk.
7	Stone curbs	Rectangular stone blocks with an elongated shape arranged in a linear pattern. Typically used to separate sidewalks and roads.
8	Bricks layer in road	Square blocks, arranged in pairs and placed side by side to form a line. They are used in Sabbioneta on some roads to indicate the centre line of the road or even the lines of the parking areas. They are used as a substitute for road markings where the road surface is made of cobblestone.
9	Stone type 2	Irregularly shaped stone blocks of roughly similar size, arranged to form a pavement. In Sabbioneta they are used for some sidewalks (few areas).
10	Rural road terrain	Rural road, made by grass and soil, without paving.
11	Brick type 2	Bricks arranged in a Herringbone pattern. Used in a few areas of the city, under some of the porches.
12	Gravel	A layer of gravel, mainly whitish, formed by very small stones.

Table 5.1: Description of the 12 classes for paving materials defined for ML predictions. The first 4 classes correspond to the ones identified for Criterion 1 in the knowledge-based segmentation method, and presented at Table 4.2.

Track	Part	Materials- Road	Materials- Sidewalk
A	part 1	Cobblestone	Bricks and some porches along the road with 'brick type 2'
	part 2	Sampietrini	Stone
	part 3	Asphalt	No sidewalk
	square	Cobblestone	Bricks (in the square) and some areas with porches with 'brick type 2'
B	part 1	Asphalt	No sidewalk
	part 2	Cobblestone	Bricks
	part 3	Rural	No sidewalk
	square	Asphalt and gravel	No sidewalk
C	part 1	Cobblestone	Bricks
	part 2	Sampietrini	Stone and one porch along the road with 'stone type 2'
D	part 1	Cobblestone	Bricks and one porch along the road with 'brick type 2'
	part 2	Sampietrini	Stone
	part 3	Rural	No sidewalks
E	part 1	Sampietrini	Stone
	part 2	Cobblestone	Bricks
F	part 1	Cobblestone	Bricks
G	part 1	Gravel	No sidewalk
H	part 1	Cobblestone	Bricks
I	part 1	Cobblestone	Bricks
	square	Cobblestone	Bricks
J	part 1	Rural road	No sidewalk
	part 2	Asphalt	Asphalt
	part 3	Gravel	No sidewalk

Table 5.2: Description of pavings for the 10 Tracks of Sabbioneta. For each Track all the sub-portions are described with the corresponding materials.

5.2.2 Training dataset

To train the ML model, a portion of the dataset together with its GT labels should be selected. To do so, the dataset must be split and it is important to prepare it with specific characteristics. The training dataset should be balanced (i.e., the same percentage of points for each class should be selected) and not biased (i.e., the points should be selected such that they represent correctly the dataset and do not create misleading predictions by the ML model).

A first attempt was to use the *scikit-learn* python library, which provides a module for this purpose, called *sklearn.model_selection.train_test_split*. This module can split the dataset into random train and test subsets. Among the parameters, it is possible to set the size of the subsets (in percentage of the whole data) and it is possible to generate a stratified training subset, in this way the data are selected using the class labels and providing a balanced training dataset.

With this approach, the training subset could be selected among the whole point cloud and the points would come (theoretically) from all over the dataset, which for Sabbioneta means from all the roads of the city. Even if, apparently, this seemed to be a good thing, actually turned out to be a mistake. In fact, doing so, the training subset was composed of points taken very close to the ones for the test set. Let's imagine having a grid of points, 20% of training data means taking one point every five on that grid. Applying the same concept on a point cloud with a high density on ground surfaces, the result is that the training points are so close to the test points that their features have substantially the same values as the test points. The result is a sort of biased training dataset.

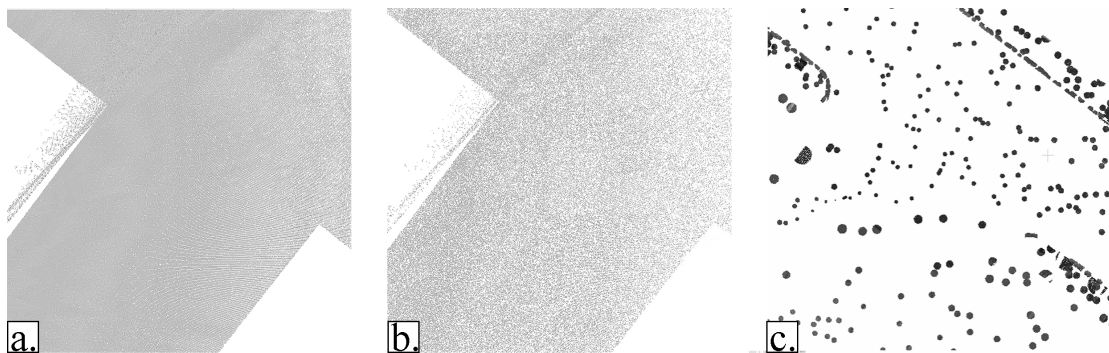


Figure 5.5: Comparison of different methods for the selection of training dataset. a: original dataset. b: random selection. c: circular crop selection.

To avoid this behaviour there is another possibility: the training subset could be selected by manually cropping portions of the full dataset. The problems, in this case, are at least two: with manual crops, it is difficult to control the percentage of points selected for each class, and the position of crops should be accurately selected to find areas that represent the whole dataset. A further step should be made. To generate a balanced dataset, with representative points of the entire city, an alternative way is to automate the process. To do so, a specific script was made, that selects points by developing several circular crops in the whole point cloud, when the points selected for a class have reached a specific percentage, no more crops are generated for collecting points of that class. The result is a balanced training dataset that should

not introduce bias and that has a large representation of the classes. Supporting what was said, Figure 5.5 shows the full dataset, the training dataset selected with the *scikit-learn* module, and the circular cropped training dataset. As it is visible, the method with the best cost-benefit ratio is the third one.

5.2.3 Peculiarities of the Sabbioneta dataset

The dataset used for the ML tests was prepared by merging all the tracks acquired in Sabbioneta, then using a threshold on point verticality, only points pertaining to ground surfaces were kept, and the other points were removed from the point cloud. The result was a very heavy dataset, composed of several millions of points, that took up 300 GB of storage space (point clouds with coordinates, colours, and point features). It is important to remark that, considering that many features were computed for each point, the file grew quickly in size. In the tests, the merged dataset and each single track dataset were used alternatively.

To perform the computation, the ML models store the entire dataset in the computer RAM. Although the computer used is high-performance and has medium-level features (specifically, 64 GB of RAM), it was not sufficient for the management of this dataset, and in fact, with a file of this size, it was not able to complete the prediction process due to a foreseeable "not enough memory" error. To overcome this issue, some solutions could be implemented.

It would be interesting to use computers with large computing capacities, made available by specific research centres, but it was considered that other solutions were more feasible, easier to implement, and cheaper. Therefore, the boundary conditions do not justify the need to use a supercomputer.

Another alternative would be to reduce the size of the dataset by subsampling the point cloud. With CloudCompare, for example, it is possible to perform, among the others, a "spatial" subsampling, which makes it possible to have a point cloud with a spatial density of points very similar to the one specified by the user. It was first decided to take full advantage of the high point density of the Sabbioneta point cloud, and for this reason, all the geometric features were computed on it. Then, to have a more uniform dataset, and to partially reduce the file size (although not by much), a subsample of the cloud was made, using 0.03m as sampling distance. This value is the average distance at which the points on the point cloud are already located. This step regularized a bit the point clouds, but still, the file size was too big for the computer.

Another solution could be to subdivide the data into various portions, which would then be analysed one at a time. Considering the actual size of the file, it would have been necessary to make many cuts, making the process very cumbersome. To deal with this problem, a strategy was implemented, that automatically splits the dataset into many portions, analyses them, and as a result, brings the dataset back together again in a single file. This procedure was developed using a library for Python that deals with parallel computing: *Dask*.

Dask is an open-source python library [49, 50], which is capable of extending the size of convenient datasets from "fits in memory" to "fits on disk". While *Dask* itself

is relatively new (it began in 2015)¹ it is built by the *NumPy*, *Pandas*, *Jupyter*, *Scikit-Learn* developer community, which is well trusted. As a result, *Dask* allows the user to extend the capability of other libraries; heavy computational workloads are managed by task graphs and executed by a series of schedulers that can work as a cluster of N possible machines. *Dask* will subdivide the dataset and the jobs to do on datasets in several chunks, give the computational task to schedulers and merge the final result. The size of chunks is selected based on machine capabilities and job requirements.

Regarding ML computations, there could be two main types of scaling issues: scaling model size, and scaling data size. The first kind of scaling challenge comes when the ML model grows so large or complex that it affects the workflow. Under this scaling challenge tasks like model training, prediction, or evaluation steps will (eventually) complete, they just take too long. To address these challenges it is possible to use *Dask Cluster* to parallelize the workload on many machines. The second type of scaling is when the datasets grow larger than RAM. Under this scaling challenge, even loading the data becomes impossible. To address these challenges, it is possible to use one of *Dask*'s high-level collections (like *Dask Array*, *Dask DataFrame* or *Dask Bag*) combined with one of the *Dask-ML* estimators that are designed to work with *Dask* collections.

In the specific case of the Sabbioneta dataset, the scaling issue was related to data size (the second type of scaling), and the implementation of *Dask DataFrame* and *Dask-ML* was the solution to overcome the problem. However, it was necessary to make some (not minor) changes to the code to be adapted to *Dask*; it allowed the management of very heavy files so it was considered a justifiable effort. The python library selected for ML computation was *scikit-learn*, and *Dask-ML* provides some meta-estimators that parallelize and scale-out certain tasks that may not be parallelized within *scikit-learn* itself. The specific module implemented for this work is called *ParallelPostFit*. So, after having converted the dataset into a *Dask DataFrame*², the *ParallelPostFit* module was implemented to load and exploit *scikit-learning* estimators. In the next paragraphs, the selection of estimators (i.e. classifiers) will be presented. The interesting feature of *Dask* is that, once the classifier has been called from *scikit-learn* via *ParallelPostFit*, it is possible to manage the classifier with the same parameters and the same dynamics as if it had been called directly from *scikit-learn*. The drawback is that having converted the dataset into a *Dask DataFrame* structure, if it is necessary to do subsequent operations on the dataset (e.g. change the colour of the points, add or modify columns), these operations can only be done if they are compatible with a *Dask DataFrame* structure. The use of *Dask* allows to make computation even on machines with a smaller RAM; to speed up the test phase, it was even possible to use a second computer with only 16 GB of RAM for the computation.

¹from: docs.dask.org/en/stable/faq.html

²A *Dask DataFrame* is a large parallel *DataFrame* composed of many smaller *Pandas DataFrames*, split along the index. These *Pandas DataFrames* may live on disk for larger-than-memory computing on a single machine, or on many different machines in a cluster. One *Dask DataFrame* operation triggers many operations on the constituent *Pandas DataFrames*. *Pandas* is another open-source python library and a *Pandas DataFrame* is a two-dimensional, size-mutable, potentially heterogeneous tabular data structure.

5.2.4 Model selection

To perform ML computation, the open-source python library used was *scikit-learn* [45]. *Scikit-learn* is an open-source ML library that supports supervised and unsupervised learning. It also provides various tools for model fitting, data preprocessing, model selection, model evaluation, and many other utilities.

Among the many supervised ML algorithms implemented in the library, four of them were selected and tested with the Sabbioneta dataset to find the most appropriate. The most appropriate is intended as the one that better balances computational cost and accuracy in the prediction. First of all several tests were made on every single track (from A to J) and then a cropped portion of the whole dataset was prepared for those tests. The points within those datasets had several geometric features; feature selection was performed later only for the selected classifier (see next section). The selected classifiers were Gaussian Naive Bayes (GaussianNB)³, K-nearest Neighbour Classifier (KNeighborsClassifier)⁴, Random Forest Classifier (RandomForestClassifier)⁵, AdaBoost classifier (AdaBoostClassifier)⁶. Additionally, also another python library that implements ML algorithms under the Gradient Boosting framework was tested: XGBoost⁷.

Dataset		Gauss NB	K-Neigh.	Random Forest	Ada Boost	XG Boost
Average all-Tracks	accuracy	78%	94%	99%	96%	99%
Train-Test (0.5%-0.5%)	time	2.31 sec	1.9 min	4.5 min	1.7 min	5.1 min
Cropped	accuracy	37%	89%	96%	48%	97%
Train-Test (0.5%-0.5%)	time	3.97 sec	1.5 h	5.6 min	1.7 min	5.8 min
	t (s/1000 pts)	0.007	10.093	0.591	0.179	0.618
Cropped	accuracy	35%	–	98%	48%	98%
Train-Test (20%-70%)	time	8,6 min	–	6.8 h	2.87 h	10.35 h
	t (s/1000 pts)	0.005	–	0.217	0.091	0.327

Table 5.3: Results of the tests made for the definition of the most suitable ML classifier. The selected one after the test is the RF.

From each dataset, a stratified sample of 0.5% of the points was drawn for training (i.e. fitting the models) and another nonoverlapping, stratified sample of 0.5% of the points was drawn for validating (i.e. estimating the generalization error). Using only 1% of the data speed up computation significantly. Then, only for the cropped dataset,

³https://scikit-learn.org/stable/modules/generated/sklearn.naive_bayes.GaussianNB.html

⁴<https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.KNeighborsClassifier.html>

⁵<https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html>

⁶<https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.AdaBoostClassifier.html>

⁷<https://xgboost.readthedocs.io/en/stable/index.html>

a test was conducted with 20% for training and 80% for testing. The tests were made only for prediction on the "paving material" classes. The results are provided in Table 5.3, where the accuracy is the ratio between correctly predicted points over the total.

It emerges from the results that Random Forest (RF) is the most accurate approach while its computational cost looks affordable. Even if its accuracy is comparable with XGBoost, the computational time seems to be much lower, and for this reason, RF was selected as the classifier for the tests.

5.2.5 Feature selection

The features for the Sabbioneta dataset consist of some geometric features computed with CloudCompare, some intrinsic features that come from the acquisition instrument, and some features that provide contextual information, computed by a specifically written python script.

The geometric features were computed with 3 different radii for the neighbourhood selection (0.05 m, 0.08 m, 1.00 m), to have more variability in the surface considered for feature computations and consequently a more representativity of the features. The computed geometric features were all the available provided by the computation tool of the software CloudCompare, here listed: Roughness, 1st order moment, Mean curvature, Gaussian curvature, Normal change rate, Number of neighbours, Surface density, Volume density, Eigenvalues sum, Omnivariance, Eigenentropy, Anisotropy, Planarity, Linearity, PCA1, PCA2, Surface variation, Sphericity, Verticality, 1st eigenvalue, 2nd eigenvalue, 3rd eigenvalue.

Intrinsic features are the Intensity, the point colour values (R, G, B), and the Normals (N_x , N_y , N_z). To give more meaningfulness to those features and also to guarantee that features have similar ranges, the Intensity was normalized. Considering that the point cloud was originally exported from Leica Cyclone, the Intensity values were divided by 2048, so that the new range was $[-1;+1]$. With the same purpose, also the colours were transformed from RGB to HSV space. Then from HSV, keeping only Hue (H) and Saturation (S), the description of colours is less sensitive to brightness (as clearly visible in Fig. 5.6) and can be less misleading than RGB in shadowed areas of the city, where the point colour is for sure affected by less brightness.

The features related to the context, and specifically computed for this purpose, are the relative distance and the relative elevation, and they were already used and described previously in this work (chapter 4). The relative distance ($d\text{-rel}$) is the orthogonal distance of a point with respect to the closer segment of the trajectory, divided by the maximum local distance, so that this value gives information on the proportion of the road environment and the relative position of each point of the point cloud within the shape of the road. The relative elevation ($Z\text{-rel}$) is the difference in Z between the selected point and the centre of the road (identified as the Z of the closer trajectory segment). The purpose of those two contextual features is to help the algorithm in finding rules that connect specific characteristics of points that describe their relation with the centre of the road and the trajectory data.

Intending to improve ML performances, it is important to carefully select the features that the algorithm uses to learn from data. The objective of feature selection is to remove irrelevant and/or redundant features and retain only relevant features. The benefits of feature selection are multifold [158]: it helps improve ML in terms of pre-

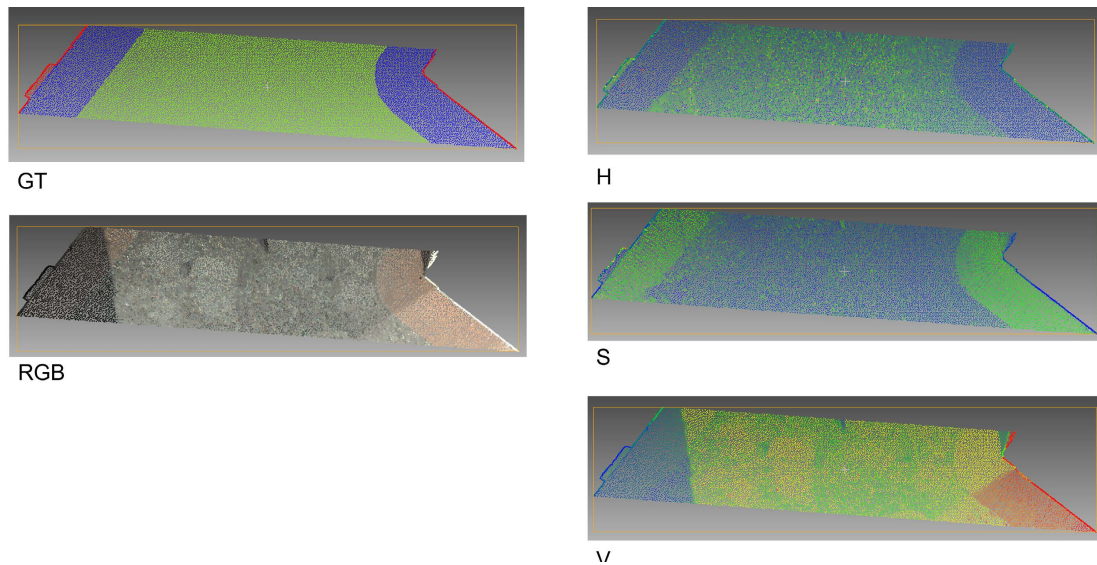


Figure 5.6: A portion of point cloud with points rendered according to different features. The difference between HSV and RGB colours is visible, and in particular, it is clear that the V-component is the most influenced by brightness and shadows. H and S (but not V) are better features for prediction than RGB.

dictive accuracy, comprehensibility, learning efficiency, compact models, and effective data collection.

To achieve this goal it is important to look carefully at all the features, think about their meaning and decide which to remove. For sure the XYZ coordinate of points are not helpful features and can create a bias, in fact, they will have a strong importance in driving the algorithm towards a result. As an example, if the training dataset has one class that always has small XYZ values, the predictions for points with similar smaller XYZ values could be labelled with that class only because XYZ are very similar and disregard other features.

Other features that should be removed a-priori are the ones related to the spatial distribution of points, such as the Number of Neighbours, Surface and Volume Density, as well as the eigenvalues-related features (1st eigenvalue, 2nd eigenvalue, etc.). Also in this case these features could be a bias for the prediction, imagine two clusters of points that, for some reason, have a low density because the instrument was not able to acquire much data (e.g., a car was passing), if those features are used, the prediction could be wrongly influenced by them.

To perform the Feature Selection, there is also a tool that can visually show the influence of a feature in the prediction, allowing the user to select only the ones that have greater influence. To use this tool it is important to have already selected the classifier. As previously written, the model selected is a RF, so the Feature Importance graph was based on a RF classifier. It is capable of describing the underlying impact of features relative to each other. The Feature importance computation was performed on the basis of feature permutation⁸. Permutation importance⁹ does not reflect the

⁸scikit-learn.org/stable/auto_examples/ensemble/plot_forest_importances.html

⁹scikit-learn.org/stable/modules/permutation_importance.html

intrinsic predictive value of a feature by itself but how important this feature is for a particular model.

As will be explained in the next section 5.3, several tests have been carried out, with different ML models, and in particular different combinations of classes have been studied, both for the "paving materials" and for the "urban elements". The most effective ML models were those with 4 classes for "paving materials" and 2 classes for "urban elements". For those ML models, the training dataset was used to produce a graph of feature importance, considering all the features previously described. Figure 5.7 and 5.8 show the two diagrams; the blue bars are the feature importance of the RF model, along with their inter-trees variability represented by the error bars. The feature selected were the ones with a higher rating, avoiding surface-based features, and selecting the one with a bigger radius (for neighbour-based features) when two are similar (i.e. select Omnivariance 0.1 instead of Omnivariance 0.8) even if they have both high importance ratings.

After the evaluation provided by the Feature Importance diagrams, also some tests were made by making predictions with RF classifier by including and excluding specific features, and comparing time-by-time the accuracy of the result. In the Result section, those several tests with a different set of features are presented, together with the point clouds with predicted labels.

Finally, the selected features for "urban elements" ML model and for "paving materials" ML model are reported in Table 5.4.

ML model	Number of classes	Selected Features
urban elements	7	Intensity, d-rel, Z-rel, Roughness (0.1), Omnivariance (0.1), H, S, Z, Normal change rate (0.1), Anisotropy (0.1), Sphericity (0.1)
paving materials	4	Intensity, d-rel, Z-rel, Roughness (0.1), Omnivariance (0.1), H, S, Z, Normal change rate (0.1), Anisotropy (0.1), Sphericity (0.1), Verticality (0.1)

Table 5.4: Selected features for the two ML models subsequently applied in the method: for paving materials and for urban elements. They were defined through a feature selection approach.

5.2. Preliminary phases

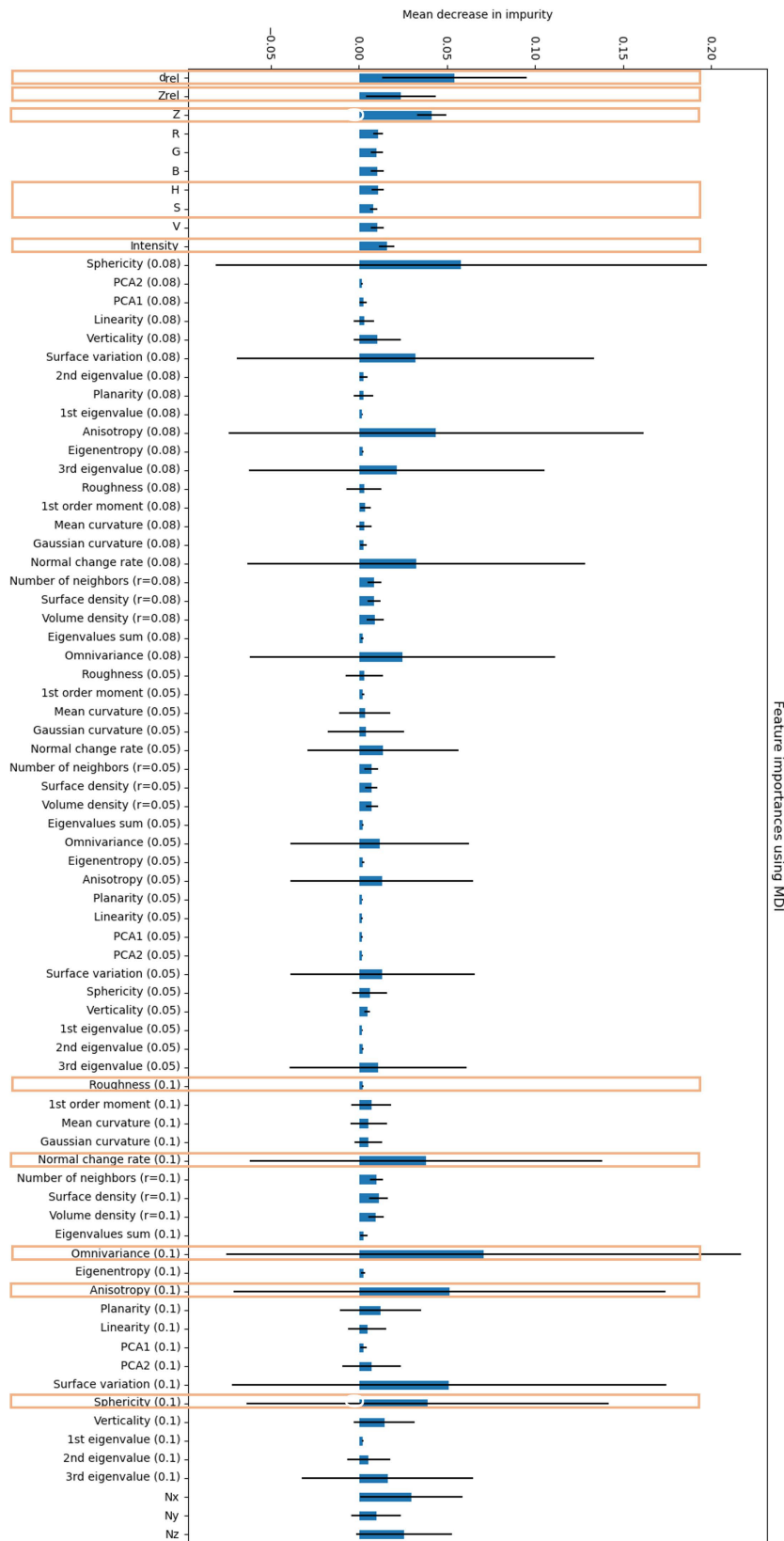


Figure 5.7: Features importance diagram computed for 2 classes: 'sidewalk' and 'road'. In orange the selected features for the classifier.

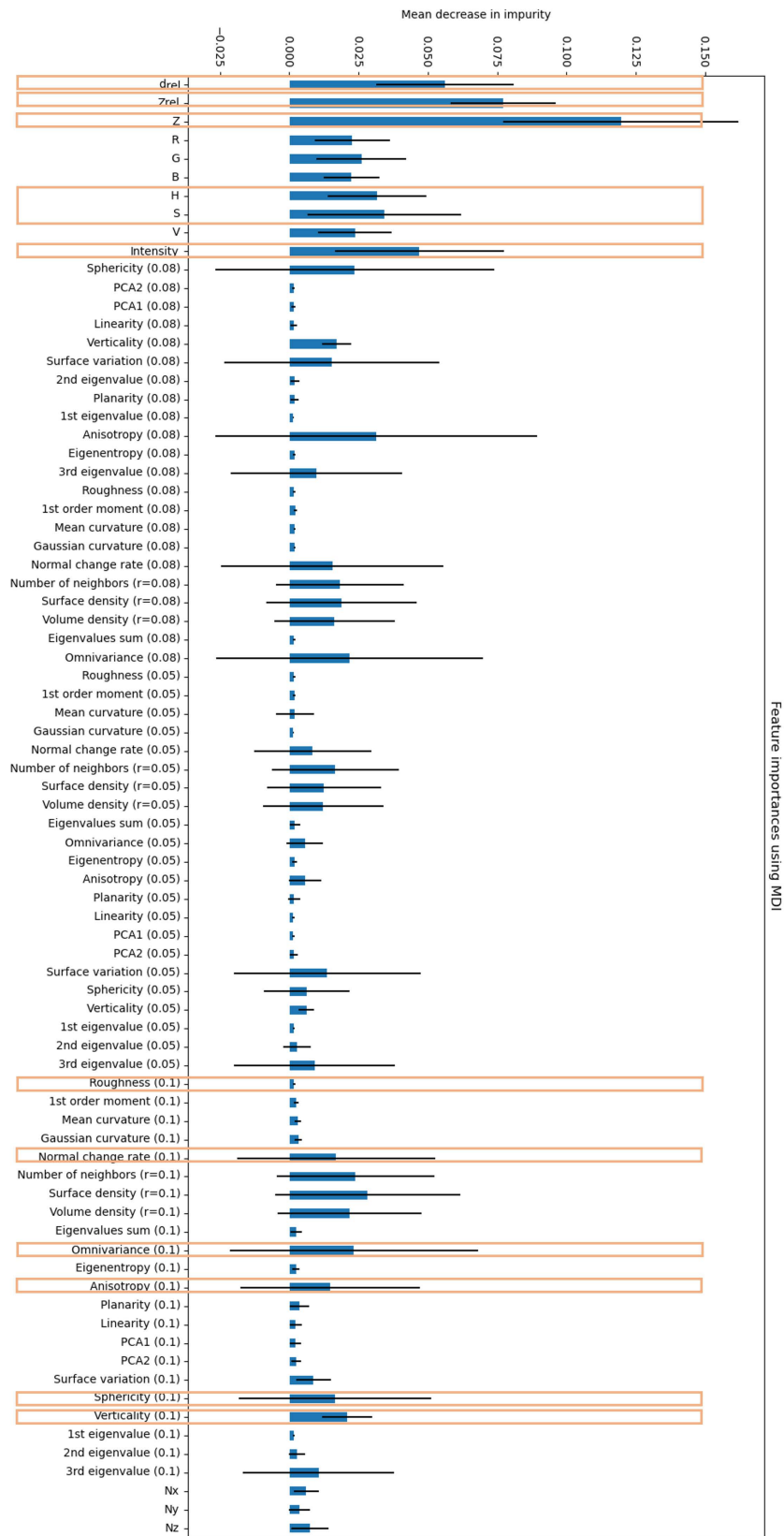


Figure 5.8: Features importance diagram computed for 12 classes: materials. In orange the selected features for the classifier.

5.3 Test phases

In this section, the tests made with the ML approach are described and the results are reported. For the purpose of making the discussion clearer, Table 5.5 summarises all the tests made, grouping them by their purpose and providing the reference tables and sections in which they are described.

The aims of the many tests done were manifold. Some of them were focused on the definition of the most appropriate ML classifier; some others were devoted to defining the importance of point cloud density and the selection of training dataset on the classification results.

After those preliminary tests, two ML models were developed, a first one for the classification of "urban elements" and a second one for the "paving materials".

The first model presented here was trained for the recognition of only two classes relative to the "urban elements", 'sidewalk' and 'road'. This model provided satisfactory results on the Sabbioneta dataset and was also tested on other datasets from some other cities.

The second ML model presented here is the one for the classification of "paving materials". The initial strategy was to test the method for the prediction of 12 classes of pavement material. Due to unsatisfactory results, a change of approach was necessary and it was later decided to use a simplified version of the model for a slightly different purpose, not the segmentation of the point cloud, but the recognition of only the materials most present on previously selected sidewalks; this model was trained only on the 4 classes of materials most present in the city.

Most of the tests here presented are based on Track-C. This track was also selected for the calibration and development of the method presented in chapter 4. Track-C is highly representative of the whole city because it has all the predominant flooring types.

Tests number	Number of classes	Table	Section	Purpose of the test
0a, 0b, 0c	12	5.3	5.2.4, 5.3.1.1	Define the most appropriate classifier
1 - 21	12	5.5	5.3.1.2	Define the influence of subsampling
22 - 38	12	5.5	5.3.1.2	Define procedures for using <i>Dask</i> on whole dataset; test the RF model; see influence of different features; define best features for 12 classes
39 c/r - 45 r	4	5.6	5.3.1.3, 5.2.5	Define best features for 4 classes; test the RF model on some datasets
46 c/r - 57 r	2	5.7	5.3.2, 5.2.5	Define best features for 2 classes ; test the RF model on all tracks and datasets

Table 5.5: Summary of all the ML tests made, their purpose, and the reference Tables and section in the text.

5.3.1 Preliminary tests

Preliminary tests to identify the most appropriate ML model were carried out using alternatively different datasets. However, all tests were performed using paving materials as training labels (12 classes). These tests were first run on each of the 10 Tracks (from A to J), selecting only 0.05% of the data for both the training and the testing. Then a test with the same stratification percentages was done with a dataset containing a larger portion of the city, obtained by cropping the complete Sabbioneta dataset. For this dataset, the portions of the data used for the train and test were respectively 20% and 80%. These preliminary tests were useful to identify an effective ML model.

As the tests progressed, it was noted that by varying the choice of train and test dataset, also the results changed. For example, using a dataset made by making circular crops of the cloud, it was noted that the accuracy of the predictions was lower. The (conceptual) difference between a dataset obtained by selecting a percentage of data and one made by making circular crops has already been discussed above. However, it was noted that, even under these conditions, the model with the best cost-benefit ratio remained the RF, which was then chosen as the model to be used for all subsequent tests.

The feature selection process was useful for skimming the many features present. However, as will be seen in the following, the outcome of the feature selection was the starting point, and in any case, various RF models were trained with different features, to try and identify those that would give the best result.

As already discussed, several datasets were selected for training and testing. Specifically, in this initial phase it was decided to use a "reduced" training and testing dataset; made by making several circular crops taking care to maintain the same percentage of points for the different classes. The aim is to obtain a slim dataset that is easy and agile to implement so that conclusions can be drawn quickly. A training dataset and a test dataset were created in this way (called "cropped dataset") and were used to make a series of preliminary tests. The purpose of these tests was twofold: to identify the influence of point density and feature selection.

Regarding the first purpose, three sets of tests were carried out for a total of 21 tests, to evaluate the impact of a subsampled dataset. The first set of tests was carried out by varying the features and applying an RF model on the so-called "circular cropped" dataset using the point cloud at maximum density. The second set of tests was carried out on the dataset with the point cloud subsampled at a spatial distance of 3 cm. The third set of tests was performed on a cloud subsampled at 5 cm. In all the tests performed, the prediction accuracy always remained almost constant, while the calculation time was shorter for the subsampled datasets. The fewer points there are in the dataset, the less time is needed for the calculation.

It should be noted that the geometric features were computed before subsampling - thus at full density - so that the informative characteristics of the cloud at maximum density are maintained even when the dataset is subsampled. So, subsampling reduces the dataset to be analysed, but the lower density of points does not affect the final result. From here on it was decided to proceed with a subsampled dataset at 0.03 m. The results of the first 21 tests are shown in Table 5.6.

Test	Training dataset	test dataset	Average Accuracy	Average time
1-7	circular crops, full density	circular crops, full density	0.8	2 h
8-14	circular crops, density = 5cm	circular crops, density = 5cm	0.8	2 h
15-21	circular crops, density = 3cm	circular crops, density = 3cm	0.8	2 h
22-33	circular crops, density = 3cm	Track-C, density = 3cm	35% - 45%	7 h
34-38	circular crops, density = 3cm	All city, density = 3cm	42% - 61%	11 h

Table 5.6: Results of the preliminary tests, made with different densities and different training datasets.

5.3.2 First approach: ML model for the classification of urban elements

The aim of this first ML model tested was to predict two urban elements classes: 'sidewalk' and 'road'. The first tests were focused on Track-C using two training datasets: circular crops (dataset-c), and stratified percentages (dataset-r). From the very beginning, the results of those tests showed high accuracy and were very promising, and in fact, it was then decided to apply the same RF trained model to all tracks.

Some first tests were made to define the best features to use for the training (that are the ones already previously presented in Figure 5.7), and in all those cases the accuracy is greater than 81% (test 49-c) with a maximum of 97.4% (test 48-r). In light of the very positive results in terms of accuracy, it was therefore decided to test the model trained on the dataset-r (from Track-C) on all the other Tracks. The results are provided in Table 5.7, and apart from Track-B, where the accuracy is low (56.85%), all the other tracks have accuracy in the range 86.2% - 92.3%. Regarding Track-B, as can be seen in Table 5.2, the path followed when acquiring this Track was almost outside the city, so there are no sidewalks in the majority of the track, and the urban ground is mostly asphalt or grass, so it is very different from the training dataset, which is based on a Track inside the historical city urban environment. For this same reason, Track-J was not analysed, this track follows a rural road outside the city and runs along the fortified walls. Similarly, also Track-G was excluded, because it was referred to a very short road behind the *Galleria degli Antichi* building, and it is all made of gravel flooring, without any sidewalk.

From those tests, the confusion matrices were computed and plotted for each Track together with a render of the point clouds with points coloured depending on the predicted class, and where red is used for prediction errors. The Figures, together with confusion matrices and performance metrics are reported in Appendix A, from page 207 Starting from the confusion matrix, it was then possible to compute the performance metrics, which are summarized in Table 5.8, from which it is possible to see that the average values of Precision, Recall and F1-score are comparable and higher with respect the previous segmentation method, presented in chapter 4. Finally, Figure 5.11, at the end of this chapter, shows a render of the point cloud with the two predicted classes and in red the erroneous points.

Test	Training dataset	test dataset	Features	Accuracy
46-c	dataset-c from Track-C	Track-C	roughness, omnivariance, d-rel,	85.6%
46-r	dataset-r from Track-C	Track-C	intensity, Z	95.5%
47-c	dataset-c from Track-C	Track-C	roughness, omnivariance, d-rel,	88.0%
47-r	dataset-r from Track-C	Track-C	intensity, Z-rel, H, S	95.2%
48-c	dataset-c from Track-C	Track-C	Intensity, d-rel, Z-rel, Roughness	87.4%
48-r	dataset-r from Track-C	Track-C	(0.1), Omnivariance (0.1), H, S, Z, Normal change rate (0.1), Anisotropy (0.1), Sphericity (0.1)	97.4%
49-c	dataset-c from Track-C	Track-E	roughness, omnivariance, d-rel,	81.1%
49-r	dataset-r from Track-C	Track-E	intensity, Z-rel, H, S	88.5%
50-c	dataset-c from Track-C	Track-I	roughness, omnivariance, d-rel,	84.7%
50-r	dataset-r from Track-C	Track-I	intensity, Z-rel, H, S	91.0%
51-r	dataset-r from Track-C	Track-E		89.16%
52-r	dataset-r from Track-C	Track-I		91.4%
53-r	dataset-r from Track-C	Track-A	Intensity, d-rel, Z-rel, Roughness	86.2%
54-r	dataset-r from Track-C	Track-B	(0.1), Omnivariance (0.1), H, S, Normal change rate (0.1),	56.5%
55-r	dataset-r from Track-C	Track-D	Anisotropy (0.1), Sphericity (0.1)	87.05%
56-r	dataset-r from Track-C	Track-F		92.3%
57-r	dataset-r from Track-C	Track-H		90.4%

Table 5.7: Results of the tests made using only two classes for prediction and focusing on urban elements.

Test	Track	Accuracy	'sidewalk' class			'road' class		
			Precision	Recall	F1-score	Precision	Recall	F1-score
53-r	A	86.2%	0.70	0.88	0.78	0.90	0.75	0.82
48-r	C	97.4%	0.9	0.9	0.9	0.9	0.9	0.9
55-r	D	87.0%	0.7	0.88	0.78	0.9	0.75	0.82
51-r	E	82.4%	0.8	0.89	0.84	0.9	0.82	0.86
56-r	F	92.3%	0.9	0.9	0.9	0.9	0.9	0.9
57-r	H	90.4%	0.8	0.89	0.84	0.90	0.82	0.86
52-r	I	81.6%	0.8	0.89	0.84	0.90	0.82	0.86
AVERAGE		88.2%	0.80	0.89	0.84	0.90	0.82	0.86

Table 5.8: Performance metrics for the ML prediction of 2 classes ('road' and 'sidewalk') for the Tracks.

Scalability

Given the high accuracy results obtained for the Sabbioneta dataset, exploiting this second ML model, it was decided to test its scalability and flexibility, by applying it to classify other datasets. For consistency with the entire work, the selected datasets were the ones already used to test the scalability of the knowledge-based method

(sections 4.6.2, 4.6.3, and 4.6.4). Therefore, with this aim, the ML model was used to predict 'road' and 'sidewalk' classes on Mantova, Sacro Monte, Porto (1, 2, 3) datasets.

Specifically, two types of tests were made: firstly using (as for the previous cases) Track-C as training dataset, secondly choosing for training one portion of each dataset at a time. The results of both cases are reported in Table 5.9 in terms of the overall accuracy of the prediction, the same table also provides the performance metrics for all the datasets. Clearly, the ML model trained using a portion of the dataset itself performed better, but on average only a few percentage points higher than the ML trained on Track-C. These results prove the applicability of the Track-C as a good representative for the training and the subsequent prediction of several urban layouts around the world. Specifically, it behaved sufficiently well not only on datasets coming from similar urban layouts, but it worked also on Porto, where the road layout is quite different.

Comparing these results to the ones previously obtained with the knowledge-based method, it is possible to say that ML approach has greater accuracy, even if it is still improvable. This result proves the efficiency and the flexibility of the ML method, which should be the preferred method as starting point for further research works.

As for the previous tests, the confusion matrices, performance metrics and renders of the point clouds are provided in Appendix A, at the end of this book.

Dataset	Training	Acc.	'sidewalk' class			'road' class		
			P	R	F1	P	R	F1
Mantova	Track-C	92%	1.00	0.91	0.95	0.90	1.00	0.95
	1/3 Mantova	92%	0.80	1.00	0.89	1.00	0.83	0.91
Sacro Monte	Track-C	93%	0.30	1.00	0.45	1.00	0.13	0.22
	1/3 Sacro M.	89%	0.60	0.86	0.71	0.90	0.69	0.78
Porto-1	Track-C	88%	0.70	0.88	0.78	0.90	0.75	0.82
	1/3 Porto-1	97%	1.00	1.00	1.00	1.00	1.00	1.00
Porto-2	Track-C	80%	0.60	0.75	0.67	0.80	0.67	0.73
	1/3 Porto-2	85%	0.50	0.36	0.42	0.91	0.95	0.93
Porto-3	Track-C	84%	0.50	0.83	0.63	0.90	0.64	0.75
	1/3 Porto-3	84%	0.50	0.83	0.63	0.90	0.64	0.75
Average	Track-C	86%	0.62	0.87	0.70	0.90	0.64	0.69
	1/3 dataset	88%	0.68	0.81	0.73	0.94	0.82	0.87

Table 5.9: Accuracy in the prediction of the tests made using the ML model for the recognition of urban classes ('road' and 'sidewalk') on other datasets, out of Sabbioneta. The same datasets were used to test the scalability of the knowledge-based method.

5.3.3 Second approach: ML model for the classification of paving materials

Tests with 12 classes

The second ML model tested, was firstly conceived for the prediction of all the 12 paving materials identified within the city. Some first tests were carried out using as training dataset the one obtained by performing circular crops (dataset-c) and as test dataset Track-C. The tests on this training/testing combination were 11 in total, varying the features and calculation parameters. The results of these preliminary tests are shown in Table 5.6 presented at page 129. Low accuracy was reached by these tests, in a range between 35% and 45% (tests 22-33). Extending the test to the whole city, using as testing dataset the one composed by the merge of all the tracks, several tests were carried out, reaching a maximum accuracy of 61% (tests 34-38). This procedure did not seem to give good results, so a change of approach was necessary.

Tests with 4 classes

A first attempt was to reduce the number of classes to be predicted. This comes from the fact that inside the city, the majority of points pertain to one of the first four classes. For this reason several tests were made training the RF model only with the first 4 classes (1 = 'sampietrini', 2 = 'bricks', 3 = 'cobblestone', 4 = 'stone'). The tests were performed on Track-C, using two types of training datasets: one obtained through circular crops on Track-C and one selecting a percentage (20% stratified through classes) of the points of Track-C. These two training datasets were subsequently called dataset-r (when selecting random points based on a percentage of the whole), and dataset-c (when selecting points using circular crops).

By varying the features used to train the model, and validating it on the whole Track-C, the accuracy changed between the dataset-r and the dataset-c, specifically in the first case the range was 86% - 92%, while for the second case 71% - 83%. The features that provided the higher accuracy are the ones also previously presented in Figure 5.8. A summary of those tests is provided in Table 5.10. The accuracy is a little lower when the model is trained using circular crops, but in both cases, it is higher than the accuracy previously obtained when considering 12 classes.

Now, since the accuracy was better, the next tests were done still considering 4 classes, and still training the models on a portion of Track-C, but validating (or testing) the model on two different tracks: Track-E and Track-I. In both cases, the accuracy of the prediction is much lower than the previous tests (in ranges 63% - 66% for dataset-r and 62%-69% for dataset-c). A summary of those tests is provided in Table 5.10.

The results from those preliminary tests show the ML approach can be a possible way to identify the paving materials, but the ML model and the training dataset need to be better calibrated and prepared to get satisfactory results. More studies will be done in the future trying to improve this approach to segment the dataset using all the 12 pavings present in the city. On the other hand, by looking at the point cloud, and analysing it visually (e.g., see Fig. 5.9), the erroneous points are more or less scattered around the point cloud and not focused on specific areas; it is also possible

Test	Training dataset	test dataset	Features	Accuracy
39-c	dataset-c from Track-C	Track-C	roughness, omnivariance, d-rel,	72.7%
39-r	dataset-r from Track-C	Track-C	intensity, Z	87.3%
40-c	dataset-c from Track-C	Track-C	roughness, omnivariance, d-rel,	73.2%
40-r	dataset-r from Track-C	Track-C	intensity, Z-rel, H, S	84.9%
41-c	dataset-c from Track-C	Track-C	Intensity , d-rel , Z-rel , Roughness	83.09%
41-r	dataset-r from Track-C	Track-C	(0.1) , Omnivariance (0.1) , H , S , Z , Normal change rate (0.1) , Anisotropy (0.1) , Sphericity (0.1) , Verticality (0.1)	92.04%
42-c	dataset-c from Track-C	Track-E	roughness, omnivariance, d-rel,	62.0%
42-r	dataset-r from Track-C	Track-E	intensity, Z-rel, H, S	63.9%
43-c	dataset-c from Track-C	Track-I	roughness, omnivariance, d-rel,	66.1%
43-r	dataset-r from Track-C	Track-I	intensity, Z-rel, H, S	69.1%
44-r	dataset-r from Track-C	Track-E	Intensity, d-rel, Z-rel, Roughness (0.1), Omnivariance (0.1), H, S, Z, Normal change rate (0.1), Anisotropy (0.1), Sphericity (0.1), Verticality (0.1)	43.1%
45-r	dataset-r from Track-C	Track-I	Intensity, d-rel, Z-rel, Roughness (0.1), Omnivariance (0.1), H, S, Z, Normal change rate (0.1), Anisotropy (0.1), Sphericity (0.1), Verticality (0.1)	40.5%

Table 5.10: Result of the tests made using only 4 classes for the training dataset.

to see that the majority of the 'sidewalks' and 'roads' areas were correctly predicted. In other words, the majority of the points on a specific ground surface were correctly predicted, with some minor erroneous points scattered on it (see the zoomed area in Fig 5.9).

A different point of view on these results led to a change of approach: even if the accuracy of the model trained with 4 classes is not satisfactory to segment the whole city, it can still be used differently. Assuming that the semantic segmentation has already been carried out previously (dividing into 'sidewalks' and 'roads'), it is, therefore, possible to select only the cluster of points related to the sidewalk.

Indeed, it is possible to operate by selecting small portions of the sidewalk (for example operating on one ROI at a time), applying the RF model to predict the classes of the sidewalk points only, and then assigning to the whole pavement the class most present in the cluster. In this way, it is possible to assume that the flooring material of the sidewalk can be assigned with a good approximation. More about that in chapter 6 at section 6.2.2.

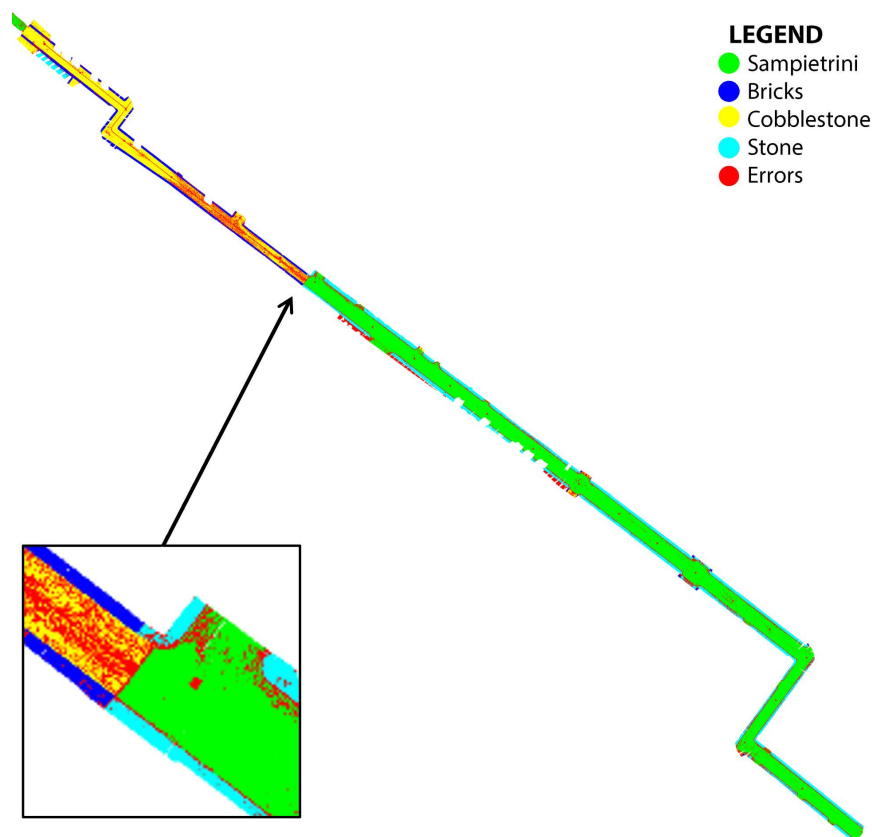


Figure 5.9: Render of the point cloud (Track-C) after prediction, for test 39-c (see Tab. 5.10), accuracy=72.7%. Note that even if accuracy is low, the points on the surface of sidewalks were, for the majority, predicted correctly.

5.4 Discussion and comparison between knowledge-based and ML methods

In this chapter, the tests aimed at verifying an ML approach for the segmentation of the ground surface of the dataset of Sabbioneta have been described. The purpose was the labelling of the points according to the pavement material. After having identified the RF as the most appropriate ML model, tests focused on the prediction of 12 classes were initially developed. Given the poor results, in terms of accuracy, it was decided to test the approach by attempting to predict a smaller number of classes, only those related to the most present materials (4 classes), again the results were not satisfactory. However, it was noted that by selecting a specific area (e.g. a piece of sidewalk), the majority of points in the area had a correct prediction. Because of this, it was decided to change the approach and to use the ML model for the prediction of the material most present within an area, and then to use this model only after having segmented the point cloud into 'sidewalk' and 'road'.

A second test was then developed attempting to use ML for 'sidewalk' and 'road' point cloud segmentation, i.e. an alternative semantic segmentation approach to the one presented in chapter 4. From the tests carried out it was possible to show that the approach works and also has results, in terms of accuracy, that are greater than

5.4. Discussion and comparison between knowledge-based and ML methods

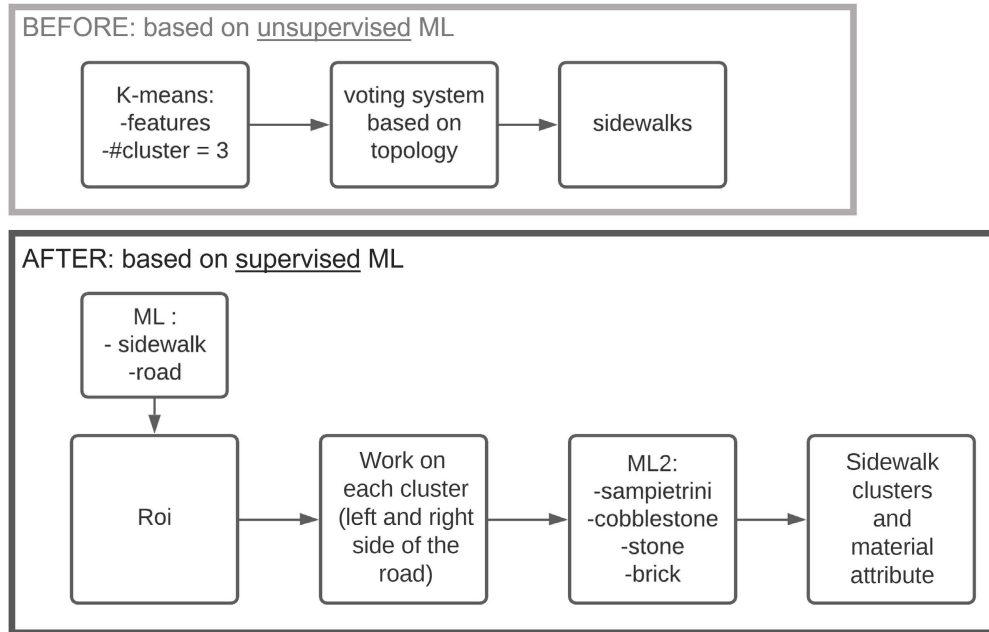


Figure 5.10: Scheme of the updated approach for segmentation, exploiting ML methods.

the knowledge-based method previously presented. This ML model was also used for the classification of some other datasets of some other cities, with good and promising results.

In the light of the results obtained, it is, therefore, possible to exploit together both the ML models tested and propose an alternative method for the semantic segmentation of the point cloud (in 'sidewalk' and 'road') and the characterisation of the pavement material of the sidewalks. This final method is summarised in Figure 5.10.

Apart from the different accuracy (in percentage, slightly higher for the case of the ML), it is possible to identify other differences and similarities between the two methods. Regarding the development of the methods, in the case of the knowledge-based one, it was developed on the basis of a single Track, Track-C, and then extended to the whole city and other Italian cities (Mantova and Sacro Monte in Domodossola), with success. In the case of ML, the final model was trained on the basis of the same track (Track-C) and then extended to all the other tracks (i.e. to the whole city) with success. However, for Track B, which has slightly dissimilar characteristics to Track-C, the accuracy turned out to be much lower than the other tracks. Nevertheless, the ML approach is still valid because it allowed the correct identification of the whole city on the basis of only one track.

Another interesting aspect is the computing power that had to be addressed for the ML. Without the implementation of the *Dask* library, it would not have been possible to work with such a large amount of data. In the case of the knowledge-based method, indeed, the analysis took place by examining small portions of the road (called ROIs) one at a time.

A final interesting aspect to note is the similarity between the features used, although in different ways, by both methods. Although the geometric features used by the ML model are more numerous than those used by the k-means algorithm in the

knowledge-based method, they contain all the features used in the case of k-means. Furthermore, the context-related features (i.e., d-rel distance from the road centre, Z-rel relative height), which in the knowledge-based method are used downstream, to provide information on the context in the voting system, proved to be fundamental for the success of the ML model predictions.

To conclude, it is clear from the results obtained that the ML approach can be useful and represents an effective alternative to the knowledge-based method previously presented. Furthermore, by carrying out further research and refinements to the ML model based on the 12 classes, it is possible to conceive of being able to achieve greater and more satisfactory accuracy. A further interesting research opportunity could be the development of tests based on DL, by rasterizing the point clouds using different scalar fields and features and performing a learning process directly on the images.

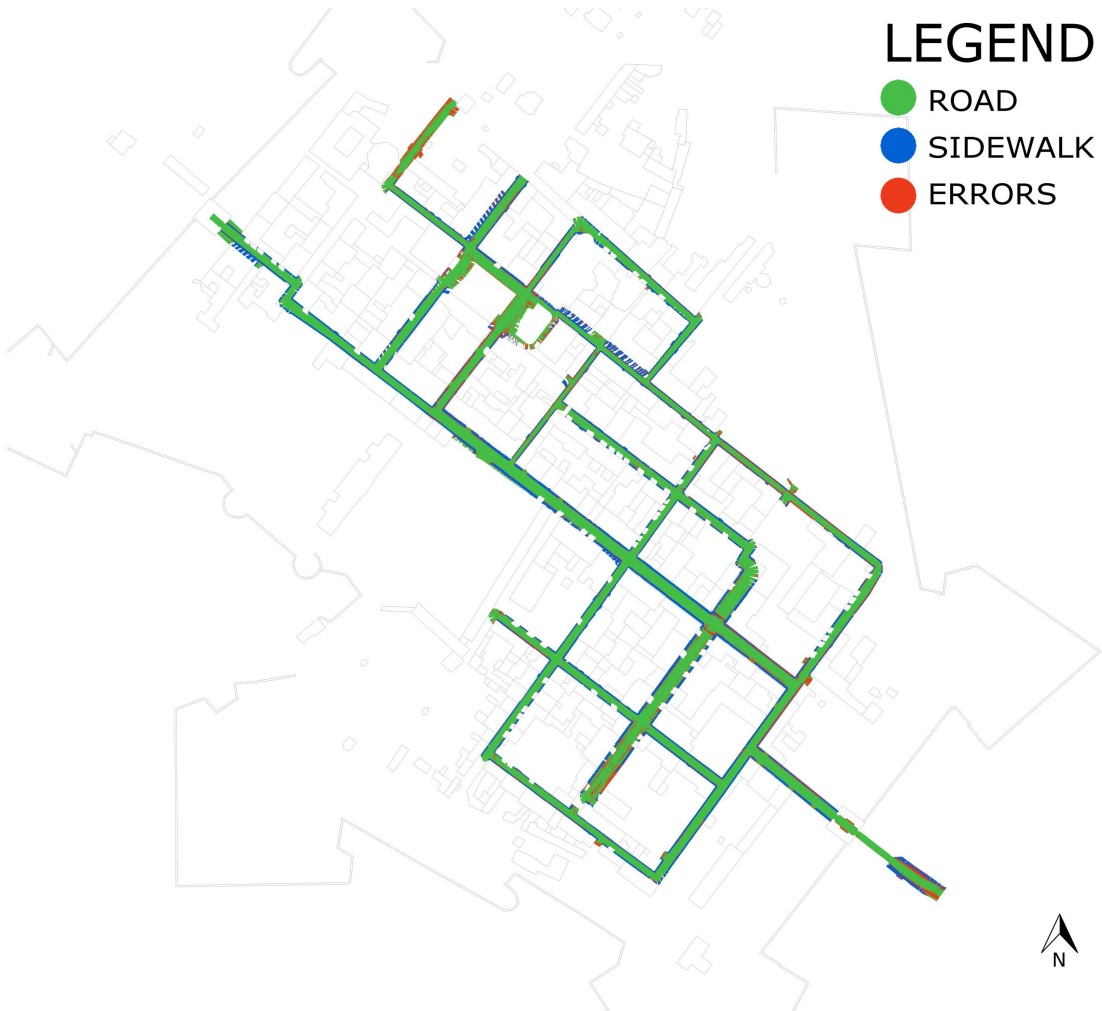


Figure 5.11: Top view of the whole Sabbioneta dataset after semantic segmentation with ML method. Red points are errors. On the top part of the image, there is a pie chart with an overview of the whole points in the point cloud. On the bottom the confusion matrix and the performance metrics.

6

Sidewalk characterization: attributes computation

The purpose of this chapter is to calculate some attributes that characterize the previously identified sidewalks. Taking advantage of the point cloud, some geometric attributes and pavement were computed, either using formulas based on cloud features or using a ML method. The results were validated by comparing them with measurements made on-site. The innovative aspect of this chapter is to have proposed and implemented a method for automated calculation of sidewalk attributes that are useful for accessibility assessments and that stand as alternatives to measurements that usually tend to be made by hand by going on-site.

The chapter is organized as follows: the first section introduces the topic; the second section describes two proposed methods for paving material characterization, one using a defined algorithm and a second one implementing ML; The third section describes the formulations implemented for the computation of other geometric attributes. The last section presents the results and a validation, comparing them with measures made on-site.

6.1 Introduction to the attributes computations

In the introduction and motivation for this research work, the importance of an inventory of all the characteristics of the urban environment that are useful for an assessment of accessibility was emphasised. A series of parameters and criteria that should be taken into account for such an assessment was also proposed and discussed (e.g., geometric features of the urban space, pavement characteristics, orientation resources, crosswalks, urban features, bus stops, and parking). In addition, the objectivity of an assessment based on a point cloud, which can then be used to extract and provide useful geometric data for decision-making, was also discussed.

It was necessary to make a choice and focus the research only on some of the parameters discussed, in this case, it was chosen to work on sidewalks, which are considered the first element of interest in the study of accessible urban routes. To this end, two methods for the identification of sidewalks were previously presented (in chapters 4 and 5). The first approach, knowledge-based, implemented k-means algorithm to segment the point clouds, so that points are grouped in clusters, then exploiting a voting system the points belonging to the surfaces of the sidewalks were classified. The second approach implemented a supervised ML method that exploited a RF model trained on a portion of the dataset, used to predict semantic labels on the entire dataset. The resulting labels assigned to points allowed the distinction between

road and sidewalk points.

This chapter aims to further characterise the sidewalks, measuring (as automatically and correctly as possible) some of the parameters useful for assessing accessibility. Specifically, these parameters are: the material of the sidewalk paving, the width of the sidewalk, its elevation in relation to the road, and the slopes in the longitudinal and transverse directions.

The points in the point cloud identified at the end of the segmentation process are the basis for these analyses. Regardless of the segmentation approach chosen, both procedures made it possible to label the points as 'sidewalk', 'road', or 'other'. Similarly to what was done for the knowledge-based segmentation (sec 4.2.3), also in this chapter, the analysis will be carried out analysing one ROI at a time and, specifically, only focusing on the points labelled as 'sidewalk'. The ROIs were selected following the trajectory of the instrument and selecting sub-clouds from the original point cloud, the selection was made using Bounding Boxes created from two-metre-long segments extracted from the trajectory line of the acquisition instrument.

6.2 Paving material detection

When assessing the accessibility of a path in an urban environment it is important to consider some features of the flooring. Primary importance is given to the non-slip properties of the material used for the paving, the coplanarity of different elements, the junction width and height, and the durability over time [8]. The detection of the flooring material, and so the type of paving, is useful to envisage the value of some of the evaluation criteria. The type of paving (i.e. the paving material) is intended in this work as the totality of the elements that, together, make up the pavement, for example, a pavement formed by bricks placed side by side with mortar joints is here considered a single element and simply called "brick" pavement.

Let's consider an example: when assessing the junction between paving elements, the max allowed junction height according to Italian law is 0.002 m. This value cannot be measured from the MLS survey because the instrument does not provide enough precision on the points and also the density of points on ground is not enough for this measure. In such a case, even if the junction cannot be measured, the identification of the paving material is useful because some of its characteristics are known a-priori. For example, considering cobblestone flooring (composed of several cobbles), it is obvious that the junction criterion is not met. Plus, it is known that this type of paving surface is not sufficiently smooth and planar for a wheelchair to move easily.

Another feasible criterion to assess is the non-slip properties. To measure this material property, some specific tests should be made, but its hazard could be estimated just from the type of paving. For stone flooring or marble flooring, it is common knowledge that on rainy days it can be a source of slipping.

These two examples are enough to understand how important it can be to define the type of pavement in an urban environment, especially when considering the need to design accessible routes for wheelchair users or pedestrians. It is therefore evident why the first attribute of interest chosen for sidewalk accessibility analyses is the type of pavement.

To reach the purpose of pavement identification, two methods are proposed, de-

scribed and tested in this chapter. The first method was based on some threshold derived from the dataset. It is also important because it was previously implemented in the voting system of the knowledge-based segmentation (section 4.4.1). The second method (section 6.2.2) exploited the ML method developed in section 5.3.3. In this case, a RF, trained on a dataset composed of the 4 classes of materials mainly present in the dataset, was used to predict the predominant material of sidewalk clusters in each ROI.

6.2.1 Method I: thresholds-based

In this approach, the input was a sample of the sidewalk cluster of which the paving material was to be identified. The strategy implemented to do that here is the same that was presented in section 4.4.1. For easy reading, this approach is also briefly described here.

The approach follows a threshold-based method. The thresholds were derived from an analysis of the probability density function distribution of values of some samples of sidewalk points. Specifically, the Intensity and the Omnivariance have proved to be very useful for the definition of thresholds that allow distinguishing between the 4 materials present in the city.

The method, firstly checks the Omnivariance mean value of the analysed sample, if it is lower than a specific threshold (THR-1) the sample is considered as 'sampietrini' or 'cobblestone'. Then, to define which is of the two pavings, the shape of the Intensity distribution curve is analysed, a narrow curve is for 'sampietrini' and a wider curve is for 'cobblestone'. The shape of the curve is retrieved by looking at the standard deviation of the Intensity value of the sample and checked with a second threshold (THR-2). If a sample has the mean Omnivariance higher than the first threshold, the sample could be 'stone' or 'bricks', to identify the correct paving material the mean value of Intensity is checked and compared with the last threshold (THR-3). For the values of the three thresholds, please refer to section 4.4.1 at page 89.

6.2.2 Method II: Machine Learning model

The input of this approach was the sidewalk cluster of points inside the ROI. Almost all the ROIs had sidewalks on both sides, anyway, it was easy to distinguish between the point of the portion of sidewalk on the left and the right. The prediction was performed on the entire sidewalk cluster of points.

Following the line of thinking presented in the previous chapter, it was possible to implement the RF model trained to predict the first 4 classes (i.e. the most present labels in the dataset) after the segmentation phase, as explained also in Figure 5.10. The RF model, features, and training dataset were already explained in the previous chapter. In this workflow, the predicted material assigned to the sidewalk portion was the most frequent class predicted among all the points of the cluster.

The steps implemented in the method involved (see Figure 6.1) the selection of the points of the sidewalk, the use of the RF model to predict the labels, the computation of the most frequent label and its assignment as paving material attribute of the analysed sidewalk.

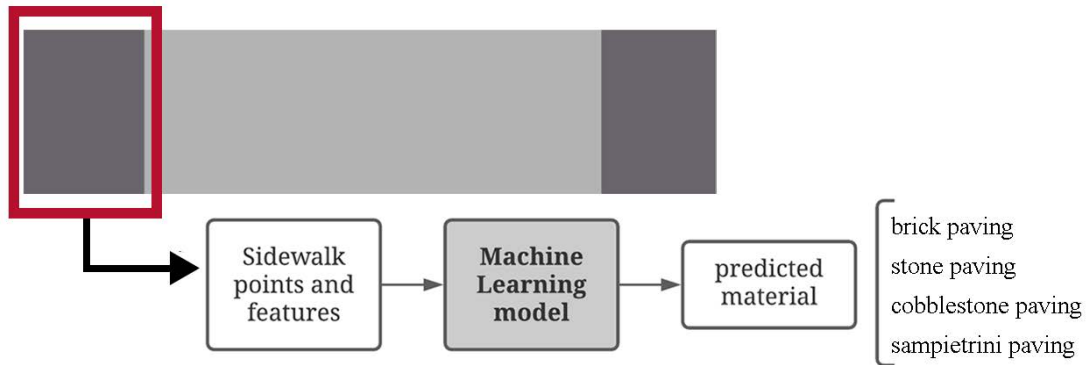


Figure 6.1: Graphic scheme of the computation of the attribute "paving material". Sidewalks in dark grey and roadway in light grey.

6.2.3 Results and comparison

The purpose of these tests was to validate the two proposed methods, to identify the most accurate, and to use it for computing the paving attribute of all sidewalks in the dataset of Sabbioneta. To properly compare the two methods, and thus base the final considerations on data with no intrinsic errors, it was decided to use the segmented point cloud with GT to divide the ROIs into sidewalk and road. Since sidewalk clusters come from the GT, these two tests were independent of errors in semantic labels. The most successful method could then be used and applied for the detection of materials on a point cloud semantically segmented exploiting one of the methods presented in previous chapters.

For each ROI, the semantic labels from the GT were used to create separate clusters: road, sidewalk left, and sidewalk right. For Method I, a sample of points in 1 square meter was selected from the centre of each semantic cluster, and then the process implemented was the one described in the previous section. For Method II, instead, the whole cluster was used for the analysis, and the prediction method was the one previously described. The processing time was, on average, 7 seconds for each ROI for Method I and an average of 4 seconds for each ROI for Method II. The results were compared with a "paving material GT" and the accuracies of predictions for both methods are presented in Table 6.1 and 6.2. The average accuracy of Method II (76.9%) is almost 10 percentage points higher than of Method I (68.1%). The two resulting datasets, represented by a sequence of single points for each ROI, coloured according to the computed paving material, are shown in Figure 6.2 providing both methods and the reference GT.

6.2.4 Final considerations

The two methods tested proved to be efficient for the identification of sidewalk pavement material. Although the method using ML gave a higher overall accuracy, it could be observed that both methods had greatly varying accuracy between different tracks, but the overall accuracies were very close and differ by only a few percentage points.

Considering Method I, thresholds-based, it depended on some parameters, which

were defined based on a small sample of the point cloud. The usefulness of the defined thresholds was proved by applying them to the whole dataset, but even if the results were promising, a better definition of this method could be studied, mainly relating it to statistical reasoning on data distribution instead of using thresholds. For this reason, Method II, which was based on the ML approach seems to be more promising, and more appropriate to reach the purpose. Further effort should be made for its improvement, by selecting a more refined training dataset, performing more specific feature selection and tuning the RF parameters.

Track	N. of analysed segments	Correctly classified	Wrongly classified	Accuracy
A	457	263	194	57.5%
B	147	74	73	50.3%
C	553	420	133	75.9%
D	717	445	272	62.1%
E	607	464	143	76.4%
F	164	148	16	90.2%
H	150	118	32	78.7%
I	128	60	68	46.9%
Average	2923	1992	931	68.1%

Table 6.1: Results of the computation of paving materials for each Track, using Method I (thresholds-based). The predicted pavings for each segment of sidewalk analysed are compared to the GT.

Track	N. of analysed segments	Correctly classified	Wrongly classified	Accuracy
A	458	358	100	78.2%
B	147	99	48	67.3%
C	553	530	23	95.8%
D	717	513	204	71.5%
E	607	448	159	73.8%
F	164	136	28	82.9%
H	150	107	43	71.3%
I	128	58	70	45.3%
Average	2924	2249	675	76.9%

Table 6.2: Results of the computation of paving materials for each Track, using Method II (ML). The predicted pavings for each segment of sidewalk analysed are compared to the GT.

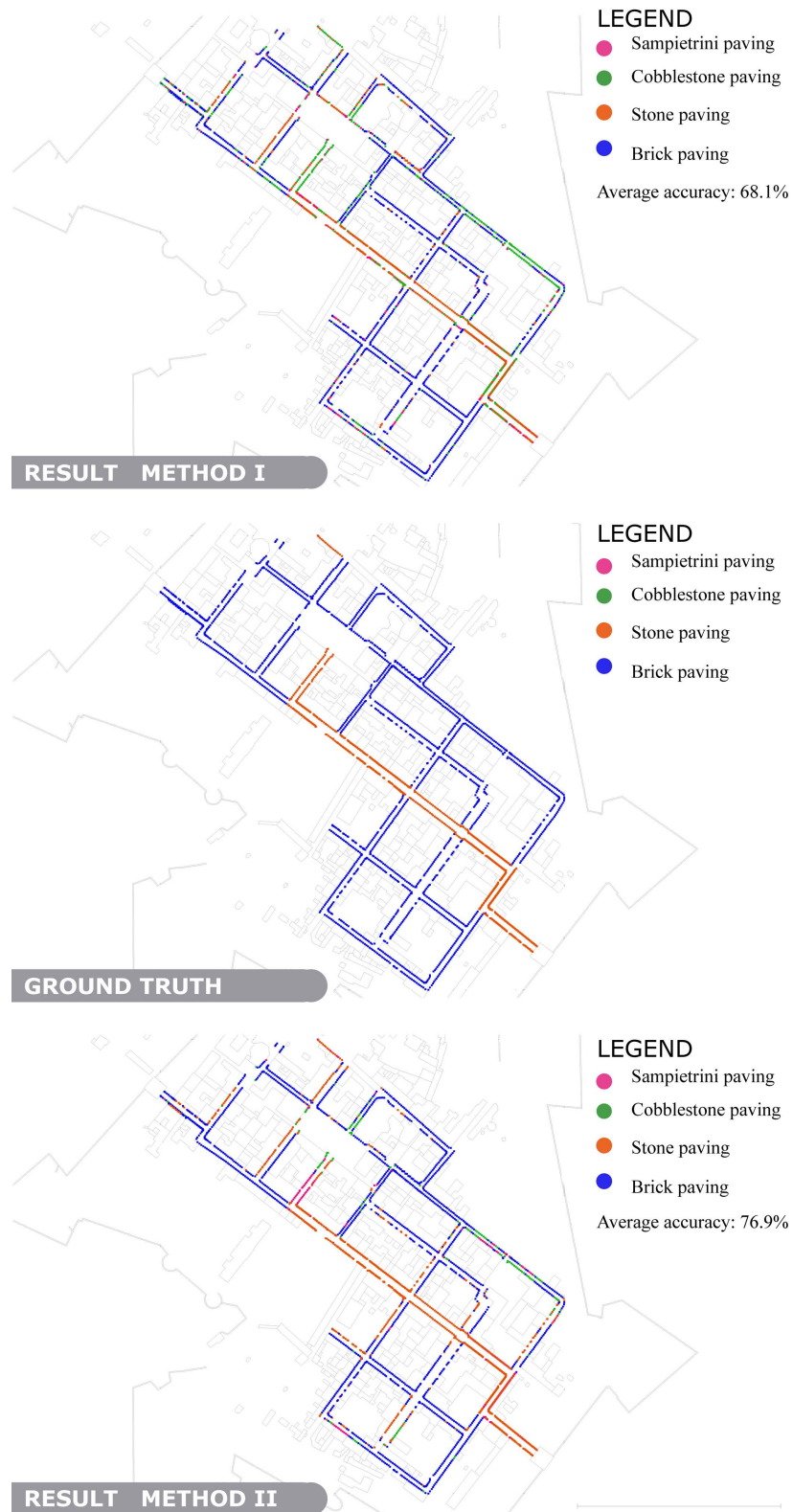


Figure 6.2: Visual result of the paving material computation with both methods. The two resulting datasets, represented by a sequence of single points for each ROI (one point for each sidewalk segment), coloured according to the computed paving material. In the middle, the GT is provided to easily compare the results.

6.3 Other geometric attributes

In addition to the material of the paving, other elements of great interest for the analysis of accessibility are some dimensional parameters (i.e., geometric attributes), such as the width of the pavement, its height in relation to the road and its slopes in the direction parallel and transverse to the direction of travel.

6.3.1 Sidewalk width

Since the trajectory segment present in each ROI is almost parallel to the sidewalks, the width of the sidewalk can be computed simply by relying on the distance between each point from the trajectory line (see Fig. 6.3). The distance is a feature which was already computed in the semantic segmentation phase, at this stage it can be retrieved as a point attribute. Considering that ROIs were created including a portion of 2 metres long sidewalks, the resulting value of width is an average value referred to the whole sidewalk portion inside the ROI.

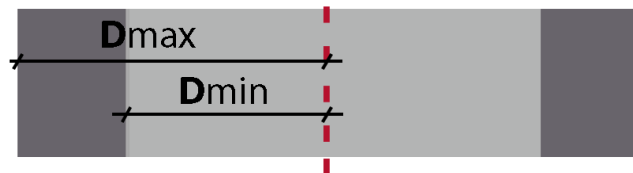


Figure 6.3: Graphic scheme of the computation of the attribute "width". Sidewalks in dark grey and roadway in light grey. The max and min distances are measured from the central line of the road in the ROI.

A first attempt to extract the width could be based on the idea of taking only two points from the sidewalk cluster: respectively the one with the max and the one with the min value of the distance from the road centerline. Then, the average width of the sidewalk can be computed simply by making the difference between these two distances.

Doing so, the resulting value will be much influenced by the presence of outliers, which can be randomly present at any distance from the centerline and can be seen as minimum or maximum distance points. To cope with that, instead of taking simply the minimum and maximum, a better solution is to take the 5th percentile as D_{min} and the 95th percentile as D_{max} . Then the average width can be extracted with the following equation:

$$Width = D_{max} - D_{min} \quad (6.1)$$

After the computation of the distance feature, it was considered inefficient to maintain the result effectively computed because neighbouring portions of the pavement could have very similar, although different, computed values. For example, as a result of the width calculation method, it may be that neighbouring portions of pavement have the following values: 2.47 m, 2.49 m, 2.52 m. To provide more effective and homogeneous information, it was decided to round off the result of the calculation to the nearest 0.05 m. For the previous example, all three values were rounded to 2.50 m.

6.3.2 Sidewalk relative elevation

The height of the sidewalk with respect the road is an important and interesting feature to be computed; in fact, the sidewalk can be at the same height as the road or even higher. The extracted attribute for this feature can be double: a Boolean variable that simply states whether the two urban elements are at the same height, and a second variable providing the actual height difference.

A possible way to do so is to analyse both sidewalk and road clusters. Exploiting (even in this case) the distance feature, it is possible to select the band of points on the road that is on the boundary with the pavement and extract the mean value of the Z-coordinate. Similarly, a limited band of pavement, adjacent to the boundary with the road cluster, can be selected and the mean value of Z can be extracted (see Fig. 6.4). It is important to select the band of points on the road that is in the immediate vicinity of the sidewalk, as the road section does not tend to be flat, but tends to be curved to encourage runoff of rainwater from the road surface. Therefore, it is precisely at the point where the road ends and the sidewalk begins that it is appropriate to check for height differences.

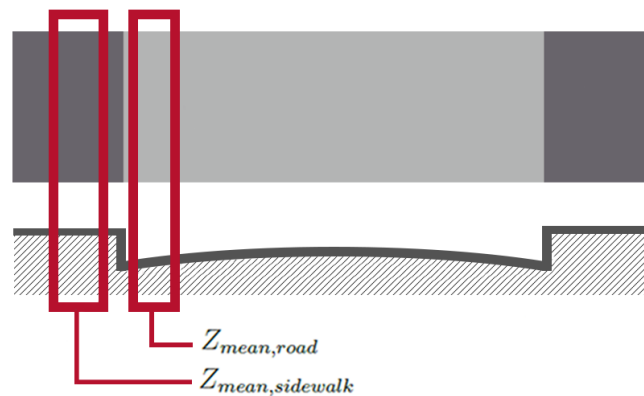


Figure 6.4: Graphic scheme of the computation of the attribute "elevation difference". Sidewalks in dark grey and roadway in light grey. The two mean elevations of sidewalk and road area (in red) are computed and compared.

To identify the two bands of points, the distance from the centre of the road can be used: the band of points on the road, close to the boundary with the sidewalk, can be identified at those points which have a distance greater than the 80th percentile (relative to the road cluster only, on the same side as the pavement). Similarly, for the sidewalk, the points with a distance lower than the 20th percentile can be selected. Compared to the previous attribute, where the intention was to be more restrictive in identifying the width, in this case, it was decided to create bands by including several points, so that the average value of Z was obtained by relying on more points.

Having defined the two reference height value ($Z_{mean,road}$ and $Z_{mean,sidewalk}$), it is possible to compute the height difference between the two urban elements in the analysed ROI. The mathematical formulation is straightforward:

$$\Delta Z = |Z_{mean,sidewalk} - Z_{mean,road}| \quad (6.2)$$

Then, the Boolean variable that defines if the two materials can be considered at

the same height, can simply compare the ΔZ with a pre-defined threshold ϵ as small as possible. The threshold can be set, for example, to the maximum difference in height defined by Italian law to consider a path accessible: 2.5 cm; it can be mathematically expressed as:

$$\Delta Z < \epsilon \Rightarrow \text{Accessible} \quad (6.3)$$

6.3.3 Sidewalk slope

To define the average longitudinal and transverse slope of the sidewalk cluster, a first possibility is to extract the points on the four corners of the (ideally) quadrilateral-shaped cluster. Actually, the identification of the four "vertices" of the quadrilateral is not a straightforward procedure. Instead, a very quick and reliable method is to take advantage of a PCA.

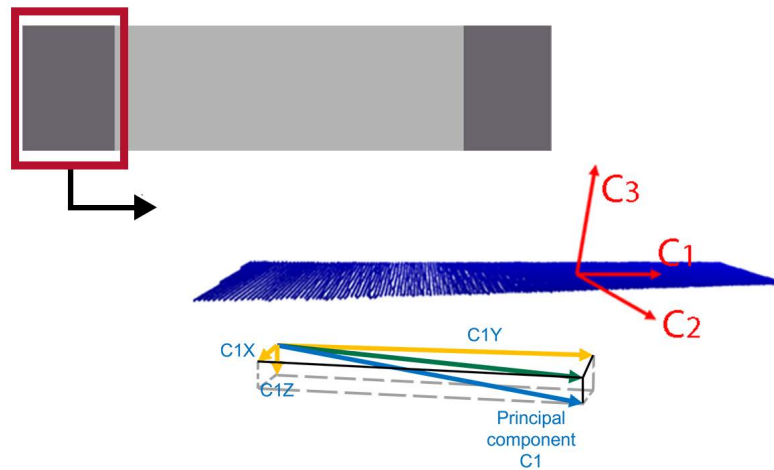


Figure 6.5: *Graphic scheme of the computation of the attribute "slopes". Sidewalks in dark grey and roadway in light grey. The cluster of points of the sidewalk (in blue) is here shown with the computed principal components (in red) from PCA. Reasoning on the inclination of components C1 and C2 it is possible to compute the Longitudinal and Transverse slopes.*

PCA is probably the best known and most widely used dimension-reducing technique, capable of extracting the principal components, that have a maximum variance for the data [159]. Computing the PCA of XYZ attributes of a point cloud, the resulting eigenvectors can be used to describe the point cloud orientation. The three principal components will have maximum variance and so the corresponding eigenvectors will be oriented in the three major directions that best fit and describe the point distribution in space. For example, usually the point normal is computed as the eigenvector corresponding to the lower eigenvalue. In the case of the sidewalk cluster of points, the three eigenvectors represent respectively the three main directions in which points are distributed, so the higher eigenvector is aligned longitudinally on the cluster, the middle one is aligned transversely and the smaller one is aligned on the normal direction. In other words, the first two vectors identify a plane that best fits the cluster of points, the third vector is its normal. An example of sidewalk cluster and its eigenvectors is provided in Figure 6.5, where a PCA of a sidewalk cluster of

points (in blue) was computed, and the three resulting eigenvectors were placed on the central point of the cluster (in red).

As it is deductible from the image, but also is confirmed by how the PCA works and by reasoning on the meaning of eigenvectors, the most suitable procedure to compute sidewalks slope is to rely on PCA and exploit the principal components vectors.

After having extracted the principal components, to compute the slopes in the two directions, a little further reasoning needs to be done. Let's consider the principal component C_1 along the longitudinal direction (Fig. 6.5), by decomposing this eigenvector on the three planes XYZ, obtaining the three components of this vector, it is then possible to compute the slope as referred to the angle between the vertical projection of C_1 component and its XY projection (green vector in the Figure):

$$\text{Longitudinal Slope } [\%] = \frac{C_{1,Z}}{\text{green vector}} * 100 = \frac{C_{1,Z}}{\sqrt{C_{1,X}^2 + C_{1,Y}^2}} * 100 \quad (6.4)$$

The slope in the transverse direction can be computed in a similar way. In this case the focus of attention is not C_1 , but C_2 , so equation 6.4 becomes:

$$\text{Transversal Slope } [\%] = \frac{C_{2,Z}}{C_{2,XY}} * 100 = \frac{C_{2,Z}}{\sqrt{C_{2,X}^2 + C_{2,Y}^2}} * 100 \quad (6.5)$$

Finally, taking into account the possibility of gross errors in the identification of the cluster of points relative to the sidewalk, it was decided to insert a limit value for the percentage of slope. If the calculated slope exceeds a limit value (which in the case of Sabbioneta has been set at 25%, since the territory is completely flat), the value automatically inserted as an attribute will be a void (NULL) value, which will leave the field empty in the database.

6.4 Results

6.4.1 Attributes

The computation of attributes (whose preliminary results were published here [160]) for all the sidewalks was straightforward. From the results, it was possible to fill a table (Tab. 6.3) that describes the ranges of the values of the attributes, and also the most frequent value.

In the same table also the reference value (for the Italian law) is reported and the percentages of nodes that respect the requirement are reported. From the Table, it appears clear that the transverse slope is the main attribute that is not respected in almost all the city. The other parameters show that in almost half of the city the requirements are met. Of course, it is also important to see the combination of those parameters and the spatial position of the corresponding nodes. The purpose of chapter 7 is to exploit the vector layer developed here and use the attributes to produce thematic maps to show the actual condition of the city.

It is important to notice that, for both the computation of the slope, in several sidewalks in corner positions the value was marked as NULL. This happened because the PCA computation exploits the three main directions of the points in the cluster, but

then the computation of the inclination is reliable if the portion of sidewalk considered is almost rectangular i.e. elongated in one direction, so that the 3 components from the PCA are a good representative of the sidewalk surface. For this reason in corners, the slopes should be removed from the attributes and a NULL value should be inserted.

From the results, it was also possible to make some diagrams and graphs that better describe the range of values. The graphs are shown in Figure 6.6. Finally, a validation of the computed values is provided in section 6.4.2, where the computed attributes are compared with manually measured values in some areas of the city.

Attribute	Ranges	Most frequent	Reference value	Accessible segments
Width	0.45 m ÷ 2.20 m	0.95 m	≥ 0.90 m	72.3%
Transverse slope	0.05% ÷ 9.75%	1.25%	≤ 1%	8.6%
Longitudinal slope	0.10% ÷ 9.77%	2.1%	≤ 5%	79.3%
Relative Z difference	0 m ÷ 0.12 m	0 m	≤ 0.025 m	66.5%

Table 6.3: Results of the Attributes computed for each sidewalk segment. For each attribute, the range of values is presented and compared with the Italian reference value. Also, the percentage of accessible segments out of the total is reported.



Figure 6.6: A series of bar charts representing the computed attributes and the distribution of their computed values. In red the values that respect the Italian reference, and in green the ones that do not respect the limit.

6.4.2 Validation

Once the attributes have been successfully computed, the intention is to validate the previously calculated data and to compare the results obtained from the semi-automatic method presented in this thesis with traditional methods that do not rely on the point cloud. As already presented in the Introduction of this thesis, a PhD of the Department of Architecture, Built environment and Construction engineering of Politecnico di Milano, architect Sebastiano Marconcini, dealt in his PhD thesis with inclusive design related also to the city of Sabbioneta [1]. In his work he carried out punctual measurements in some areas of the city, to assess their accessibility characteristics.

In this section, therefore, the results obtained from this thesis are compared with the measurements carried out in the field by the architect Sebastiano Marconcini, whose areas of expertise are accessibility management and inclusion. To this end, different portions of pavement were identified in different areas of the city, and their attributes were compared (the validation results were also discussed in this paper [161]). The areas identified as references for these comparison analyses are represented by the map in Figure 6.7. The architect, on-site, measured only the width of the sidewalks, making one or two measures that he considered representative of long portions of sidewalks. He also measured some slopes, but not for all the cases. He also deduced, by visual analysis, the state of conservation of the pavings.

Comparing the two approaches, of course, the one presented here can give much more attributes, with a high density of data. The comparison of attributes and measures on the selected areas of the city are summarized by Table 6.4, which shows that the width attributes derived from the two methods are perfectly compatible and correspondent.

The average difference between the two methods is 0.04 m with picks of 0.07 m for some of the tested areas. In addition, it was noted that even local variations, e.g. the width of the pavement decreasing at protruding parts of the building, are well recognised by the method presented in this thesis. In the case of on-site measurements, since one or two measurements are often taken and considered to be representative of long stretches of pavement, it may happen that recesses or protrusions are missed and not reported.

In this regard, Figure 6.8 shows the example of Via Assunta in Sabbioneta, where it is possible to observe the values of the pavement width in a thematic map generated in QGIS¹ and the value measured on-site (in the Figure 6.7, it is position 10). It is interesting to notice that from the result of the computation of this thesis, the sidewalk is accessible (in terms of width) for a certain portion, and then it is no more accessible. This scenario is confirmed also by the measures taken on-site.

Therefore, it emerges that the method presented here allows not only to obtain objective and correct information, but also allows to have a greater number of information that are more effective when it is necessary to identify even local variations of the parameter of interest.

¹the explanation of how the attributes were exported into a shapefile and manipulated in QGIS is provided in chapter 7

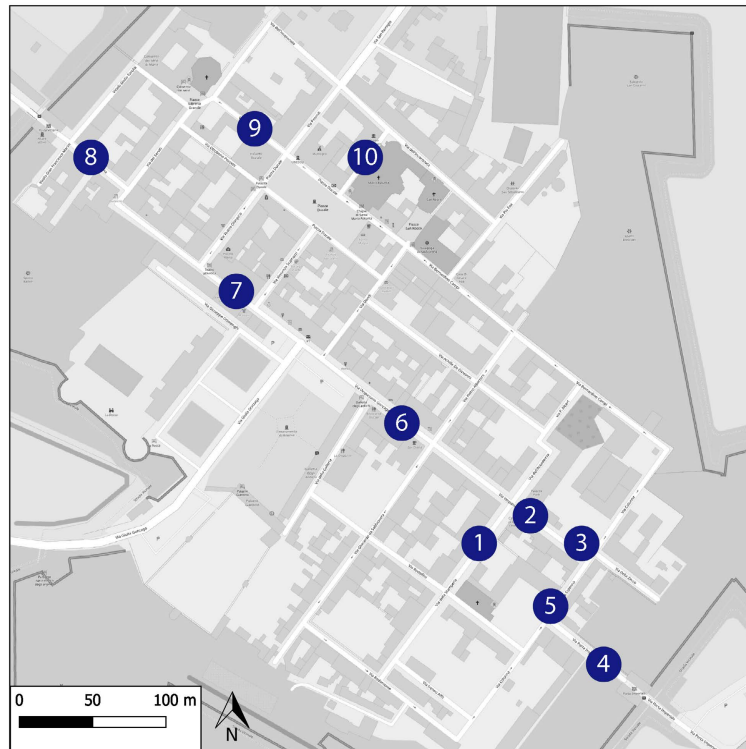


Figure 6.7: Map of the position of sidewalks used for the comparison between computed width and measured width.



Figure 6.8: An example of direct comparison between computed and measured width. In the middle a portion of the sidewalk network with edges coloured according to the width if accessible ($>0.9\text{m}$) in green, or not accessible ($<0.9\text{m}$) in red. Comparing the results of the computation with the photos and measures taken on-site, it is possible to see the strengths of the method presented here.

Check Point	Width computed (m)	Width Measured (m)	Difference (m)
1	1.55	1.53	+0.02
2	1.30	1.37	-0.07
3	1.20	1.16	+0.04
4	1.25	1.20	+0.05
5	1.00	0.99	+0.01
6	1.38	1.31	+0.07
7	0.85	0.83	+0.02
8	1.25	1.25	0.00
9	0.70	0.73	-0.03
10	0.75	0.68	+0.07

Table 6.4: Comparison between the attribute "width" computed by the method presented in this thesis work and measured on-site by an expert. The difference between the measures is also reported. The maximum difference is 0.07 m and the minimum is 0.00 m, the average difference is 0.04 m. The positions of check-points are reported on the map in Figure 6.7.

6.5 Discussion

In this chapter a workflow to compute sidewalks attributes was presented. The method started from a point cloud that was previously classified as 'sidewalk' and 'road'; the attributes were then calculated considering small portions of the sidewalk (a portion of 2 metres along the trajectory).

The computed attributes, referred to a portion of the sidewalk, were of two types: the identification of the paving surface, and the measure of several geometric attributes. For the definition of the paving material, both a thresholds-based method and an approach based on ML techniques were analysed and tested, the ML approach proved to be more efficient.

Regarding the geometric attributes they were retrieved by applying some specific equations on the cluster of points analysed. The equation parameters involved the distances of the point with respect the road centerline, the Z-coordinate, and the exploitation of PCA.

The method has proved to be an effective tool for objective data collection, based on an objective source of information: the point cloud. The resulting computed attributes are comparable with data collection carried out by an experienced technician on-site. In addition, the amount of reliable data collected was far greater than that generally collected in the field by hand.

Concluding, the method presented here allowed the computation of several sidewalks' attributes, with a high continuity along the sidewalk network, and with reliable measurements. The data extraction is not the final phase of this work, a suitable way of communicating the extracted information to the end user must be found. This will be addressed in chapter 7.

7

Sidewalk network vectorization and some possible implementations

The purpose of this chapter is twofold: the transformation of previously computed sidewalk information into a vector format representing a spatial datum containing the city's network of sidewalks, and the demonstration that such a file can be used for a variety of purposes. The developed file contains the sidewalks of the city of Sabbioneta with their attributes. The results of this chapter demonstrate that the initial purpose of this work has been achieved: the implementation of a procedure for extracting information from an initial point cloud data, processing it, and then conveying the extracted data in a format that can reach a variety of users and is easily interoperable.

The chapter is organized as follows: the first section introduces the topic and discusses the point cloud representation techniques; the following section presents the method for the vectorization of the sidewalk network. The last section presents some possible uses of the vector file and a critical discussion.

7.1 Motivation for the vectorization

In the previous chapters, and since the beginning of this thesis work, a procedure that allows the extraction of geometric and informative attributes related to the sidewalks of an urban environment of a historical city has been presented. The information was collected and analysed starting from a raw point cloud, which was processed and elaborated, to retrieve specific sidewalks' attributes.

For the purpose of this research, once the attributes have been identified following specific methods (chapter 6), a way must be found to convey this information correctly to the end-user. The starting data are two: the first is the segmented point cloud (classified into sidewalks and road), from which additional information was extracted for a specific category of elements (the sidewalks); the second is the information itself (the attributes of the sidewalks). What is important now to define is the best strategy for the dissemination of this information.

A possibility is to enrich the point cloud with semantic meaning, thinking about interoperable scenarios in which one point cloud could be used by many users from different domains, each having a different need [162]. In the scientific literature, it is possible to find researches that have used the point cloud itself with the dual function of a container for the information and a means of disseminating and providing it to the end user. It is possible, for example, to find references to the concept of smart cities

and city modelling addressed directly on the point cloud [163–165]. In addition, in other cases it is possible to find examples of pathfinding computation exploiting directly the point cloud, both in urban [38, 39, 41] and indoor [36, 166, 167] environments.

It cannot be denied that the direct use of the point cloud for both information computation and the final product has numerous operational and practical advantages, including the fact that the point cloud itself represents a highly detailed and metrically correct model of reality. However, although at the research level, the topic of representation through point clouds is of growing interest, it must also be remembered that point clouds are often very large depending on how much data is collected (typically Gigabytes, if not Terabytes), and especially for this reason are often destined to be exploited only as a base to create new types of data and products.

The purpose of this work is to provide support for urban accessibility management, undoubtedly this means that the end users may be diverse (different skills and needs), and may use the data for different purposes (consulting, planning, decision making). It, therefore, turns out to be necessary to disseminate the extracted information in a format that is as interoperable as possible, takes up little memory space in the computer, and can be easily utilized without the need for high expertise. For all these reasons, although direct use of the point cloud is considered of great interest, it was decided not to use the point cloud as the output product of this work, but to move to a vector representation through the GIS environment. Future developments in this research could investigate the possibilities given by the direct use of the point cloud for the dissemination of the extracted information.

One of the final products that can be obtained from a processing of an urban point cloud is a vector file. For example, it can be used when the element of interest is the road framework [168, 169]. As seen in the state of the art chapter (at par. 2.4), it is not new to the scientific literature that research related to accessibility and spatial data produces vector files as output from starting from point clouds [9, 27, 31, 32, 35, 113]. Since a vector file (i.e., a shapefile) is a widely used industry standard defined by ESRI, it was decided to use it as the output file in this research as well, so the sidewalks and their attributes were converted to a network (made by nodes and edges) and inserted into a shapefile that could then be for example visualized, exploited, and used through GIS. The generation of the vector layer in its preliminary form was previously discussed at a conference [160], the test area was only the city of Sabbioneta.

7.2 Vectorization method

The purpose of vectorization is to collect all the data extracted in the previous phases and convey them into a single system, which must therefore collect and display all the information effectively. Not only, since the initial data is a point cloud georeferenced in a precise reference system (in the case of Sabbioneta it is EPSG:32632 WGS 84 / UTM zone 32N), also the vectorization of the extracted data can, and must, be presented in the same world reference system. The result of this operation is a vectorial representation (in the specific case the file is of shapefile type) in which the georeferenced sidewalks and their geometric and pavement attributes are present.

To vectorise data from the results presented in the previous sections, it was neces-

sary to use a specific python library that allows shapefile type files to be written. The one used in this work is pyshp [46], and the open-source software QGIS was chosen for the visualisation and manipulation of the resulting files.

7.2.1 Representation: points or lines

Before proceeding to vectorisation, it is necessary to define how to represent the sidewalks. The starting point is a cluster of points whose coordinates, as mentioned, are georeferenced. To insert the sidewalk in the vector file, it is not possible to insert all the points in the cluster, both because it would be complex in terms of the weight of the data, and also because it would not allow effective and immediate use of the data provided. Therefore, sidewalks must be represented with a simpler geometric element.

The geometric entities that can be inserted in a shapefile are basically three: points, lines, and polygons. Therefore the choice must fall on one of these three. If the choice was to use a simple point, it could be, for example, the central point of the cluster. If, on the other hand, the decision was to use a line, this could be identified using PCA, as presented above in paragraph 6.3.3, and then using the eigenvector in the longitudinal direction of the sidewalk. If, finally, the decision was to use a polygon, this could be a quadrilateral enclosing all the points of the cluster.

Among the three possibilities, the one that would seem to be the most logical is the polygon. In fact, with it, one could completely represent the area occupied by the sidewalk. However, to be able to use the vector file for routing simulations, the polygon is not the ideal choice, since routing algorithms tend to be based on nodes and links between nodes (i.e. edges). Therefore, to give the possibility of further development of the work, starting from the vector data, it is appropriate to use either lines or points. It can also be said that starting from a line and creating an offset on both sides or on one side only, it is easy to create a polygon. The reverse operation, on the other hand, is more complex and there is a greater probability of error, so again, representing the sidewalk with a polygon does not seem to be the most advantageous option.

Whether it is chosen to use points or lines, it is still necessary to find a way to connect the sidewalk segments. It is indeed important to remember that all segmentation and classification work is based on ROIs of 2 metres in width (see section 4.2.3). Therefore, the starting point is a set of many sidewalk segments of 2 metres in length. If it was decided to represent them with points, or with lines, it is still necessary to connect the consecutive elements.

At this stage, some tests were carried out using both the point representation and the representation with lines derived from the main direction identified with the PCA method. Figure 7.1 and Figure 7.2 show the results of vectorizing the sidewalks on the Sabbioneta Track-C dataset using both methods.

Note that in the case of the lines (Figure 7.1), in the programming code that creates them from the clusters, a method has also been inserted that automatically connects them (identifying the point of contact and creating new segments that form a single polyline).

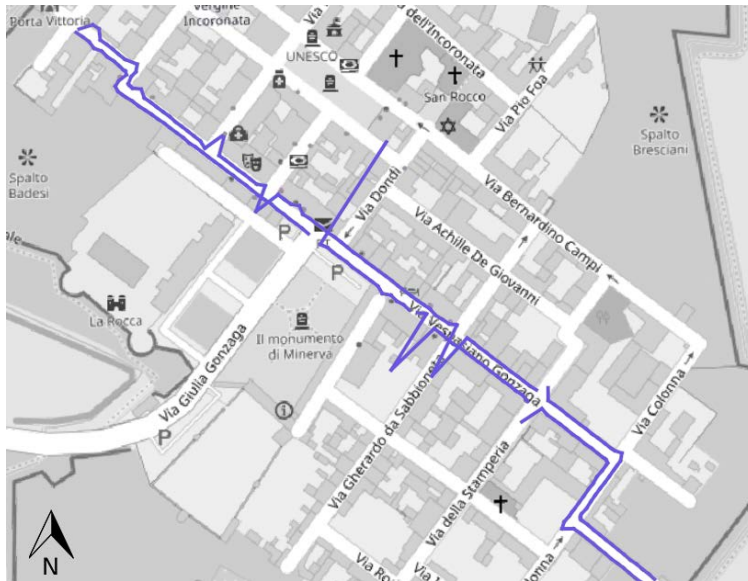


Figure 7.1: Results from the vectorization of sidewalks on Track-C using PCA to transform each sidewalk segment in line. When connecting consecutive lines, some errors arise, in particular, some lines have a strong inclination respect to the roadway.

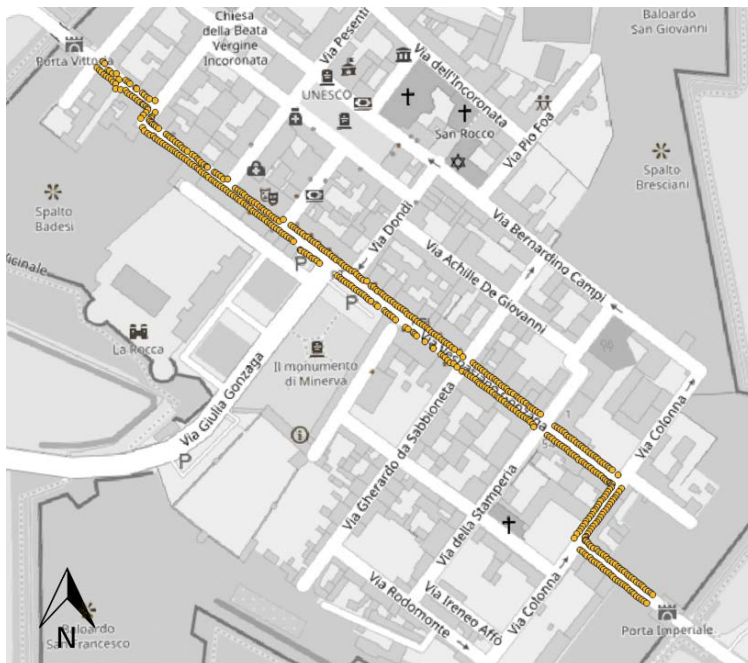


Figure 7.2: Results from the vectorization of sidewalks on Track-C representing each sidewalk segment with a point. Here the points were not yet connected, but the position of points is more coherent than using lines.

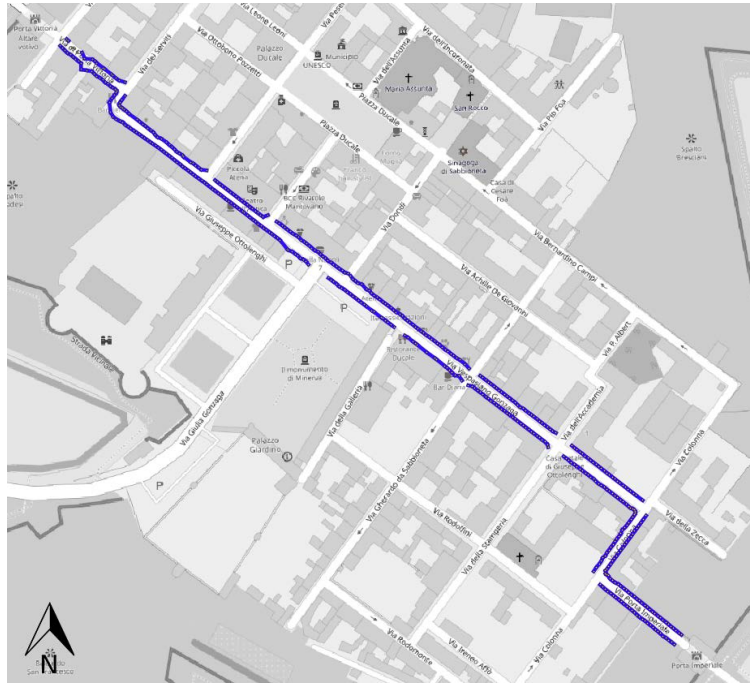


Figure 7.3: Result of the complete vectorization of Track-C into a sidewalk network. Firstly sidewalk segments are represented by a single point (i.e. nodes), and then points are connected by lines (i.e. edges); lines that cross the road are avoided.

As can be seen, it often happens that some lines had an incorrect inclination, which may be due to sidewalk clusters that were incomplete, or that the sidewalks had not been correctly recognised in the specific ROI, or that the ROI included a sidewalk end which occupies a smaller area. All of these situations mean that the principal component calculated by PCA did not represent the longitudinal direction; fortunately, this happens in a few isolated cases. However, this does not seem to be the correct way to proceed.

In the case of the points (Figure 7.2), however, it can be seen that they were correctly aligned on the axis of the sidewalk. However, having only the central point of the sidewalk is not sufficient for the purposes discussed above. Therefore, even for the points, it is necessary to find a way to connect them. The connection is straightforward. Since the ROIs are sequential, the extracted points are also sequential, so the consecutive points can simply be linked together. This idea may work; however, some errors may occur in the case of intersections with other roads. If the method is to link the points in sequence, there is no way of knowing which of the links should be broken.

To overcome this problem, it is necessary to know in advance where the roads are located, so that links between points that intersect the road must be eliminated. This operation is very simple since the polylines representing the streets of the city can be easily found from OSM, as it has already been done previously for the footprint of buildings or the positioning of T, L, X intersections. The result of such an operation is shown in Figure 7.3, and it is noticeable that, as expected, where there is a crossing with another road, sidewalks network interrupts.

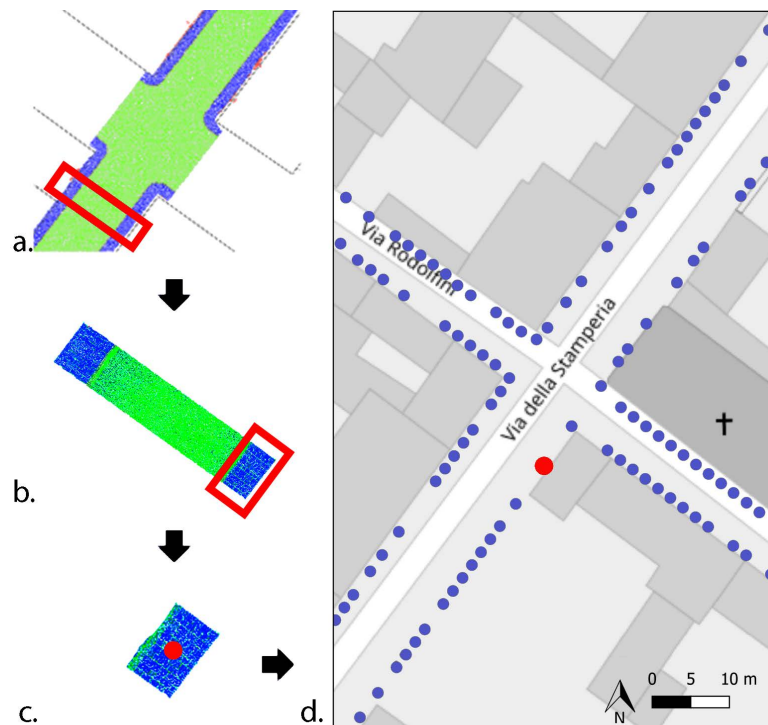


Figure 7.4: Graphic scheme of the vectorization of sidewalk segments into points (i.e. edges).

7.2.2 Generation of the Sidewalk network

As concluded in the previous section, the best way to represent sidewalk portions is a set of single points. To generate a sidewalk network from the previously generated points and lines, the points could be the nodes of the network, while the connection lines between points could be the edges.

To do so, the sidewalk clusters of points from all the ROIs of all the tracks were selected, then each cluster was represented as a single point, with the computed central point (average XYZ coordinates) of each cluster of points serving as the representative for the sidewalk segment itself (Fig. 7.4).

At this time, every sidewalk was represented by a series of points that were nearly aligned and parallel to the road. A refinement was required here, thus the points were slightly changed to achieve a more regularized result, taking into account the contextual information. To do so (see Fig. 7.5), the three types of road crossings found in the case study were examined: X crossing, T crossing, and corner, which is usually at 90 degrees, and thus designated L crossing. The OSM dataset was used to determine the position of the crossings; specifically, the road lines and their vertices were used to identify crossing positions and types.

The points on the vector layer were then selected for each crossing type, the best-suited line was computed using the least square approach, and the points were slightly adjusted on X and Y to finally be on the computed line. Lastly, consecutive points (i.e. nodes) were linked together by lines (i.e. edges) forming a network. The approach followed for the generation of edges was already described in the previous section; the

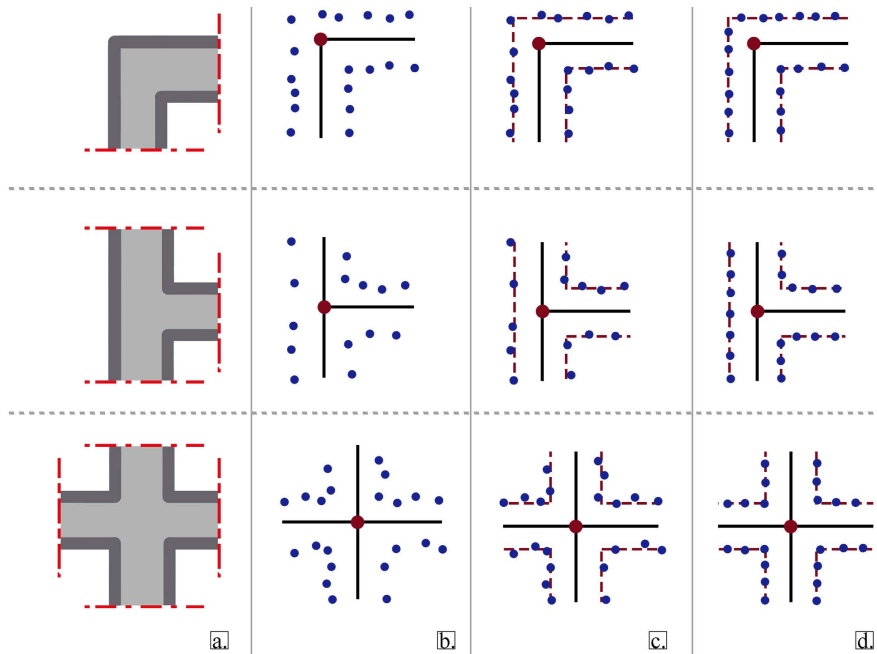


Figure 7.5: Scheme of the regularization workflow for three types of crossing: X-shape, T-shape, L-shape. a: graphic scheme of the crossing with sidewalks and road. b: vector nodes representing sidewalks. c: interpolated lines (dashed, in red). d: regularization of the nodes by slightly moving them onto the interpolated lines.

basic idea is that the two nearest nodes must be connected by an edge, but if the connection crosses a road, it must not be made.

After that, a last manual revision was planned, to fix possible minor errors and to add extra details. For example, the very few zebra crossings in the city were manually added. In fact, despite many zebra crossing recognition methodologies exist in the literature, automatic zebra crossing detection was not implemented here.

Another factor to consider is that in the case study chosen, there are just a few (actually only three) zebra crossings, and in such cases, according to Italian law, pedestrians are allowed to cross the road in any location in most of the city. A second vector network (see Fig 7.6) was created to replicate this behaviour, connecting not only the closer sidewalk nodes but also the closer node across the road. The reasons that led to the creation of this second network are described in section 7.3.2.

7.2.3 Linking the attributes

Once the most effective vectorisation method has been identified, it is necessary to link the values of the attributes extracted from the point cloud to the spatial position attribute of the element (i.e., the point). The attributes extracted, and already discussed in chapter 6, are paving material, sidewalk width, relative height respect road, slope in the two main directions (longitudinal and transverse).

Once the attributes have been computed, following specific methods, the operation of linking them to the respective point in the vector file is immediate. All that is required is to add a few new fields to the shapefile which, besides identifying the



Figure 7.6: A portion of the city of Sabbioneta, green circles are the nodes, representing the sidewalks segments, blue lines are the edges connecting the sidewalks along the sidewalk path, and purple lines are edges that connects sidewalk nodes crossing the road, considering the possibility for a pedestrian to cross everywhere inside the historical centre of the city.

spatial position (i.e, X and Y coordinates) of the point, also assigns the values of all the other attributes. Then, to easily convey and exploit the data file, the same attributes stored in the nodes are passed to the edges.

Doing so, when the shapefile is viewed, for example with QGIS, selecting an edge makes it possible to read its characteristics in the attributes table. Figure 7.7 shows a screenshot of QGIS with the attributes of a selected edge highlighted, i.e. the attributes corresponding to the sidewalk segment identified by that specific point.

7.2.4 Results

The spatial position of sidewalk segments was calculated and added to a vector layer (see Fig. 7.4). The resulting vector network has 1780 nodes and 1720 edges, and it took overall 3.5 minutes to compute on a medium-quality computer (16GB RAM, NVIDIA GeForce 950M). A second type of edge was also estimated, which took into account the possibility of crossing the road at any time. This second network had the same 1780 nodes as the first, but added 1357 extra edges for a total of 3077. Figure 7.6 depicts a section of the two networks.

After the computation, several small problems, such as a few missing edges or some additional edges that needed to be eliminated, required manual refinement. Only 23 edges needed refining, which correspond to 1.3% of the total, resulting in an accuracy of 98.7%. Furthermore, because the point cloud only covered practically the entire city centre, certain short roads were not scanned, and thus their walkways were not included in the network. They were manually inserted.

Figure 7.8 shows the whole network, in blue the edges and in yellow the points.



Figure 7.7: A screenshot of QGIS with the attributes of a selected edge highlighted, i.e. the attributes corresponding to the sidewalk segment identified by that specific edge.



Figure 7.8: Result of the vectorization of the whole Sabbioneta dataset. In yellow the nodes and in blue the edges. A zoomed area helps to better see the results.

7.3 Possible uses

The final result of previous steps is an accurate, realistic, and high-resolution shapefile containing the spatial position of the city's sidewalk network, filled with its attributes.

In this section, a demonstration of what is possible to do with the shapefile is provided. Of course, just some examples are made, and the main objective is to prove that the shapefile can be efficiently used. For this reason, these final products are extensive but not definitive, and further similar deliverables can be produced, if necessary.

In specific, first, the use of the shapefile in QGIS is presented, through the generation of some thematic maps, and the computation of paths between Points of Interest. Then the upload of the vector file and the thematic maps on a webGIS platform is presented and described. Finally, the use of the shapefile for the updating of existing cartographic data is described; in specific, the procedure to upload it in OSM is described, and the result is presented.

7.3.1 Thematic maps

The first use for the vector file was to import, visualise and exploit it through a GIS desktop application, in this case, it was imported into QGIS. In particular, it was initially used for the creation of thematic maps, which are presented and discussed in this section.

A first attempt was the generation of simple maps with the choice of giving different themes to the lines in the shapefile. The themes that have been represented were intended to show the different attributes, it was then decided to give a colouring style to the lines by dividing them into different gradations according to the value of the specific attribute. The thematic maps created were therefore related to width (Fig. 7.9), relative Z difference (Fig. 7.10), longitudinal slope (Fig. 7.11), transverse slope (Fig. 7.12), paving materials (Fig. 7.13).

Then, it was possible to create a map showing which portions of pavement can be considered accessible and which cannot, considering a specific attribute. In this specific case, it was chosen to refer to the width, using as limit value the one imposed by the Italian law (0.9 m). This map is shown in Figure 7.14. The same result could be obtained for all the other parameters.

Following the same line of reasoning, the last map was created to show the severity (in terms of non-accessibility) of each portion of the pavement. Then, considering every single attribute and the respective limit value, for each edge in the shapefile a further attribute was inserted given by the sum of the previous attributes intended with value 0 if "accessible" and with value 1 if "not-accessible". Therefore low values of this parameter mean that the portion of pavement considered has a low severity, if the value is high, it has a high severity. This new attribute was then inserted in the map in Figure 7.15.

In conclusion, these thematic maps can be useful to give information with a high level of detail on the whole city and could be useful for different purposes (spatial planning, decision of accommodation interventions, planning of effective routes) and for different users (administrators, citizens, tourists).

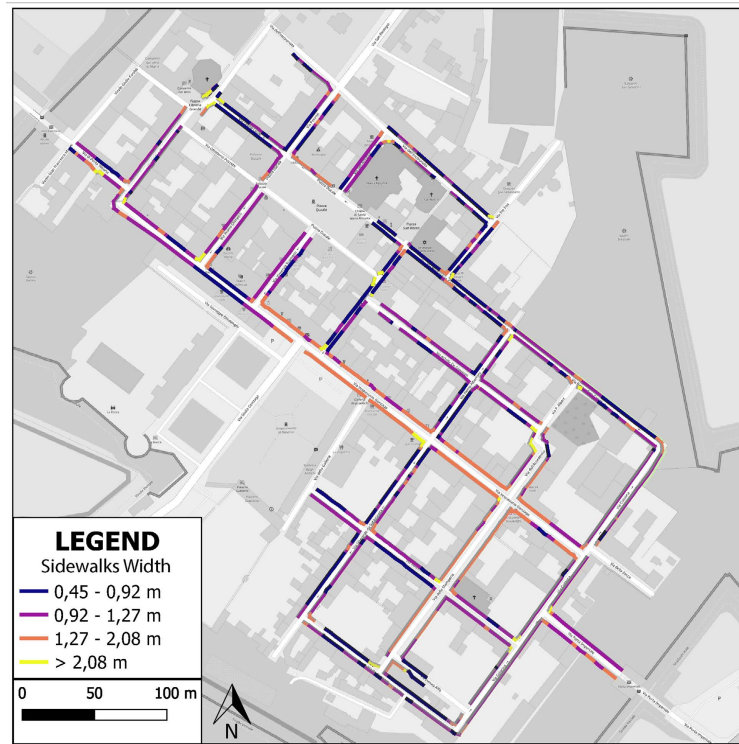


Figure 7.9: Thematic map generated from the shapefile generated in this thesis work. The theme here is the sidewalks width.

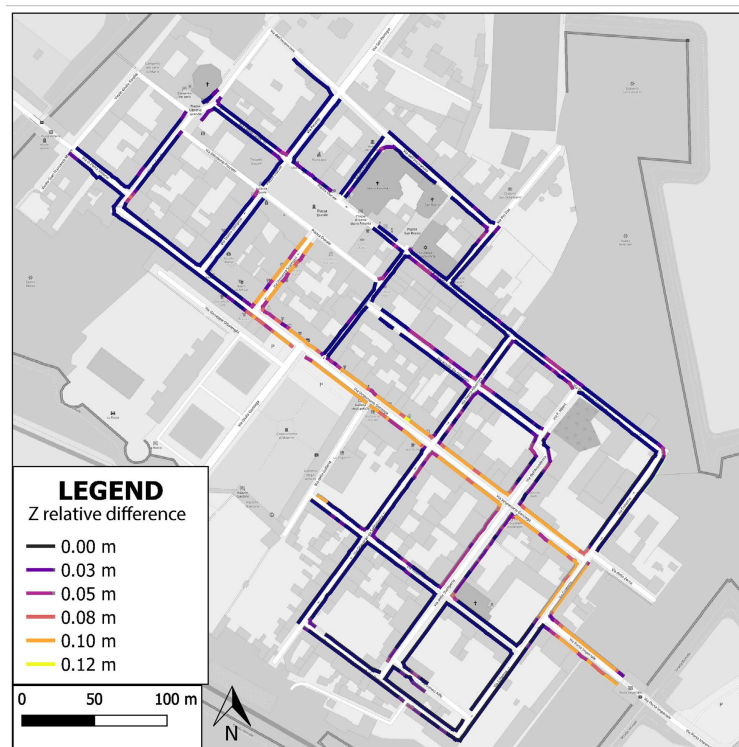


Figure 7.10: Thematic map generated from the shapefile generated in this thesis work. The theme here is the sidewalks relative elevation.

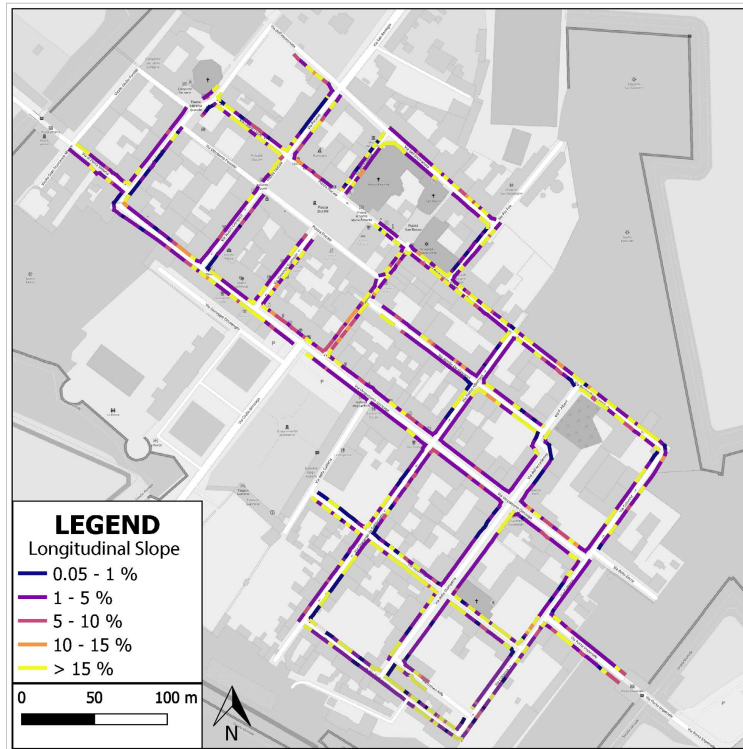


Figure 7.11: Thematic map generated from the shapefile generated in this thesis work. The theme here is the sidewalks longitudinal slope.

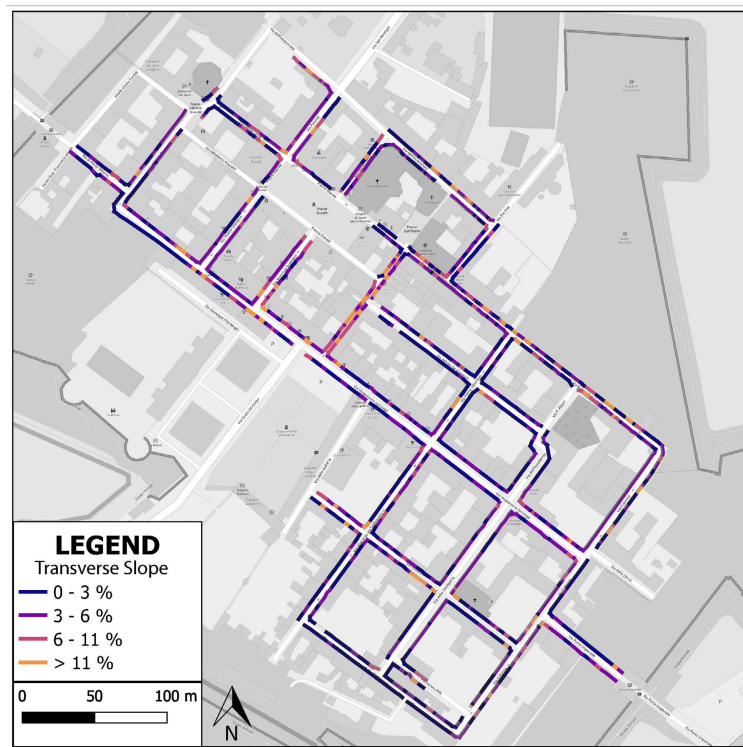


Figure 7.12: Thematic map generated from the shapefile generated in this thesis work. The theme here is the sidewalks transverse slope.

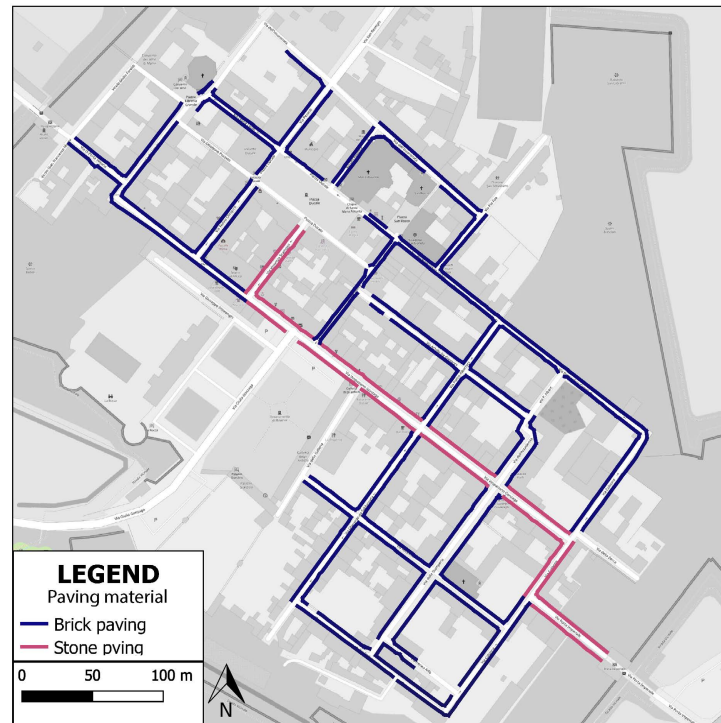


Figure 7.13: Thematic map generated from the shapefile generated in this thesis work. The theme here is the sidewalks paving materials.

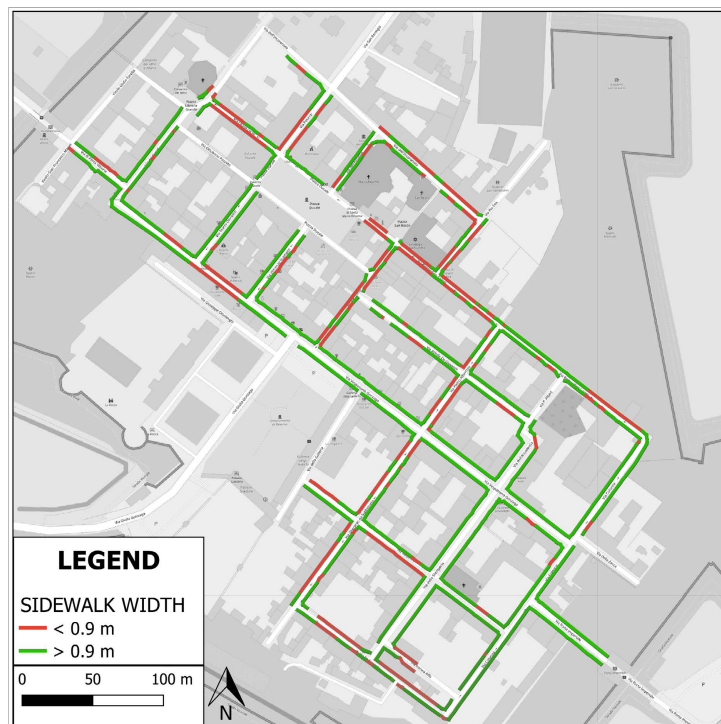


Figure 7.14: Thematic map generated from the shapefile generated in this thesis work. The theme here is the sidewalks width coloured according to the accessibility of the sidewalk segment according to Italian law. Accessible segments (>0.9m) are green, and not accessible (<0.9m) are red.

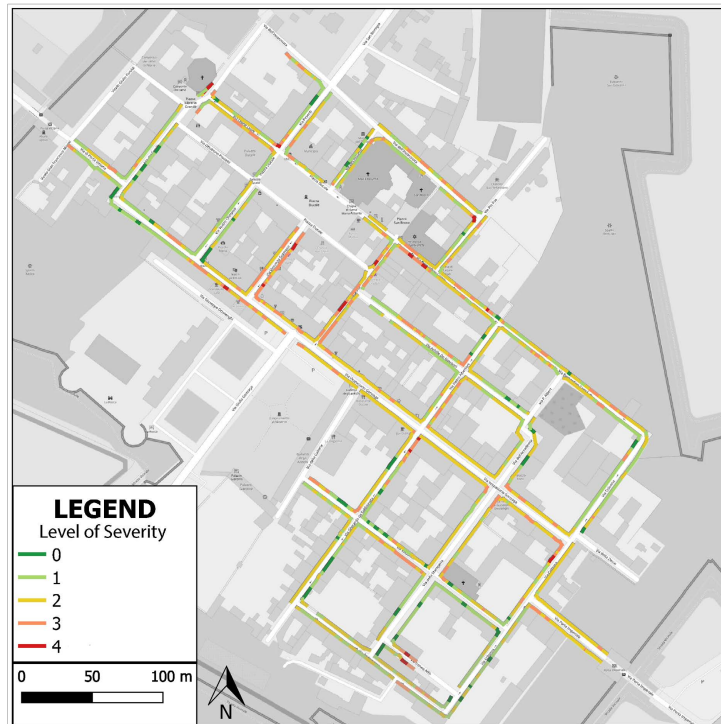


Figure 7.15: Thematic map generated from the shapefile generated in this thesis work. The theme here is the level of severity of each segment, according to accessibility. The level of severity was computed based on the other accessibility parameters and their values respect the Italian law.

7.3.2 Path computation

Having a vector file containing the sidewalk network with attributes, it is possible to think about using it for the calculation of routes within the city. These routes should obviously be designed for sidewalk users, and it could also be possible to use the attributes calculated for sidewalks to compute accessible routes.

There are various software tools available on the market for calculating routes, and for simulating pedestrian flows. In this case it was decided to use the tools already installed in QGIS. In fact, as already mentioned for the creation of the thematic maps, the shapefile has been imported into a QGIS project.

QGIS was used to import a vectorized sidewalk network saved in a shapefile. Two types of networks were imported: one representing solely the sidewalks, and another taking into account the fact that, in the case study under discussion, people can cross the road wherever in the city centre (by paying attention to the vehicular traffic, of course). This occurs because there are no zebra crossings in the city centre, hence Articles 190 and 191 of the Italian highway code apply in this case. According to the code, in this situation *"pedestrians may only cross the road in a perpendicular direction, taking care to avoid situations that endanger themselves or others"*. The second network has connections between nodes also crossing the road. Considering how the nodes are generated, it is not possible to have edges connection in perpendicular, so when developing the "crossing-the-road" edge, only the ones with the shortest length were added to the network.

The route analysis was carried out in QGIS with the help of a processing toolbox program. The "Shortest path (point to point)" tool, which was able to compute both the shortest and the fastest path between two points specified directly on the map, was chosen for the tests from the several options available under "Processing > network analysis."

As previously stated, this method is capable of calculating the shortest or fastest path. The "shortest" clearly considered simply the length of network edges, whereas the "fastest" computed the fastest route to the end destination using the "speed" value of each edge. The "fastest path" approach was employed to find the most accessible path in this study. It was feasible to drive the algorithm to compute a path that only used accessible sidewalk edges by establishing a new column in the edges database and filling it with a fictitious speed (i.e. the speed acts as a weight) based on the sidewalk geometric attribute.

For example, considering a sidewalk width of 0.9 m as accessible (as enforced by Italian laws), and by assigning a speed = 1 Km/h to all edges with "width" < 0.9 m and speed = 2 Km/h to all edges with "width" > 0.9 m; it was possible to compute the "fastest" path that was forced to be as much as possible on sidewalks with a width which was considered accessible.

Similarly, each edge speed value can be used to limit the number of crossing edges (just by increasing the speed on non-crossing-edges). Crossing-edges should be used only if the Z difference between the road and the sidewalk is less than 0.0025 m when designing a path for wheelchair users (to do so crossing edges where the difference in Z is higher should have a very slow speed).

Two separate examples were used to test path computation. In the first case study, the shortest and fastest paths were calculated based on the width of the sidewalks. In the second test, the presence of ramps on sidewalks with higher elevations than the road was taken into account while calculating the path.

Test 1

Figure 7.16 illustrates the first test. In this example, a path was computed from a general location of the city (point A, a crossroads) to a city POI: the *Galleria degli Antichi* (point B). The shortest path was computed in Figure 7.16b, whereas the fastest path considering accessibility (width) was determined in Figure 7.16c. For the second scenario, a separate speed was assigned to sidewalks with an accessible width (4 km/h) and those with an inaccessible width (0.001 km/h). Given the large difference in speeds, the algorithm avoided sidewalks with inaccessible widths as much as possible when calculating the fastest path. When comparing the predicted path in 7.16c to the thematic map in 7.16a, it can be seen that the suggested path avoided a small section of inaccessible sidewalk in the top right corner, crossing the street only for a few metres.

Test 2

Figure 7.17 illustrates the second test. The path from *Piazza Ducale* (point A) to *Galleria degli Antichi* (position B) was calculated in this example. The route was designed with the changing elevations of sidewalks in relation to the road in consideration, making it impossible for wheelchair users to cross the road at all times. As shown in Figure 7.17a, the sidewalks connecting points A and B were all characterized by a Z difference between road and sidewalk that was greater than the permissible value (which for the Italian law is 0.025 m). In this scenario, it was essential to add two extra edges to the network, which corresponded to ramps on sidewalks, to allow path computation. Figure 7.17a depicts the two added edges (in blue, close to point A and on the right respect point B). The computed path, as shown in Figure 7.17b, used the two ramps to overcome the inaccessibility issue of sidewalk elevation.

In both situations, the system was able to predict a path that passed exclusively on sidewalks that were judged accessible based on the weight's individual properties. In the first test, two distinct speeds were used: 0.001 km/h and 4 km/h. It is crucial to note that a smaller variation in speed (e.g., 1 km/h and 2 km/h) would have resulted in an unsatisfactory algorithm result. Because of the large speed disparity between the two, the computed path was fully accessible.

Future work could investigate a possible combination of different weights derived from all geometric features. Other QGIS existing path computation algorithms, as well as a custom-made algorithm, could be investigated. In addition, the possibilities offered by the use of specific software to calculate routes can also be considered interesting. In any case, the starting point would be the same shapefile.

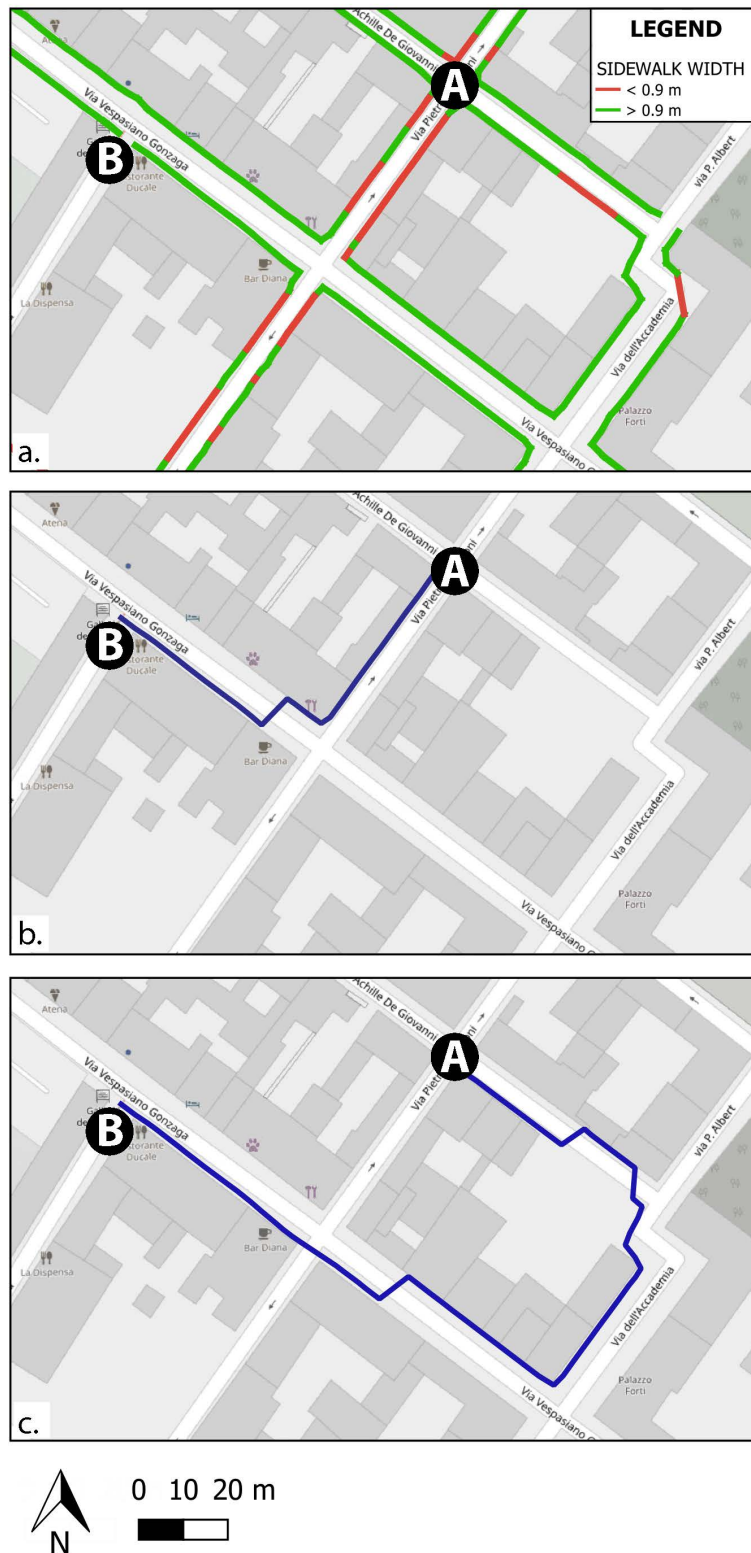


Figure 7.16: Test 1: computation of path considering widths of sidewalks as constraints. The path is computed from point A to point B. (a) sidewalk width thematic map; (b) shortest path; (c) fastest path considering accessible width.

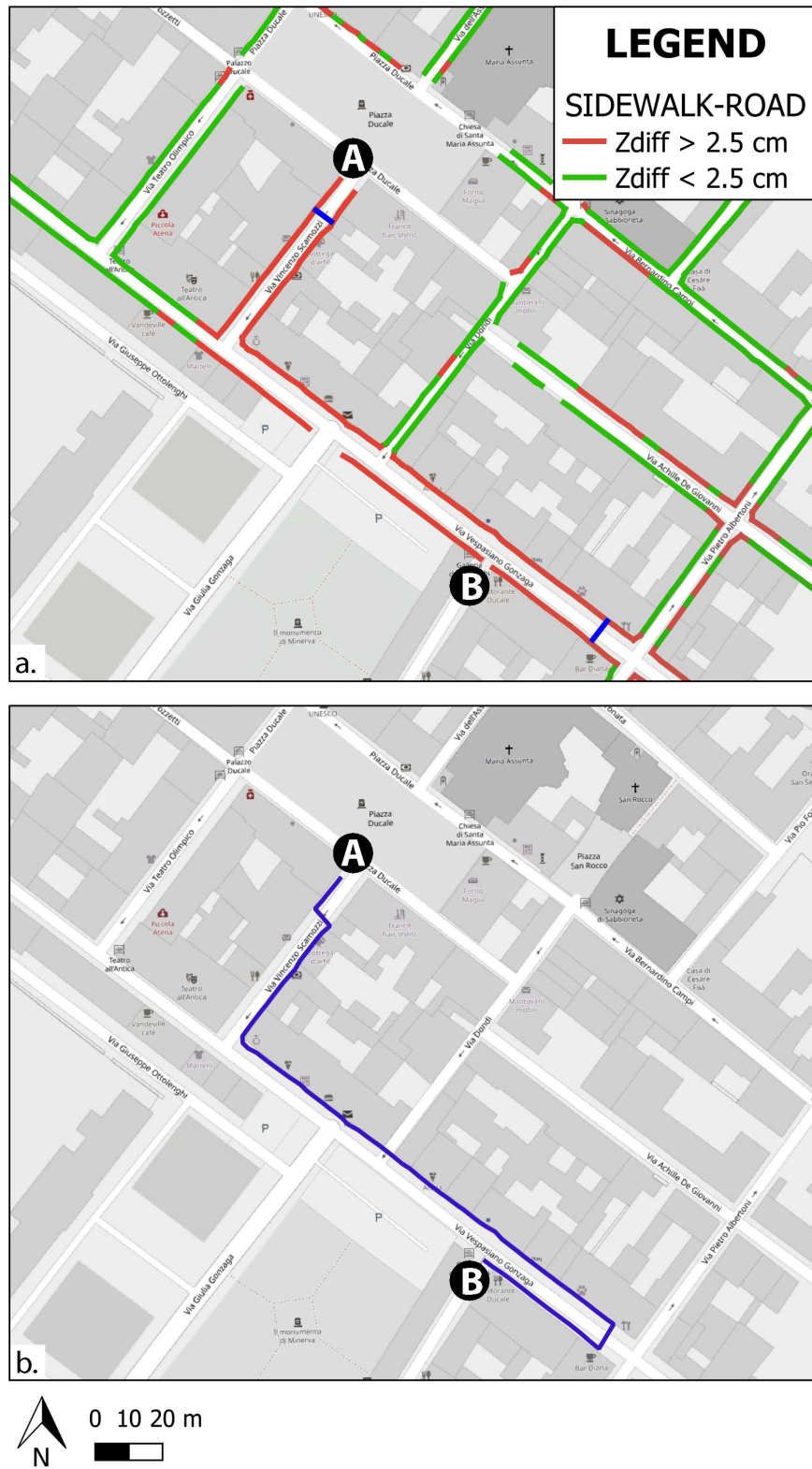


Figure 7.17: Test 2: computation of path considering Z differences between sidewalks and roads as constraints. The path is computed from point A to point B. (a) sidewalk Z -difference thematic map; (b) shortest path considering also the presence of ramps on sidewalks.

7.3.3 WebGIS platform

The generated shapefile is flexible and can be easily uploaded to webGIS platforms. To demonstrate that this is possible, it was decided to make use of a simple webGIS platform developed during a PhD course. It was done using a virtual machine (OSGeoLive 14.0¹) developed specifically to try a wide variety of open source geospatial software without installing anything. It provides pre-configured applications for a range of geospatial use cases, including storage, publishing, viewing, analysis and manipulation of data. Within this virtual machine, the GeoServer service was used for the publication and implementation of a webGIS service. Geoserver² is a server for sharing geospatial data, designed for interoperability, it publishes data from any major spatial data source using open standards.

The developed webGIS platform is local and only works offline and on the PC on which it was developed and tested. Nevertheless, the element of interest is precisely how it has been developed and the type of geospatial data it can communicate. It was developed as part of a PhD course, before the final product of this thesis work, therefore the vector files included were only temporary placeholder files. Since the result of the thesis work is a shapefile, it is possible to think about uploading it to the developed platform and thus show how potentially easily it can be used and accessed online.

The platform was designed (see Fig. 7.18) with a side column in which information about the selected elements appears, and a large map on which several layers appear. Specifically, there was a layer with the Points of Interest; a layer with information on sidewalks; a layer with information on shaded areas; and a layer with information on ramps and pedestrian crossings. It was also planned to be able to choose between a background map and a satellite view.

Many layers were planned, but at the end of the thesis work, only the layer for sidewalks was actually finished and correctly inserted. This, therefore, leaves open the possibility of further development and implementation of further methods for the extraction of accessibility attributes from the point cloud.

The vector file has been inserted as a Web Feature Service (WFS) layer, therefore by selecting the single pieces of pavement a pop-up (see Fig. 7.19) appears with the information (i.e. the attributes) of that pavement. The layer relating to points of interest, on the other hand, actually identifies certain areas of interest in the city, is a Web Map Service (WMS) layer and if selected allows the attributes relating to the selected POI to be displayed in the left-hand column (see Fig. 7.20).

In conclusion, this brief example is intended to highlight the multiple uses of the vector file created. In addition, it can be noted that, if required, all the thematic maps created earlier in this chapter can easily be loaded into the webGIS platform.

¹<https://live.osgeo.org/en/index.html>

²<http://geoserver.org>

Chapter 7. Sidewalk network vectorization and some possible implementations

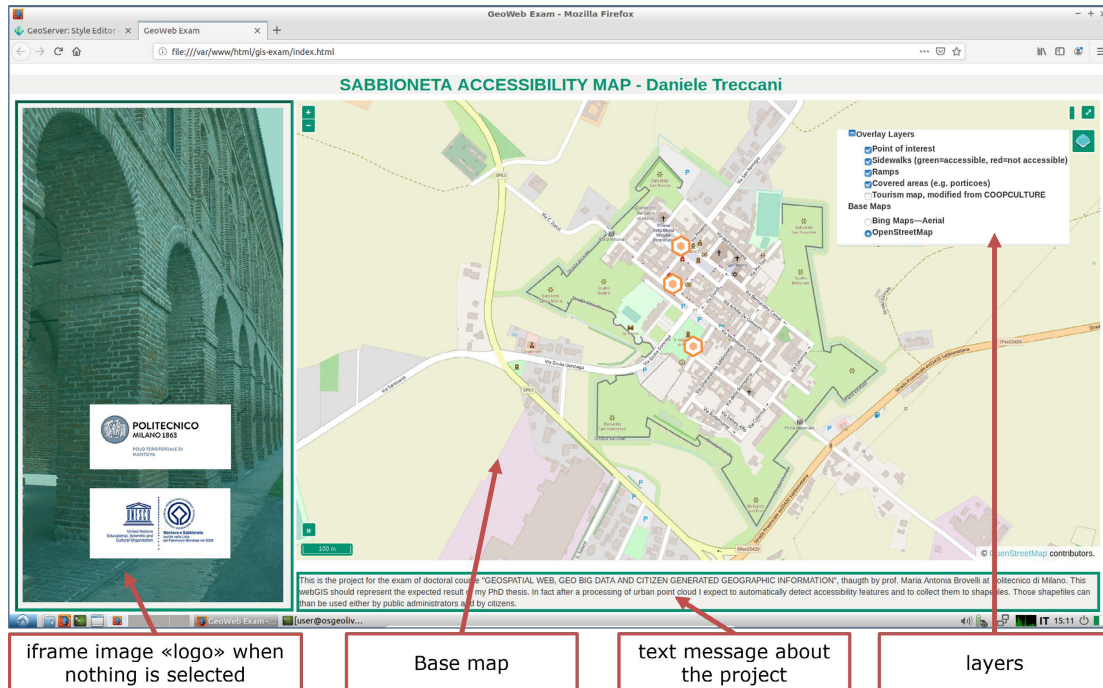


Figure 7.18: Screenshot of the webGIS platform developed within the doctoral course "Geospatial web, geo big data and citizen-generated geographic information", held by Prof. Maria Antonia Brovelli. The main page of the webGIS platform is here briefly described.

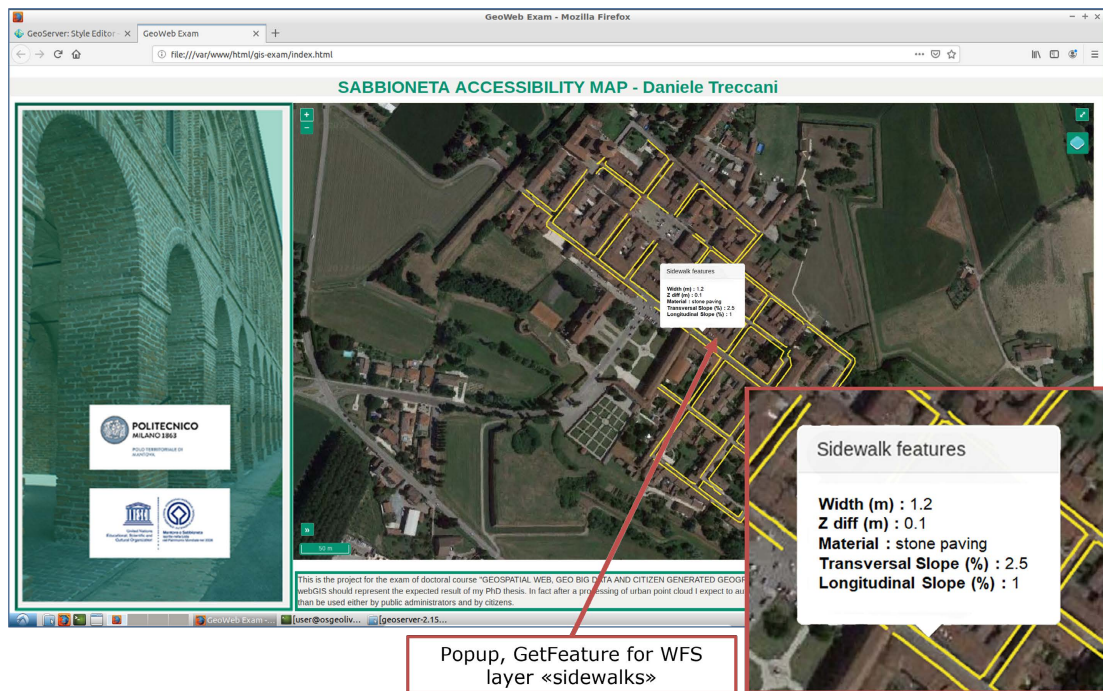


Figure 7.19: Screenshot of the webGIS platform with a sidewalk segment selected. Note the pop-up with sidewalk attributes.

SABBIONETA ACCESSIBILITY MAP - Daniele Treccani

Point of interest description

- Name: Palazzo Ducale
- Accessibility: Main entrance: 6 footsteps. Entrance with ramp from Piazza Libreria Grande, there is a lift inside the building.
- Open: Fri-Sun 9:00am - 7:00pm phone:+39 0375 221044
- Url: [Website](#)
- Pic:

iframe with customized HTML
getFeatureInfo for WMS

This hyperlink works

Comune di Sabbioneta

PALAZZO DUCALE

Figure 7.20: Screenshot of the webGIS platform with a POI selected. Note the information provided on the left side of the screen.

7.3.4 Upload on OpenStreetMap

Since the procedure presented in this paper uses, in several steps, data freely downloaded from the OSM database, it seemed interesting (and also consistent with the open source principles) to try to understand if the shapefile created can be used in some way to update the existing OSM data on Sabbioneta. In fact, at the moment on the OSM dataset, focusing on the historical city centre of Sabbioneta, the elements that can be found are related to land uses (e.g., parking areas, parks, gardens), to buildings, and to roads, but there are not any information on sidewalks. The most recent data loaded³ dates back to 2018 and was related to buildings footprints, taken from a municipal technical map made available by the public administration.

In OSM, pavements can be mapped in two ways⁴: they can be considered as refinement data for a highway or they can be mapped as a separate footway. In the first case, an additional property is added to the existing road element, indicating whether or not there is a sidewalk and on which side it is located (specifically: *sidewalk = both/left/right/no*), additional information can also be added referring to the left or right sidewalk or both⁵, using the schema, e.g. *sidewalk:left:width = 3 m* to indicate that the left pavement has a width of 3 metres. Other information that can be entered is about the type of paving (*surface = **), the roughness of the pavement (*smoothness = **), whether it is bikeable (*bicycle = yes/no*), the inclination (*incline = **), whether there is a kerb (*kerb = **), whether a wheelchair can pass (*wheelchair = **), whether there is a sign nearby (*traffic_sign = **), or whether the pavement is tactile (*tactile_paving = **).

In the second case, each sidewalk is mapped as a separate highway, specifically, the tag for OSM consists of the spatial element, with associated *highway = footway* and then *footway = sidewalk*. In this way, sidewalks can be mapped as separate elements. The footway key⁶, when characterised by *footway = sidewalk*, may contain all the same parameters as described above for the sidewalk tag. It can be observed that the sidewalk key has attributes compatible with those obtained from the workflow of this thesis, and is, therefore, an excellent starting point for data entry into OSM.

After consultation with the Italian OSM community⁷, the possibility of using the shapefile to add pavement data was investigated. The vector file was examined by the experts of the community and their evaluation was positive. The file is appropriate, but it may be necessary to homogenise the data, to reduce the number of lines present. In this sense, it may be useful to represent a long strip of pavement with one line instead of many small pieces, and then average the value of the attributes. The data upload procedure⁸ involves several steps:

1. prerequisite check: familiarise with OSM procedures, identify the type of data to be uploaded and its correspondence with the OSM database, be aware of how licences work;

³<https://wiki.openstreetmap.org/wiki/Sabbioneta>

⁴<https://wiki.openstreetmap.org/wiki/Sidewalks>

⁵<https://wiki.openstreetmap.org/wiki/Key:sidewalk>

⁶<https://wiki.openstreetmap.org/wiki/Key:footway>

⁷<https://wiki.openstreetmap.org/wiki/Italy>

⁸<https://wiki.openstreetmap.org/wiki/Import/Guidelines>

2. community buy-in: make contact with the community to verify interest in uploading the proposed data type, discuss uploading plans, prepare a wiki-page for the data upload project, be prepared to have a constructive dialogue with the community about the import job, be prepared for a review of the import job by local community volunteers;
3. license approval: get permission from the data owner to upload data to OSM;
4. documentation: record the permissions obtained and describe step-by-step procedures for editing and uploading the data;
5. import review: have the data (ready to be uploaded, but not yet uploaded) reviewed by the community;
6. Uploading: following the pre-approved plan for uploading the data, keeping track of progress and informing the community when the process is complete.

As can be seen, this is not an immediate process and is subject to several stages of verification by the community before the actual upload can take place. At the moment the process has started, stage 3 was reached, the vector file was published on a public repository on github⁹, the OSM wiki-page was created¹⁰, a licence was selected. The next steps are in charge of the OSM Italian community and it is expected that the data will be uploaded within a few months. It is important to note that the procedures that will then lead to the uploading of the vector file have begun, but the further steps are beyond the scope of this chapter. What must be emphasised, however, is that the vector file is useful and is considered suitable for uploading to OSM.

7.4 Discussion

In this chapter, a method that allows representing the network of the pavements of the city of Sabbioneta in a vector file through nodes and edges has been presented. Moreover, it allows obtaining useful attributes related to the pavement. In specific (see Fig. 7.21), it started from a semantically segmented point cloud, subdivided into 'sidewalk' and 'road', the sidewalk's attributes were computed and a vector network was generated. The vector network, composed of nodes and edges, was produced based on the several sidewalk portions analysed, converted into nodes, and then the nodes are regularized and connected by edges.

From the results obtained, the method is effective and efficient. The resulting shapefile opens the possibility of making several analyses of the urban environment related to mobility inside the city, but it also allows the creation of thematic maps and other similar final products. The file created can be easily exploited by different software, making it highly inter-operable; plus, it can also be easily inserted into operational workflows. Furthermore, it can be considered very flexible, as its current content (spatial position of the network, and attributes) can be easily expanded with new parameters and additional spatial data. For example, with various surveys, it may be possible to make detailed measurements of pavements to estimate the state

⁹<https://github.com/HeSuTech/Sabbioneta-sidewalks>

¹⁰https://wiki.openstreetmap.org/wiki/Import/Catalogue/Sabbioneta_Sidewalk_Import

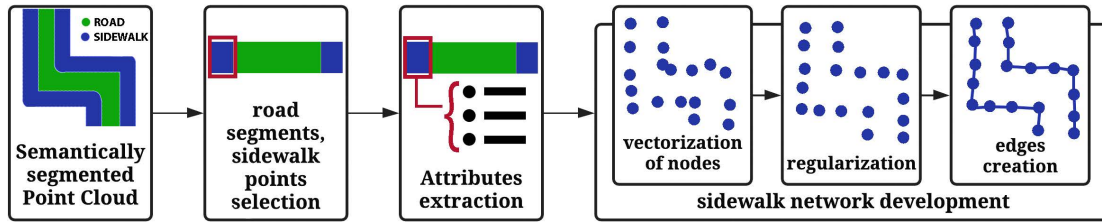


Figure 7.21: Workflow of the sidewalk attributes extraction and vectorization.

of maintenance of the pavement surface. On the basis of the same network, other information could also be added, such as the position of benches, bus stops etc. Therefore, the result of this process is a file that is intended to be a first step in the direction of accessibility management, but also a container ready to receive new and continuous information over time.

This chapter also presented several examples of possible applications of the vector file. The purpose of these examples was to demonstrate the multiple possibilities of use given by this file, and they certainly represent only some of the possible uses. These final products have not been fully refined or developed, what is important to take from them is only to see that the shapefile is effective and functional for them.

Starting from the creation of thematic maps, it was shown how visualisations with colour gradations could be generated, to quickly perceive the trend of a specific attribute in the city. But it was also possible to go further, starting from the values of the attributes and using the limits prescribed by law, effective visualisations could be created that allow to clearly see where this parameter was respected, and where it was not. A further possibility was to grade the severity of each portion of the sidewalk by combining the assessments of its attributes. In this specific case, a simple method for assessing severity has been developed, but more specific and reasoned calculation formulas could also be used by referring to the literature.

Based on the same attributes and network, routes could then be calculated. As seen, using a tool pre-installed in QGIS, the calculation of such routes was very easy and effective. By acting on the "speed" parameter and using it as a weight, it was possible to force the algorithm to propose accessible routes, which then used sidewalks with good accessibility characteristics. Clearly, there is a wide range of software, plugins and algorithms on the market and in the literature that allow the calculation of routes based on a network of nodes and edges. What has been demonstrated here was that the file could work with such systems.

Furthermore, it has been demonstrated that the vector file could be easily exploited to be inserted in webGIS platforms that would allow its use on a large scale, by diversified users. For example, it was possible to imagine that on the basis of the same vectorial file, different types of users could access it with different types of restrictions on data display. This means that the same platform could be accessed by different users with different purposes, whether related to city management, planning, or even for the sole purpose of receiving useful information for urban navigation within the city.

The last use shown was for uploading data to OSM. Thanks to the cooperation with the Italian community, the file realised in this thesis has been effectively inserted

in the OSM database and will therefore be available together with the other contents already present. This demonstrated the flexibility of the realised file and increases its possibilities of use. OSM was only one example of cartographic data that could be imagined to be extended, but there are many others, for example, the regional topographic database (for Italy), or cadastral data.

In conclusion, it can be said with confidence that this thesis produced a file that is both lightweight and easy to use by all those who may be interested in information about the city's pedestrian mobility network. Whether they are administrators, planners, members of associations for the valorisation of the territory and culture, citizens, or tourists, they can make use of the information that is located on the same base (the shapefile), but can (for example) be provided in different ways and formats, even with different restrictions depending on the users.

8

Conclusions

The work developed by this thesis aimed at taking a first step in the realisation of an integrated and interdisciplinary system to tackle the analysis of urban accessibility. Starting from objective data measured with state-of-the-art surveying instruments and exploiting AI, several methods and semi-automated approaches were implemented. In the introductory chapter, this applied research thesis was presented as a complex workflow divided into several Working Packages (Fig. 1.4). In light of the various tests carried out, the several methodologies developed, and the many results obtained, it was then possible to outline a schematic summary of the methodology proposed by this research thesis, from the input data to the various outputs realized (Fig. 8.1).

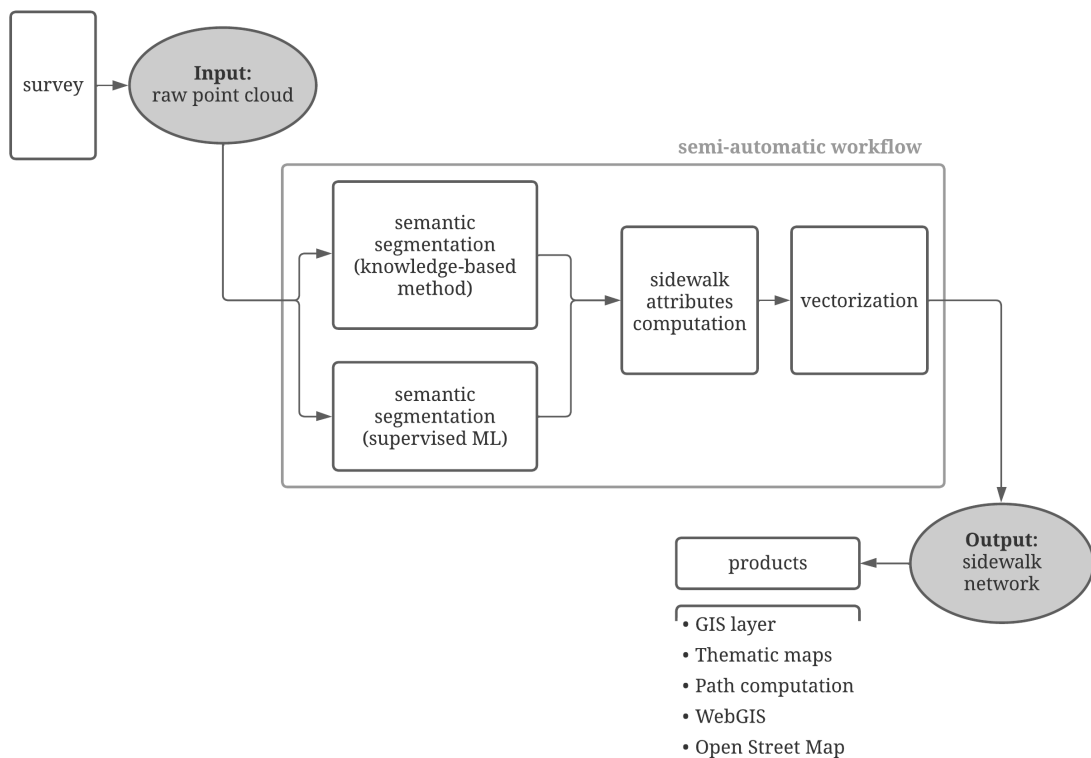


Figure 8.1: General workflow of the thesis. From the survey, to the semi-automatic processing, to the final products.

Despite the existence of several benchmark datasets of urban area point clouds in the scientific literature, it was decided to focus on a very specific case study. The research was developed -and tested- on a historical city: Sabbioneta. The city, originally built in 16th century, still retains today its chessboard urban layout, with narrow streets, widening areas, alleys, porticoes and squares. These peculiarities made the realisation of a method for the identification and collection of useful information for urban accessibility management even more challenging.

In fact, existing methods in the literature were often based on assumptions that do not typically occur in historical cities. For example, often in the literature in order to separate the road from the sidewalk, authors went in search of curbs and therefore of a quick jump in height. In the city examined, as well as in many other historical Italian or European cities, many roads do not have curbs and the sidewalks are at the same height as the road. Another typical peculiarity is the use of different paving materials to highlight the main urban elements. This peculiarity was used and exploited to develop the segmentation method presented in this thesis.

8.1 Critical discussion

8.1.1 The survey

Summary

Firstly, a survey of the city was planned and conducted with a MMS, Leica Pegasus:Two. The survey was performed in approximately 40 minutes, acquiring 7.71 Km of roads of the city, covering almost the entire historical centre within the fortified walls of the city. The survey was composed of 10 acquisition missions (called tracks) and was post-processed with Leica Pegasus:Manager, proprietary software for managing MLS surveys. The resulting point clouds were composed of 1.21 billion points, with an average spacing on the ground of 0.03 m. The processed dataset, provided by Leica, was then analysed to validate the data and to guarantee their quality and accuracy. To do so three evaluations were performed: the point clouds were compared with some check points measured with a GNSS, the value of the "quality parameter" was checked, some slices were made on the point clouds to see eventual misalignments between consecutive tracks. After the validation, the point clouds were considered appropriate to be used for the thesis purposes.

Discussion

The dataset used for this thesis work was acquired with a MMS. The well-considered choice of this measurement system proved to be effective. The point clouds obtained with this instrument were a good compromise between the quality of the data needed, the precision required, and the acquisition time.

An observation on the density of points and the accuracy on the ground is here necessary: although it was high, it was not sufficient to be able to measure millimetre variations in surfaces. For example, the measurement of the height of the joints between different portions of the pavement (which according to Italian accessibility regulations is fixed at 2 mm) could not be detected by the instrument as it was below the declared precision of the instrument itself (in the order of centimetres, see technical

sheet in Fig. 3.14). The required precision to measure also the junction height could be reached only by changing survey instruments or techniques.

Finally, it must be taken into account that if the data acquisition phase was very fast, a post-processing phase of the raw measured data was conducted to obtain point clouds that could be used for subsequent purposes. The post-processing phase, anyway, is something that is always part of any survey, disregarding the method implemented to perform the survey itself. Different instruments and techniques used need a different amount of time for the post-processing phase, but considering the wideness of the object of the survey (almost the entire historical centre of the city), the MMS proves to be the best choice.

8.1.2 The knowledge-based segmentation

Summary

To make the procedure more efficient, computations were performed on consecutive portions of the point clouds, obtained by cropping them following the trajectory of the acquisition instrument. Then, based on the idea that different pavements were used for different ground areas (sidewalks and roads), a method allowing the semantic segmentation of the ground surface of the point clouds was implemented. This approach used unsupervised ML to subdivide the point cloud on the basis of some of its characteristics. In specific, the algorithm used was k-means, subdividing the points into 4 groups on the basis of the following features: Omnivariance, Sphericity, Roughness, Intensity, Z-coordinate, and distance from the road centerline. After the segmentation of the sub-clouds, each cluster was identified as a 'sidewalk' or 'road' implementing a voting system based on topological relations between the elements and the context.

This approach was tested on the Sabbioneta dataset with an average F1-score of 78% for 'sidewalk' class. Out of 30 million points analysed, those wrongly classified were only 11% of the total. Since the method was conceived for sidewalk detection, only results for 'sidewalk' class are reported here. The same method was tested on a smaller test dataset of other historical urban environments: a road in Mantova and a road in Sacro Monte (Domodossola). For both datasets, the results were similar to the ones in Sabbioneta; respectively the F1-score for Mantova and Sacro Monte were 83% and 72% for sidewalk class. The method was then applied to a dataset of a modern environment: some roads in Porto (Portugal). The results from this last test were much lower respect the previous ones (35%). The reasons for this result could be found in both the road layout (very different from the method assumptions) and in the different (lower) quality of the data collected.

Discussion

The semantic segmentation method implemented, based on a knowledge-based approach, that also exploits an unsupervised ML tool, was proven to be effective. Despite the presence of localized errors, it is necessary to remember the final aim of the work: the identification of the sidewalks (their spatial position and their attributes) for subsequent vectorization. Therefore, thinking of the final aim, some localised errors could be considered of minor interest, as they have little influence on the final result.

Although the result is satisfactory, the method presented is certainly not simple and requires a strong component of reasoning in the calibration -above all steps- of the voting system. This step turns out to be strongly linked to the urban context surrounding the sidewalks and, just because the urban context of a historical city is characterized by various complexities, the voting system, trying to cooperate with these situations, turns out to be formed by many variables and therefore is highly complex. However, analysed in more detail, it is easily demonstrated how all its components link it to the urban context, providing optimal results in the final point classification.

This method has also proven to be effective in historical urban environments in other cities. However, the similarity of the boundary conditions is a fundamental element to guarantee the functioning of the method. It should be noted that the errors that occurred with the Porto dataset are due not only to a very different urban layout but also to the different density and quality of the data collected.

8.1.3 The segmentation exploiting supervised ML

Summary

Considering the growing interest of the scientific community in approaches using AI, and in view of the many scientific papers with very positive results, also in the field of Geomatics, it was decided to test an approach for segmentation using ML techniques. The aim was to evaluate the potential of this approach for the case study in object, and to compare it with the knowledge-based method already presented and discussed.

This alternative approach for the segmentation of the point cloud was studied by implementing supervised ML techniques. Two ML models were tested, a first one for the classification of urban elements, and a second one for the classification of paving materials.

In particular, the first mathematical model was trained to recognise, on the Sabbioneta dataset, the sidewalk and the road. After testing different classifiers and features, the Random Forest classifier with the following features was selected: Intensity, relative distance with respect the road centerline, Zrel, Roughness (0.1), Omnivariance (0.1), H, S, Z, Normal change rate (0.1), Anisotropy (0.1), Sphericity (0.1). The classification performed using this model had average results in terms of accuracy much greater than the knowledge-based method (average accuracy of 88% and average F1-score of 84%).

It was then decided to train a second mathematical model for the classification of urban pavings, but the results were poor and a different approach was defined: the second model was used only for the identification of the most probable material of a pre-selected specific cluster of points. In this way, the identification of the sidewalk paving material was possible through a ML approach.

The first ML model, the one that performed better on the Sabbioneta dataset, was also tested on the other datasets previously used for the knowledge-based method: Mantova, Sacro Monte di Domodossola, and Porto. In these tests, the accuracy of the prediction was on average 89%, but then looking at the F1-score of the 'sidewalk' class, the average value was 78%.

Discussion

This alternative test, carried out using supervised ML techniques, allowed to experiment a different approach, to achieve the same goal.

A first element to report is the need to implement strategies to be able to analyze the Sabbioneta point cloud in its entirety, characterized by a large number of points, which required excessive computing capacity, exceeding the available computer RAM. The choice of a specific library to manage these situations, Dask, has proved effective.

A second point to note is that there was a need to change the point of view, and the focus is on the importance that a different approach gave to the meaningfulness of the research itself. In fact, although initially the aim was the subdivision of the whole point cloud according to the paving materials, this approach did not turn out to be the correct one, but a change of perspective made it possible to identify a more effective and successful workflow. Considering only the urban elements, the classifier was able to make predictions with high accuracy and to reach more satisfactory results. Indeed, it is correct to say that in terms of classification accuracy, this second method is better than the knowledge-based method, but still improvable.

A consideration must then be made for the use of ML to predict the material of the sidewalk cluster. Following what was said before, it was necessary to change the approach as the results were not satisfactory in terms of accuracy. In fact, probably for a too large number of classes, the material classification on the entire dataset failed. By visually looking at the classified points, it was possible to note that the wrongly classified points were evenly distributed over the entire point cloud, making it possible to see and correctly hypothesize the material present in an area. So, asking the model to predict only the material of the sidewalk points and then selecting the material predicted more frequently in the cluster considered, the result was that the material was correctly defined in many cases. It is interesting, however, to note that this method is promising and therefore greater attention in the future should be placed in trying to better calibrate its parameters, in particular for the prediction of the whole dataset.

Lastly, the ML model trained to classify urban elements gave high accuracy results, not only in Sabbioneta but also in other datasets. Specifically, it performed well not only for the ones with similar urban environment, but also for the ones with different urban layouts. This fact is a good result because opens the possibility of developing a more comprehensive ML workflow that can be applied to a big variety of urban layouts, not only for historical cases. The particularity is that, if the existing methods (in literature) work on modern layouts and not on historical layouts, this method has the capability to work in both the cases.

8.1.4 Comparison of the two segmentation methods

The goal of correctly classifying the point cloud was reached through two different methods; it is, therefore, possible to draw some considerations by comparing the two approaches. It is interesting to note that in both cases context-related information was somehow conveyed within the method. In fact, in the case of the knowledge-based method, the voting system was certainly essential in order to consider the context. In the second case, instead, pieces of information on the context were somehow conveyed

through the features considered: on the distance of the points from the centre line of the road and in the value of relative Z-coordinate from the mean Z value of the road. In both cases, therefore, together with intrinsic geometric information related to the distribution of points, there was also a connection to the urban context of the point cloud analysed.

It is now possible to make also some considerations about the complexity of the development of the two methods and their possible versatility. Although the knowledge-based method proved to be effective in achieving its purpose, it required the development of a procedure consisting of several steps, and with various parameters that were optimised in order to maximise the final result. The use of the same model in different case studies may require that some of these parameters could have to be re-optimised to be adapted to different conditions. For example, considering the application of the Porto case study, where the results were not convincing, the cause was a big difference between the Porto road layout and the assumptions that led to the development of the knowledge-based method. Nevertheless, this method remains valid and effective for urban areas similar to the one used to develop it, i.e. areas with characteristics comparable to Sabbioneta. In fact, the method has proved its effectiveness in two other case studies in Italian cities.

On the other hand, the ML method required the optimisation of far fewer parameters. After initial testing and a few refinements, it was possible to achieve satisfactory results, and, in terms of segmentation into 'road' and 'sidewalk', even better than the knowledge-based method. The main improvement of the approach was related to the choice of the training dataset and the features to be used for training the model. The advantage of ML is that there are far fewer parameters to optimise for positive results. As already mentioned, the tests done here for ML are very promising and, therefore, further tests are necessary to refine the results.

Concerning the classification performances, Table 8.1 shows the accuracies and performance metrics of all the tested datasets for the two segmentation methods developed in this work. Considering that for the thesis the importance is given to sidewalks, the precision, recall and F1-score reported are only referred to class 'sidewalk'. From the table is clear that the ML method had better results not only on average, but also in each specific case. Anyway, subsequent and more deep studies focused on an improvement of the ML method presented seem to be necessary.

Dataset	Knowledge-based				Machine Learning			
	Accuracy	P	R	F1	Accuracy	P	R	F1
Sabbioneta	89%	0.77	0.79	0.78	88%	0.80	0.89	0.84
Mantova	85%	0.76	0.92	0.83	92%	0.80	1.00	0.89
Sacro Monte	80%	0.85	0.62	0.72	89%	0.60	0.86	0.71
Porto	51%	0.33	0.39	0.35	88%	0.60	0.82	0.69
Average	76%	0.67	0.68	0.67	89%	0.70	0.89	0.78

Table 8.1: Comparison of the performances of the knowledge-based and ML methods, referred only to class 'sidewalks'. The value reported are the averages for the datasets: Sabbioneta, Mantova, Sacro Monte, and Porto. The performance metrics reported are the average accuracy, Precision (P), Recall (R), and F1-score.

Since ML has brought promising results, the use of DL should also be investigated; for example, the ground surfaces of the point cloud could be rasterized and a DL approach could be implemented for the classification of paving materials, or for the classification of urban elements. Plus, if material recognition procedures were further improved to give results with greater accuracy, for example by exploiting DL techniques, it is possible to imagine that material classification and urban element classification could be performed in a single integrated step. Therefore, future research developments will consider the benefits of a method that combines sidewalk material definition and street and sidewalk classification together.

The last element that should be noted, however, is related to the intrinsic functioning of an AI model: to have good results the model must be trained on a sufficiently large amount of data. In the case of Sabbioneta, the dataset was very large and so this was possible. This last note opens the thought to the possibility of creating very large datasets with many different types of urban areas, with various pavements on the ground surfaces. This could be a useful basis for developing and testing further ML and DL approaches to the classification of such pavements.

8.1.5 The sidewalks attributes computation

Summary

In both the previous cases the result was a classified point cloud, where it was possible to select 'sidewalk' and 'road' points. Focusing only on 'sidewalk' points, some geometric attributes were computed for all sidewalks, subdividing them into several small segments of 2 metres in width, so that the values of the computed features were more representative of the local situation, instead of a single global value. The computed characteristics were: paving material, width, relative elevation of sidewalk from the road, and inclination in the longitudinal and transverse directions. The extracted data were then linked to the sidewalks segments as their specific attribute.

Discussion

The computation of the attributes was done by considering small portions of the sidewalk, which made it possible to have punctual and highly localised information. The result was a high level of detail, with attribute values that globally speaking were variable, even by a few units, along the development of the same sidewalk. Depending on the final purpose for which this information will be used, it might be necessary to make these values more homogeneous and, for example, calculate an average value over a longer section of sidewalk. This is easily achievable and therefore allows information to be suitable and adaptable for several scales of representation.

In general, the extracted values proved to be correct, which was also evident from the comparison with sample measurements carried out on-site by an expert technician. However, there were still some values that were significantly out of scale, as a result of a calculation error, especially in certain sections of the sidewalk in the corner areas. In these areas, the error arises from the fact that the method of calculation of the attributes assumed that the piece of sidewalk analysed was almost rectangular in shape. In corner areas, it was possible that the section was not rectangular and therefore the slope and width values were distorted. To overcome this problem, a limit

value has been added to the attributes which, if exceeded, invalidated the calculated value. These erroneous values were very limited, so a manual intervention quickly solved the problem. Further study for angular zones will certainly provide methods for the correct calculation of these attributes.

The majority of information calculated was correct and quickly extracted, and therefore the method and its result was considered satisfactory and in line with the intended purpose. Of course, the extracted attributes were only a few if compared to the many possibilities given by a point cloud. Comparing for example with the table proposed in the introduction of this thesis (Tab. 1.1) all the information has been calculated; however, going into more detail about the Italian standard and the good practices [8] it is possible to note that many other parameters are worth considering. The parameter related to the height of joints between pavements, for example (max 0.002 m), cannot be calculated as it is below the accuracy of the instrument used. Different surveying instruments should be used to take this into account. Moreover, not only do not all instruments achieve such accuracies, but even a very high-resolution survey of the entire city becomes, in practical terms, difficult to handle, process and elaborate. Vice versa, other parameters, such as the presence of benches for sitting along the route, can be calculated in various ways from the point cloud (e.g., using ML and DL for the classification of urban furniture). Therefore, the presented method is effective because it allows the calculation of some useful attributes for the management of physical accessibility in urban areas, but certainly can be adapted and modified to calculate and include many others.

8.1.6 The vectorisation of sidewalk network and its applications

Summary

Sidewalks segments and their attributes were used to produce a sidewalk network composed of nodes and edges, stored in a vector layer composed of 1780 nodes and 1720 edges. The accurate, realistic, and high-resolution shapefile generated by this thesis work, was then used as a basis for the generation of several products. It was inserted in a GIS project (using the open-source software QGIS), where it was possible to produce technical tables with thematic maps. This also allowed the computation of accessible routes connecting two selected points on the map, taking into account weights related to sidewalks attributes and the type of route chosen. The vector layer itself was also uploaded into an example of a webGIS application, developed during a PhD course on Geospatial web, and used as a queryable vector file; the purpose was to show a possible application of it, which is ideally extensible to a real and publishable webGIS. A final achievement was the possibility of using the spatial data and attributes to enrich the existing Open Street Map database. The procedure of uploading the data in OSM was done by working together with the Italian OSM community.

Discussion

The process of vectorizing the data was done in a very simple and straightforward way: the centre of each portion of the sidewalk becomes a node, by joining neighbouring nodes with edges a network is then created. Although the procedure worked correctly, it was still necessary to manually refine the data: to fix small errors, to add undetected

parts (e.g., some minor roads that were not measured in the data acquisition phase), and to insert missing elements that the method described here was not developed to recognize (e.g., ramps, pedestrian crossings). The system is promising and leads research more and more in the direction of AI-based methods for the development of accessible pedestrian mobility maps.

As well as a semi-automatic procedure for analysing point clouds, a more concrete result of this thesis is a vector file in which the spatial and geometrical information relating to the sidewalk of the urban area of a historical city is conveyed. The choice of the vector file (i.e. a shapefile) has been well thought out and is the result of balancing the desire to reach the largest number of end users with maintaining a rigorousness of the final data. In fact, even if point cloud representation techniques exist, it was considered that a vector file is easier to be understood and used by a wider range of final users independently of their experience in Geomatics. Going on with the comparison with a point cloud, the vector file results to be easier to interoperate with different software, in terms of user skills, required storage capacity, file size, and also in terms of technical requirements for the computer used. Proof of this is the fact that with the vector file alone, it has been possible to produce various finished products ranging from route calculations to maps, to filling web platforms and even to upgrading existing data. The last note is concerning the use of data in OSM, and it can be guessed that the same vector file can similarly be used to update other cartographic datasets, like topographic databases (regional or national) or cadastral maps.

The multiple results (in terms of final products) obtained are only some of the possible uses that can be made of the vector file resulting from the entire process. Further possibilities of use, and the relevant considerations, are set out in the following sections of this concluding chapter.

8.1.7 Summary of the achievements

In this thesis work, a variety of results has been obtained, which goes beyond the shapefile created and the related final products. It is of greater importance to note that in the development of this research work, several methods and working approaches useful for processing point clouds were realized and tested, with the ultimate goal of improving the management of urban physical accessibility. In particular, these results range from simple procedures that are part of larger workflows, to far more complex and comprehensive methods, and can be summarized as:

- a method for the generation of fictitious trajectories exploiting OSM dataset and MMS point clouds;
- a procedure for the definition of the type of crossings between roads (X-shape, T-shape, L-shape), by leveraging on OSM dataset;
- a knowledge-based method for the detection of material of paved surface;
- a method for the same purpose, but implementing supervised ML;
- a workflow for the segmentation of urban point clouds, by classifying it in 'sidewalk' and 'road', implementing unsupervised ML and a voting system;

- a workflow that reaches the same purpose implementing supervised ML techniques;
- a method to implement ML models within large size files;
- methods for the computation of sidewalk geometric attributes (widths, relative elevation, longitudinal and transverse slopes);
- a procedure for the vectorization of a semantically segmented urban point cloud;
- a workflow for the generation of a sidewalk network, complete of attributes.

8.2 Perspectives for the future

The experience gained during the work carried out in this thesis, the workflows developed, the reasoning done and the results obtained allow to outline some possible future developments. In addition, new tests can be planned using approaches that derive from the work presented, and that could improve it. Further applications can be envisaged both in the field of surveying and in the field of data processing, and it can also be imagined to extend the reasoning and fields of application for purposes other than physical accessibility management. In particular, this section will address possible approaches, possibilities of further use for accessibility management, and possibilities of interoperability and integration with City Models and BIM processes.

8.2.1 Survey opportunities

The point cloud analysed in this thesis was generated with a survey conducted with a MMS. This instrument was considered adequate for the purposes of the work, but some considerations could be made. Other surveying instruments can be just as effective and, indeed, can increase the possibilities (in terms of data extraction) of the method.

For example, the use of a different surveying technique, e.g. drone photogrammetry, under appropriate conditions of flight, like altitude sufficiently close to the ground, could allow a much denser point cloud to be obtained. Furthermore, if particular care is taken in the environmental lighting conditions, colour channels could become useful features in segmentation processes.

Another device that would be interesting to test is a PMMS, which could allow to directly measure only the area of the urban surface defined for the pedestrian network. However, the density of points or the accuracy could be lower with respect to the photogrammetric technique mentioned above. A similar discussion could take place for LiDAR mounted on drones.

Developing surveys with the above methods, it could be interesting to evaluate the possibility of deducing information about the state of preservation of the pavement and thus predicting any necessary maintenance. This would also add further information to manage accessibility.

Alternative measuring techniques can then be integrated to better characterise the data already obtained. Existing systems such as Walkbot [28], or wheelchair-mounted accelerometers [171, 172], or pavement distress detection methods [173], or even obstacle and obstruction detection on sidewalk pavement [174] could be used to capture additional data that can then be easily incorporated into the workflow

presented in this thesis. Similarly, other sidewalk-data gathering techniques [175] could be implemented to increase the sidewalk attributes stored in the resulting vector file.

8.2.2 Accessibility-related possibilities

The results obtained in this thesis allow some considerations regarding possible research developments in fields related to accessibility management. In particular, it has already been demonstrated how the vector file and the geometric attributes extracted from the point cloud represent an objective source of information. Therefore, some possible uses (immediate or not) of the extracted data can be outlined. Considering the attributes currently extracted, and those that in fact can potentially be extracted (following modifications and improvements to the current method, or following integration with data from other sources), it is possible to create basic cartographic data that can allow the realisation of Plans for the Elimination of Architectural Barriers (*Piano per l'Eliminazione delle Barriere Architettoniche*, PEBA) or, from a more urban point of view, the Plan for Urban Accessibility (*Piani di Accessibilità Urbana*). In this way, companies, organisations and professional studios involved in the realisation of methods for planning the elimination of urban architectural barriers can have an objective source of information useful for making informed decisions.

Considering the push by the local public administration towards social inclusion and environmental well-being, there is a growing need for tools for a rapid inventory of possible problems or relevant elements concerning physical accessibility in urban areas. This interest is underlined by the guidelines for the drafting of PEBA's recently issued by the Lombardy region, in a resolution of a regional council meeting on 23 November 2021 (resolution no. XI/5555¹). In the same line, other Italian regions have already undertaken similar initiatives, for example, Friuli Venezia Giulia region published in June 2020 guidelines for the implementation of PEBA's in its territories².

Similarly, the same data extracted from this work can be used for purposes related to the service sector. For example, the creation of applications (web-based or for smartphones) that allow end-users (citizens, tourists) to plan their visit to the city, or to be able to navigate in real-time, moving along accessible routes. The realisation of applications of this type has already been developed in the literature, using directly the vector data or merging them with data from other sources [9]. Applications of this kind can be useful for accessible tourism and the planning of accessible routes within historical cities. For example, they can be useful for the planning of artistic-cultural itineraries within historical cities (the Prince's Route, in the case of the city of Mantova³).

Finally, with a broader perspective and also wanting to be foresighted and ready for future changes, even in terms of policy-making, it is important to note that the issue of accessibility is strongly discussed and emphasized by the National Plan for

¹https://www.ledha.it/allegati/LED_t2_notizie_allegati/4004/FILE_Allegato_DGR_5555_del_23nov21_LG_PEBA.pdf

²https://www.regione.fvg.it/rafv/export/sites/default/RAFVG/infrastrutture-lavori-pubblici/lavori-pubblici/FOGLIA30/allegati/lineeGuida_PEBA_14072020.pdf

³<https://www.comune.mantova.it/index.php/scoprire-mantova-top/itinerari/mantova-citta-d-arte-e-cultura/268-il-percorso-del-principe>

Recovery and Resilience⁴ (PNRR), issued by the Italian government, following the covid-19 pandemic. It describes which projects Italy intends to implement with funds received from the European community. In the PNRR, "equal opportunities for people with disabilities" is a strongly felt theme, and addressed in several "missions". In particular, the elimination of architectural barriers and accessible mobility are topics discussed in "missions" 1, 2 and 3. From this perspective, it is therefore important to note that a push, even an economic one at the institutional level, will be given to urban accessibility issues in the near future.

Lastly, it is important to remember that interventions to improve accessibility, even those that are focused and local, can lead to positive economic effects in the long term. For example, a greater flow of tourists into the city may have positive effects on economic activities based on the third sector.

8.2.3 Further accessibility attributes to treat

As mentioned in the Introduction, when dealing with accessibility in the urban environment - in its broadest sense - many factors can be considered that go beyond the geometric attributes of pavements. Considering the list presented on page 6, taken from [8], it can be affirmed that this thesis has focused mainly on the themes more linked to a geometric and dimensional aspect of urban space.

As discussed in detail before, the method described has many potentials for modification and implementation to identify other elements as well. Discussing these issues, it is important to recall them briefly here: dimensional features of urban space, flooring features, presence of wayfinding solutions, crosswalks, urban furniture, public transport stops, and reserved parking lots for people with disability.

The first two have been effectively addressed by this thesis work. Continuing, it can be observed that the presence of wayfinding solutions can be computed on the basis of a possible city model realised starting from the point cloud. On the 3D City Model, given the presence of the third dimension, the possible algorithms developed can better verify and exploit the presence of natural guides, or search for possible obstacles.

The evaluation of crosswalks, which is an interesting point, although not treated in this thesis, has certainly been addressed in the literature, e.g. in [32].

Regarding street furniture, ML algorithms on urban point clouds are also trained in the recognition of benches, lampposts, and other similar elements. Once the furniture element is recognised, inserting a point -which acts as a placeholder- into the vector file is an immediate procedure.

Concerning the analysis of the bus stop, and also the reserved parking spaces, in both cases it is perhaps more interesting to assess whether it is only useful to indicate their absence or, perhaps more interesting, to assess their characteristics. If in the first case it is only important to understand how to locate them quickly (e.g., by making a survey or deriving the data from existing cartographic databases), in the second case it is necessary to implement a series of procedures aimed at searching for certain characteristics by analysing a survey or photographs.

⁴<https://www.governo.it/sites/governo.it/files/PNRR.pdf>

Lastly, it is interesting to consider the possibility of including among the information derived from the point cloud, also the presence of porches (i.e. covered areas) and the computation of shaded areas. This information can be useful for a more informed estimate of the accessible paths, including also the behaviour of the people. For example, in case of rainy or very hot weather, the shaded or covered areas will be the ones preferred by the users of the urban area. If information on these areas is present, it is also possible to make realistic and useful simulations for a better design of spaces and paths.

8.2.4 Other implementations for the method

The method presented in this paper has proved effective for the analysis of a point cloud of an urban environment to identify useful characteristics for accessibility. It is possible to translate the same approach to analyse point clouds of other objects: for example, derived methods can be developed for the analysis of rooms of a building, or other kinds of indoor environments.

Up to now, the reasoning has been developed based on a set of rules (deriving from regulations and good practices) that would make it possible to guarantee and quantify the accessibility of an urban environment. Starting from these requirements, the method has been developed for the specific research of certain elements. Once an effective process has been identified, theoretically it should be possible to apply the same workflow -modifying some steps of the method- for the identification of other elements that may be useful for other purposes, other than accessibility management.

Alternatively, it may be possible to continue and adapt the workflows for an analysis carried out on a point cloud of the interior of a building. PMMSs allow entire buildings to be acquired quickly and would therefore provide support for the development of tests in this direction. Continuing to work in the indoor environment, the same approach can be transferred to Building Information Modelling (BIM). Once the rules and the elements or attributes to be identified have been established, if the base support is no longer the point cloud but becomes a BIM model, the procedures must be adapted to work on parametric models instead of point clouds. It is therefore interesting to think about this comparison and the possibility of duality in the application of the procedure presented.

An interesting discourse can then take place by reasoning on the possible applications in the field of parametric models, therefore talking about BIM models for infrastructures (i.e. InfraBIM), which can be the next step of the research, after the vectorial file generated in this thesis work. The vectorial file can be the starting point for the creation of an IFC file interchangeable and compatible with BIM processes. The extension of the concept of a digital model of the infrastructures to a digital model of the city (i.e. City Models) further opens the fields of application and the possibilities of future tests. Indeed, there are now methods for the rapid, efficient and automatic realisation of city models based on cartographic vector datasets [176].

Lastly, although the method presented produces as output a vector file, it might be interesting to evaluate and test the possibilities given by the direct use of the point cloud, enriched with the computed information related to sidewalks, as the output file of the whole procedure presented. The point cloud, in fact, being a detailed and metrically correct representation of (the historic city in this case), and further

enriched with accessibility information, could become an excellent medium and meeting point for the different figures gravitating around accessibility management. It is clear, however, that further studies are still needed in order to reach the same range of users (i.e., the most diverse) that can be reached to date from the vector file.

8.3 Final comments

In a broad line of reasoning dealing with accessibility issues, referring to architectural barriers, it can be argued that their elimination within public buildings and buildings of cultural interest is of fundamental importance. This is true, but it must be remembered that while the topic and the objective are of great interest and fundamental towards a more inclusive society, it is also important to identify effective and barrier-free routes to connect the different buildings and areas of interest in a city. The usefulness of the system developed by this thesis fits perfectly into this context.

In addition, machine learning approaches for cultural heritage were tested in this work. The need to abstract the problem sufficiently in order to be able to analyse it and make it replicable, becomes a challenge when considering cultural assets, which by their very definition have the character of uniqueness. The work presented here demonstrates the possibility given by AI technologies for cultural heritage as well, and the good results obtained push for more and more study and use of such technologies.

In conclusion, this applied research has shown that it is possible and feasible to develop semi-automatic methods for the rapid extraction of information from a point cloud. In this specific case, the field of application was accessibility management. The results have shown how effective and repeatable the method is, and the arguments made in the discussions point to the first steps towards possible diverse applications of the method. It is possible to change the basis on which the calculations are made (point cloud, BIM model, City Models), or to change the object analysed (indoor environment, urban environment, historical or otherwise), or even readjust the set of rules and elements researched according to the final purpose, which may therefore be different from physical accessibility.

Some final remarks should be done, specifically about:

- Usefulness of the system:
 - analysis of an entire Town;
 - objective data (not subjective interpretations);
 - convey information in a finished and manageable format.
- Final users:
 - Public Administrators, Public Entities;
 - urban planners, designers;
 - citizens, tourists, social associations.

- Scalability of the work:
 - applicable to different data sources;
 - suitable for indoor environments, with minor/medium changes;
 - appropriate for workflow related to BIM, 3D GIS for infrastructures.
- Accessibility opportunities:
 - method focused on sidewalks, but the workflows developed and the method tested may be adapted and exploited also for other urban physical accessibility issues;
 - in the context of PNRR and the recent push for PEBA adoption, the interest in urban accessibility is gaining more attention from policymakers.

Finally, an element that should certainly be underlined and kept in mind is the increasingly necessary interdisciplinarity of research, which can lead to useful results only if developed as the result of the encounter of experts with different skills, sensitivities and capacities, coming from different sectors. The results produced by this group could have practical and applicative effects on the community and could mark a turning point and contribute to the world of research. In this specific case, it was necessary to involve experts from the field of Geomatics, but also data processing and accessibility experts in order to develop processes that can lead to a technological and methodological improvement of some urban management procedures.

Bibliography

- [1] Sebastiano Marconcini. *The project of the "City for All" in cultural heritage sites*. PhD thesis, Dipartimento di architettura, ingegneria delle costruzioni e ambiente costruito, Politecnico di Milano, Milano, March 2020.
- [2] The International Classification of Functioning. *Disability and Health*. World Health Organization, Geneva, 2001.
- [3] World Health Organization. *World Report on Disability 2011*. WHO, Malta, 2011.
- [4] A. Lauria. *Persone reali e progettazione dell'ambiente costruito. L'accessibilità come risorse per la qualità ambientale*. Maggioli Editore, Dogana, 2003.
- [5] S. Marconcini and V. Pracchi. Inclusive Cultural Heritage Sites: Ict As A Tool to Support the Design Process and Share Knowledge. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 42(2/W11):793–800, 2019.
- [6] Susanne Zimmermann-Janschitz. Geographic information systems in the context of disabilities. *Journal of Accessibility and Design for All*, 8(2):161–192, Nov. 2018.
- [7] L. Biagi, M. A. Brovelli, and L. Stucchi. Mapping the accessibility in openstreetmap: A comparison of different techniques. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLIII-B4-2020:229–236, 2020.
- [8] S. Marconcini, D. Treccani, L. Díaz-Vilariño, and A. Adami. A data collection framework for managing accessibility and inclusion in urban heritage. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, VIII-M-1-2021:101–108, 2021.
- [9] Miguel R. Luaces, Jesús A. Fisteus, Luis Sánchez-Fernández, Mario Muñoz-Organero, Jesús Balado, Lucía Díaz-Vilariño, and Henrique Lorenzo. Accessible routes integrating data from multiple sources. *ISPRS International Journal of Geo-Information*, 10(1), 2021.
- [10] E. Estellés-Arolas, R. Navarro-Giner, and F. González-Ladrón-de Guevara. *Crowdsourcing Fundamentals: Definition and Typology*, pages 33–48. Springer International Publishing, Cham, 2015.
- [11] Amin Mobasheri, Jonas Deister, and Holger Dieterich. Wheelmap: the wheelchair accessibility crowdsourcing platform. *Open Geospatial Data, Software and Standards*, 2(1):27, 2017.
- [12] Manaswi Saha, Michael Saugstad, Hanuma Teja Maddali, Aileen Zeng, Ryan Holland, Steven Bower, Aditya Dash, Sage Chen, Anthony Li, Kotaro Hara, and Jon Froehlich. Project sidewalk: A web-based crowdsourcing tool for collecting sidewalk accessibility data at scale. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, pages 1–14, New York, NY, USA, 2019. Association for Computing Machinery.
- [13] Federico Prandi, Marco Soave, Federico Devigili, Raffaele Amicis, and Alkis Astyropoulos. Collaboratively collected geodata to support routing service for disabled people. In George Gartner and Haosheng Huang, editors, *Proceedings of the 11th International Symposium on Location-based Services*, 11 2014.
- [14] Christian Voigt, Susanne Dobner, Mireia Sanz, Stefan Hahmann, and Karsten Gareis. Community engagement strategies for crowdsourcing accessibility information: Paper, wheelmap-tags and mapillary-walks. In *15th International Conference on Computers Helping People with Special Needs*, 07 2016.
- [15] European Commission. Employment, Social Affairs & Inclusion. Persons with disabilities. <https://ec.europa.eu/social>. Accessed on March 2020.
- [16] UN Department of Economic and Social Affairs Disability. <https://www.un.org/development/desa/disabilities/>. Accessed on March 2020.
- [17] Decreto Ministeriale 14 giugno 1989, n. 236, 1989.
- [18] Decreto del Presidente della Repubblica 24 luglio 1996, n. 503, 1996.
- [19] Regione Lombardia. Legge Regionale 20 febbraio 1989, n.6.
- [20] A. Lauria. L'accessibilità come "sapere abilitante" per lo sviluppo umano: il piano per l'accessibilità. *Techne*, 7:125–131, 2014.
- [21] A. Lauria. *I piani per l'accessibilità. Una sfida per promuovere l'autonomia dei cittadini e valorizzare i luoghi dell'abitare*. Gangemi Editore, Roma, 2012.
- [22] I. Argentin, M. Clemente, and T. Empler. *Eliminazione barriere architettoniche. Progettare per un'utenza ampliata*. DEI, Roma, 2008.

Bibliography

- [23] Andrés Serna and Beatriz Marcotegui. Urban accessibility diagnosis from mobile laser scanning data. *ISPRS Journal of Photogrammetry and Remote Sensing*, 84:23–32, 2013.
- [24] J. Balado, L. Díaz-Vilariño, P. Arias, and H. González-Jorge. Automatic classification of urban ground elements from mobile laser scanning data. *Automation in Construction*, 86(September 2017):226–239, 2018.
- [25] Kiichiro Ishikawa, Daisuke Kubo, and Yoshiharu Amano. Curb detection and accessibility evaluation from low-density mobile mapping point cloud data. *International Journal of Automation Technology*, 12(3):376–385, 2018.
- [26] Chengbo Ai and Yichang (James) Tsai. Automated sidewalk assessment method for americans with disabilities act compliance using three-dimensional mobile lidar. *Transportation Research Record*, 2542(1):25–32, 2016.
- [27] Qing Hou and Chengbo Ai. A network-level sidewalk inventory method using mobile LiDAR and deep learning. *Transportation Research Part C: Emerging Technologies*, 119(December 2019):1–14, 2020.
- [28] Miguel Costa, Paulo Cambra, Filipe Moura, and Manuel Marques. Walkbot: A portable system to scan sidewalks. In *2019 IEEE International Smart Cities Conference (ISC2)*, pages 167–172, 2019.
- [29] Ayman Halabya and Khaled El-Rayes. Automated compliance assessment for sidewalks using machine learning. In *Construction Research Congress*, pages 288–295, 2020.
- [30] E. Angelats, M. E. Parés, and P. Kumar. Feasibility of smartphone based photogrammetric point clouds for the generation of accessibility maps. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLII-2:35–41, 2018.
- [31] Daniel Rodrigues, Rui Ramos, and Maisa Tobias. A spatial multicriteria model for urban accessibility mapping. *Proceedings of the ICE - Urban Design and Planning*, 169:1–13, 07 2015.
- [32] Mario Soilán, Belén Riveiro, Ana Sánchez-Rodríguez, and Pedro Arias. Safety assessment on pedestrian crossing environments using MLS data. *Accident Analysis and Prevention*, 111(August 2017):328–337, 2018.
- [33] Tsubasa Maruyama, Satoshi Kanai, and Hiroaki Date. Tripping risk evaluation system based on human behavior simulation in laser-scanned 3d as-is environments. *Automation in Construction*, 85:193–208, 2018.
- [34] J. Balado, L. Díaz-Vilariño, P. Arias, and M. Soilán. Automatic building accessibility diagnosis from point clouds. *Automation in Construction*, 82(March):103–111, 2017.
- [35] J. Balado, L. Díaz-Vilariño, P. Arias, and I. Garrido. Point clouds to indoor/outdoor accessibility diagnosis. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, IV-2/W4:287–293, 2017.
- [36] Lucía Díaz-Vilariño, Pawel Boguslawski, Kourosh Khoshelham, and Henrique Lorenzo. Obstacle-aware indoor pathfinding using point clouds. *ISPRS International Journal of Geo-Information*, 8(5), 2019.
- [37] Tsubasa Maruyama, Satoshi Kanai, and Hiroaki Date. Vision-based wayfinding simulation of digital human model in three dimensional as-is environment models and its application to accessibility evaluation. In *ASME 2016 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, 08 2016.
- [38] Jesús Balado, Lucía Díaz-Vilariño, Pedro Arias, and Henrique Lorenzo. Point clouds for direct pedestrian pathfinding in urban environments. *ISPRS Journal of Photogrammetry and Remote Sensing*, 148(January):184–196, 2019.
- [39] G. López-Pazos, J. Balado, L. Díaz-Vilariño, P. Arias, and M. Scaioni. Pedestrian pathfinding in urban environments: Preliminary results. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, IV-5/W1:35–41, 2017.
- [40] Juan Corso Sarmiento and Jordi Casals Fernandez. Obtaining optimal routes from point cloud surveys. *ACE-Architecture City and Environment*, 11(33), 02 2017.
- [41] Jesus Balado Frias, Lucía Díaz-Vilariño, Ernesto Frías, and Elena González. Generation and enrichment of pedestrian maps with vertical shadows in urban environments from mobile laser scanning data. *Surveying and Geospatial Engineering Journal*, 1:14 – 20, Dec. 2020.
- [42] Qian-Yi Zhou, Jaesik Park, and Vladlen Koltun. Open3D: A modern library for 3D data processing. *arXiv:1801.09847*, 2018.

-
- [43] Charles R. Harris, K. Jarrod Millman, Stéfan J. van der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, Julian Taylor, Sebastian Berg, Nathaniel J. Smith, Robert Kern, Matti Picus, Stephan Hoyer, Marten H. van Kerkwijk, Matthew Brett, Allan Haldane, Jaime Fern'andez del R'io, Mark Wiebe, Pearu Peterson, Pierre G'erald-Marchant, Kevin Sheppard, Tyler Reddy, Warren Weckesser, Hameer Abbasi, Christoph Gohlke, and Travis E. Oliphant. Array programming with NumPy. *Nature*, 585(7825):357–362, September 2020.
- [44] Wes McKinney. Data Structures for Statistical Computing in Python. In Stéfan van der Walt and Jarrod Millman, editors, *Proceedings of the 9th Python in Science Conference*, pages 56 – 61, 2010.
- [45] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.
- [46] The python shapefile library (pyshp). <https://pypi.org/project/pyshp/>. Accessed on March 2021.
- [47] Sean Gillies et al. Shapely: manipulation and analysis of geometric objects. <https://github.com/Toblerity/Shapely>, 2007.
- [48] J. D. Hunter. Matplotlib: A 2d graphics environment. *Computing in Science & Engineering*, 9(3):90–95, 2007.
- [49] Dask Development Team. Dask: Library for dynamic task scheduling. <https://dask.org>, 2016.
- [50] Matthew Rocklin. Dask: Parallel computation with blocked algorithms and task scheduling. In Kathryn Huff and James Bergstra, editors, *Proceedings of the 14th Python in Science Conference*, pages 130 – 136, 2015.
- [51] Martin Weinmann. *Reconstruction and Analysis of 3D Scenes*. Springer International Publishing, Cham, 2016.
- [52] Yuxing Xie, Jiaojiao Tian, and XiaoXiang Zhu. Linking Points With Labels in 3D: A Review of Point Cloud Semantic Segmentation. *IEEE Geoscience and Remote Sensing Magazine*, (March), 2020.
- [53] Emmanuel P. Baltsavias. A comparison between photogrammetry and laser scanning. *ISPRS Journal of Photogrammetry and Remote Sensing*, 54(2):83–94, 1999.
- [54] M.J. Westoby, J. Brasington, N.F. Glasser, M.J. Hambrey, and J.M. Reynolds. Structure-from-motion photogrammetry: A low-cost, effective tool for geoscience applications. *Geomorphology*, 179:300–314, 2012.
- [55] Jie Shan and Charles K. Toth, editors. *Topographic Laser Ranging and Scanning*. Taylor & Francis, CRC Press, Boca Raton, feb 2018.
- [56] Rongjun Qin, Jiaojiao Tian, and Peter Reinartz. 3d change detection - approaches and applications. *ISPRS Journal of Photogrammetry and Remote Sensing*, 122:41–56, 2016.
- [57] Richard Bamler, Michael Eineder, Nico Adam, Xiaoxiang Zhu, and Stefan Gernhardt. Interferometric potential of high resolution spaceborne sar. *Photogrammetrie - Fernerkundung - Geoinformation*, 2009(5):407–419, 11 2009.
- [58] K. Kraus, I.A. Harley, and S. Kyle. *Photogrammetry: Geometry from Images and Laser Scans*. De Gruyter Textbook. De Gruyter, 2011.
- [59] Liang Yuan, Jingjing Guo, and Qian Wang. Automatic classification of common building materials from 3D terrestrial laser scan data. *Automation in Construction*, 110(October 2019):103017, 2020.
- [60] Zhizhong Kang and Juntao Yang. A probabilistic graphical model for the classification of mobile lidar point clouds. *ISPRS Journal of Photogrammetry and Remote Sensing*, 143:108–123, 2018.
- [61] Quan Li and Xiaojun Cheng. Comparison of different feature sets for TLS point cloud classification. *Sensors*, 18(12), 2018.
- [62] M. Weinmann, B. Jutzi, C. Mallet, and M. Weinmann. Geometric features and their relevance for 3d point cloud classification. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, IV-1/W1:157–164, 2017.
- [63] R. Blomley, M. Weinmann, J. Leitloff, and B. Jutzi. Shape distribution features for point cloud analysis - a geometric histogram approach on multiple scales. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, II-3:9–16, 2014.
-

Bibliography

- [64] Martin Weinmann, Boris Jutzi, Stefan Hinz, and Clément Mallet. Semantic point cloud interpretation based on optimal neighborhoods, relevant features and efficient classifiers. *ISPRS Journal of Photogrammetry and Remote Sensing*, 105:286–304, 2015.
- [65] L. Díaz-Vilariño, H. González-Jorge, M. Bueno, P. Arias, and I. Puente. Automatic classification of urban pavements using mobile LiDAR data and roughness descriptors. *Construction and Building Materials*, 102:208–215, 2016.
- [66] E. Grilli, F. Menna, and F. Remondino. A review of point clouds segmentation and classification algorithms. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences - ISPRS Archives*, 42(2W3):339–344, 2017.
- [67] Anh Nguyen and Bac Le. 3D Point Cloud Segmentation. *2013 6th IEEE Conference on Robotics*, pages 1–6, 2013.
- [68] Yusheng Xu and Uwe Stilla. Toward building and civil infrastructure reconstruction from point clouds: A review on data and key techniques. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 14:2857–2885, 2021.
- [69] Weiping Liu, Jia Sun, Wanyi Li, Ting Hu, and Peng Wang. Deep learning on point clouds and its application: A survey. *Sensors (Switzerland)*, 19(19):1–22, 2019.
- [70] Saifullahi Aminu Bello, Shangshu Yu, Cheng Wang, Jibril Muhammad Adam, and Jonathan Li. Review: Deep learning on 3d point clouds. *Remote Sensing*, 12(11), 2020.
- [71] T Rabbani, F.A. Heuvel, and George Vosselman. Segmentation of point clouds using smoothness constraint. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 36, 01 2006.
- [72] D Tóvári and Norbert Pfeifer. Segmentation based robust interpolation- a new approach to laser data filtering. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 36, 05 2012.
- [73] P.J. Besl and R.C. Jain. Segmentation through variable-order surface fitting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10(2):167–192, 1988.
- [74] Abdul Nurunnabi, David Belton, and West Geoff. Robust segmentation for multiple planar surface extraction in laser scanning 3d point cloud data. In *Proceedings - International Conference on Pattern Recognition*, 11 2012.
- [75] D.H. Ballard. Generalizing the hough transform to detect arbitrary shapes. *Pattern Recognition*, 13(2):111–122, 1981.
- [76] Martin A. Fischler and Robert C. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24(6):381–395, jun 1981.
- [77] Geoffrey E. Hinton. Connectionist learning procedures. *Artificial Intelligence*, 40(1):185–234, 1989.
- [78] R. Qi Charles, Hao Su, Mo Kaichun, and Leonidas J. Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 77–85, 2017.
- [79] C. Qi, L. Yi, Hao Su, and L. Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In *NIPS*, 2017.
- [80] Yangyan Li, Rui Bu, Mingchao Sun, Wei Wu, X. Di, and Baoquan Chen. Pointcnn: Convolution on X-Transformed points. *arXiv: Computer Vision and Pattern Recognition*, 2018.
- [81] E. Grilli, E. Özdemir, and F. Remondino. Application of machine and deep learning strategies for the classification of heritage point clouds. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLII-4/W18:447–454, 2019.
- [82] E. Grilli, E. M. Farella, A. Torresani, and F. Remondino. Geometric features analysis for the classification of cultural heritage point clouds. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLII-2/W15:541–548, 2019.
- [83] Simone Teruggi, Eleonora Grilli, Michele Russo, Francesco Fassi, and Fabio Remondino. A hierarchical machine learning approach for multi-level and multi-resolution 3d point cloud classification. *Remote Sensing*, 12(16), 2020.

-
- [84] Valeria Croce, Gabriella Caroti, Livio De Luca, Kévin Jacquot, Andrea Piemonte, and Philippe Véron. From the semantic point cloud to heritage-building information modeling: A semiautomatic approach exploiting machine learning. *Remote Sensing*, 13(3), 2021.
- [85] Yanjun Wang, Qi Chen, Qing Zhu, Lin Liu, Chaokui Li, and Donyong Zheng. A survey of mobile laser scanning applications and key techniques over urban areas. *Remote Sensing*, 11(13):1–20, 2019.
- [86] Mohamed Boussaha, Eduardo Fernandez-Moral, Bruno Vallet, and Patrick Rives. On the Production of Semantic and Textured 3D Meshes of Large scale Urban Environments from Mobile Mapping Images and LiDAR scans. In *RFIAP 2018, Reconnaissance des Formes, Image, Apprentissage et Perception*, Marne la Vallée, France, June 2018.
- [87] Pouria Babahajiani, Lixin Fan, Joni-Kristian Kämäräinen, and Moncef Gabbouj. Urban 3D segmentation and modelling from street view images and LiDAR point clouds. *Machine Vision and Applications*, 28(7):679–694, oct 2017.
- [88] Rongjun Qin and Armin Gruen. 3d change detection at street level using mobile laser scanning point clouds and terrestrial images. *ISPRS Journal of Photogrammetry and Remote Sensing*, 90:23–35, 2014.
- [89] Charles Thomson and Jan Boehm. Automatic geometry generation from point clouds for BIM. *Remote Sensing*, 7(9):11753–11775, 2015.
- [90] Erzhuo Che, Jaehoon Jung, and Michael J. Olsen. Object recognition, segmentation, and classification of mobile laser scanning point clouds: A state of the art review. *Sensors*, 19(4), 2019.
- [91] Lingfei Ma, Ying Li, Jonathan Li, Cheng Wang, Ruisheng Wang, and Michael A. Chapman. Mobile laser scanned point-clouds for road object detection and extraction: A review. *Remote Sensing*, 10(10):1–33, 2018.
- [92] Mario Soilán, Ana Sánchez-Rodríguez, Pablo del Río-Barral, Carlos Perez-Collazo, Pedro Arias, and Belén Riveiro. Review of laser scanning technologies and their applications for road and railway infrastructure monitoring. *Infrastructures*, 4(4), 2019.
- [93] Manohar Yadav, Ajai Kumar Singh, and Bharat Lohani. Extraction of road surface from mobile lidar data of complex road environment. *International Journal of Remote Sensing*, 38(16):4655–4682, 2017.
- [94] Jenny Guo, Meng-Ju Tsai, and Jen-Yu Han. Automatic reconstruction of road surface features by using terrestrial mobile lidar. *Automation in Construction*, 58:165–175, 2015.
- [95] Mario Soilán, Belén Riveiro, Joaquín Martínez-Sánchez, and Pedro Arias. Segmentation and classification of road markings using mls data. *ISPRS Journal of Photogrammetry and Remote Sensing*, 123:94–103, 2017.
- [96] Chenglu Wen, Xiaotian Sun, Jonathan Li, Cheng Wang, Yan Guo, and Ayman Habib. A deep learning framework for road marking extraction, classification and completion from mobile laser scanning point clouds. *ISPRS Journal of Photogrammetry and Remote Sensing*, 147:178–192, 2019.
- [97] Zhao, He. Recognizing features in mobile laser scanning point clouds towards 3d high-definition road maps for autonomous vehicles. Master’s thesis, 2017.
- [98] Lingfei Ma, Ying Li, Jonathan Li, Zilong Zhong, and Michael A. Chapman. Generation of horizontally curved driving lines in HD maps using mobile laser scanning point clouds. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 12(5):1572–1586, 2019.
- [99] Yi-Chang Tsai, Vivek Kaul, and Russell M Mersereau. Critical assessment of pavement distress segmentation methods. *Journal of transportation engineering*, 136(1):11–19, 2010.
- [100] Ying Cheng, Zhang Xiong, and YH Wang. Improved classical hough transform applied to the manhole cover’s detection and location. *Opt. Tech*, 32(S1):504–508, 2006.
- [101] Naoki Tanaka and Kenji Uematsu. A crack detection method in road surface images using morphology. *MVA*, 98:17–19, 1998.
- [102] Haiyan Guan, Yongtao Yu, Jonathan Li, Pengfei Liu, Haohao Zhao, and Cheng Wang. Automated extraction of manhole covers using mobile lidar data. *Remote sensing letters*, 5(12):1042–1050, 2014.
- [103] T Saar and O Talvik. Automatic asphalt pavement crack detection and classification using neural networks. In *2010 12th Biennial Baltic Electronics Conference*, pages 345–348. IEEE, 2010.
- [104] Yongtao Yu, Haiyan Guan, and Zheng Ji. Automated detection of urban road manhole covers using mobile laser scanning data. *IEEE Transactions on Intelligent Transportation Systems*, 16(6):3258–3269, 2015.

Bibliography

- [105] Chenglu Wen, Jonathan Li, Huan Luo, Yongtao Yu, Zhipeng Cai, Hanyun Wang, and Cheng Wang. Spatial-related traffic sign inspection for inventory purposes using mobile laser scanning data. *IEEE Transactions on Intelligent Transportation Systems*, 17(1):27–37, 2015.
- [106] Hanyun Wang, Cheng Wang, Huan Luo, Peng Li, Ming Cheng, Chenglu Wen, and Jonathan Li. Object detection in terrestrial laser scanning point clouds based on hough forest. *IEEE Geoscience and Remote Sensing Letters*, 11(10):1807–1811, 2014.
- [107] Lin Li, Dalin Li, Haihong Zhu, and You Li. A dual growing method for the automatic extraction of individual trees from mobile laser scanning data. *ISPRS Journal of Photogrammetry and Remote Sensing*, 120:37–52, 2016.
- [108] Xinhui Zou, Ming Cheng, Cheng Wang, Yan Xia, and Jonathan Li. Tree classification in complex forest point clouds based on deep learning. *IEEE Geoscience and Remote Sensing Letters*, 14(12):2360–2364, 2017.
- [109] Daniel Munoz, J. Andrew Bagnell, Nicolas Vandapel, and Martial Hebert. Contextual classification with functional max-margin markov networks. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 975–982, 2009.
- [110] Andrés Serna, Beatriz Marcotegui, François Goulette, and Jean-Emmanuel Deschaud. Paris-rue-Madame database: a 3D mobile laser scanner dataset for benchmarking urban detection, segmentation and classification methods. In *4th International Conference on Pattern Recognition, Applications and Methods ICPRAM 2014*, Angers, France, March 2014.
- [111] Bruno Vallet, Mathieu Brédif, Andres Serna, Beatriz Marcotegui, and Nicolas Paparoditis. TerraMobilita/iQmulus urban point cloud analysis benchmark. *Computers and Graphics (Pergamon)*, 49:126–133, 2015.
- [112] Xavier Roynard, Jean Emmanuel Deschaud, and François Goulette. Paris-Lille-3D: A large and high-quality ground-truth urban point cloud dataset for automatic segmentation and classification. *International Journal of Robotics Research*, 37(6):545–557, 2018.
- [113] Rolando Arenas, Blanca Arellano, and Josep Roca. City without barriers, ICT tools for the universal accessibility. study cases in barcelona. In *International Conference on Virtual City and Territory. "Back to the Sense of the City"*, Krakow, Poland, 07 2016.
- [114] Angelo Lorenzi. *Sabbioneta. Progetto di una città*. LetteraVentidue, Siracusa, 2020.
- [115] K.W. Forster. *From "Rocca" to "Civitas": urban planning at Sabbioneta*. Istituto Editoriale Italiano, 1969.
- [116] Susanna Sassi. *Mantova e Sabbioneta: linee guida per il progetto dello spazio pubblico urbano*. Ediguida, Mantova, 2015.
- [117] Grazia Tucci and Valentina Bonora. New technologies for cultural heritage documentation and conservation: The role of geomatics. In *ICONARCH - I Architecture and technology international congress*, Konya, Turkey, 2012.
- [118] Fabio Remondino and Alessandro Rizzi. Reality-based 3D documentation of natural and cultural heritage sites-techniques, problems, and examples. *Applied Geomatics*, 2(3):85–100, 2010.
- [119] Sara Gonizzi Barsanti, Fabio Remondino, and Domenico Visintini. Photogrammetry and laser scanning for archaeological site 3D modeling - Some critical issues. *CEUR Workshop Proceedings*, 948(January), 2012.
- [120] Ruisheng Wang, Jiju Peethambaran, and Dong Chen. LiDAR Point Clouds to 3-D Urban Models : A Review. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 11(2):606–627, 2018.
- [121] Mahdi Javanmardi, Ehsan Javanmardi, Yanlei Gu, and Shunsuke Kamijo. Towards high-definition 3D urban mapping: Road feature-based registration of mobile mapping systems and aerial imagery. *Remote Sensing*, 9(10), 2017.
- [122] Fabio Remondino, Isabella Toschi, and Simone Orlandini. Mobile mapping systems: recenti sviluppi e caso applicativo. *GEOmedia*, 19(4), 2015.
- [123] Bharat Lohani and Manohar Yadav. Mobile lidar systems today and tomorrow: The promising future of mobile mapping and laser scanning. *GIM international*, 07 2018.
- [124] Cheng Wang, Chenglu Wen, Yudi Dai, Shangshu Yu, and Minghao Liu. Urban 3D modeling with mobile laser scanning: a review. *Virtual Reality & Intelligent Hardware*, 2(3):175–212, 2020.

-
- [125] I. Puente, H. González-Jorge, J. Martínez-Sánchez, and P. Arias. Review of mobile mapping and surveying technologies. *Measurement*, 46(7):2127–2145, 2013.
- [126] E.D. Kaplan and C. Hegarty. *Understanding GPS/GNSS: Principles and Applications, Third Edition*. GNSS Technology and Applications Series. Artech House Publishers, 2017.
- [127] Paul Groves. *Principles of GNSS, Inertial, and Multisensor Integrated Navigation Systems, Second Edition*. 2013.
- [128] M.S. Grewal, L.R. Weill, and A.P. Andrews. Fundamentals of satellite and inertial navigation. In *Global Positioning Systems, Inertial Navigation, and Integration*, chapter 2, pages 18–52. John Wiley Sons, Ltd, 2007.
- [129] Conor Cahalane, Paul Lewis, Conor P. McElhinney, and Timothy McCarthy. Optimising mobile mapping system laser scanner orientation. *ISPRS International Journal of Geo-Information*, 4(1):302–319, 2015.
- [130] Manuel Rodríguez-Martín, Pablo Rodríguez-Gonzálvez, Esteban Ruiz de Oña Crespo, and Diego González-Aguilera. Validation of portable mobile mapping system for inspection tasks in thermal and fluid-mechanical facilities. *Remote Sensing*, 11(19), 2019.
- [131] Erica Nocerino, Pablo Rodríguez-Gonzálvez, and Fabio Menna. Introduction to mobile mapping with portable systems. In Belén Riveiro and Roderik Lindenbergh, editors, *Laser Scanning*, chapter 4. CRC Press, oct 2019.
- [132] Grazia Tucci, Domenico Visintini, Valentina Bonora, and Erica Isabella Parisi. Examination of indoor mobile mapping systems in a diversified internal/external test field. *Applied Sciences (Switzerland)*, 8(3), 2018.
- [133] Andrea di Filippo, Luis Javier Sánchez-Aparicio, Salvatore Barba, José Antonio Martín-Jiménez, Rocío Mora, and Diego González Aguilera. Use of a wearable mobile laser system in seamless indoor 3D mapping of a complex historical site. *Remote Sensing*, 10(12), 2018.
- [134] Ville V. Lehtola, Harri Kaartinen, Andreas Nüchter, Risto Kaijaluoto, Antero Kukko, Paula Litkey, Eija Honkavaara, Tomi Rosnell, Matti T. Vaaja, Juho Pekka Virtanen, Matti Kurkela, Aimad El Issaoui, Lingli Zhu, Anttoni Jaakkola, and Juha Hyypä. Comparison of the selected state-of-the-art 3D indoor scanning and point cloud generation methods. *Remote Sensing*, 9(8):1–26, 2017.
- [135] F. Fassi and L. Perfetti. Backpack mobile mapping solution for DTM extraction of large inaccessible spaces. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences - ISPRS Archives*, 42(2/W15):473–480, 2019.
- [136] Francini S Giannetti F Nicolaci A Travaglini D Massai L Giambastiani Y Terranova C D8217;Amico G, Vangi E and Chirici G. Are we ready for a national forest information system? state of the art of forest maps and airborne laser scanning data availability in italy. *iForest - Biogeosciences and Forestry*, (2):144–154, 2021.
- [137] M. Yadav, B. Lohani, and A. K. Singh. Road surface detection from mobile lidar data. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 4(5):95–101, 2018.
- [138] D. Treccani, L. Díaz-Vilariño, and A. Adami. Sidewalk detection and pavement characterisation in historic urban environments from point clouds: Preliminary results. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLIII-B4-2021:243–249, 2021.
- [139] Timo Hackel, Jan D. Wegner, and Konrad Schindler. Contour detection in unstructured 3d point clouds. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1610–1618, 2016.
- [140] Douros I. and Buxton B. Three-dimensional surface curvature estimation using quadric surface patches. In *Computer science*, 2002.
- [141] Maria Rosaria De Blasiis, Alessandro Di Benedetto, and Margherita Fiani. Mobile laser scanning data for the evaluation of pavement surface distress. *Remote Sensing*, 12(6), 2020.
- [142] S. P. Lloyd. Least squares quantization in pcm. *IEEE Trans. Inf. Theory*, 28:129–136, 1982.
- [143] Xin Jin and Jiawei Han. *K-Means Clustering*, pages 563–564. Springer US, Boston, MA, 2010.
- [144] Trupti M Kodinariya and Prashant R Makwana. Review on determining number of cluster in k-means clustering. *International Journal*, 1(6):90–95, 2013.
- [145] Joerg Sander. *Density-Based Clustering*, pages 270–273. Springer US, Boston, MA, 2010.
-

Bibliography

- [146] Martin Ester, Hans-Peter Kriegel, Jörg Sander, Xiaowei Xu, et al. A density-based algorithm for discovering clusters in large spatial databases with noise. In *kdd*, volume 96, pages 226–231, 1996.
- [147] Berit Schmitz, Christoph Holst, Tomislav Medic, Derek D. Lichti, and Heiner Kuhlmann. How to efficiently determine the range precision of 3d terrestrial laser scanners. *Sensors*, 19(6), 2019.
- [148] Kai Ming Ting. Confusion matrix. In Claude Sammut and Geoffrey I. Webb, editors, *Encyclopedia of Machine Learning*, pages 209–209, Boston, MA, 2010. Springer US.
- [149] Kai Ming Ting. Precision and recall. In Claude Sammut and Geoffrey I. Webb, editors, *Encyclopedia of Machine Learning*, pages 781–781, Boston, MA, 2010. Springer US.
- [150] Virginia Dignum. What is artificial intelligence? In *Responsible Artificial Intelligence: How to Develop and Use AI in a Responsible Way*, pages 9–34, Cham, 2019. Springer International Publishing.
- [151] Sebastian Raschka and Vahid Mirjalili. Giving computers the ability to learn from data. In *Python Machine Learning - Third Edition*, pages 1–17. Packt, 2019.
- [152] Gopinath Rebala, Ajay Ravi, and Sanjay Churiwala. Machine learning definition and basics, 2019.
- [153] Supervised learning. In Claude Sammut and Geoffrey I. Webb, editors, *Encyclopedia of Machine Learning and Data Mining*, pages 1213–1214, Boston, MA, 2017. Springer US.
- [154] Unsupervised learning. In Claude Sammut and Geoffrey I. Webb, editors, *Encyclopedia of Machine Learning and Data Mining*, pages 1304–1304, Boston, MA, 2017. Springer US.
- [155] Sandro Skansi. Machine learning basics. In *Introduction to Deep Learning: From Logical Calculus to Artificial Intelligence*, pages 51–77, Cham, 2018. Springer International Publishing.
- [156] Gopinath Rebala, Ajay Ravi, and Sanjay Churiwala. Deep learning. In *An Introduction to Machine Learning*, pages 127–140, Cham, 2019. Springer International Publishing.
- [157] Jürgen Schmidhuber. Deep learning in neural networks: An overview. *Neural Networks*, 61:85–117, 2015.
- [158] Huan Liu. *Feature Selection*, pages 402–406. Springer US, Boston, MA, 2010.
- [159] Ian Jolliffe. Principal component analysis. In Miodrag Lovric, editor, *International Encyclopedia of Statistical Science*, pages 1094–1096, Berlin, Heidelberg, 2011. Springer Berlin Heidelberg.
- [160] D. Treccani, L. Díaz-Vilariño, and A. Adami. Accessible path finding for historic urban environments: Feature extraction and vectorization from point clouds. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLVI-2/W1-2022:497–504, 2022.
- [161] Daniele Treccani and Sebastiano Marconcini. Innovative accesibility data inventory tools for urban environments in historic sites. *Studies in health technology and informatics*, 297:515–522, September 2022.
- [162] Florent Poux. *The Smart Point Cloud: Structuring 3D intelligent point data*. PhD thesis, 06 2019.
- [163] Stephan Nebiker, Susanne Bleisch, and Martin Christen. Rich point clouds in virtual globes – a new paradigm in city modeling? *Computers, Environment and Urban Systems*, 34(6):508–517, 2010. GeoVisualization and the Digital City.
- [164] G.-A. Nys, A. Kharroubi, F. Poux, and R. Billen. An extension of cityjson to support point clouds. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLIII-B4-2021:301–306, 2021.
- [165] Ole Wegen, Jürgen Döllner, and Rico Richter. Concepts and challenges for 4d point clouds as a foundation of conscious, smart city systems. In Osvaldo Gervasi, Beniamino Murgante, Sanjay Misra, Ana Maria A. C. Rocha, and Chiara Garau, editors, *Computational Science and Its Applications – ICCSA 2022 Workshops*, pages 589–605, Cham, 2022. Springer International Publishing.
- [166] Florian W. Fichtner, Abdoulaye A. Diakitè, Sisi Zlatanova, and Robert Voûte. Semantic enrichment of octree structured point clouds for multi-story 3d pathfinding. *Transactions in GIS*, 22(1):233–248, 2018.
- [167] B. R. Staats, A. A. Diakitè, R. L. Voûte, and S. Zlatanova. Automatic generation of indoor navigable space using a point cloud and its scanner trajectory. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, IV-2/W4:393–400, 2017.
- [168] E. Barçon, T. Landes, P. Grussenmeyer, and G. Berson. Vectorization of urban mls point clouds: A sequential approach using cross sections. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLIII-B2-2022:351–358, 2022.

- [169] H.-Y. Pai, J.-C. Zeng, M.-L. Tsai, and K.-W. Cheng. Automated modelling of road for high-definition maps with.opendrive format utilizing mobile mapping measurements. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLIII-B1-2022:263–270, 2022.
- [170] D. Treccani, J. Balado, A. Fernández, A. Adami, and L. Díaz-Vilariño. A deep learning approach for the recognition of urban ground pavements in historical sites. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLIII-B4-2022:321–326, 2022.
- [171] Hyunsoo Kim, Changbum R. Ahn, and Kanghyeok Yang. A people-centric sensing approach to detecting sidewalk defects. *Advanced Engineering Informatics*, 30(4):660–671, 2016.
- [172] Satoshi Kobayashi, Ren Katsurada, and Tatsuhito Hasegawa. Estimation of sidewalk surface type with a smartphone. In *Proceedings of the 2019 7th International Conference on Information Technology: IoT and Smart City*, ICIT 2019, page 497–502, New York, NY, USA, 2019. Association for Computing Machinery.
- [173] Tom B.J. Coenen and Amir Golroo. A review on automated pavement distress detection methods. *Cogent Engineering*, 4(1):1374822, 2017.
- [174] Nicholas A. Coppola and Wesley E. Marshall. Sidewalk static obstructions and their impact on clear width. *Transportation Research Record*, 2675(6):200–212, 2021.
- [175] Chang Sun, Jia Su, Wenpeng Ren, and Yong Guan. Wide-view sidewalk dataset based pedestrian safety application. *IEEE Access*, 7:151399–151408, 2019.
- [176] Hugo Ledoux, Filip Biljecki, Balázs Dukai, Kavisha Kumar, Ravi Peters, Jantien Stoter, and Tom Commandeur. 3dfier: automatic reconstruction of 3D city models. *Journal of Open Source Software*, 6(57):2866, 2021.



Semantic Segmentation results

This appendix shows the results of the semantic segmentation of the point cloud, obtained both using the knowledge-based method (Chapter 4) and a Machine Learning approach (Chapter 5). Confusion matrices, precision metrics (Precision, Recall, F1-score), and various renders of the tested dataset are presented, with the points coloured according to the class to which they belong (they are coloured in red in the case of misclassification). In the following, the results are then broken down by dataset: Sabbioneta with all its tracks, Mantova, Sacro Monte di Domodossola, and Porto with its three cases.

The classes for the knowledge-based approach were 'sidewalk', 'road', 'other'. For the ML approach, instead, the 'other' class was not considered, and only 'sidewalk' and 'road' were predicted. Then, for the ML approach, for the Sabbioneta datasets the classification predicted with the model trained on Track-C was presented; for the other datasets also a classification with the model trained on a portion of the dataset itself is presented, and the training dataset is highlighted in the related image.

A.1 Sabbioneta - Track A

This track starts from where via Vespasiano Gonzaga faces Piazza d'Armi, follows the road towards Piazza Ducale, passing via Scamozzi, crosses the square and follows via Assunta, skirting the block of buildings behind the church, then returns to Piazza Ducale, follows via Leoni, turning right into via Incononata.

This route includes several 90-degree bends, passing (totally or partially) through squares, streets of different widths, and passing in front of porticoed areas, for a total of 534 metres of road analysed and for a total of 32 million points on the ground. The sidewalks are different: in the first section, up to Piazza Ducale, the road is paved with cobblestones and the sidewalk raised and with a kerb, is made of stone; from Piazza Ducale onwards the road is paved with cobblestone, with sidewalks at the same level as the road, made of bricks; the final street of this route, Via Incononata, is asphalted and without a sidewalk.

Here are the confusion matrices and precision metrics for the knowledge-based method (with both the cases: with and without via Incononata) and the ML approach.

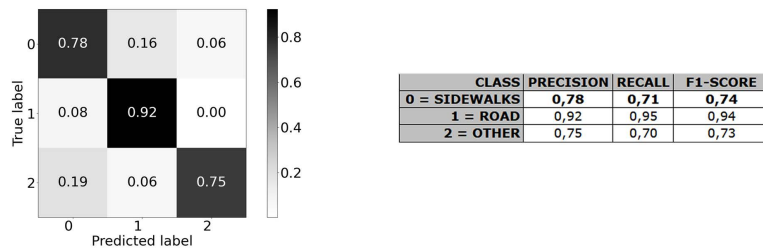


Figure A.1: Confusion matrix and precision metrics for track-A with knowledge-based approach, considering also via Incononata portion of dataset.

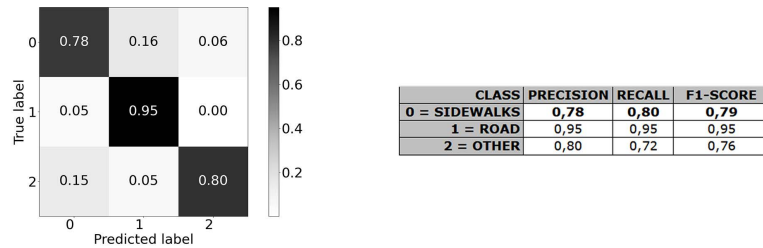


Figure A.2: Confusion matrix and precision metrics for track-A with knowledge-based approach, without considering via Incononata portion of dataset.

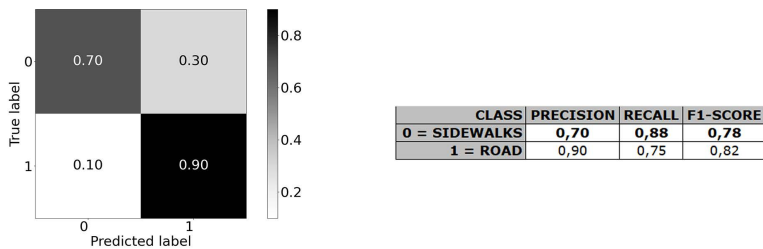


Figure A.3: Confusion matrix and precision metrics for track-A with ML approach.

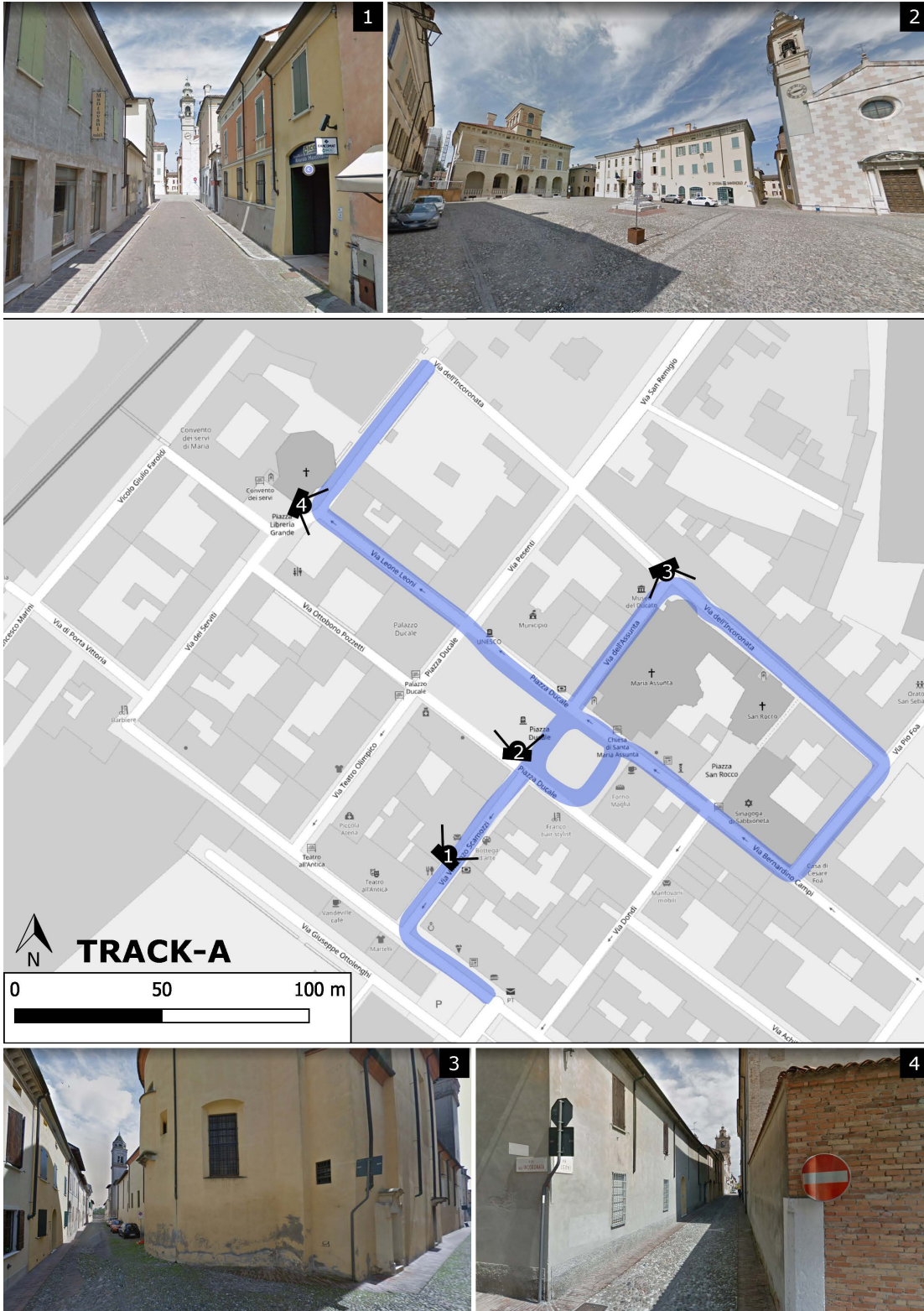


Figure A.4: Map of the path followed for the survey along Track-A, together with some images of the road and the city. Images were elaborated from Google Earth.

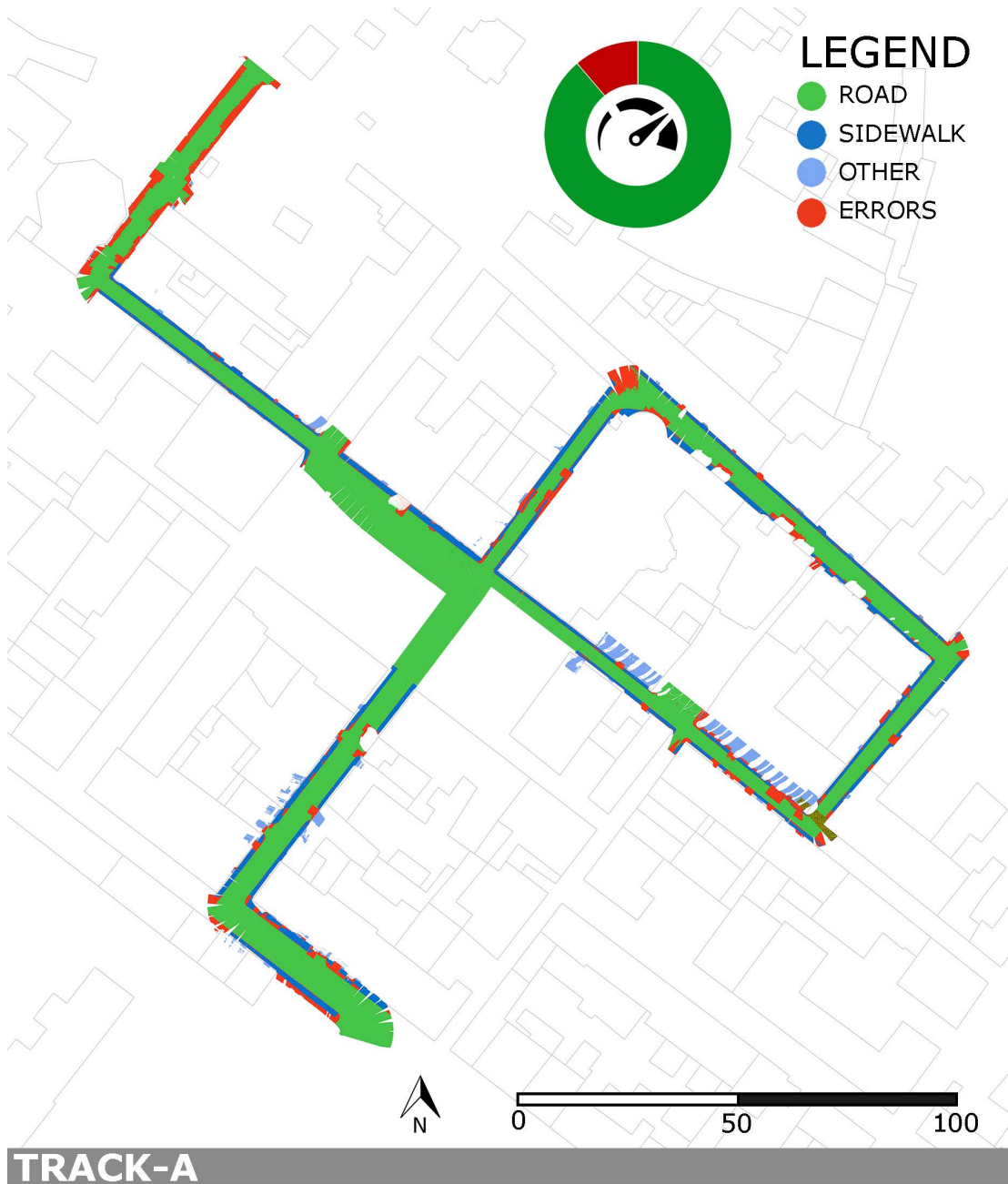


Figure A.5: Top view of the point cloud of Track-A after semantic segmentation with the knowledge-based method. Red points are errors. On the top part of the image, there is a pie chart with an overview of the whole points in the point cloud.



Figure A.6: Top view of the point cloud of Track-A after semantic segmentation with ML method. Red points are errors. On the top part of the image, there is a pie chart with an overview of the whole points in the point cloud.

A.2 Sabbioneta - Track B

This track starts from via Incononata and moves in the opposite direction with respect the previous track. After via Incononata, the path passes through Piazza Libreria Grande, via dei Serviti and concludes on via Porta Imperiale, just in front of the door in the fortified walls.

This route is a short path, 258 metres (13 million points), composed of straight portions, some curves and the passage to a small square. The paving surface is only asphalt without sidewalks for via Incononata, cobblestone for the Piazza and the road, with brick sidewalks. As for the previous track, via Incononata is a source of errors, because its shape is such that the algorithm assumptions are not met, and it can fail. Looking at the rendered image of the point cloud, it is very clear that the first half of the track (via Incononata and piazza Libreria Grande) has high errors in sidewalks recognition, for the reasons mentioned before.

Here are the confusion matrices and precision metrics for the knowledge-based method (with both the cases: with and without via Incononata). The ML approach was not tested on track B.

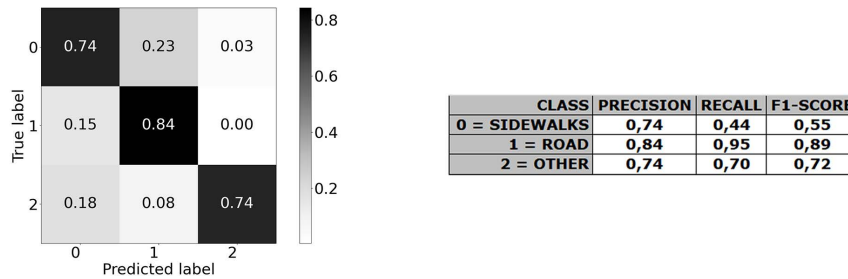


Figure A.7: Confusion matrix and precision metrics for track-B with knowledge-based approach, considering also via Incononata portion of dataset.

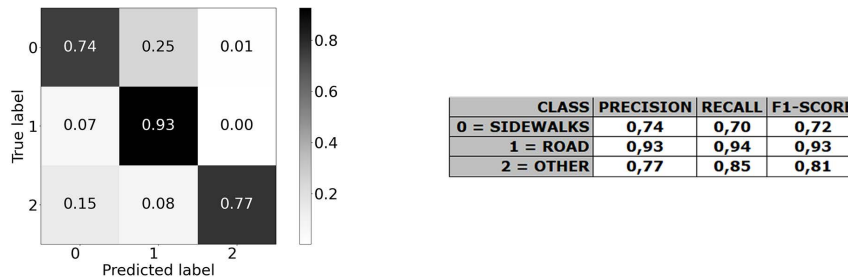


Figure A.8: Confusion matrix and precision metrics for track-B with knowledge-based approach, without considering via Incononata portion of dataset.

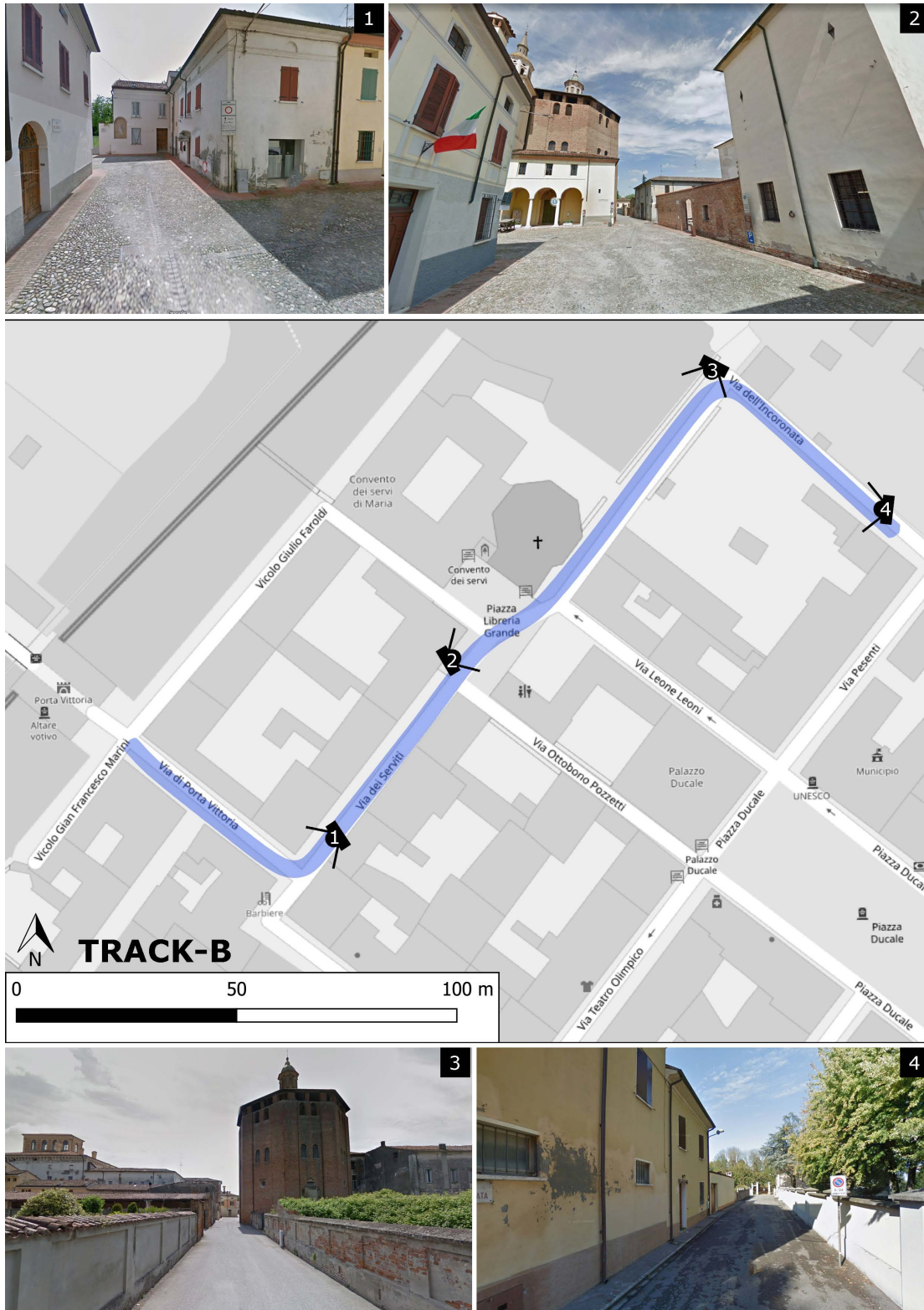


Figure A.9: Map of the path followed for the survey along Track-B, together with some images of the road and the city. Images were elaborated from Google Earth.



Figure A.10: Top view of the point cloud of Track-B after semantic segmentation with the knowledge-based method. Red points are errors. On the top part of the image, there is a pie chart with an overview of the whole points in the point cloud.

A.3 Sabbioneta - Track C

This track rides along the decumanus maximum of the city. It starts from Porta Vittoria, moves along via Vespasiano Gonzaga, passes on via Colonna and concludes its path on Porta Imperiale.

Track C is one of the most representative of Sabbioneta, because it has several strange and interesting conditions, and it was used for the first tests while developing the algorithm. It has a length of 576 metres, for a total of 31 million points. Along its path there are narrow roads, 90 degrees curves, crossings with other roads, it passes near porches and there is also an area where suddenly the road doubles its width. Plus, sidewalks are sometimes at the same height as the road and sometimes at a higher level. The paving surfaces are various; for the road, cobblestone is used in the first half of the path, and sampietrini for the second half; while sidewalks pavings are made of bricks in the first portion and stone in the second one.

Here are the confusion matrices and precision metrics for the knowledge-based method and the ML approach.

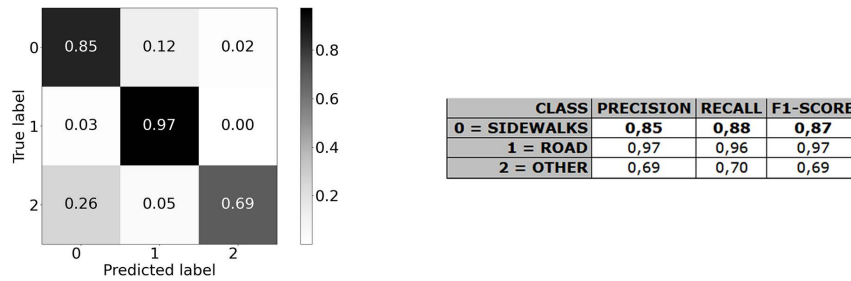


Figure A.11: Confusion matrix and precision metrics for track-C with knowledge-based approach.

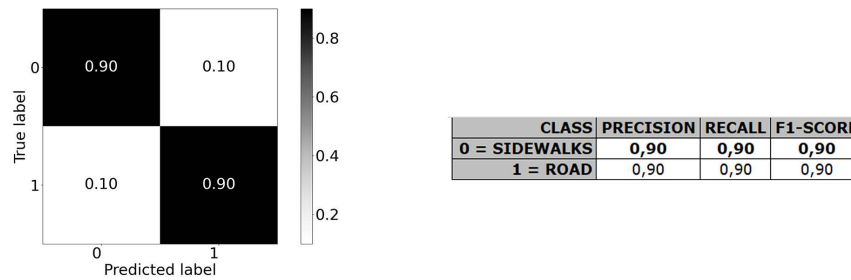


Figure A.12: Confusion matrix and precision metrics for track-C with ML approach.

Appendix A. Semantic Segmentation results

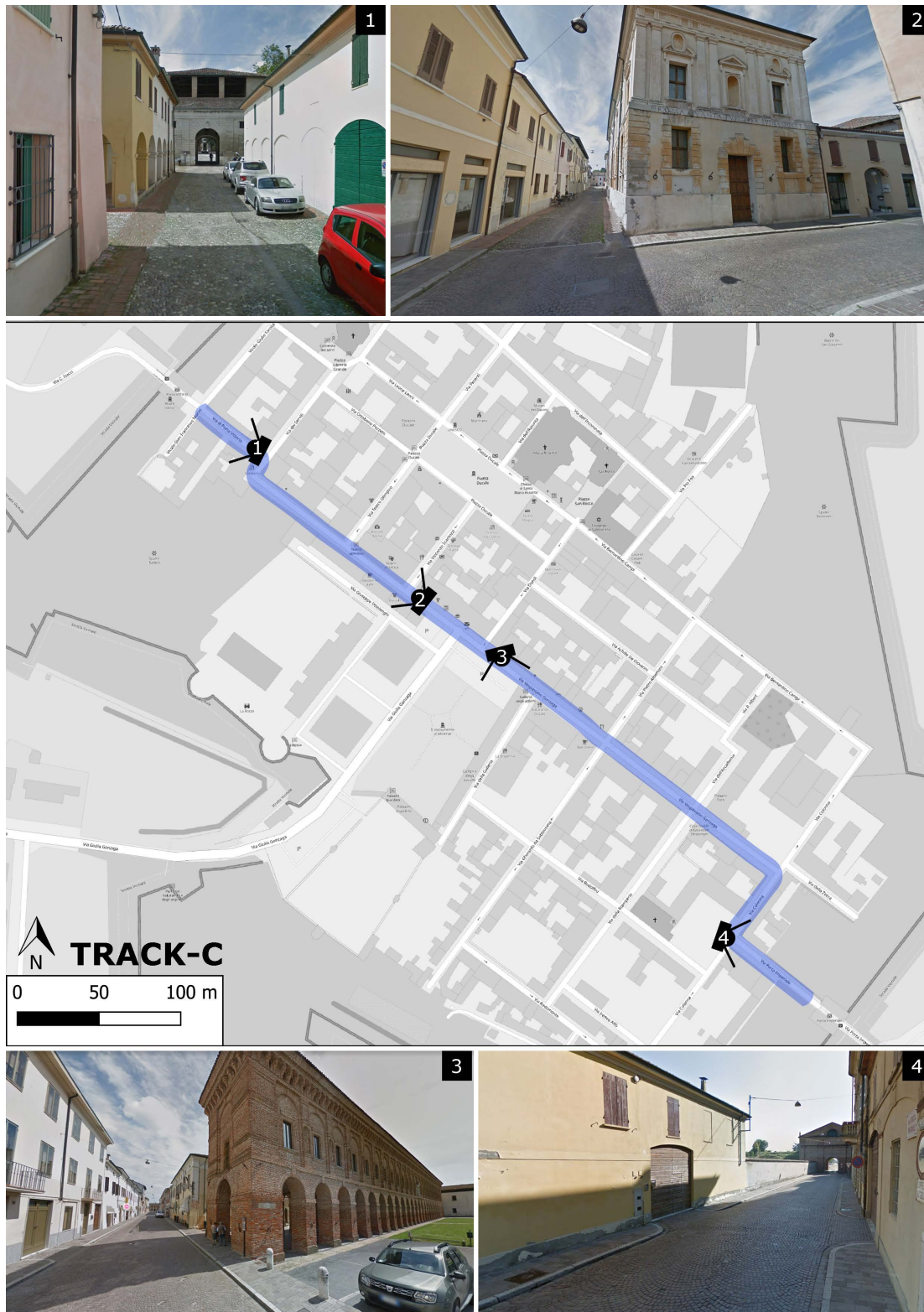


Figure A.13: Map of the path followed for the survey along Track-C, together with some images of the road and the city. Images were elaborated from Google Earth.

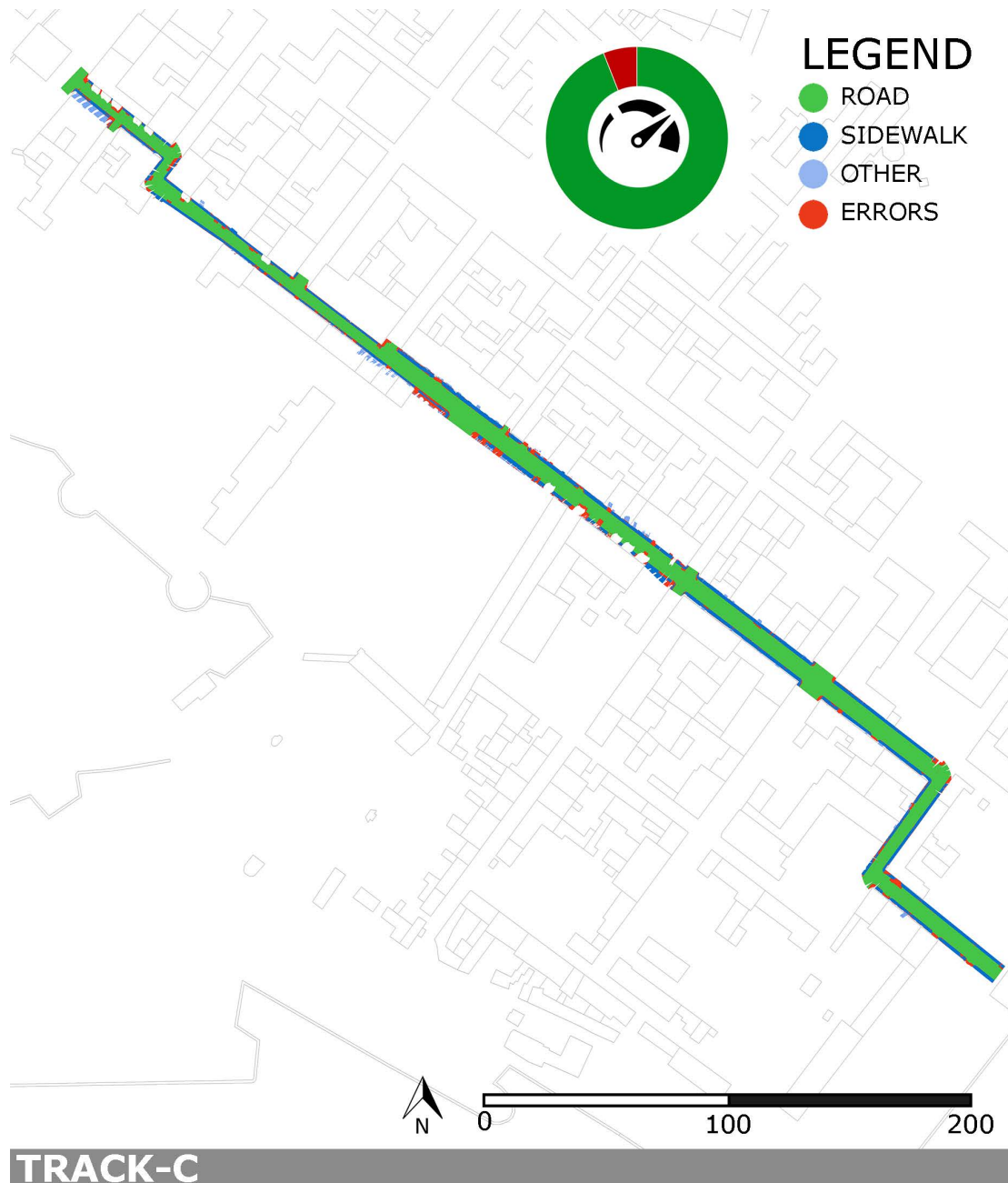


Figure A.14: Top view of the point cloud of Track-B after semantic segmentation with the knowledge-based method. Red points are errors. On the top part of the image, there is a pie chart with an overview of the whole points in the point cloud.



Figure A.15: Top view of the point cloud of Track-C after semantic segmentation with ML method. Red points are errors. On the top part of the image, there is a pie chart with an overview of the whole points in the point cloud.

A.4 Sabbioneta - Track D

This track starts from Porta Imperiale, moves along via Colonna towards the north direction, then turns left to via Bernardino Campi, which is a road parallel to the Decumanus of the city (via Vespasiano Gonzaga), then, at the synagogue, turn left onto via Dondi, then turn left again and return to the decumanus Maximus, via Vespasiano Gonzaga, which follow for a short distance; after about ten metres, turn left again onto via Albertoni and encircle the district, ending on via de Giovanni.

The path is quite long and complex, composed of long straight portions and several 90 degrees curves. In the middle of via Bernardino Campi, the path entered, on the right, the spalto Bresciani, an area with grass and earth, which surrounded the patrol path of the fortified wall. Since there are no pavements in this area and it does not represent an urban area, at the entrance to the spalt, the calculation was interrupted and then resumed. For this reason, an interruption in that area will be seen in the Figure with the rendered point cloud. The roads of this track present the familiar combinations of paving materials: cobblestone or sampietrini for the road and stone or bricks for the sidewalks.

This track has a total length of 812 metres, with 43 million points.

Here are the confusion matrices and precision metrics for the knowledge-based method and the ML approach.

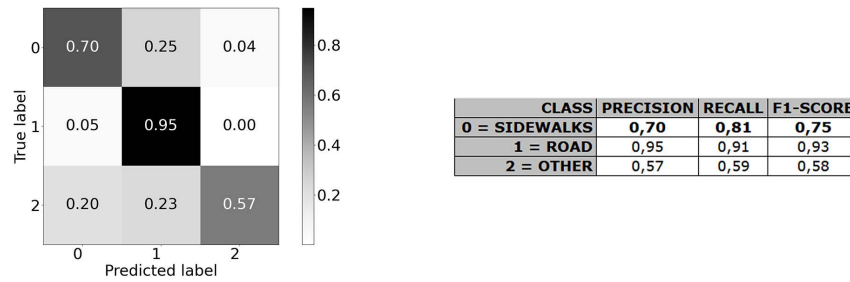


Figure A.16: Confusion matrix and precision metrics for track-D with knowledge-based approach.

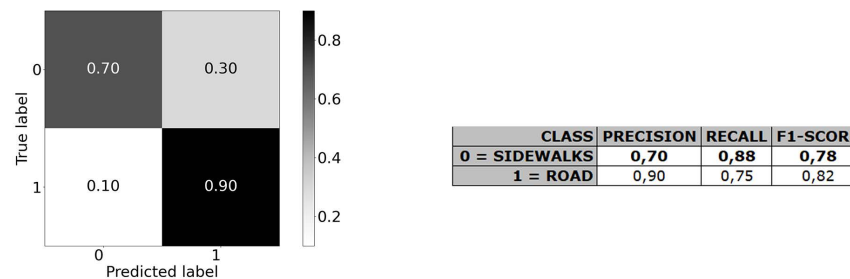


Figure A.17: Confusion matrix and precision metrics for track-D with ML approach.

Appendix A. Semantic Segmentation results

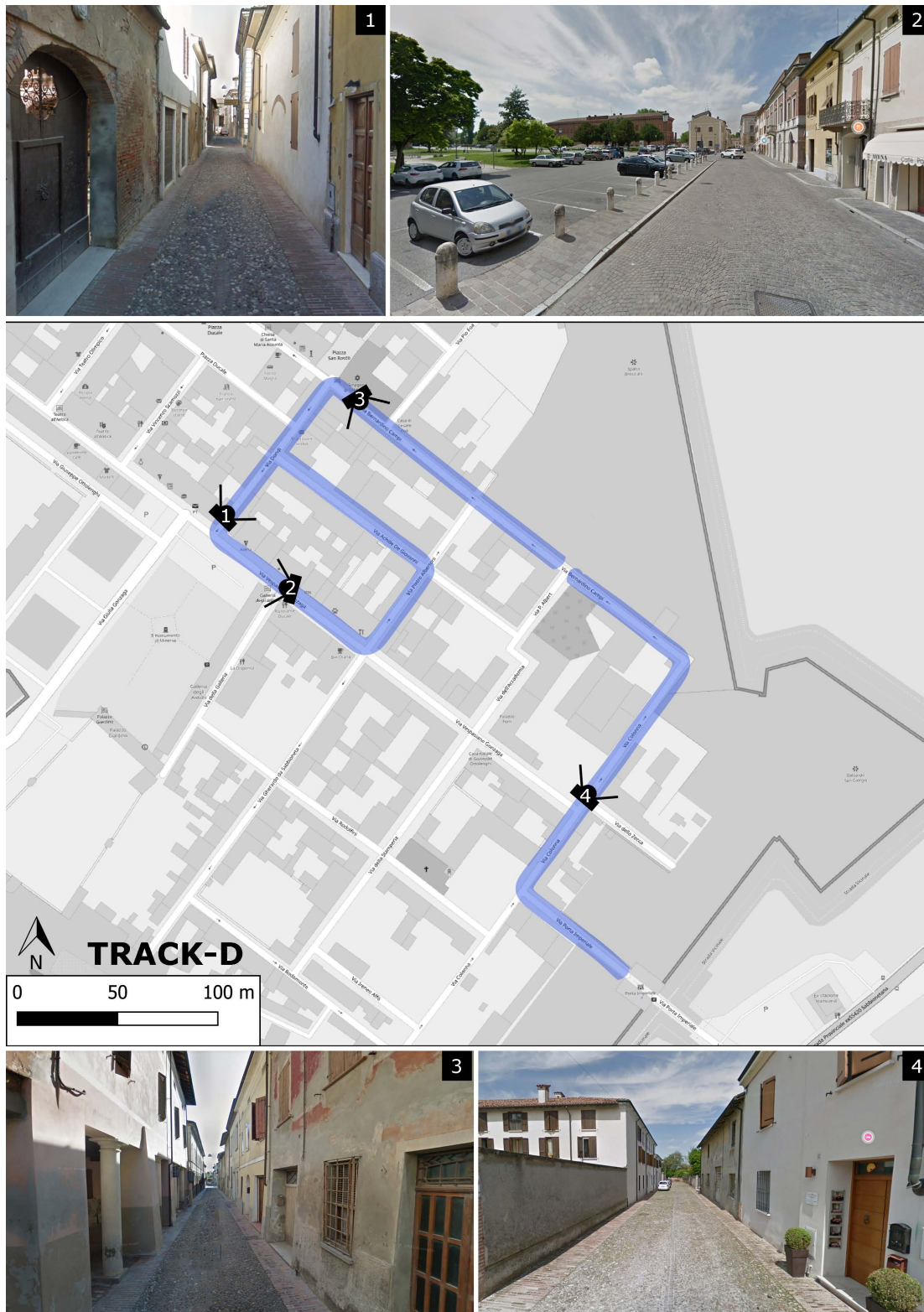


Figure A.18: Map of the path followed for the survey along Track-D, together with some images of the road and the city. Images were elaborated from Google Earth.

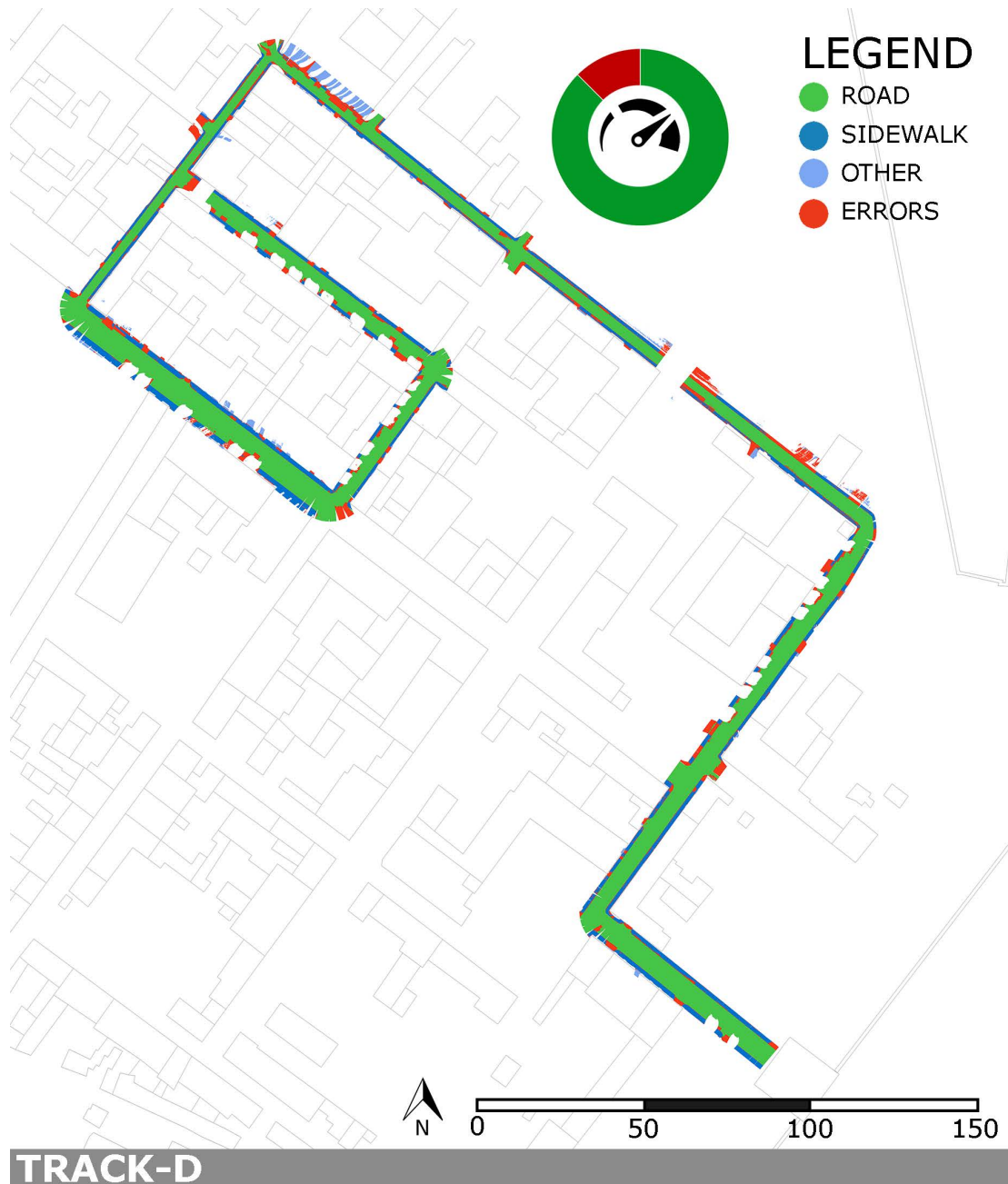


Figure A.19: Top view of the point cloud of Track-D after semantic segmentation with the knowledge-based method. Red points are errors. On the top part of the image, there is a pie chart with an overview of the whole points in the point cloud.

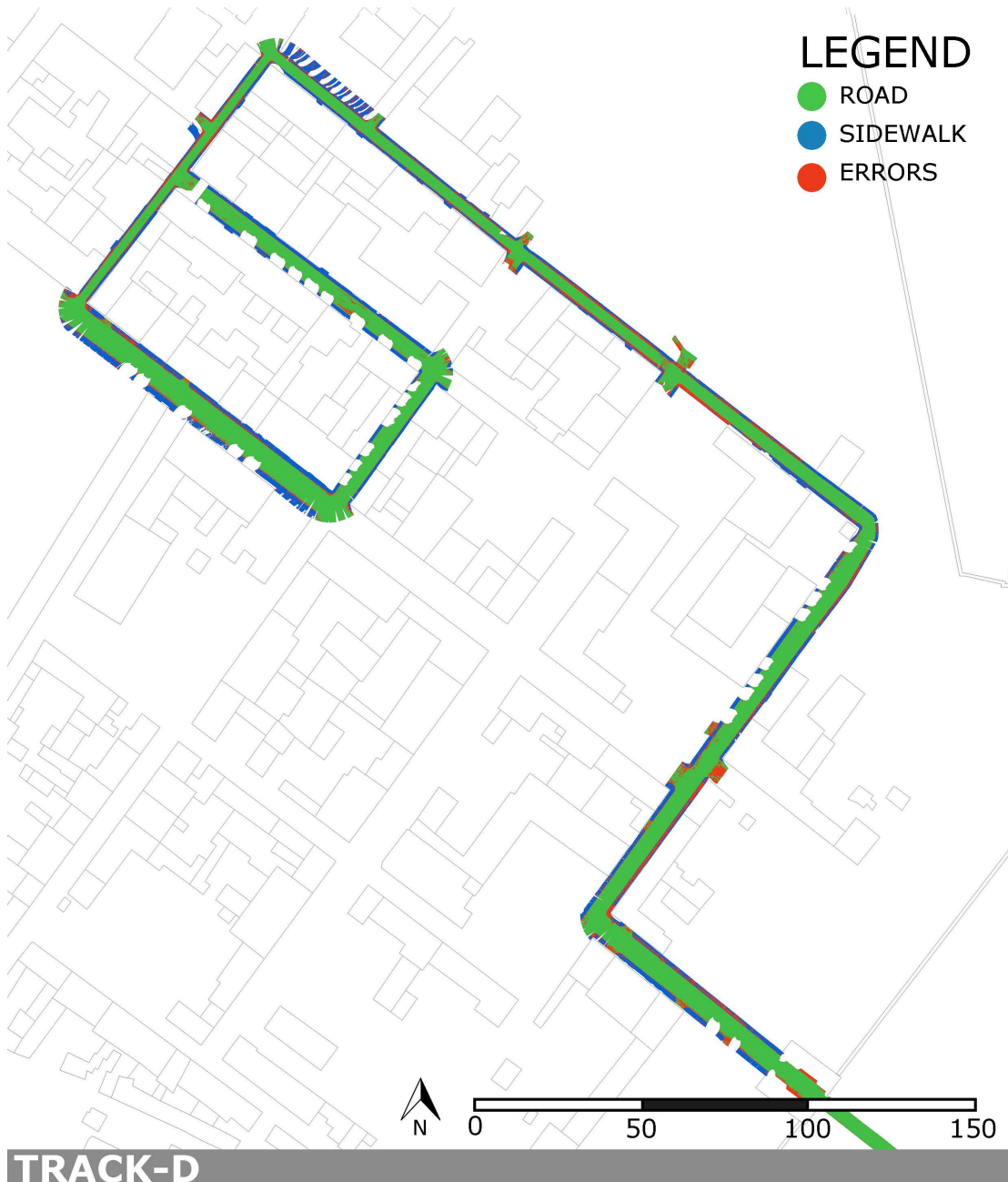


Figure A.20: Top view of the point cloud of Track-D after semantic segmentation with ML method. Red points are errors. On the top part of the image, there is a pie chart with an overview of the whole points in the point cloud.

A.5 Sabbioneta - Track E

This track surveyed the blocks in the southern part of the city. It starts in via Vespasiano Gonzaga, then after a few metres, it moves south, turning onto Via Gherardo da Sabbioneta, turns left onto Via Rodomonte, and returns to Via Colonna and then again Via Vespasiano Gonzaga. It runs for about ten metres and then turns left onto via della Stamperia, ending in a minor, less wide street, via Affo.

The shape of this path is a sort of spiral, capable of acquiring almost all the blocks at south respect the decumanus maximum of the city. There is only one road missing (via Rodolfini), which is surveyed by the next track.

This track has a total length of 812 metres, with 40 million points.

Here are the confusion matrices and precision metrics for the knowledge-based method and the ML approach.

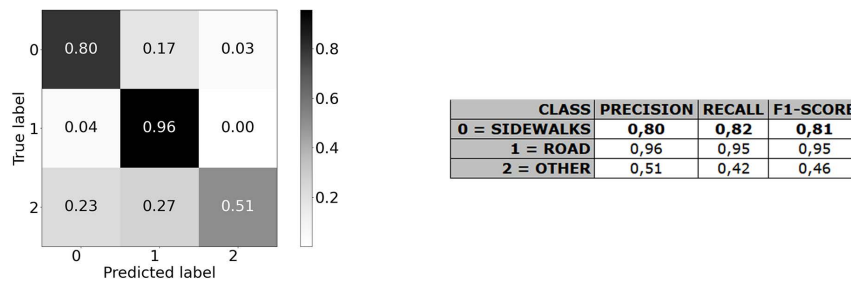


Figure A.21: Confusion matrix and precision metrics for track-E with knowledge-based approach.

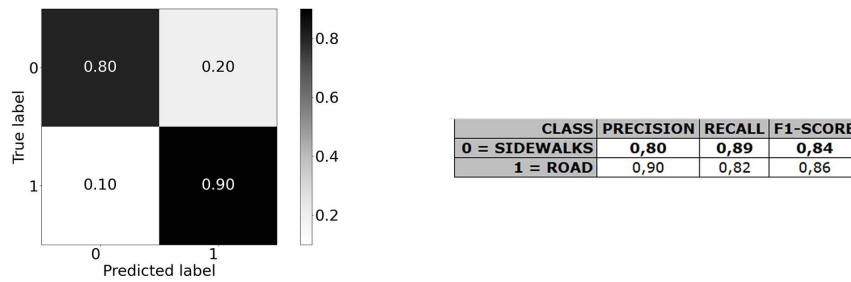


Figure A.22: Confusion matrix and precision metrics for track-E with ML approach.

Appendix A. Semantic Segmentation results

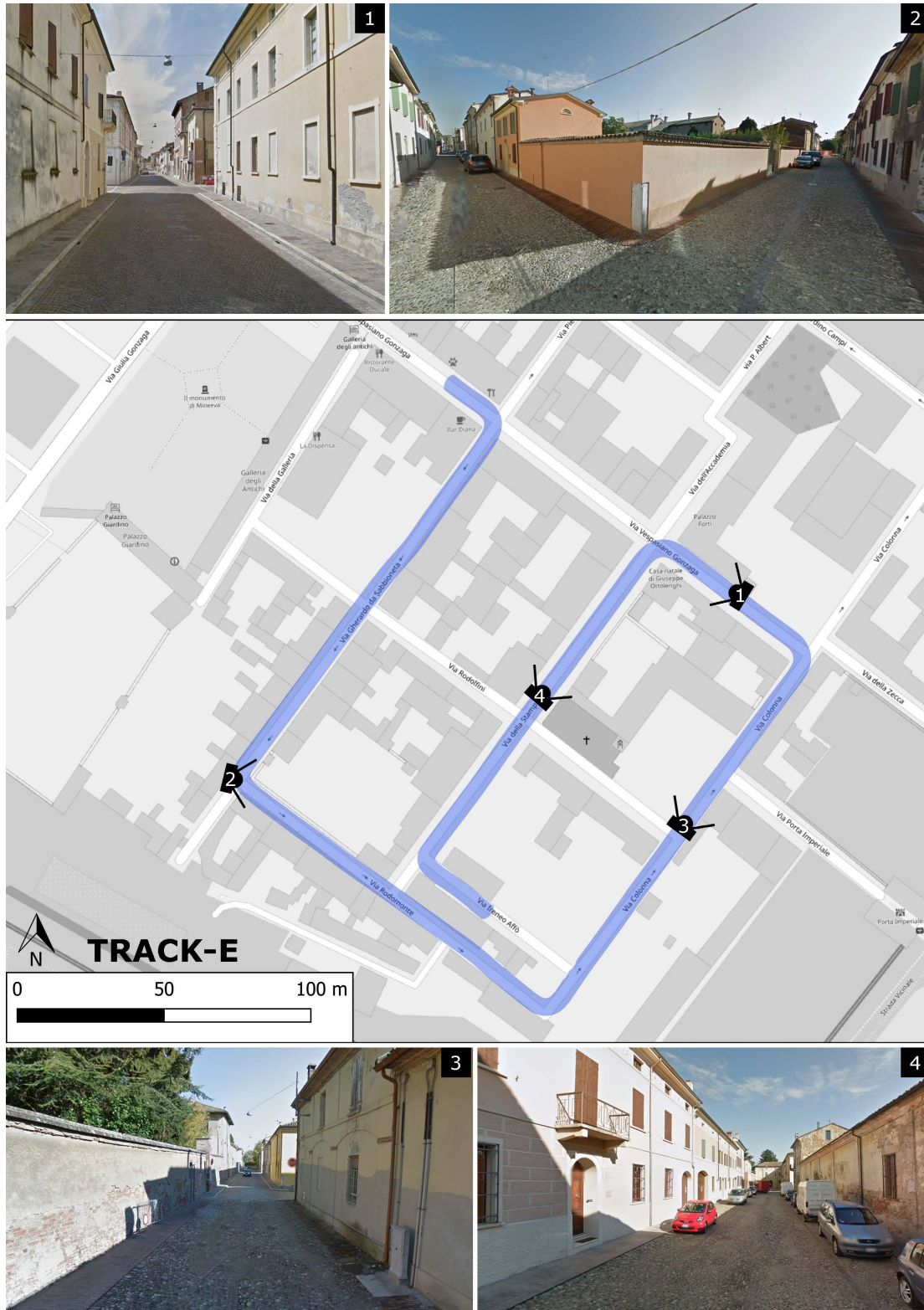


Figure A.23: Map of the path followed for the survey along Track-E, together with some images of the road and the city. Images were elaborated from Google Earth.



Figure A.24: Top view of the point cloud of Track-E after semantic segmentation with the knowledge-based method. Red points are errors. On the top part of the image, there is a pie chart with an overview of the whole points in the point cloud.



Figure A.25: Top view of the point cloud of Track-E after semantic segmentation with ML method. Red points are errors. On the top part of the image, there is a pie chart with an overview of the whole points in the point cloud.

A.6 Sabbioneta - Track F

This track acquired only one road: via Rodolfini, which together with the previous track composes the southern blocks of the city. This road is paved with cobblestone for the road and bricks for the sidewalks, which are placed at the same height as the road.

This track has a total length of 172 metres, with 10 million points.

Here are the confusion matrices and precision metrics for the knowledge-based method and the ML approach.

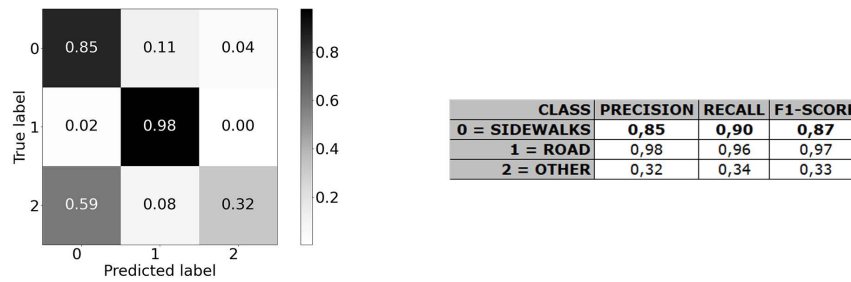


Figure A.26: Confusion matrix and precision metrics for track-F with knowledge-based approach.

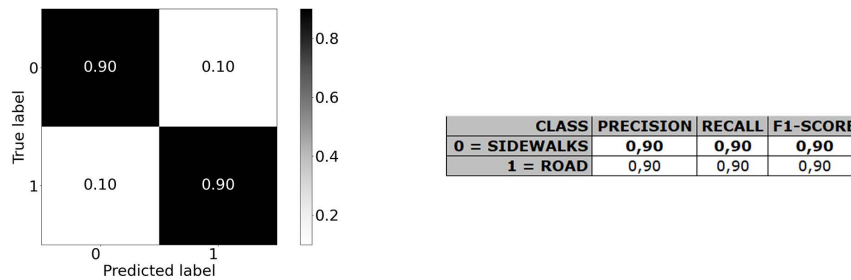


Figure A.27: Confusion matrix and precision metrics for track-F with ML approach.

Appendix A. Semantic Segmentation results

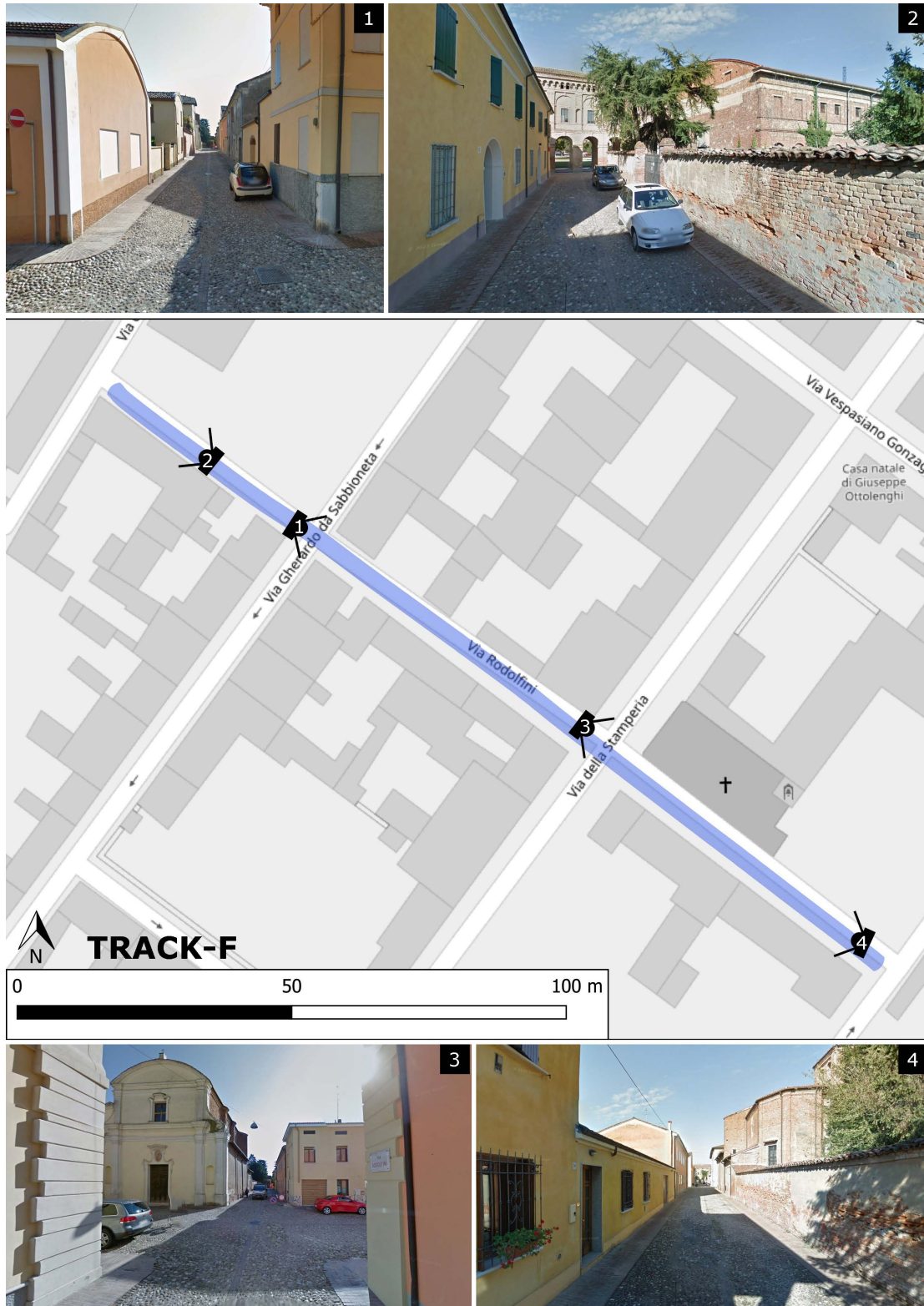


Figure A.28: Map of the path followed for the survey along Track-F, together with some images of the road and the city. Images were elaborated from Google Earth.

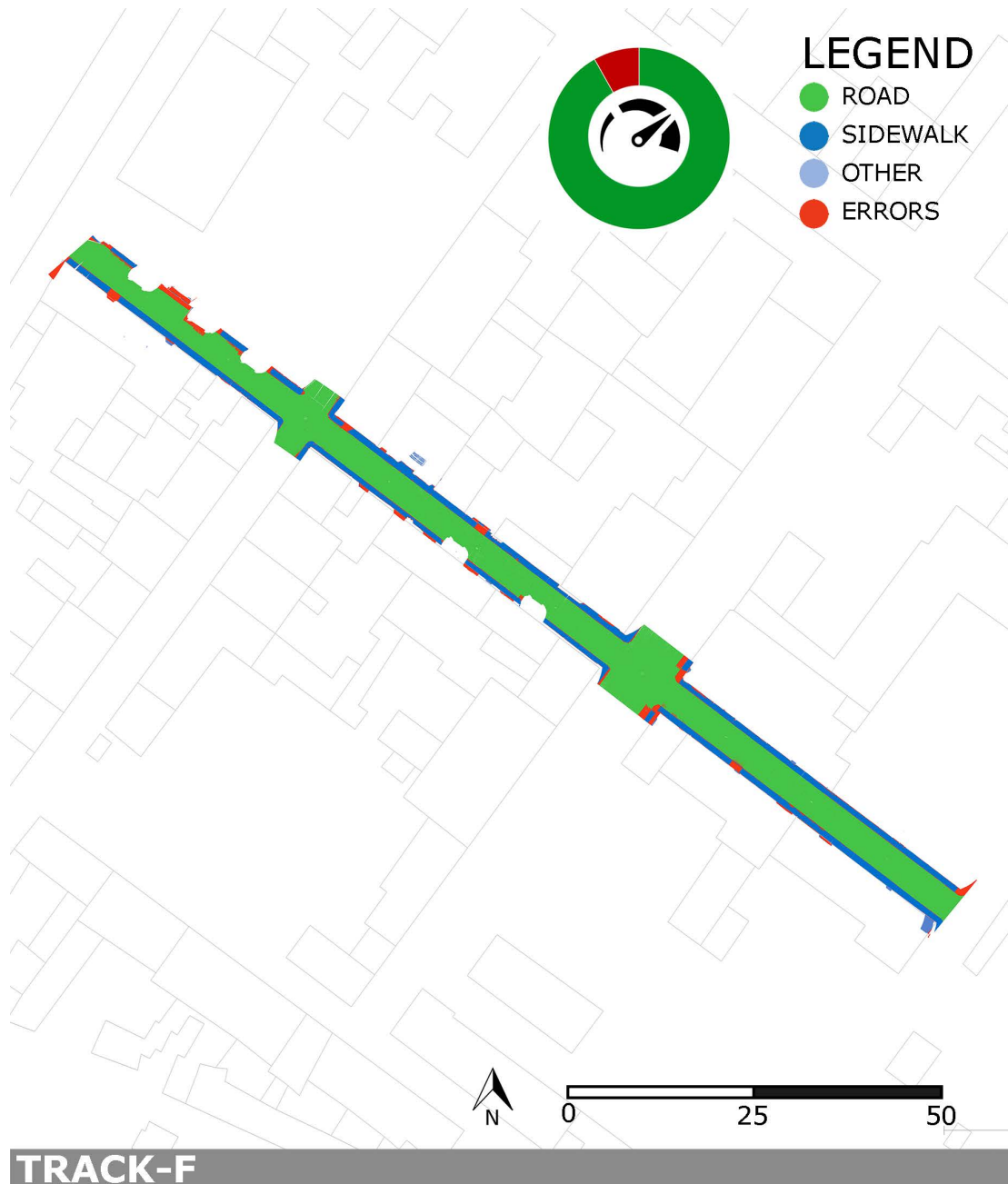


Figure A.29: Top view of the point cloud of Track-F after semantic segmentation with the knowledge-based method. Red points are errors. On the top part of the image, there is a pie chart with an overview of the whole points in the point cloud.



Figure A.30: Top view of the point cloud of Track-F after semantic segmentation with ML method. Red points are errors. On the top part of the image, there is a pie chart with an overview of the whole points in the point cloud.

A.7 Sabbioneta - Track H

This track is useful to complete the city blocks at north respect via Vespasiano Gonzaga. Starting from Via della Stamperia, at Via Vespasiano Gonzaga, heading north, it passes in front of the municipal library, following the road that curves slightly to follow the façade of the building, then turns left onto Via Achille de Giovanni, to end its course at Via Albertoni, heading north. The streets in this track are paved with cobblestone for the road and bricks for the sidewalks, which are placed at the same height as the road.

The particularity of this route is the presence of three very close elbow curves which, as can be seen from the Figure with the cloud render, have been computed fairly well.

This track has a total length of 162 metres, with 16 million points.

Here are the confusion matrices and precision metrics for the knowledge-based method and the ML approach.

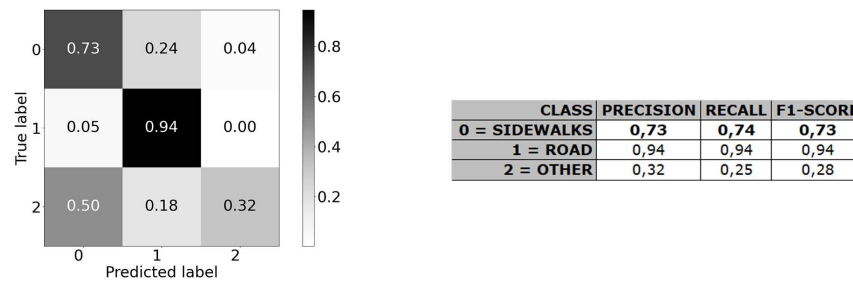


Figure A.31: Confusion matrix and precision metrics for track-H with knowledge-based approach.

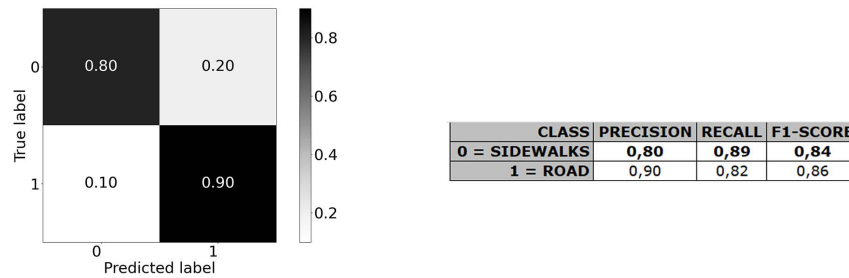


Figure A.32: Confusion matrix and precision metrics for track-H with ML approach.

Appendix A. Semantic Segmentation results

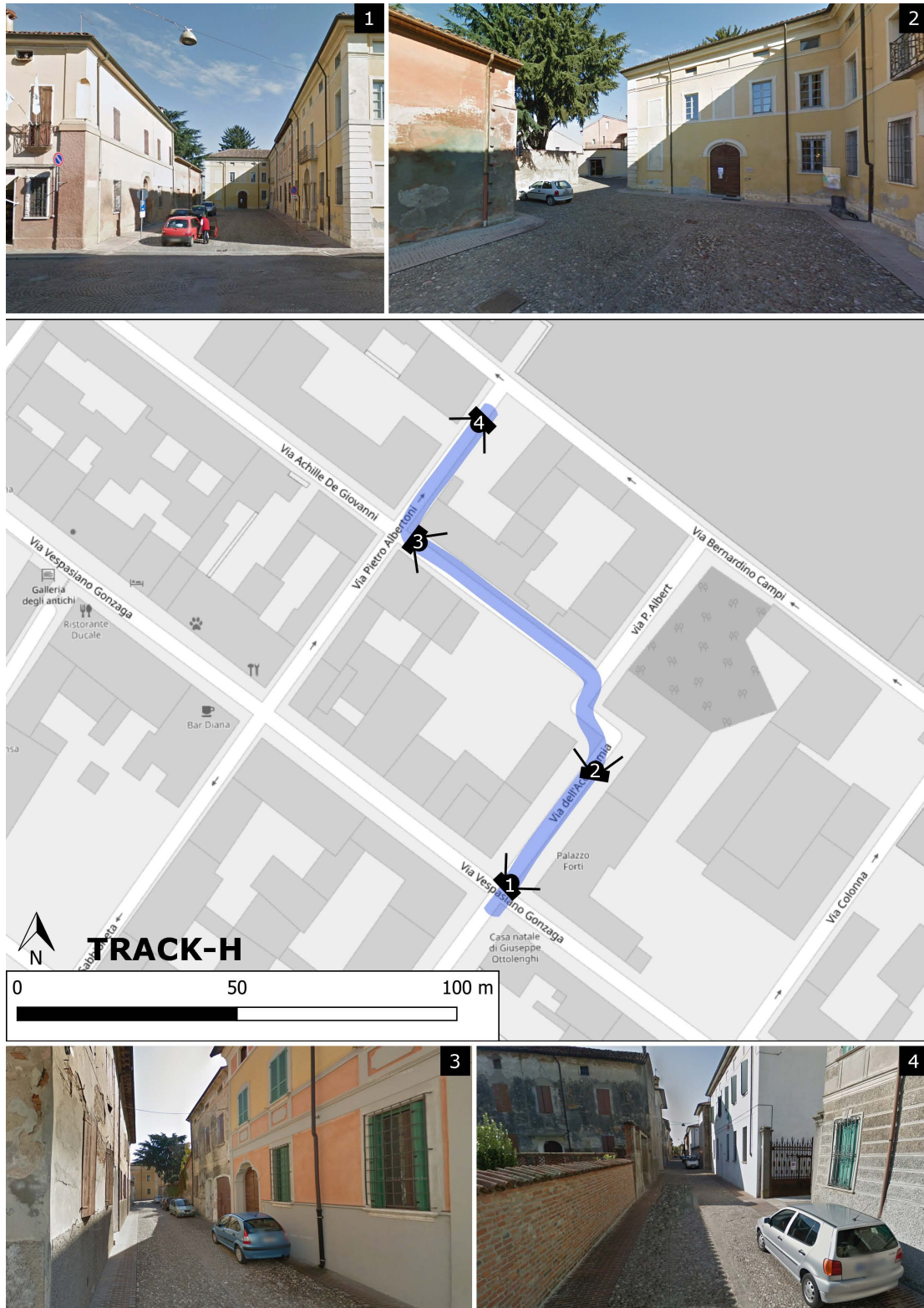


Figure A.33: Map of the path followed for the survey along Track-H, together with some images of the road and the city. Images were elaborated from Google Earth.



Figure A.34: Top view of the point cloud of Track-H after semantic segmentation with the knowledge-based method. Red points are errors. On the top part of the image, there is a pie chart with an overview of the whole points in the point cloud.



Figure A.35: Top view of the point cloud of Track-H after semantic segmentation with ML method. Red points are errors. On the top part of the image, there is a pie chart with an overview of the whole points in the point cloud.

A.8 Sabbioneta - Track I

This track serves to complete the city's northeastern districts. It starts from via Pesenti, with porticoes on the left-hand side, passes through piazza Ducale, in front of Palazzo Ducale, and ends on via Teatro, as far as the junction with via Vespasiano Gonzaga.

The path is completely straight. Even if it does not have curves, the source of error is the Palazzo Ducale. In front of the palace, there is a staircase that acts as an entrance to the building; this staircase interrupts the pavements and joins the street.

It is a short path, only 142 metres and 10 million points.

Here are the confusion matrices and precision metrics for the knowledge-based method and the ML approach.

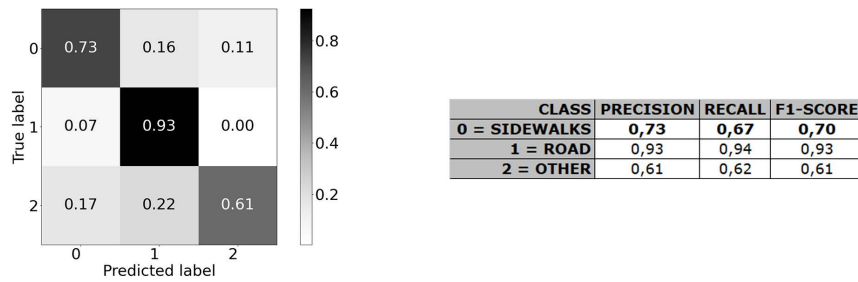


Figure A.36: Confusion matrix and precision metrics for track-I with knowledge-based approach.

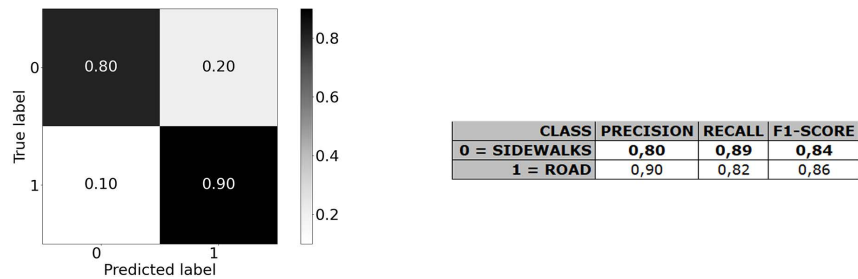


Figure A.37: Confusion matrix and precision metrics for track-I with ML approach.

Appendix A. Semantic Segmentation results

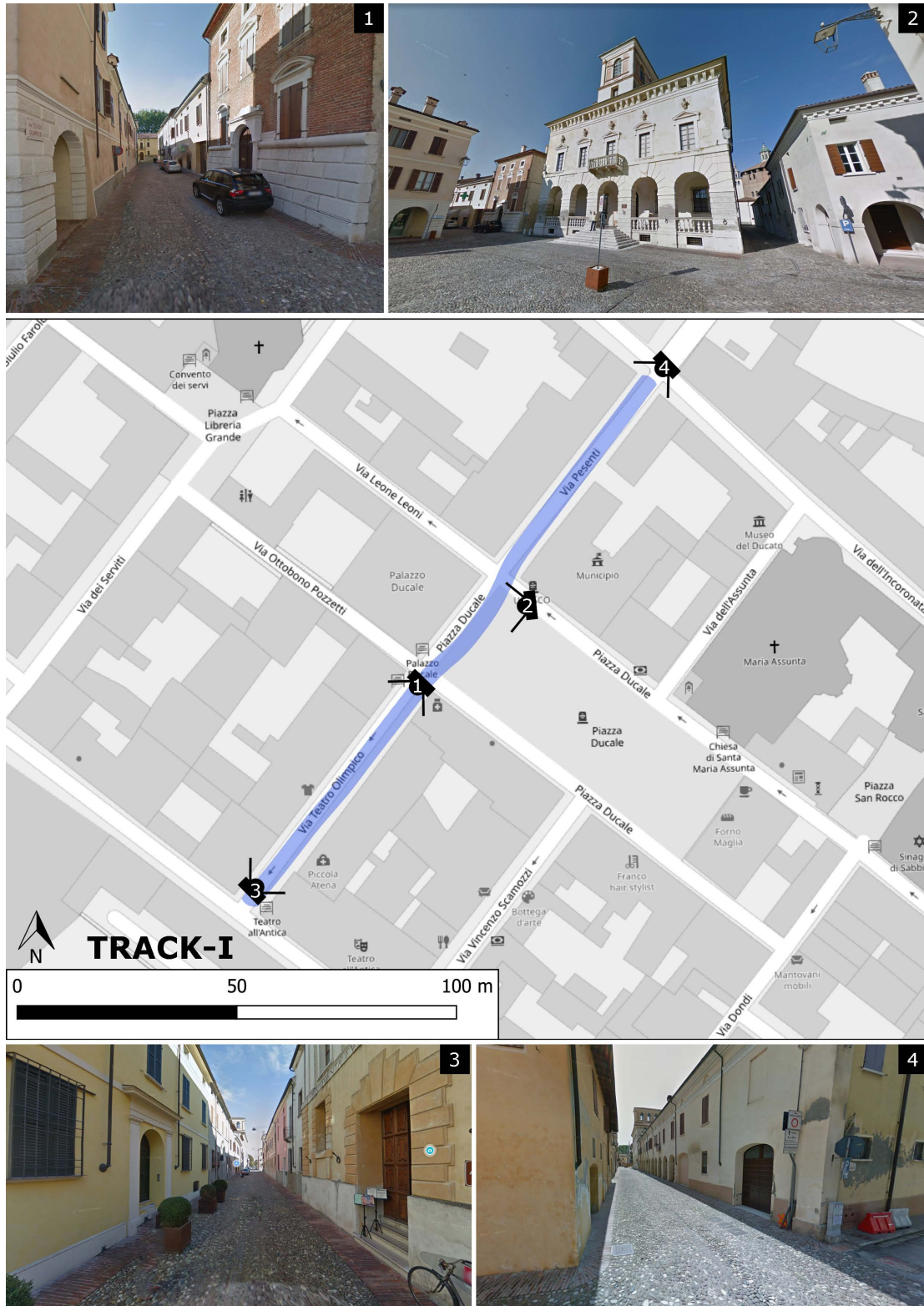


Figure A.38: Map of the path followed for the survey along Track-I, together with some images of the road and the city. Images were elaborated from Google Earth.

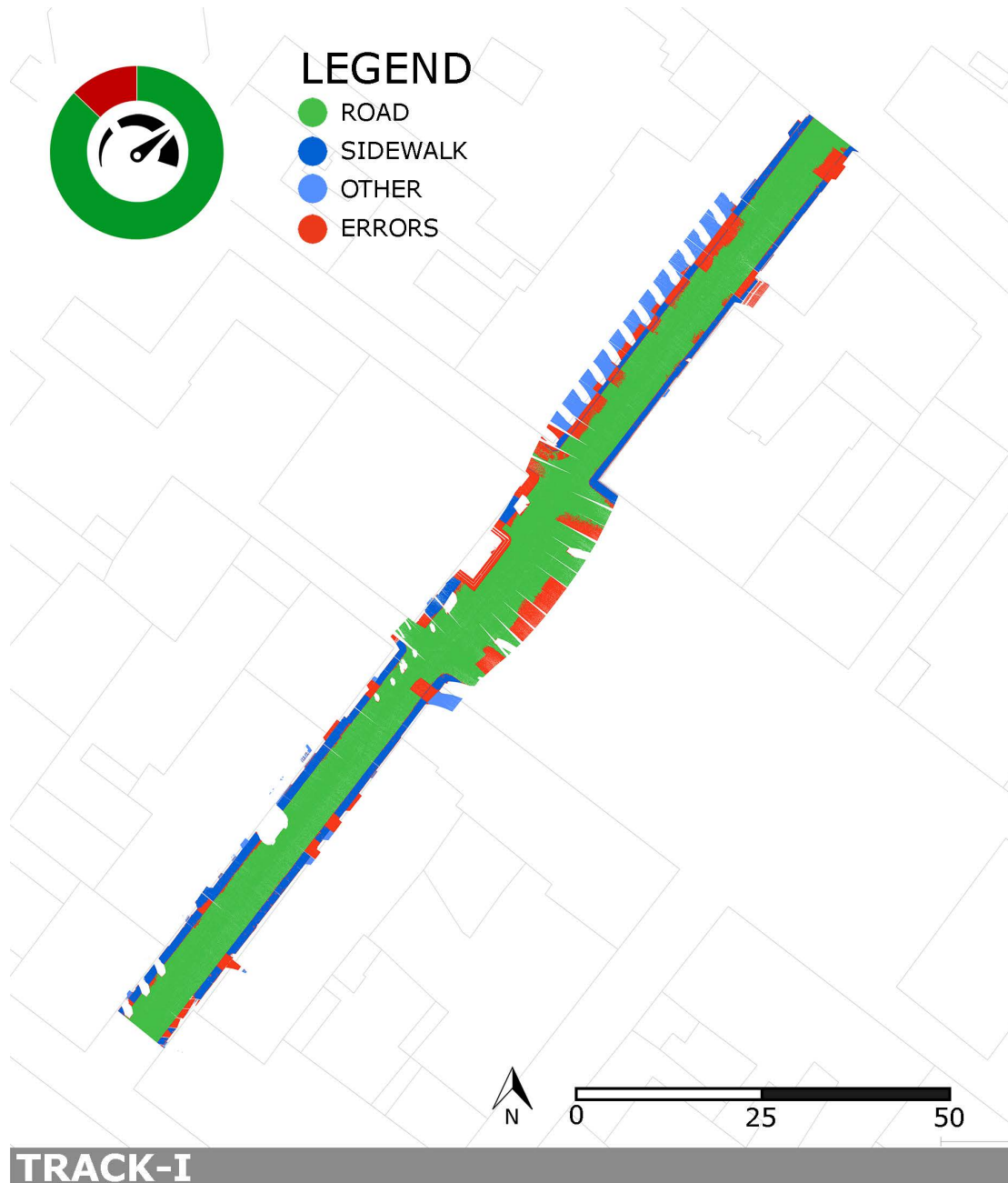


Figure A.39: Top view of the point cloud of Track-I after semantic segmentation with the knowledge-based method. Red points are errors. On the top part of the image, there is a pie chart with an overview of the whole points in the point cloud.



Figure A.40: Top view of the point cloud of Track-I after semantic segmentation with ML method. Red points are errors. On the top part of the image, there is a pie chart with an overview of the whole points in the point cloud.

A.9 Mantova

The surveyed area is a straight road in front of the Mantova Campus of Politecnico di Milano. It starts in Piazza Carlo d'Arco and concludes in via Angelo Scarsellini. There are some peculiar elements: it passes near a piazza and in front of a historic building (Palazzo d'Arco, facing Piazza d'Arco); there are some parking areas aligned with the road on one side of via Scarsellini that separate the road from the sidewalk. The road is all paved with cobblestone, the sidewalks are different: some of them are made of white stone (in front of Palazzo d'Arco) and at the same level of the road, some others are in porphyry stone and at a higher level of respect the road, some others are made by a different type of stone and with a higher level. The path has a length of 142 metres, for 7 million points.

Here are the confusion matrices and precision metrics for the knowledge-based method and the ML approach (using both Track-C and a portion of the dataset itself as training for the model).

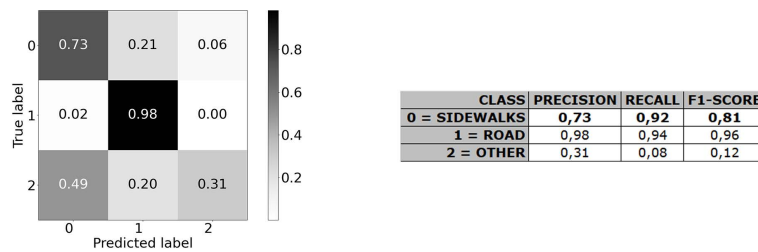


Figure A.41: Confusion matrix and precision metrics for Mantova dataset with knowledge-based approach.

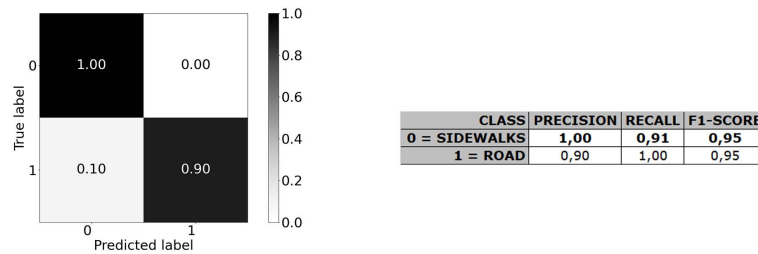


Figure A.42: Confusion matrix and precision metrics for Mantova dataset with ML approach. The ML model was trained on Track-C of Sabbioneta.

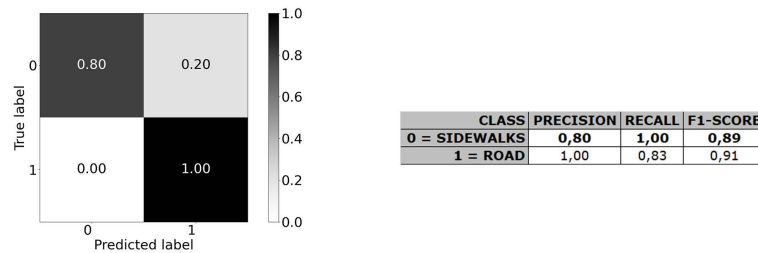


Figure A.43: Confusion matrix and precision metrics for Mantova dataset with ML approach. The ML model was trained on a portion (about 1/3) of Mantova dataset itself.

Appendix A. Semantic Segmentation results

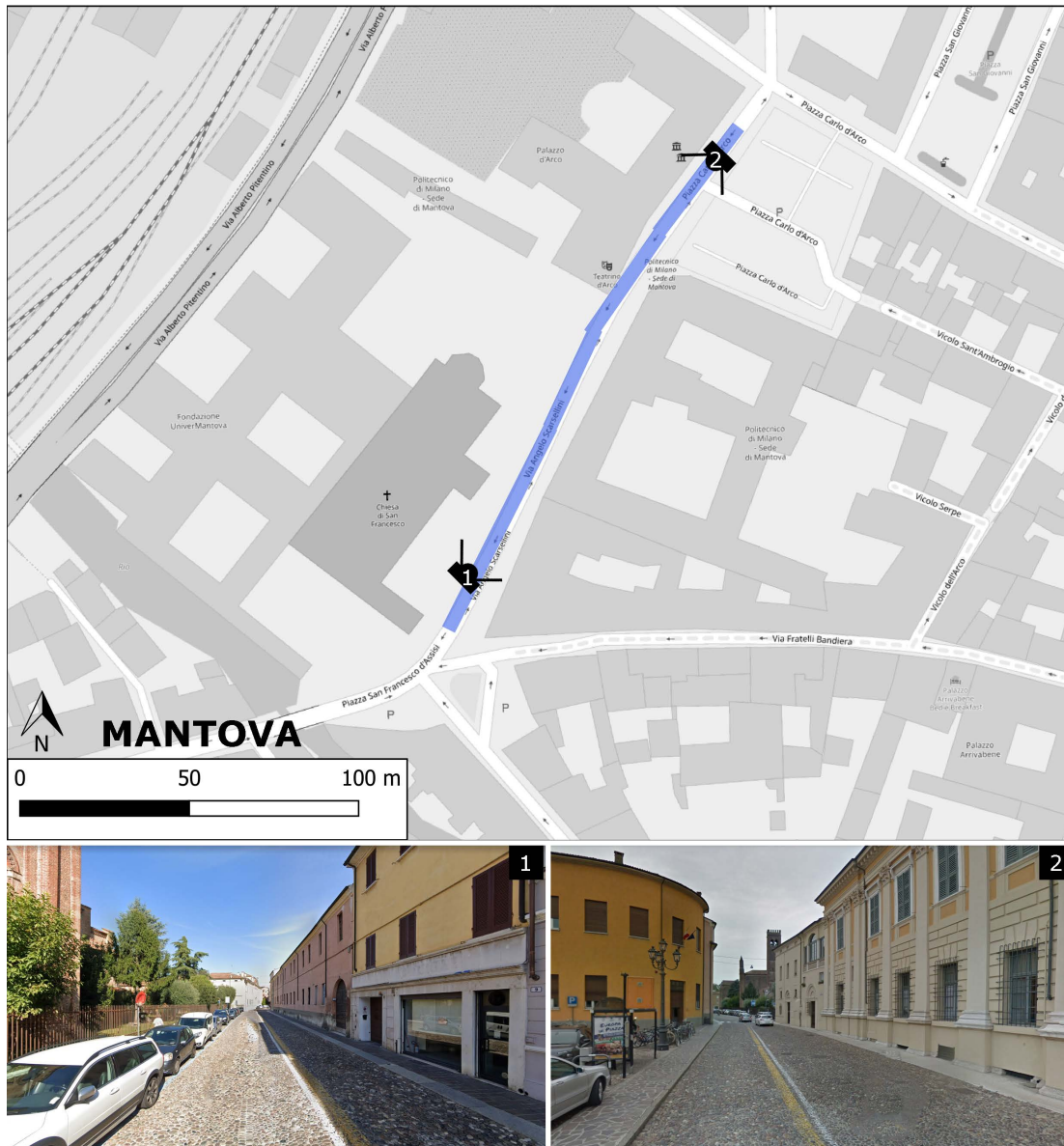
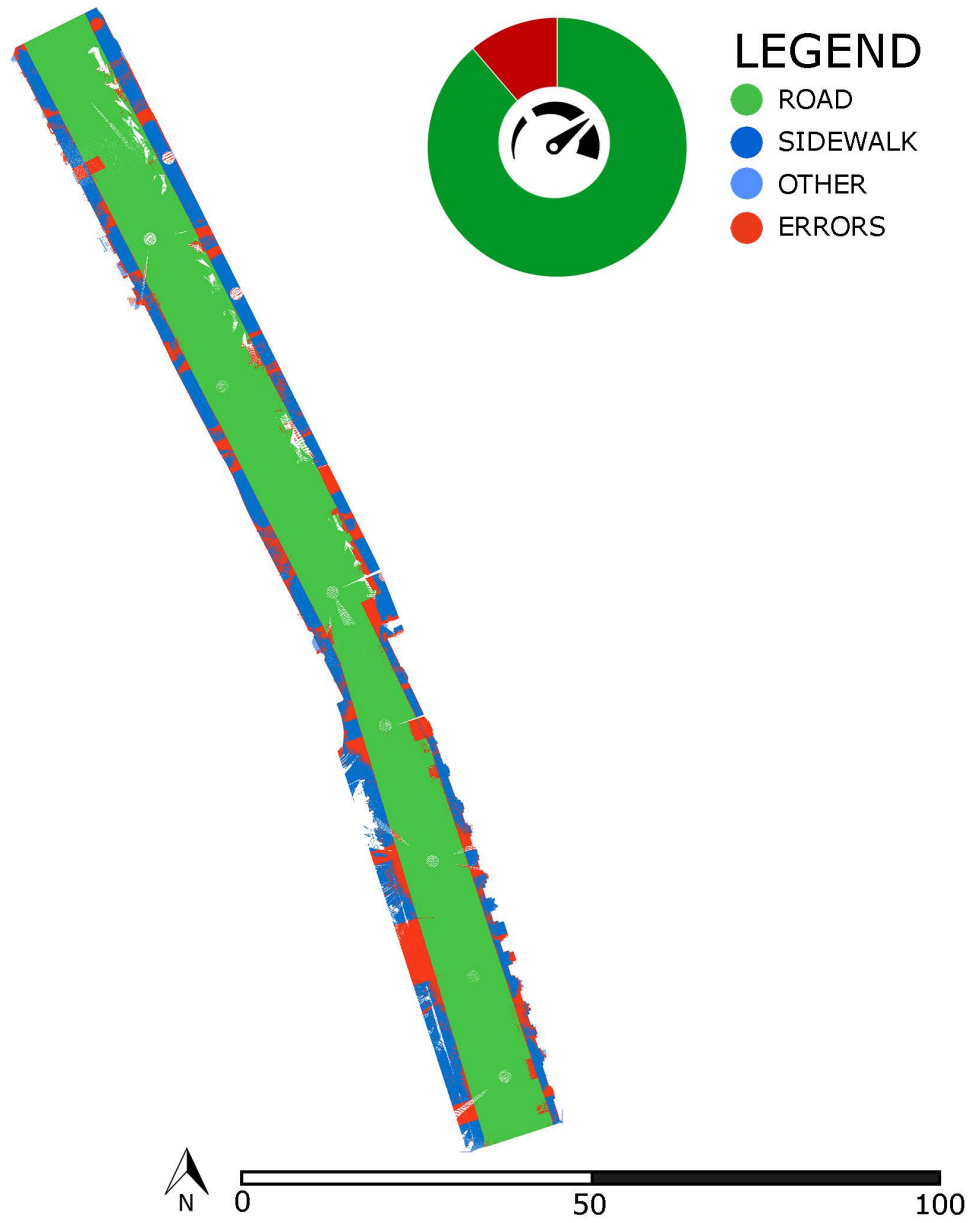
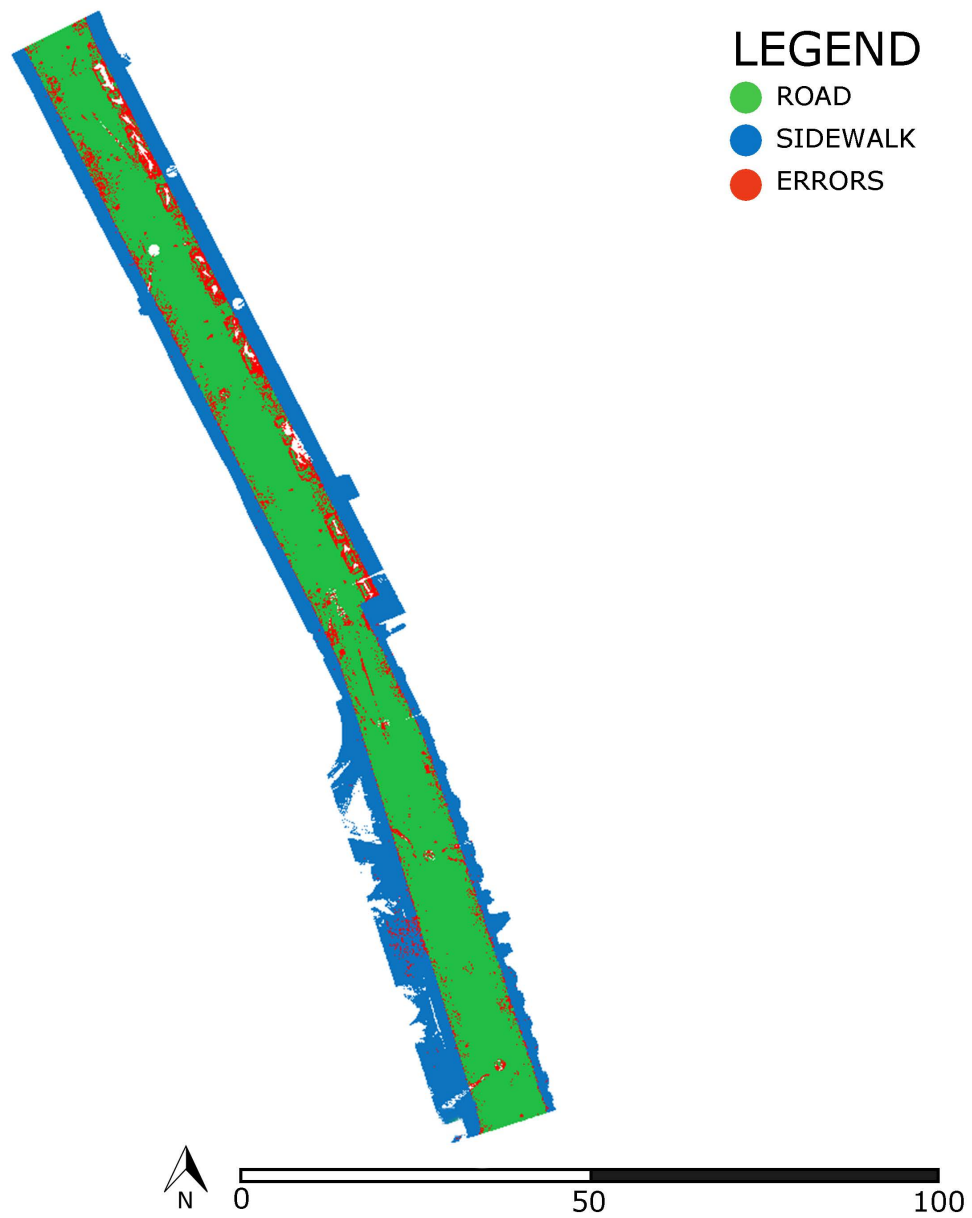


Figure A.44: Map of the path followed for the survey along the dataset in Mantova, together with some images of the road and the city. Images were elaborated from Google Earth.



MANTOVA- VIA SCARSELLINI

Figure A.45: Top view of the point cloud of Mantova after semantic segmentation with the knowledge-based method. Red points are errors. On the top part of the image, there is a pie chart with an overview of the whole points in the point cloud.



MANTOVA- VIA SCARSELLINI

Figure A.46: Top view of the point cloud of Mantova after semantic segmentation with the ML method. Red points are errors. The training dataset was Track-C of Sabbioenta.

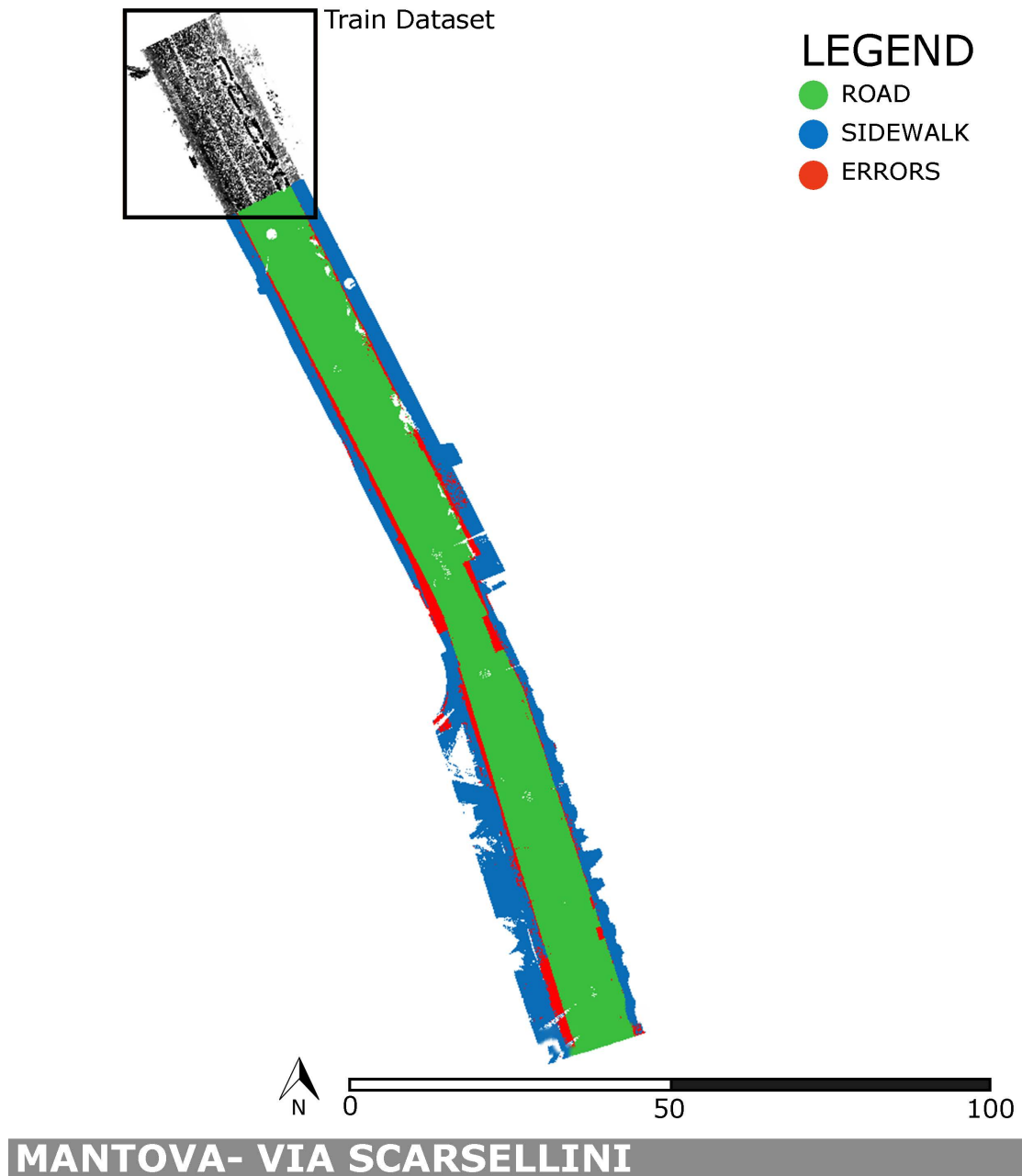


Figure A.47: Top view of the point cloud of Mantova after semantic segmentation with the ML method. Red points are errors. The ML model was trained on a portion (about 1/3) of Mantova dataset itself.

A.10 Sacro Monte

The dataset selected is Sacro Monte di Domodossola, the path winds from the village of Domodossola up to the hills of Mattarella, a place of very ancient origins where Roman and Lombard building materials, ceramics and utensils were found along with a marble headstone dating back to 539 A.D.¹; Sacro Monte of Domodossola is a Unesco site since 2003. This dataset was kindly provided by prof. Francesco Fassi, 3D survey group, Politecnico di Milano. In the case of the Sacri Monti dataset, most of the route is on a footpath going up the mountain and therefore does not have the necessary characteristics for this test. However, in the dataset, there is an initial portion (a few tens of metres) that was acquired along the road of the village located further down the valley. This portion of the dataset has the right characteristics for the test and was therefore selected. The selected portion of the point cloud has a length of 30 metres, for a total of 6 million points on the ground. The point density is homogeneous and sufficiently high for the algorithm to perform correctly, the Intensity attribute is provided. Both the road and the sidewalks have asphalt pavement, the sidewalks are slightly raised above the road level.

Here are the confusion matrices and precision metrics for the knowledge-based method and the ML approach (using both Track-C and a portion of the dataset itself as training for the model).

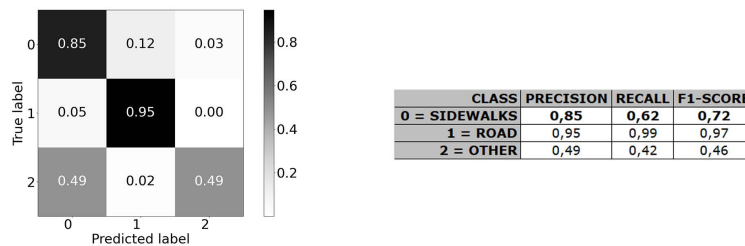


Figure A.48: Confusion matrix and precision metrics for Sacro Monte dataset with knowledge-based approach.

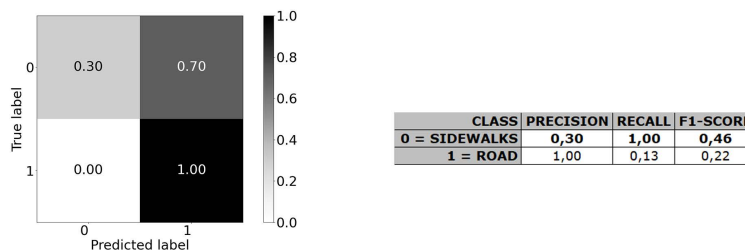


Figure A.49: Confusion matrix and precision metrics for Sacro Monte dataset with ML approach. The ML model was trained on Track-C of Sabbioneta.

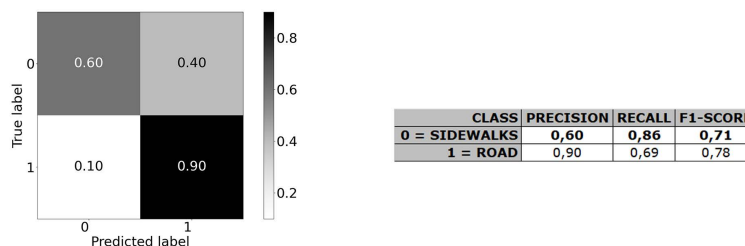


Figure A.50: Confusion matrix and precision metrics for Sacro Monte dataset with ML approach. The ML model was trained on a portion (about 1/3) of Sacro Monte dataset itself.

¹www.sacrimonti.org/en/sacro-monte-di-domodossola



Figure A.51: Map of the path followed for the survey along the dataset in Domodossola, together with some images of the road and the city. Images were elaborated from Google Earth.

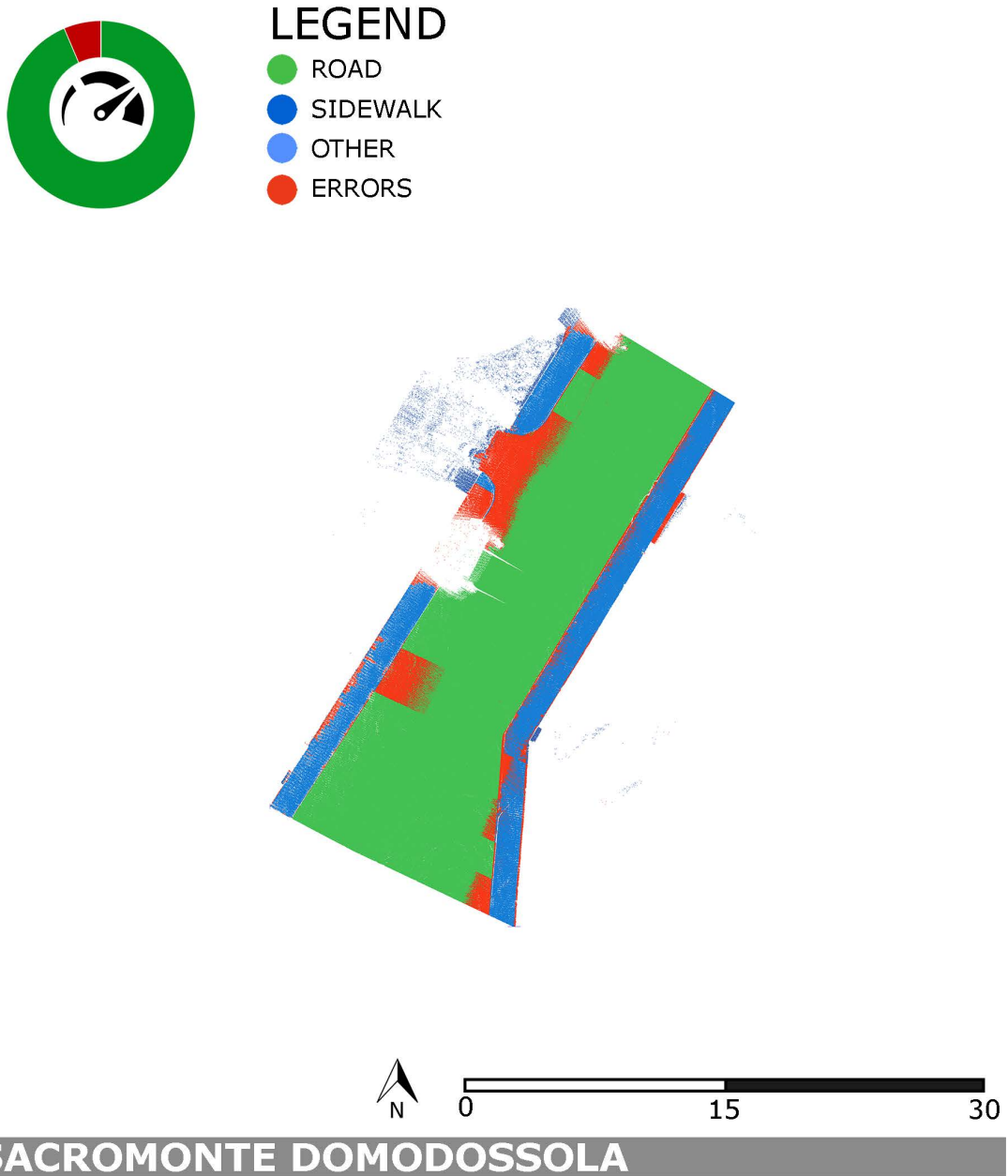
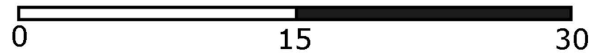


Figure A.52: Top view of the point cloud of Domodossola after semantic segmentation with the knowledge-based method. Red points are errors. On the top part of the image, there is a pie chart with an overview of the whole points in the point cloud.

LEGEND

- ROAD
- SIDEWALK
- ERRORS



SACROMONTE DOMODOSSOLA

Figure A.53: Top view of the point cloud of Sacro Monte after semantic segmentation with the ML method. Red points are errors. The training dataset was Track-C of Sabbioenta.

LEGEND

- ROAD
- SIDEWALK
- ERRORS

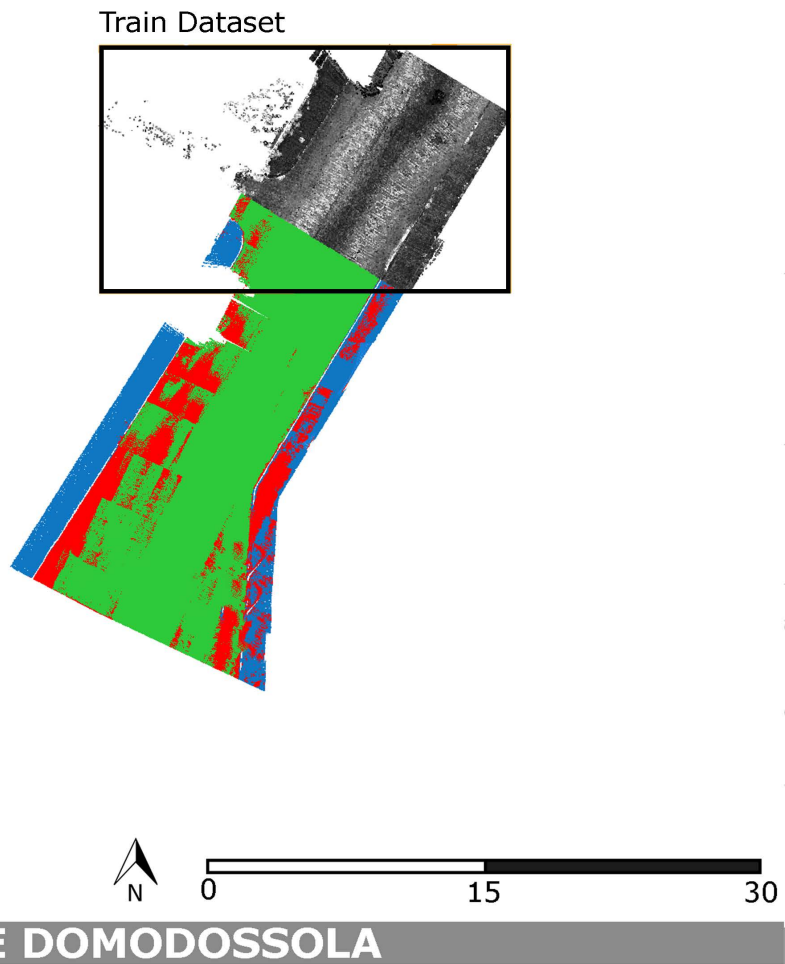


Figure A.54: Top view of the point cloud of Sacro Monte after semantic segmentation with the ML method. Red points are errors. The ML model was trained on a portion (about 1/3) of Sacro Monte dataset itself.

A.11 Porto - 1

The first area investigated is a portion of Campo 24 de Agosto, a wide road that passes in front of a bus station on one side, and near a park on the other side. The length of the path is 122 metres, with 5 million points on the ground surface. The sidewalk paving is in sampietrini, while the road is in asphalt.

Here are the confusion matrices and precision metrics for the knowledge-based method and the ML approach (using both Track-C and a portion of the dataset itself as training for the model).

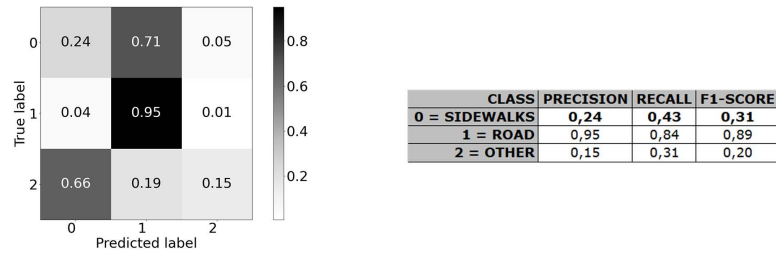


Figure A.55: Confusion matrix and precision metrics for Porto-1 dataset with knowledge-based approach.

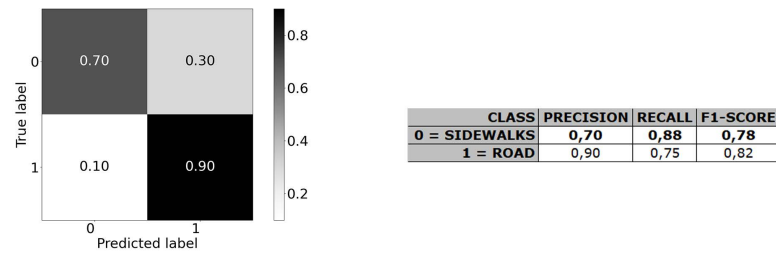


Figure A.56: Confusion matrix and precision metrics for Porto-1 dataset with ML approach. The ML model was trained on Track-C of Sabbioneta.

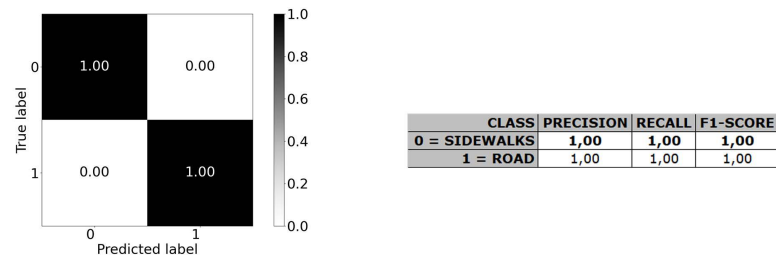


Figure A.57: Confusion matrix and precision metrics for Porto-1 dataset with ML approach. The ML model was trained on a portion (about 1/3) of Porto-1 dataset itself.

Appendix A. Semantic Segmentation results

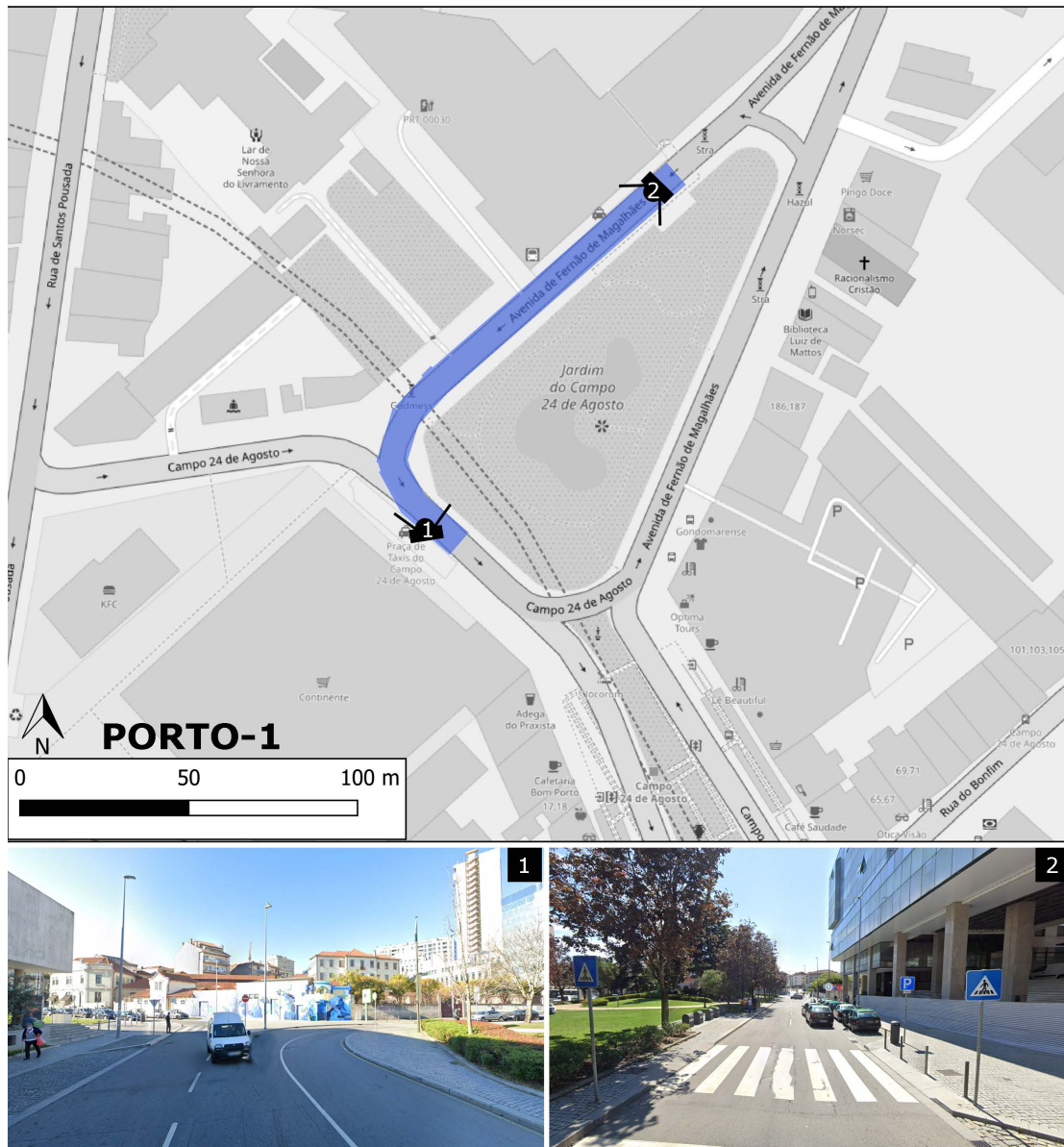


Figure A.58: Map of the path followed for the survey along the dataset in Porto-1, together with some images of the road and the city. Images were elaborated from Google Earth.



Figure A.59: Top view of the point cloud of Porto-1 after semantic segmentation with the knowledge-based method. Red points are errors. On the top part of the image, there is a pie chart with an overview of the whole points in the point cloud.



Figure A.60: Top view of the point cloud of Porto-1 after semantic segmentation with the ML method. Red points are errors. The training dataset was Track-C of Porto-1.

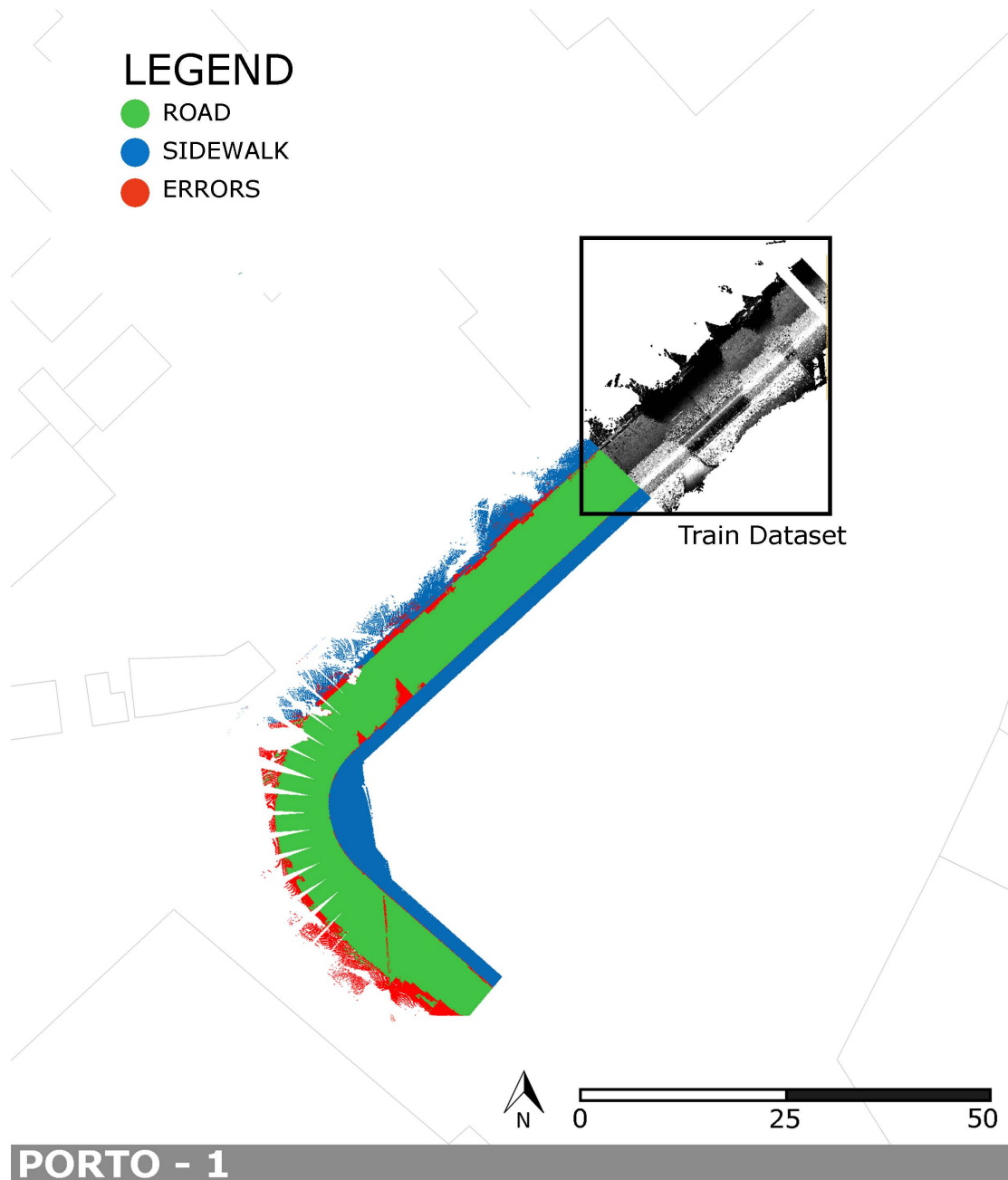


Figure A.61: Top view of the point cloud of Porto-1 after semantic segmentation with the ML method. Red points are errors. The ML model was trained on a portion (about 1/3) of Porto-1 dataset itself.

A.12 Porto - 2

The second area is a crossing between three roads: Rua de Fernandes Tomás, Rua de Santos Pousada, and Rua de Cohelo Neto. This crossing is very close to the previous test area; the three roads are very large, and in their surroundings there are buildings, but also an open space area. Again the urban configuration is very different from the hypotheses. The length of the path is 216 metres, with 11 million points on the ground surface. The sidewalk paving is in concrete, while the road is in asphalt.

Here are the confusion matrices and precision metrics for the knowledge-based method and the ML approach (using both Track-C and a portion of the dataset itself as training for the model).

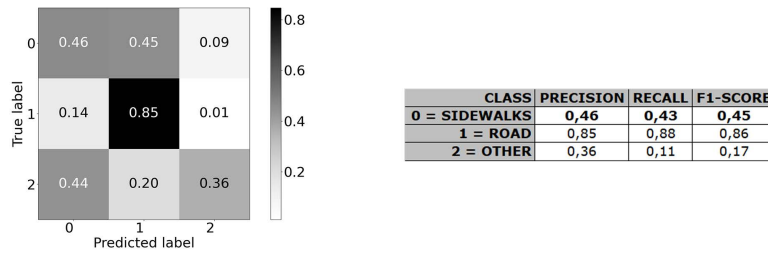


Figure A.62: Confusion matrix and precision metrics for Porto-2 dataset with knowledge-based approach.

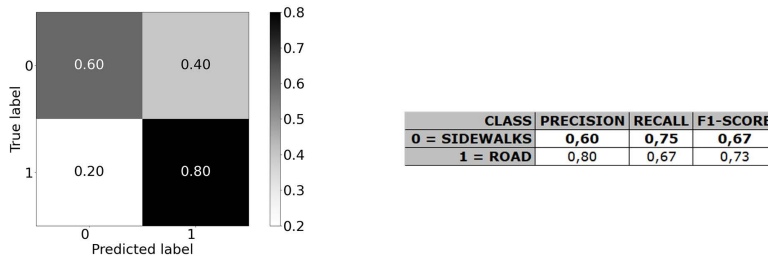


Figure A.63: Confusion matrix and precision metrics for Porto-2 dataset with ML approach. The ML model was trained on Track-C of Sabbioneta.

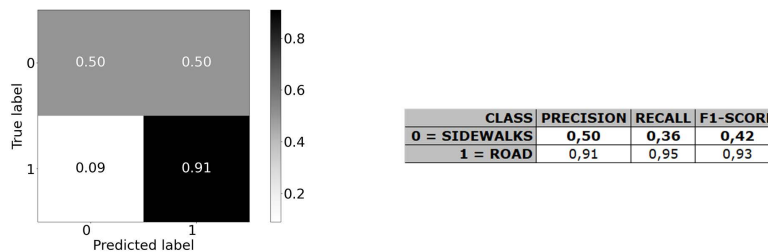


Figure A.64: Confusion matrix and precision metrics for Porto-2 dataset with ML approach. The ML model was trained on a portion (about 1/3) of Porto-2 dataset itself.

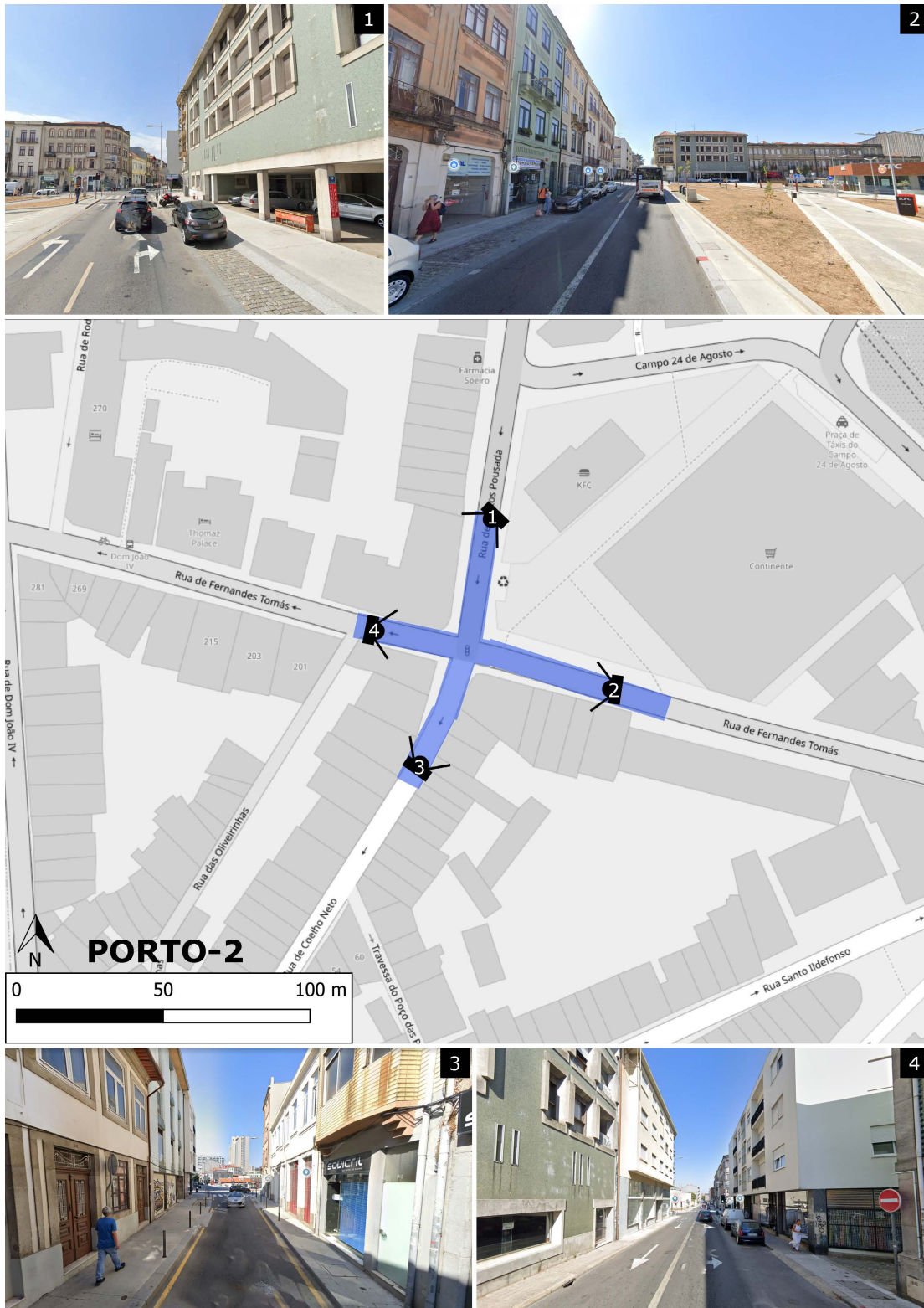


Figure A.65: Map of the path followed for the survey along the dataset in Porto-2, together with some images of the road and the city. Images were elaborated from Google Earth.

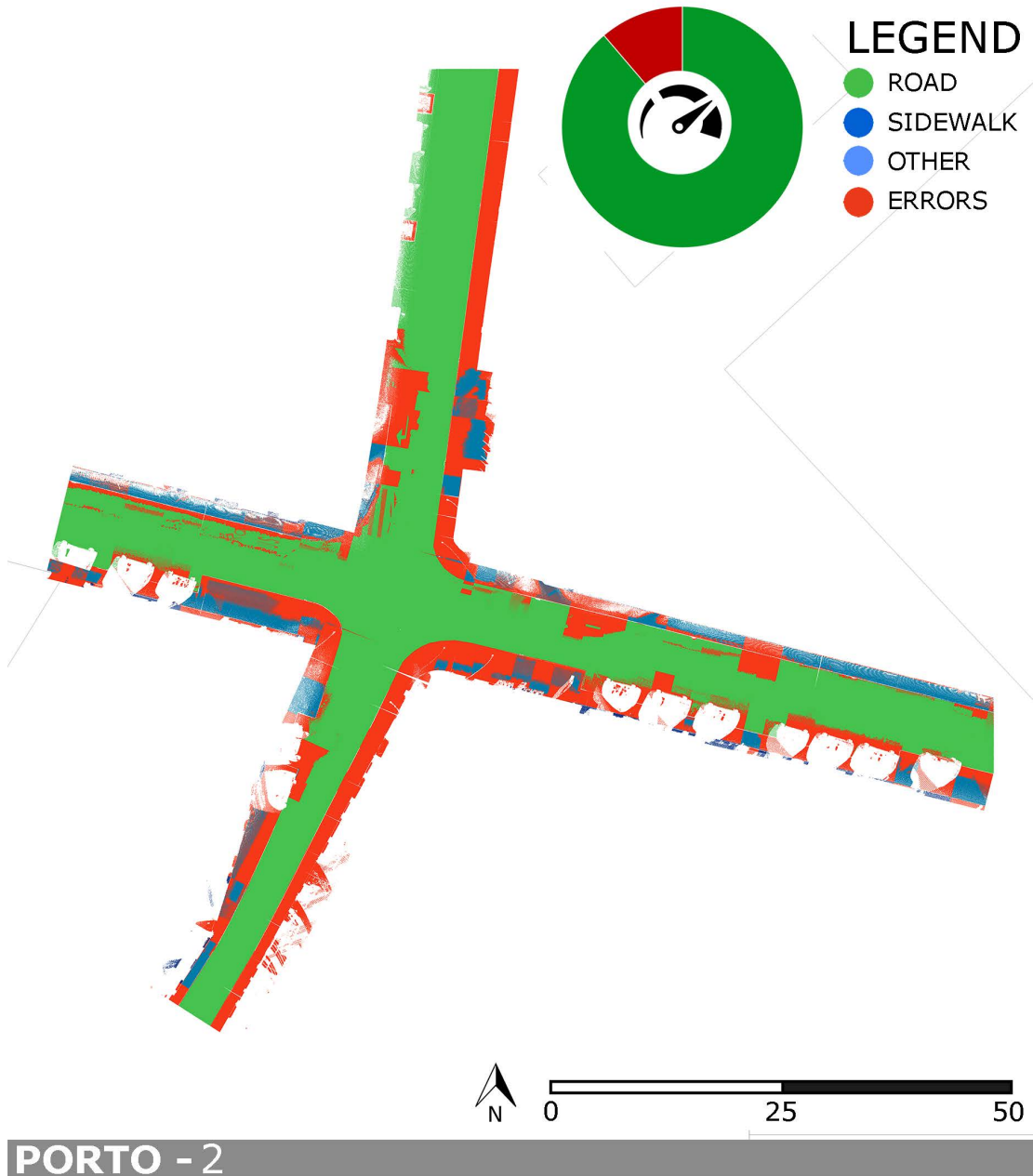


Figure A.66: Top view of the point cloud of Porto-2 after semantic segmentation with the knowledge-based method. Red points are errors. On the top part of the image, there is a pie chart with an overview of the whole points in the point cloud.



Figure A.67: Top view of the point cloud of Porto-2 after semantic segmentation with the ML method. Red points are errors. The training dataset was Track-C of Porto-2.

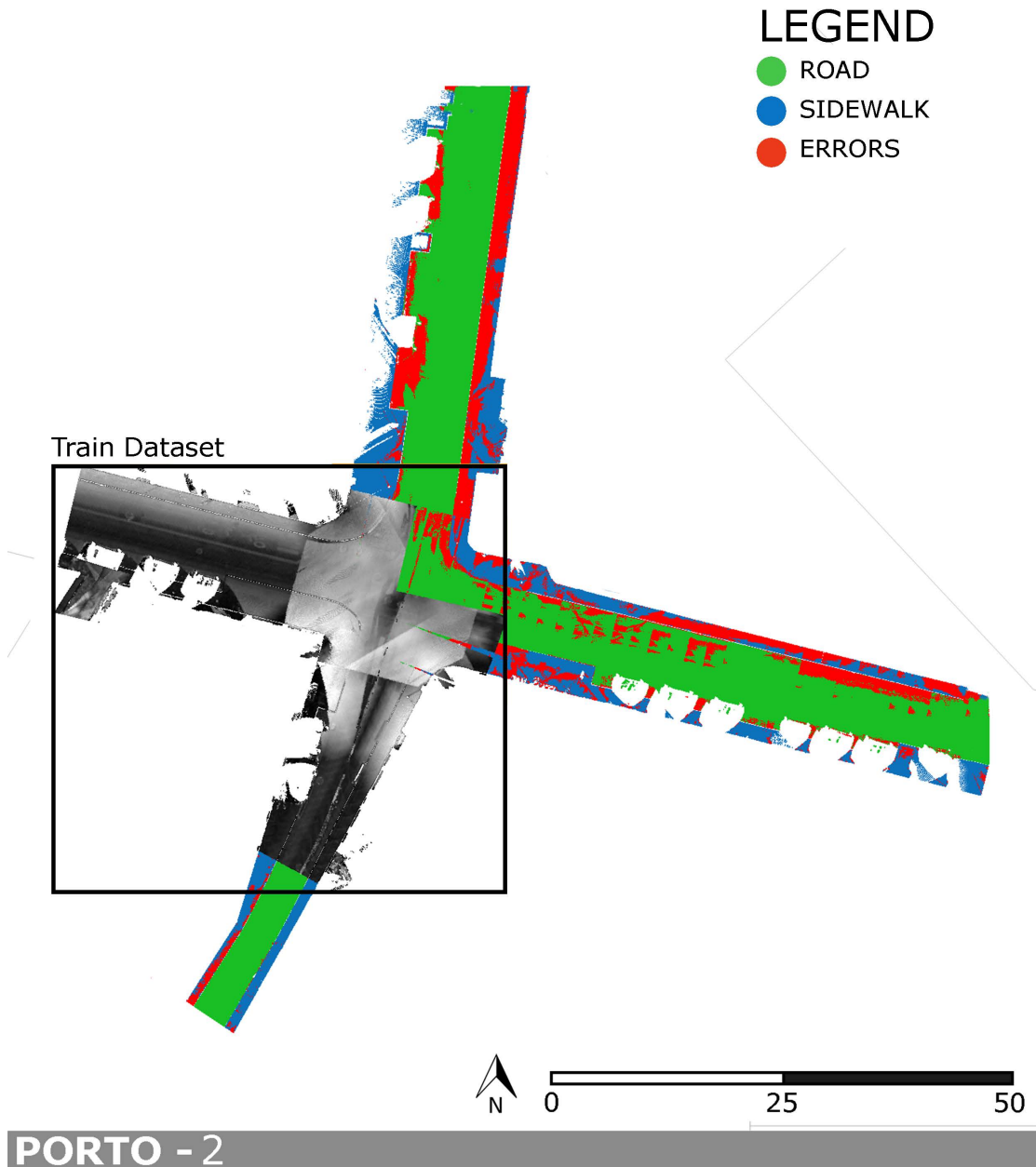


Figure A.68: Top view of the point cloud of Porto-2 after semantic segmentation with the ML method. Red points are errors. The ML model was trained on a portion (about 1/3) of Porto-2 dataset itself.

A.13 Porto - 3

The third area is a portion of Rua de São Filipe de Nery, in a segment which passes near the church *Igreja dos Clérigos*. The road is modestly large, with a large pedestrian area in the final portion. The length of the path is 104 metres, with 11 million points on the ground surface. The sidewalk paving is in stone, while the road is in sampietrini.

Here are the confusion matrices and precision metrics for the knowledge-based method and the ML approach (using both Track-C and a portion of the dataset itself as training for the model).

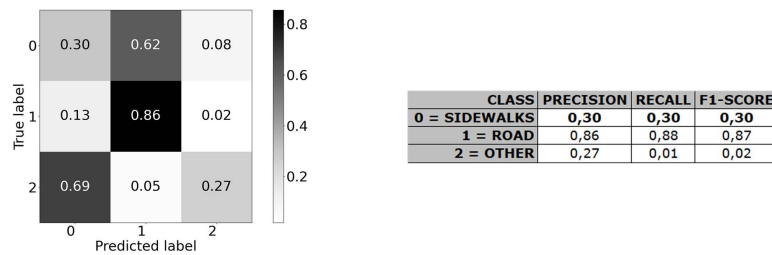


Figure A.69: Confusion matrix and precision metrics for Porto-3 dataset with knowledge-based approach.

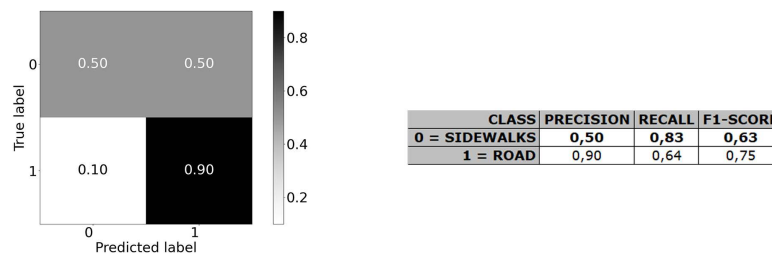


Figure A.70: Confusion matrix and precision metrics for Porto-3 dataset with ML approach. The ML model was trained on Track-C of Sabbioneta.

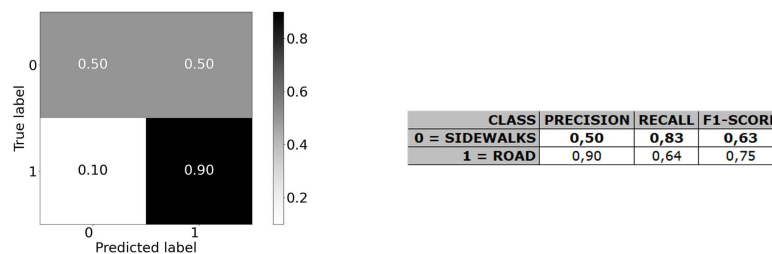


Figure A.71: Confusion matrix and precision metrics for Porto-3 dataset with ML approach. The ML model was trained on a portion (about 1/3) of Porto-3 dataset itself.

Appendix A. Semantic Segmentation results

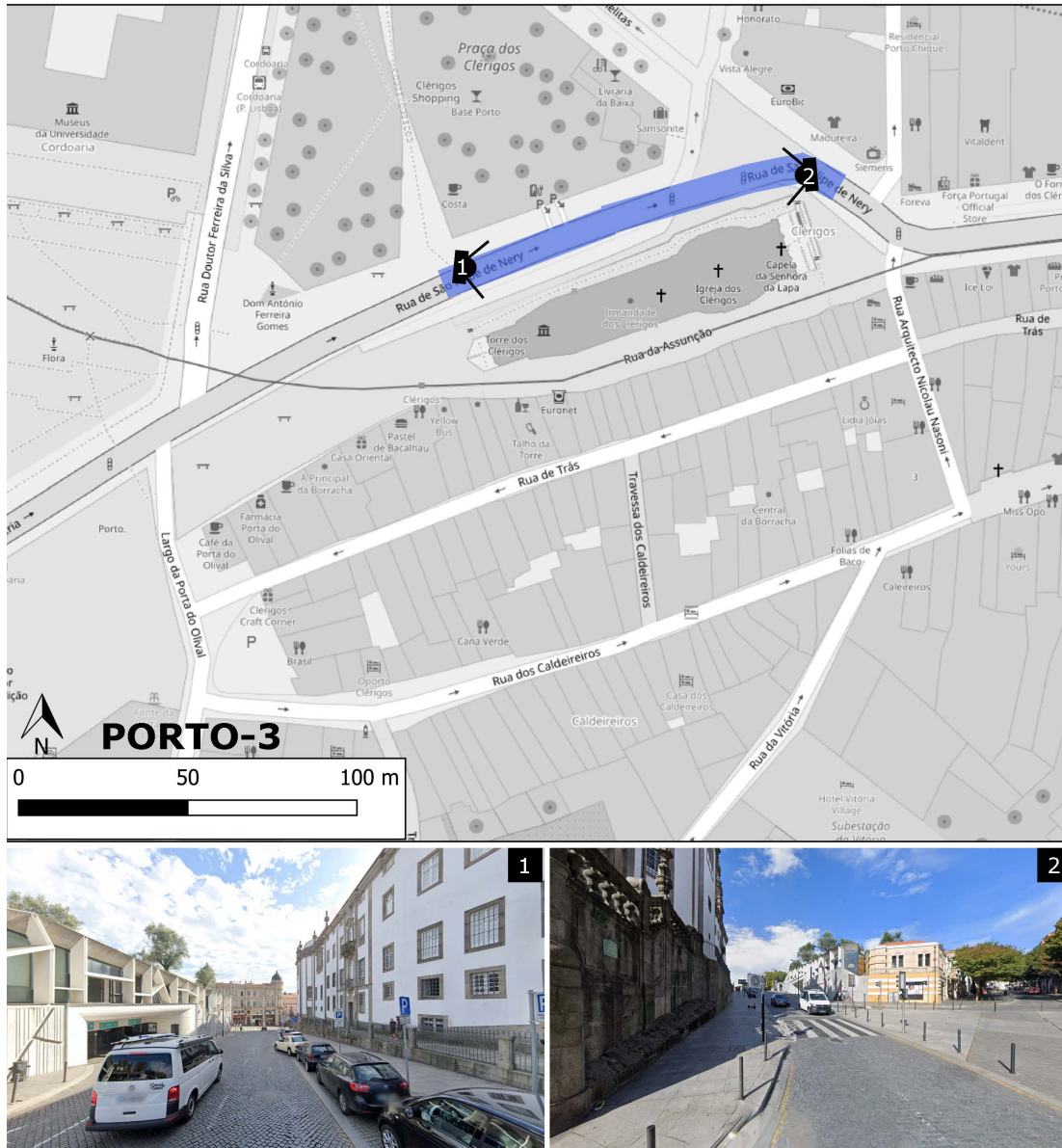


Figure A.72: Map of the path followed for the survey along the dataset in Porto-3, together with some images of the road and the city. Images were elaborated from Google Earth.



Figure A.73: Top view of the point cloud of Porto-3 after semantic segmentation with the knowledge-based method. Red points are errors. On the top part of the image, there is a pie chart with an overview of the whole points in the point cloud.

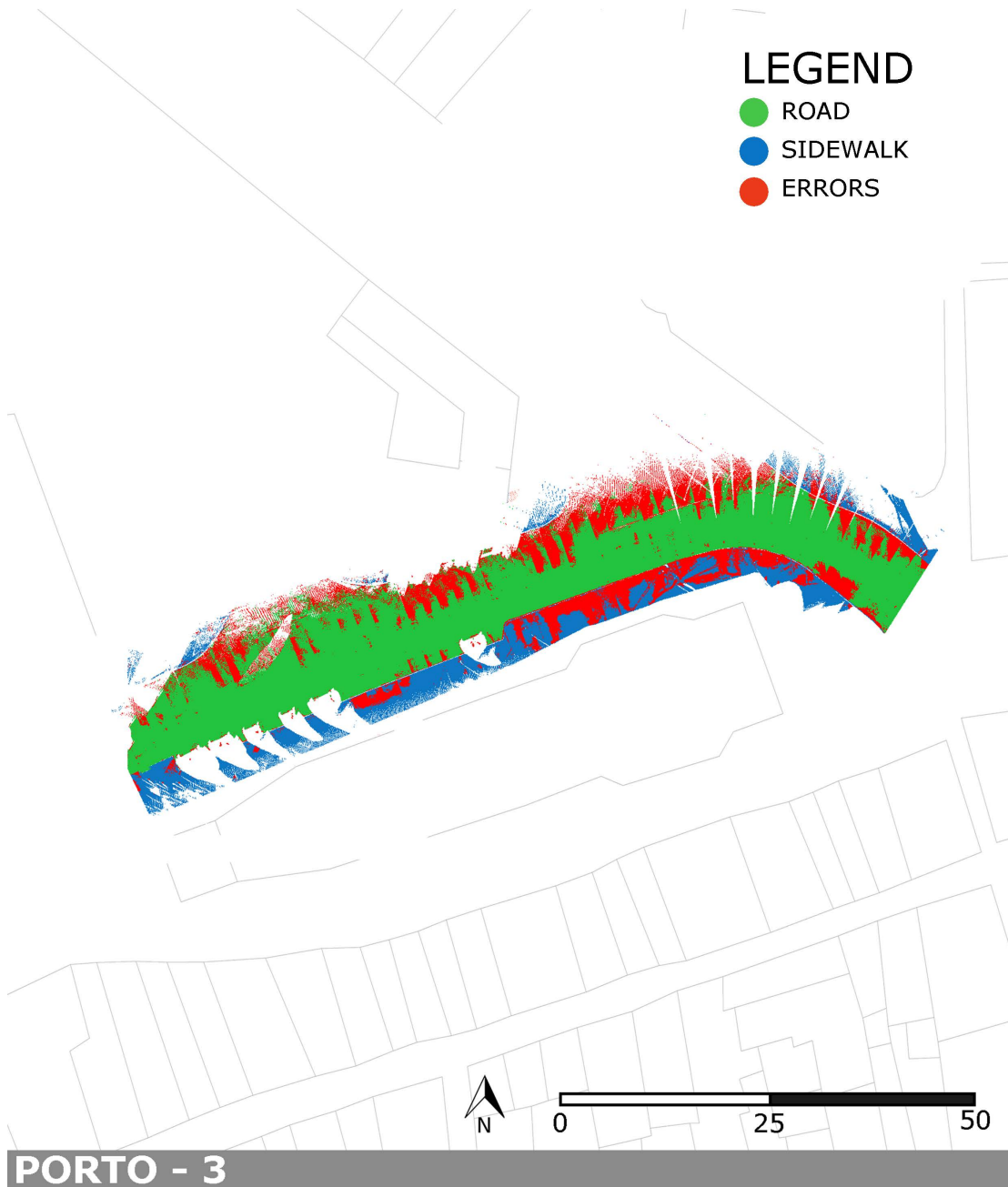


Figure A.74: Top view of the point cloud of Porto-3 after semantic segmentation with the ML method. Red points are errors. The training dataset was Track-C of Porto-1.

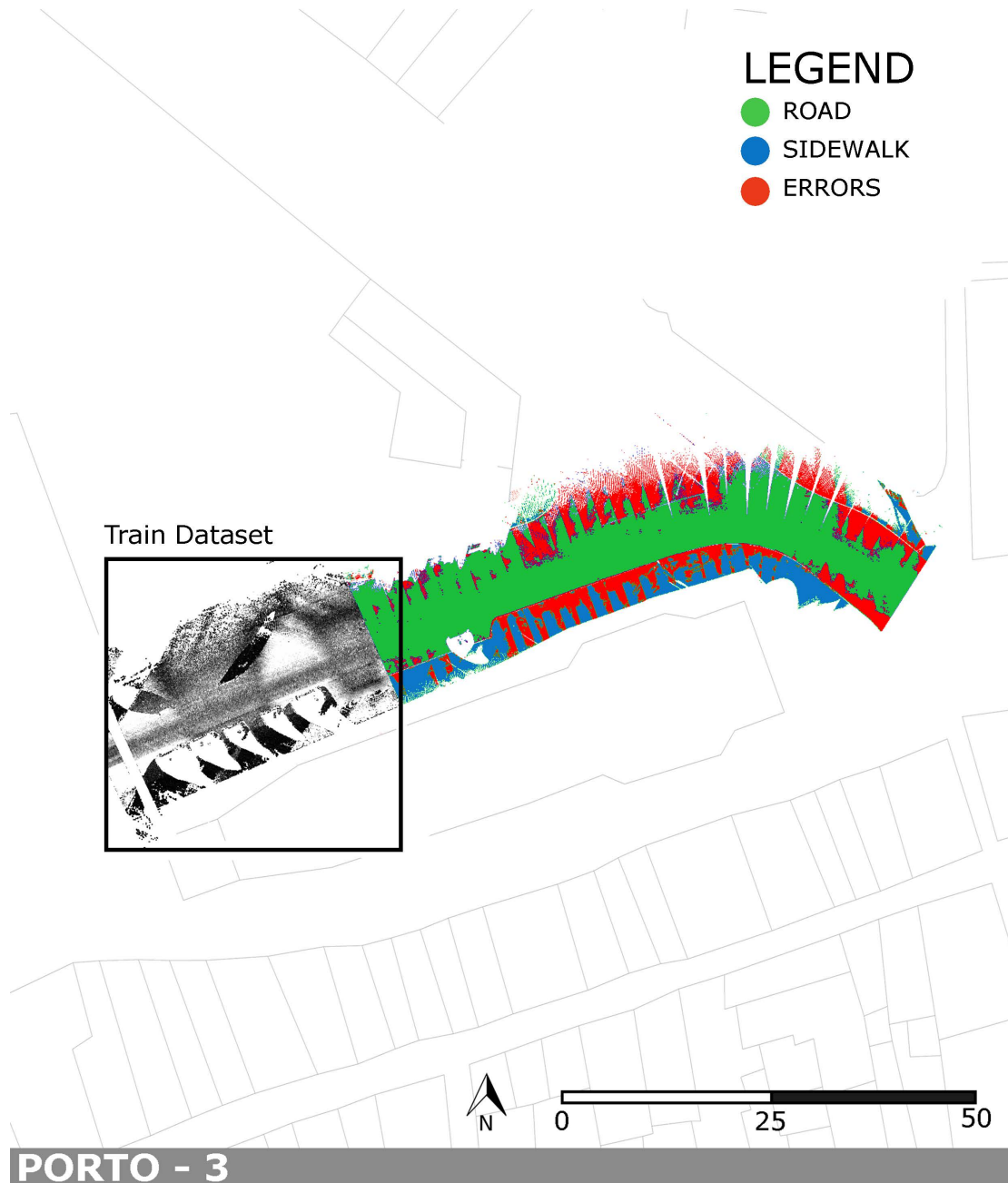


Figure A.75: Top view of the point cloud of Porto-3 after semantic segmentation with the ML method. Red points are errors. The ML model was trained on a portion (about 1/3) of Porto-3 dataset itself.