Politecnico di Milano

School of Civil, Environmental and Land Management Engineering

Master Degree in Environmental and Land Planning Engineering

POLITECNICO
MILANO 1863

ETH zürich

Evaluation of Airborne Image Velocimetry approaches using low-cost UAVs in riverine environments

Supervisor at Politecnico di Milano: Prof. Livio Pinto
Supervisor at ETH Zürich: Dr. Martin Detert

MSc Thesis of: Francesco Ioli
Matriculation ID: 899445

Thesis version: 2.0

Academic Year 2019-2020

Acknowledgments

At the conclusion of this work, I would like to express my gratitude to my supervisor at ETH Zürich Dr. Martin Detert for guiding me throughout the topic of Airborne Image Velocimetry and for helping me during the whole thesis. I would like to extend my special thanks to my supervisor at Politecnico di Milano Prof. Livio Pinto for his interest in the topic and for supporting me during the work, especially during the last few months in Milano.
I wish to acknowledge Prof. Robert Boes, head of the Laboratory of Hydraulics, Hydrology and Glaciology (VAW) at ETH Zürich, for having accepted me as a visiting student at VAW department.
I would like to thank Magali Jodeau, Jérôme Le Coz, Alexander Hauet, developers of Fudaa-LSPIV, and Antoine Patalano, developer of RIVeR, for their interest in my work and their constructive feedbacks given.
Thanks also to Jörg Hammer and to the Swiss Federal Office for the Environment (FOEN) for the data acquired on river Limmat at the Zürich Unterhard gauging station. These were essential as references against which compare my AIV results.
Finally, I would like to say thank you to Francesco Avanzi, for supporting me in the decision to move to Zürich in order to develop my thesis, and to all the friends of mine, in particular Daniele Moncecchi with whom I shared my time there.

Francesco

Abstract

Measuring flow velocities is one of the main issues of hydraulic engineering. Traditional flow measurements require contact with the fluid and are usually costly, time-consuming and, sometimes, even dangerous. The image-based technique Particle Image Velocimetry (PIV) allows the flow velocity field to be remotely characterized from the shift of intensity patterns of sub-image areas in at least two video frames with a known time lag. Recently, Airborne Image Velocimetry (AIV) has enabled the surface flow velocity of large-scale water bodies to be determined by analysing videos, recorded by cameras mounted on Unmanned Aerial Vehicles (UAVs), with PIV. This work presents a comparison of three AIV approaches: BASESURV, Fudaa-LSPIV and RIVeR. For the evaluation, two nadiral videos were acquired on river Limmat (Switzerland) with a low-cost UAV DJI Phantom 4 Pro. The first was recorded under low flow and seeded conditions, the second during a flood event. According to the results obtained, BASESURV is found to be an accurate and complete research oriented AIV approach but it is time-consuming and neither a GUI nor documentation are yet provided. Fudaa-LSPIV is a well-developed AIV software package, with a user-friendly GUI and good documentation, however it lacks some features and the source code is closed. RIVeR may be suitable for fast processing as well as for real time monitoring thanks to the efficient rectification of the velocity vectors only. Overall, all the codes are found to be effective in performing AIV in riverine environments using images taken from low-cost UAVs. In addition to the AIV codes comparison, a further experiment was carried out on river Lambro (Italy) in order to test an orthorectification approach based on Structure from Motion. This technique is promising, as it performs the rectification by estimating one 3D-to-2D transformation based on the collinearity equations for each video frame.

Keywords: AIV, PIV, low-cost UAV, image-based velocimetry, river surface velocity, BASESURV, Fudaa-LSPIV, RIVeR

Abstract in italiano

Misurare le velocità di un flusso fluido è uno dei principali problemi dell'ingegneria idraulica. Le misure tradizionali richiedono il contatto diretto con il flusso e sono solitamente costose, lunghe e, talvolta, pericolose. *Particle Image Velocimetry* (PIV) è una tecnica basata su immagini che consente di caratterizzare da remoto il campo di velocità di un fluido, a partire dallo spostamento di pattern caratteristici in almeno due frame di un video, acquisiti con un intervallo temporale noto. Recentemente, *Airborne Image Velocimetry* (AIV) ha permesso di determinare il campo di velocità superficiale di corpi idrici su larga scala, analizzando video acquisiti da camere montate su drone. A questo scopo, i video frame devono essere innanzitutto stabilizzati e rettificati. Successivamente, il campo di velocità può essere calcolato con la tecnica PIV. Questo lavoro presenta un confronto e una valutazione di tre diversi approcci di AIV: BASESURV, Fudaa-LSPIV e RIVeR. Per il confronto sono stati acquisiti due video nadirali sul fiume Limmat (Svizzera), usando un drone commerciale economico DJI Phantom 4 Pro. Il primo video è stato registrato durante condizioni di deflusso stabili e confrontabili con la portata media mensile di ottobre. Sono stati usati inoltre dei traccianti biodegradabili, composti di amido di mais, per aumentare il contrasto nelle immagini. Il secondo video è stato acquisito durante una piena confrontabile con l'evento di periodo di ritorno annuale e non è stato usato alcun tracciante. Sulla base dei risultati ottenuti, BASESURV si è rivelato il miglior software per applicazioni scientifiche: è l'approccio più accurato e completo, ma anche quello che richiede maggior tempo computazionale. Inoltre, non è stata ancora implementata un'interfaccia grafica nè è stata redatta una documentazione. Fudaa-LSPIV è risultato essere il miglior approccio per applicazioni professionali: è infatti un software ben sviluppato, con un'interfaccia grafica semplice da usare e una buona documentazione. Nonostante ciò, è carente di alcuni strumenti utili come il pre-processamento delle immagini e dei filtri per l'identificazione degli outlier basati sulle serie temporali. Inoltre, il codice sorgente del software non è pubblico. RIVeR, infine, potrebbe essere adatto per applicazioni in cui è necessario un'elaborazione rapida delle immagini, come nel campo del monitoraggio in tempo reale, grazie al suo efficiente approccio di rettificazione dei vettori velocità invece che delle immagini. Nel complesso, tutti i tre approcci di AIV sono risultati efficaci per il calcolo della velocità superficiale in ambiente fluviale usando immagini acquisite da droni economici. In aggiunta al confronto dei codici di AIV, è stato sviluppato un ulteriore esperimento sul fiume Lambro (Italia), finalizzato alla valutazione di un metodo di ortorettifica delle immagini basato sulla fotogrammetria *Structure from Motion*. Questa tecnica si è rivelata promettente in quanto consente la rettifica di ogni video frame mediante una trasformazione geometrica *3D-to-2D* basata sulle equazioni di collinearità.

Parole chiave: AIV, PIV, droni, immagini, velocità fluviale superficiale, BASESURV, Fudaa-LSPIV, RIVeR.

Table of Contents

1 Introduction **1**
- 1.1 Particle Image Velocimetry (PIV) . . . 2
- 1.2 Airborne Image Velocimetry (AIV) . . . 3
- 1.3 Goals of the study . . . 5
- 1.4 Thesis outline . . . 5

2 Overview of current AIV approaches **7**
- 2.1 BASESURV . . . 7
- 2.2 Fudaa-LSPIV . . . 7
- 2.3 RIVeR . . . 8

3 Experiments setup **11**
- 3.1 River Limmat (CH) . . . 11
  - 3.1.1 Study area . . . 11
  - 3.1.2 Datasets A and B . . . 11
  - 3.1.3 Spatial framework of the surveys and reference SfM models . . . 12
- 3.2 River Lambro (IT) . . . 13
  - 3.2.1 Study area . . . 14
  - 3.2.2 AIV survey and dataset C . . . 14
  - 3.2.3 Topographic survey and reference SfM model . . . 15
  - 3.2.4 Simulations of the photogrammetric blocks . . . 16

4 Processing of videos A and B **19**
- 4.1 Video frames extraction and undistortion . . . 19
- 4.2 Video stabilization and rectification . . . 19
- 4.3 Image pre-processing . . . 23
- 4.4 PIV . . . 24
  - 4.4.1 BASESURV . . . 24
  - 4.4.2 Fudaa-LSPIV . . . 24
  - 4.4.3 RIVeR . . . 25
  - 4.4.4 Time Lag . . . 25
- 4.5 Filtering of spurious vectors . . . 27
- 4.6 Computation of the time-averaged velocity fields . . . 28
- 4.7 Velocity time series and cross-sections . . . 28
- 4.8 Estimation of the discharge . . . 30

5 Results of the AIV approaches comparison **31**
- 5.1 Dataset A . . . 31
- 5.2 Dataset B . . . 37

6 Discussion **43**

7 Studies on SfM-based orthorectification using video C **49**
7.1 Orthophotos generation 49
7.2 Orthorectification procedure C1 and C2 50
7.3 Orthorectification accuracy assessment 54

8 Summary and conclusions **57**

Appendix **58**

A Time-averaged surface velocity fields **59**

B Comments from the Fudaa-LSPIV developers team **69**

C Comments from the RIVeR developers team **71**

D Comments from the BASESURV developer **73**

Bibliography **77**

List of Figures

1.1 Scheme of a traditional PIV experiment (Raffel, Willert, Scarano et al. 2018). 2

1.2 (a) Example of integer displacement estimation computed as the location of highest peak of the correlation intensity distribution over an IA with size 64×64 px. The PIVlab using the provided sample data. (b) Example of sub-pixel displacement estimation using 3-point Gaussian fitting along the *x-axis* only (figure taken and adapted from Thielicke and Stamhuis (2014)). 3

1.3 Typical workflow to be followed to perform an AIV analysis. 4

2.1 Screenshot of the typical Fudaa-LSPIV software window. 8

2.2 Screenshot of the typical RIVeR software window. 9

3.1 Photo of the Limmat - Zürich Unterhard gauging station. On the image are marked: the flow direction (arrows); the area within which the surface velocity is computed through AIV (sold line); the rope used to control monitoring instruments such as the ACDP (dashed line). 12

3.2 MEDEWO FILL-PAC Bio chips. The diameter of the chips were about 3 cm 13

3.3 The quadcopter DJI Phantom 4 Pro+ used for the two surveys on river Limmat. 13

3.4 Screenshot of the SfM model M1 produced with Agisoft Metashape with the data acquired on 10/10/2018 by M. Detert and co-workers. The blue flags visible on the model are some of the markers used either as GCP or check points. 14

3.5 Photo of the study area on river Lambro in proximity to Diga delle Fornaci in Inverigo (IT). On the image are marked: the flow direction (arrows); the area within which the surface velocity is computed through AIV (sold line); 8 of the 16 GCPs used to georeference the survey (crosses). 15

3.6 Mesh model built with Agisoft Metashape by using a combination of nadiral and oblique images. On the model are marked: the GCPs (flags with labels); the flow direction (arrows); the region of interest (solid line). 16

3.7 Geometry of the simulations: the world reference system used is parallel to the GCPs plane (a); acquisition geometry of simulation S1 with only GCPs (b); acquisition geometry of simulation S2 with GCPs and TPs (b). 17

3.8 Results of the simulations: (a) S1; (b) S2. For each EO parameter, the RMS and the maximum value of the simulated standard deviations of the 15 cameras are plotted. 18

4.1 The binary mask applied on the rectified frame of the video A in the BASESURV rectification tool: only the RGB area corresponds to the visible part of the binary mask and therefore has been used in interesting points research. The south-west area in which there are no points at the water surface level along the riparian side is marked in the box. 20
4.2 Mask applied in Fudaa on the first frame of video A. The red area, marked by the software as *Flow area*, is the one excluded by the mask. . 21
4.3 Drift error in the stabilization of video A performed with RIVeR: (a) the first and last stabilized frames are plotted in false colours, respectively in magenta and green, to enhance their differences; (b) cut-out at the location marked in (a) in which the shift between the frames is evident. 22
4.4 Example of image pre-processing: (a) Cut-out at the location marked in (b) of a video frame after the background subtraction. The white pixels are the moving objects (i.e. seeding particles) which are clearly distinguishable from the nearly-black background. 23
4.5 Simplified scheme of the cross-correlation algorithm implemented in BASESURV. For each grid point, the cross-correlation function between an IA of e.g. 32×32 px in image i_1 and the corresponding IA in i_2 is computed in the frequency domain. The most probable displacement vector is given by the location of the correlation peak. Images taken and adapted from Raffel, Willert, Scarano et al. (2018). 24
4.6 Scheme of the cross-correlation algorithm implemented in Fudaa-LSPIV for a sample grid point a_{ij}. The IA (green square) centred in a_{ij} on the first image is searched on the second image within the Search Area (blue rectangle), by maximizing the correlation coefficient $R(a_{ij}, b_{ij})$ (Jodeau et al. 2019). 25
4.7 Scheme of PIV processing sequence in (a) BASESURV and (b) Fudaa and RIVeR if, for instance, an interval $n_f = 4$ frames is used. 26
4.8 (a) Cut-out of instantaneous velocity field computed by BASESURV between frames # 151 and #155 for dataset A at the location marked in (b). The light red vectors are the ones rejected by PIVlab instantaneous vector filters. 28
4.9 Example of time-averaged surface velocity field over a window of 35 s. The median velocity vector computed on each grid point is superimposed to its magnitude, represented with a scale of colours. 29
4.10 Location of the cross-sections and the grid points at which the vector time-series is extracted, respectively for: (a) Dataset A; (b) Dataset B. . 29

5.1 Time-averaged surface velocity field obtained from dataset A with the three different AIV approaches: (a) BASESURV median velocity field; (b) RIVeR average velocity field; (c) Fudaa average velocity field; (d) Fudaa median velocity field. For larger figures, see App.A 32
5.2 Time series and scatter plot of the two cartesian velocity components u and v at grid points as follows: (a) Pt. A1, upstream of the ACDP rope; (b) Pt. A2, downstream the ACDP rope. The location of the two points is given in Fig. 4.10a. 33
5.3 (a) Cut-out at the location marked in (b) of the BASESURV time-averaged velocity field. The noise caused by the shadows of the trees is clearly visible; (b) Location of the cut-out. 34
5.4 Velocity profiles at the cross-sections: (a) AA'; (b) BB'; (c) CC'. 35

5.5 Velocity profile measured by the hydrometric impeller (0.19 m below the water surface) compared against the surface velocity profiles obtained by AIV along the cross-section BB' (its location is given in Fig. 4.10a). These have been resampled in the same points as the bathymetry depth profile and extrapolated with the approach proposed by Le Coz, Hauet et al. (2010) to fill missing data. 36

5.6 Time-averaged surface velocity field obtained from dataset B with the three different AIV approaches: (a) BASESURV median velocity field; (b) RIVeR average velocity field; (c) Fudaa average velocity field; (d) Fudaa median velocity field. All the results are computed using a time lag of 0.042 s. For larger figures, see App.A 38

5.7 Time series and scatter plot of the two cartesian velocity components u and v at grid points as follows: (a) Pt. B1 on the left-hand side; (b) Pt. B2 on the right-hand side. The location of the two points is given in Fig. 4.10b. 39

5.8 (a) Sample frame of video B. The orange arrow shows the flow direction. The water coming from river Shil, carrying lot of suspended sediments, can be clearly distinguished. (b) Orthophoto of the city of Zürich showing the confluence of river Sihl into river Limmat, just ∼200 m upstream of the Zürich Unterhard gauging station. 40

5.9 Time-averaged surface velocity field computed with BASESURV using different time lag in PIV: (a) $\Delta t = 0.042\,\mathrm{s}$; (b) $\Delta t = 0.501\,\mathrm{s}$. 41

5.10 Comparison of the velocity profiles along the cross section DD' obtained by AIV (the location is given in Fig. 4.10b). These have been resampled in the same points as the bathymetry depth profile and extrapolated with the approach proposed by Le Coz, Hauet et al. (2010) in order to fill missing data. Both the velocity profiles obtained with BASESURV using the two different time lags are plotted. 41

7.1 Example of (a) raw video frame and (b) orthorectified frame. The railing is still visible in the orthophoto because it was not properly reconstructed in the mesh model due to a lack of oblique photos (see Sec. 3.2.3). . . . 50

7.2 Sfm products computed with the C1 approach. (a) Sparse cloud generated by ∼400 TPs by solving the bundle block adjustment with 20 video frames. The colorbar refers to the RMS of standard deviations of the TPs coordinates in the world reference system. TPs with the largest standard deviations were previously removed. (b) Perspective view of the estimated position and orientation of the 899 images with respect to the mesh model. The image is distorted by a perspective effect in order to enhance the different location of the cameras. For a scaled plot of the camera projective centres, one may refer to Fig. 7.4. 51

7.3 For each EO parameter, the RMS and the maximum value of the standard deviations of the 899 cameras, estimated by Metashape, are plotted. The coordinates are expressed in the local reference system defined by the GCPs, which is different than that used for the simulations S1 and S2. 52

7.4 Centres of projection of the cameras estimated in C1 (blue) and C2 (red). 53

7.5 Boxplots of the differences DX, DY, DZ of the coordinates of the projection centres of each camera estimated with the two approaches, computed as $\xi_{C1} - \xi_{C2}$, where ξ is a generic coordinate. The red line marks the median of the differences in each coordinate. The left and right edges of the box (blue lines) indicate respectively the 25^{th} and 75^{th} percentiles. The whiskers extend to the most extreme data points not considered outliers. The outliers are individually plotted using the '+' symbol. . . . 53
7.6 Time-averaged surface velocity field computed with the approaches: (a) C1; (b) C2. The two cutout areas, in which PIV is employed for the orthorectification error assessment (see Fig. 7.7), are marked with coloured bounding boxes. 55
7.7 Results of PIV computation over the Left and Right hand cutout for both the approaches C1 and C2. The standard deviation of the velocity vector time-series at each grid point is plotted with a scale of colour. In order from top-right up to bottom left: (a) Left hand cutout computed with the approach C1; (b) Left hand cutout computed with the approach C2; (c) Right hand cutout computed with the approach C1; (d) Right hand cutout computed with the approach C2. 55

List of Tables

3.1 Summary of the main characteristics of the two datasets used in this study. The hydrological information about the river Limmat at the time of the surveys is taken from Swiss Federal Office for the Environment FOEN (2020). 12
3.2 Geometrical accuracy of the 3D models M1 and M2, computed through SfM from the data acquired respectively on 10/10/2018 and 10/10/2019 13

4.1 Rectification error of the first frames of video A and B computed from the reprojection error of the double points. BASESURV rectification is based on homography transformation, Fudaa on the full DLT. 20
4.2 Summary of the main differences in stabilization and rectification for the three AIV approaches. IP stands for Interesting Points; VF for the generic Video Frame; RF (Reference Frame) for the first, manually rectified, frames. 22
4.3 Main PIV parameters used in BASESURV, Fudaa and RIVeR respectively for dataset A and B. Column n_f refers to interval in terms of number of frames within every pair of images used for PIV and Δt is the relative time lag. For video B, two PIV analysis have been performed with BASESURV using different time lag. 27

5.1 Comparison between the discharge measured by the hydrometric impeller and those estimated by AIV. For the latter ones, the interval of possible values, depending on the parameter β as explained in Sec. 4.8, is given. 36
5.2 Comparison between the discharge values estimated by AIV against those estimated with the rating curve of river Limmat. For the first ones, the interval of possible values, depending on the parameter β as explained in Sec. 4.8, is given. 40

6.1 Comparison of the time needed by the three AIV approaches to process video A, normalized by the number of frames involved in the computation. 44

Chapter 1

Introduction

Measuring flow velocity field is one of the main issues of hydraulic engineering. In contrast to laboratory conditions, field measurements are usually difficult due to the large spatial and temporal variability of the flow, costly, time-consuming and sometimes even dangerous. In the nineteen eighties, an image-based technique named *Particle Image Velocimetry* (PIV) was developed to capture the whole velocity field without being directly in contact with the fluid, but recording the flow with a camera (Adrian 1991; Raffel, Willert and Kompenhans 1998). The instantaneous velocity field is computed from the shifts of characteristic intensity patterns of subimage areas between two frames with a known time lag, obtained through a cross-correlation analysis (Keane and Adrian 1992). PIV has been successfully used under laboratory conditions over the past 35 years (Adrian 2005) and, more recently, this technique has been adapted to measure the velocities at the free surface of a water body: since this technique allows wide areas to be investigated, it has been labelled by Fujita and co-workers as *Large-Scale Particle Image Velocimetry* (LSPIV) (Muste, Fujita and Hauet 2008).

In the traditional PIV technique, the camera is required to be fixed at an elevated position compared to the river flow in order not to have any apparent ground displacement due to its shaking. However, the possibility of performing the analysis from a non-fixed station, but flying or hovering above the river is promising because it provides more degrees of freedom and more flexibility compared to a ground based station. *Airborne-based Image Velocimetry* (AIV) was first developed in Japan by Fujita and Hino (2003) and Fujita and Kunita (2011), who used images taken from a helicopter to compute the surface velocity of a riverine environment under both low flow and flood conditions.

In the past ten years, UAVs (i.e. Unmanned Aerial Vehicles) and especially off-the-shelf multi-copter drones have become affordable instruments to be used for several civil engineering purposes. First UAV-based AIV experiments were conducted by Fujita, Notoya and Shimono (2015) and Detert and Weitbrecht (2015). Their works were mostly focused on video stabilization based on projective transformation and on PIV.

More recently, Detert, Johnson and Weitbrecht (2017) have developed a proof-of-concept for low-cost AIV by employing Structure from Motion (SfM) techniques to obtain the orthophotos for velocimetry and discharge computation. A similar approach was used by Detert, Cao and Albayrak (2019) to measure a large surface velocity field in proximity to the Schiffmühle hydropower plant on river Limmat, Switzerland. In both studies, the cost of the equipment used for the measurements was less than €2000.

In this work, a comprehensive overview over available AIV approaches is carried out in order to evaluate their performances in computing the surface velocity of a river using low-cost UAVs.

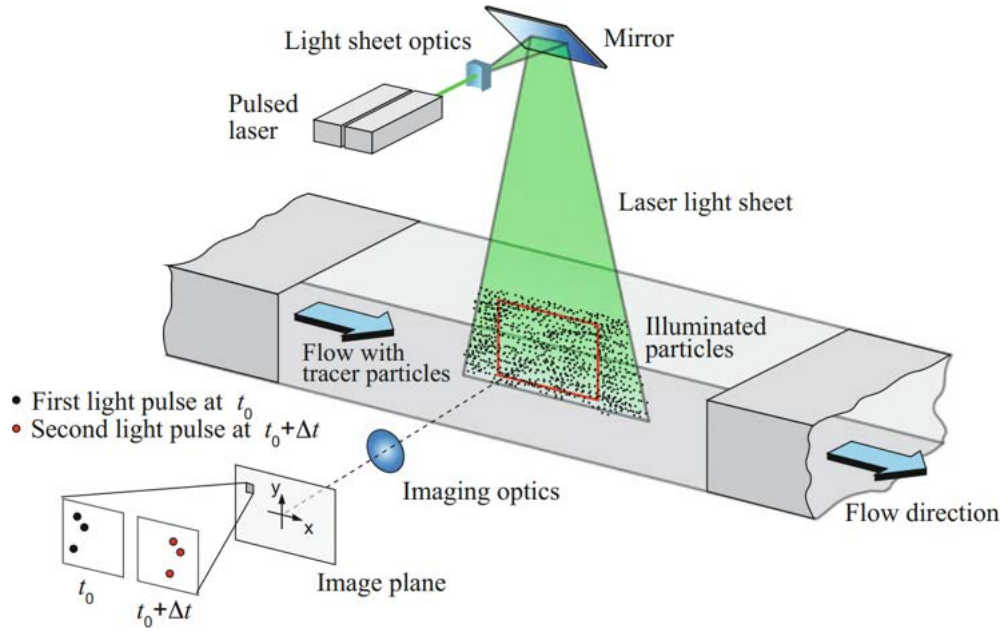
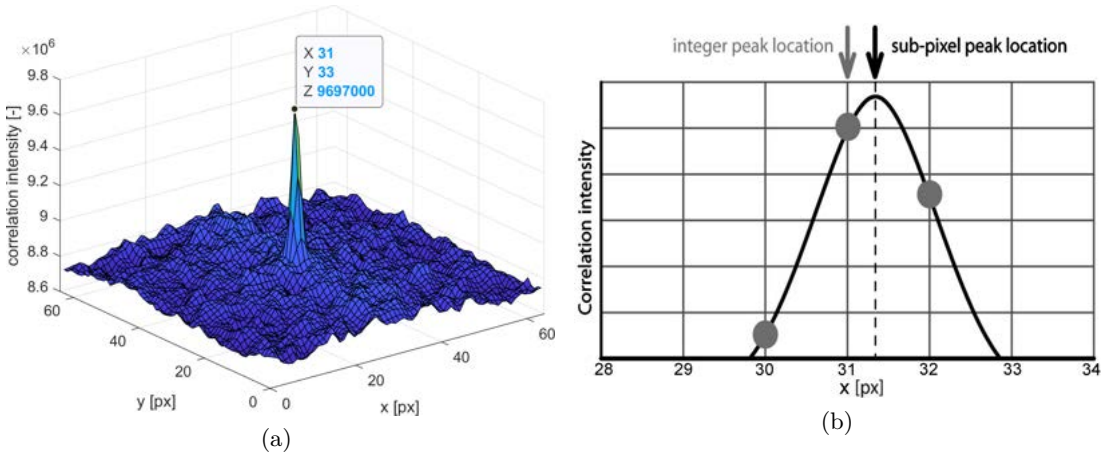


Figure 1.1: Scheme of a traditional PIV experiment (Raffel, Willert, Scarano et al. 2018).

1.1 Particle Image Velocimetry (PIV)

PIV is traditionally used under laboratory conditions to measure the whole flow velocity field of a fluid. A thin sheet of fluid, typically containing high reflective particles, is illuminated by laser light pulses at least two times, spaced at a known interval Δt. A camera is positioned parallel to the illuminated sheet to capture the movement of the tracer particles (Fig. 1.1). The local displacement vectors of the particles between two consecutive images are computed by a cross-correlation statistical method. Each image is divided into small sub-areas called Interrogation Areas (IAs) in such a way that a large enough number of particles is present therein. A minimum number of 5 particles within the IA can be considered as a rule of thumb (Keane and Adrian 1992). The cross-correlation can be performed either in the spatial domain by computing the correlation matrix or in the frequency domain by multiplying the Fourier spectra of the two IAs (Raffel, Willert, Scarano et al. 2018). The most probable displacement vector is computed from the position of highest peak of the correlation function (Fig. 1.2a). The integer displacement obtained may then be refined using a sub-pixel estimation technique (Nobach, Damaschke and Tropea 2005; Nobach and Honkanen 2005). The standard procedure is to use a 3-point Gaussian sub-pixel estimation: two one-dimensional Gaussian functions are fitted on the integer intensity distributions for both axes independently. Their maxima are used to determine the displacement of the particles with sub-pixel precision (Fig. 1.2b) (Thielicke and Stamhuis 2014). This procedure is repeated for all the IA in which the images are divided, resulting in one displacement vector for each area. Moreover, some PIV codes implement an iterative multi-pass approach (Scarano and Riethmuller 1999). This is an iterative procedure for which cross-correlation is applied several times, called passes, on the same area and the integer result of the first pass is used to offset the IA in the following ones. At the same time, the size of the IA may be gradually reduced at every iteration. Such technique allows for both a finer resolution in space and a higher dynamic range of the resolvable velocities. Finally, the flow velocity field is derived from the displacement vectors and



(a) (b)

Figure 1.2: (a) Example of integer displacement estimation computed as the location of highest peak of the correlation intensity distribution over an IA with size 64×64 px. The PIVlab using the provided sample data. (b) Example of sub-pixel displacement estimation using 3-point Gaussian fitting along the *x-axis* only (figure taken and adapted from Thielicke and Stamhuis (2014)).

the time interval Δt between the two images. For a more detailed explanation about the PIV methodology, one can refer to Adrian (1991), Keane and Adrian (1992) and Raffel, Willert, Scarano et al. (2018).

Large-scale flow measurements in natural riverine environments, however, are more challenging as the set-up described before is not longer applicable. First of all, it is not possible to illuminate the fluid with a laser beam, but the images are subjected to natural light conditions. Also distributing the reflective tracers uniformly over the area of interest can be difficult (and sometimes also dangerous for the operators). Moreover, the areas to be investigated are usually significantly larger than those typical of laboratory experiments. It is therefore more difficult to distinguish the intensity pattern (e.g. given by the tracers) from the background and the signal must be extracted adequately from the noise, which is inevitably widely present in the images.

1.2 Airborne Image Velocimetry (AIV)

Airborne Image Velocimetry (AIV) enables the surface flow velocity of a river to be characterized by recording a video while flying or hovering over it by a camera mounted on an Unmanned Aerial Vehicle (UAV), for instance. The typical AIV workflow to be followed is summarized in Fig. 1.3.

Although the performance of the recent UAVs in keeping a stable position while they are hovering have improved in the past few years, the vibrations induced by the propellers and the wind still introduce some shaking effects in the recorded video. This must be corrected before applying PIV, in order to avoid spurious displacement vectors due to the apparent movement of the images. Therefore, video frames have to be first stabilized and rectified using GCPs or existing orthophotos of the study area. In order to stabilize the video, several techniques have been investigated: despite some differences, those mainly consist in warping the video frames by applying a geometric transformation (usually a projective transformation), estimated on the basis of double points detected on pairs of video frames.

Once the video is stabilized, the frames have to be rectified in order to correct the

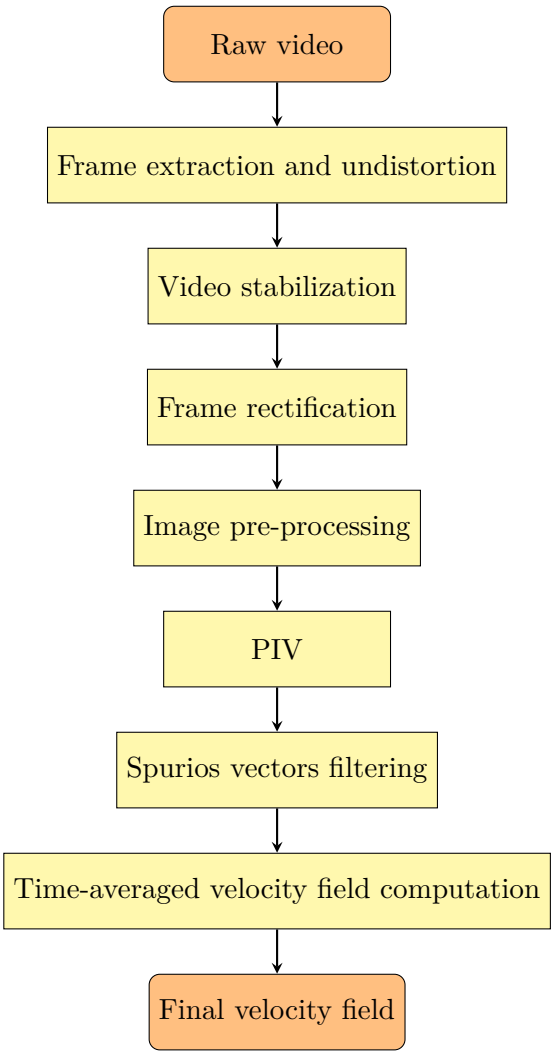


Figure 1.3: Typical workflow to be followed to perform an AIV analysis.

perspective effect due to the camera central projection and to provide metric information to the pixels. In order to achieve this, a transformation between the 2D image reference system and the 3D world reference system (e.g. a Direct Linear Transformation, DLT) has to be estimated. Under the assumption that the video is properly stabilized, only one transformation can be estimated and applied to all the frames.

Several image enhancing techniques may be applied to the video frames to improve the correlation analysis (e.g. by subtracting the video background to the images in order to emphasize the moving particles). PIV is then applied to compute velocity vectors over a regular grid of points for each pair of video frames with a known time lag, as described in Sec. 1.1. Therefore, a sequence of instantaneous surface velocity fields is built up. In the natural environment, generally, a large number of spurious velocity vectors result from the PIV analysis. Those should be identified and filtered out in order not to have noisy or biased results. Finally, the instantaneous velocity fields can be averaged to compute a time-averaged velocity field over a certain time window.

1.3 Goals of the study

In recent years, several studies on the application of PIV to riverine environments or open channel have been developed, but just a small number of software packages suitable for AIV analysis have been released to the public. Therefore, the goal of this work is to compare the performance of three different AIV codes in computing the surface velocity field of natural river flow using a low-cost UAV platform.

The following AIV approaches are considered:

- *BASESURV*, developed recently by M. Detert and co-workers;
- *Fudaa-LSPIV*, developed by M. Jodeau, A. Hauet, J. Le Coz and co-workers (Le Coz, Jodeau et al. 2014);
- *RIVeR* by A. Patalano and co-workers (Patalano, García and Rodriguez 2017).

The comparison was carried out by performing the full AIV process based on two videos acquired on river Limmat (Switzerland) under different flow conditions (i.e. low and high flow rate) and both with and without seeding particles, used as tracers. The framework adopted for this study allows for the evaluation of the performance of the three codes in the different steps of the AIV process. At the same time, this work aims to test the ability of the AIV technique, based on low-cost equipment, to provide reliable results in terms of surface velocity field, without being in contact with the river flow.

In addition to the comparison of the AIV approaches described above, a further experiment was set up on river Lambro (Italy) to test a further orthorectification approach based on a complete Structure from Motion technique. The method lays the foundations on the work of Detert, Huber and Weitbrecht (2016) and Detert, Johnson and Weitbrecht (2017). They computed the external orientation of the video frames by using the SfM software package Agisoft Photoscan (nowadays called Metashape) and produced the orthophotos by means of an existing 3D model. The study presented in this work aims to go further and to make use of the ability of Agisoft Metashape to automatically detect codec targets in the images. This would enable the possibility to robustly estimate the external orientation of every video frame, not only on the basis of tie points with the other images, but also with a large enough number of ground control point.

1.4 Thesis outline

The thesis is organized as follows:

- Chapter 1 provides an introduction to the study and the topics of PIV and AIV. The goal of the work and its structure are also presented.
- Chapter 2 gives an overview of the existing AIV approaches (BASESURV, Fudaa-LSPIV and RIVeR), focusing on the general logic behind each code.
- In Chapter 3, the area under study on river Limmat (CH), the equipment used and the two collected datasets are described. An overview of the spatial reference frame in which the work has been conducted is also presented. Moreover, the setup of the experiment carried out on river Lambro (IT) and the simulation of the photogrammetric blocks are described.

- In Chapter 4 the processing workflow performed with BASESURV, Fudaa-LSPIV and RIVeR for the two Limmat datasets is illustrated.
- Chapter 5 presents the results of the comparison of the three AIV approaches.
- In Chapter 6, the main strengths and limitations of the three codes are discussed.
- Chapter 7 describes the study on SfM-based orthorectification carried out on river Lambro (IT), the results obtained and possible future developments of this approach.
- In Chapter 8, a summary of the work is presented and some recommendations for possible practical application of AIV are provided.

Chapter 2

Overview of current AIV approaches

In this chapter, an overview of the three different AIV approaches considered in this study is provided.

2.1 BASESURV

BASESURV (acronym that stands for *BASic Environment for SURface Velocity computation*) is a research-oriented software developed recently by M. Detert at the Laboratory of Hydraulics, Hydrology and Glaciology (VAW) at ETH Zürich. The software allows the surface velocity of a fluid to be resolved starting from a video or a sequence of images acquired with a known time interval under laboratory conditions as well as in natural riverine environment. It is written in MATLAB®, but the code is currently not released to the public because it is still under development by the author. Neither a GUI nor documentation about the code are yet available. This implies that the user must go to code in depth and know the programming language to handle the software.

According to the typical AIV workflow explained in Sec.1.2, four main processing blocks can be identified in the software:

a) a tool for the video frames orthorectification and stabilization, performed at the same time

b) the image pre-processing block

c) the PIV part based on the code taken from PIVlab 1.42, developed by William Thielicke (Thielicke and Stamhuis 2014) and released on 10/01/2018

d) an effective time-based filtering tool to exclude spurious velocity vectors from the computation

2.2 Fudaa-LSPIV

Fudaa-LSPIV is an *all-in-one* software package developed by M. Jodeau, A. Hauet, J. Le Coz and co-workers for processing flow image sequences to calculate surface velocity fields and discharge across certain cross-sections. Its development has been executed by DeltaCAD since 2010 with financing and under the direction of EDF and Irstea (France) (Jodeau et al. 2019). Fudaa-LSPIV has a Java GUI (Fig. 2.1), which

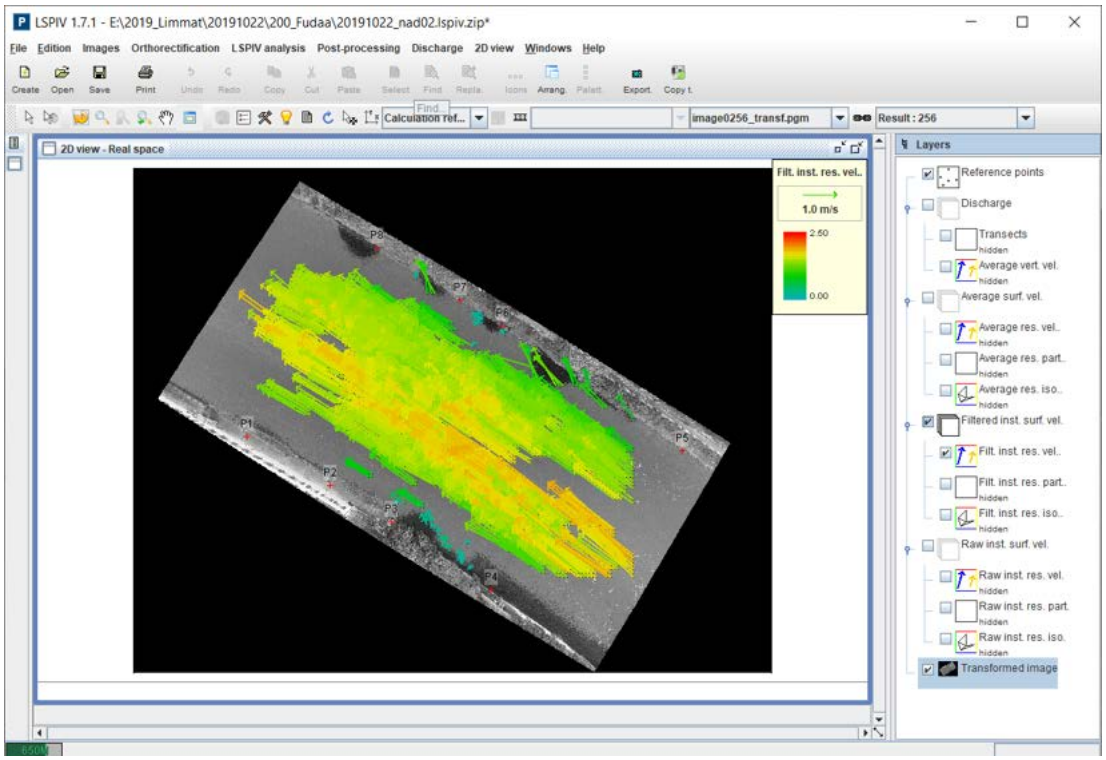


Figure 2.1: Screenshot of the typical Fudaa-LSPIV software window.

calls executables written in Fortran. It is diffused to the public freely under GPL licence. The version used in this work is Fudaa-LSPIV 1.7.1, released on 03/06/2019. Fudaa-LSPIV is equipped with a good user manual, that guides the user in the use of the software as well as provides detailed explanations about the algorithms implemented. This, together with the user-friendly graphical interface, makes Fudaa-LSPIV easy to be used.

The AIV workflow with Fudaa-LSPIV is straight forward and the processing can be fully performed inside the software. The main processing steps are the following (Jodeau et al. 2019):

a) Video stabilization (optional)

b) Frame orthorectification

c) PIV analysis, performed with an internal code

d) Post-processing with some vector filtering tools and computation of time averaged surface velocity field

e) Discharge computation on the basis of a bathymetric transect and a velocity correction coefficient (this module is not considered in this work).

The author is aware of the existence of Fudaa as a distributed integration platform for scientific codes. However, for the sake of brevity, the large-scale PIV code with its GUI, named Fudaa-LSPIV, will be called just Fudaa from now on.

2.3 RIVeR

RIVeR (Rectification of Image Velocity Results) is a stand-alone application developed by A. Patalano and co-workers in the Center for Water Research and Technology (CETA)

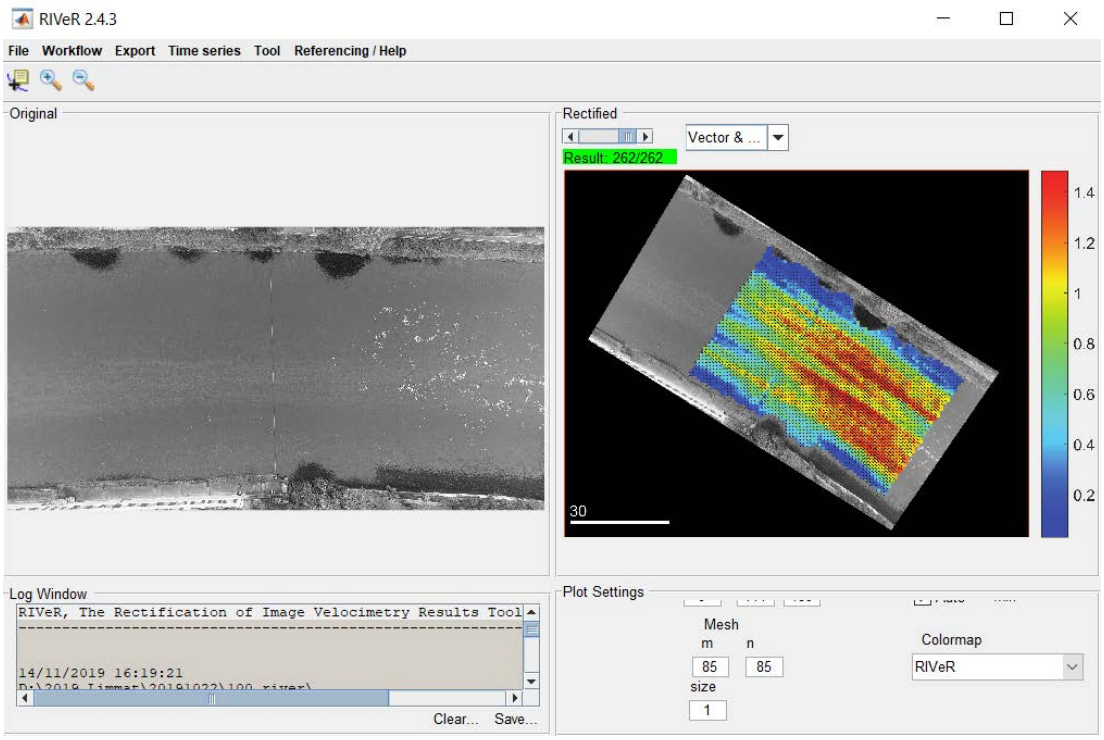


Figure 2.2: Screenshot of the typical RIVeR software window.

at the National University of Cordoba (Argentina) in 2013. The aim of RIVeR is to provide a new and fast approach in large scale water surface characterization by applying PIV first and then performing the rectification of the computed velocity vectors. It was initially designed for processing video (or images) taken from an oblique view (Patalano, García and Rodriguez 2017). Like BASESURV, RIVeR is based on PIVlab for the PIV part. Alternatively to PIV, individual trajectories of floating tracers can be computed by Particle Tracking Velocimetry (PTV). It is written in MATLAB® and released to be used free of charge with MATLAB® Runtime 8.5. At the time of this work, the latest version of RIVeR is 2.4.3, released on 13/11/2019 (Fig. 2.2). The first version of the User Manual was published on 12/11/2019 and helps the final user to learn how to use the software. However, it does not provide almost any technical detail about the algorithms implemented in the software.

The main difference of RIVeR compared to BASESURV and Fudaa is that it does not rectify all the images on which the PIV analysis is performed, but it computes the instantaneous surface velocity fields on the raw frames and then rectifies the velocity vectors afterwards. This approach can, in principle, be useful in reducing the computational time, especially in case the video stabilization is not necessary (e.g. when the camera is fixed). Another feature of RIVeR is the independence of the processing blocks, as these are developed in autonomous modules, launched from the main code.

RIVeR's workflow can be summarized as follow:

a) Video stabilization (optional)

b) PIV analysis with PIVlab 2.31, released on 4/10/2019 by William Thielicke. During this step, image pre-processing and some post-processing filters can be also applied.

c) Vector rectification

d) Discharge estimation in different cross sections (the module is not considered in this work).

Chapter 3

Experiments setup

3.1 River Limmat (CH)

3.1.1 Study area

During this work, two surveys were carried out on 10/10/2019 and 22/10/2019, on the river Limmat, in proximity to the Zürich Unterhard gauging station (Fig. 3.1). This is located at 400 m a.s.l and has a surface of the catchment area of 2714 km^2; the gauging station is managed by the Swiss Federal Office for the Environment (FOEN), which records water level data and estimates the discharge through the specific rating curve for river Limmat every 10 minutes. Furthermore, close to the gauging station, a rope is suspended across the river to control monitoring instruments such as a hydrometric impeller or an Acoustic Doppler Current Profiler (ADCP) along the section.

3.1.2 Datasets A and B

In order to test the performance of the three AIV approaches, two datasets under different conditions were acquired. The main characteristics of each one are summarized in Tab. 3.1. Dataset A was acquired on 22/10/2019 under regular and very stable flow conditions with a discharge comparable to the October daily average, computed during years 1938-2018 (Swiss Federal Office for the Environment FOEN 2020). To obtain a significant grayscale contrast on the frames, the flow was seeded with tracers from an upstream bridge: $\sim$ 2 m^3 of 100% biodegradable chips FILL-PAC Bio, produced by MEDEWO, were used (Fig. 3.2). These chips were made out of corn starch and therefore they were 100% biodegradable (certified according to EN 13432 as a proof of their compostability). Dataset B, on the other hand, was acquired on 10/10/2019 when the discharge was comparable to the 1-year return period flood event (ibid.). In this case, no seeding particles were used. Both the videos were acquired from a nadiral viewing angle (i.e. with the camera facing plumb vertical to the ground).

The equipment used for all the surveys consisted of a quad-copter UAV DJI Phantom 4 Pro+ (Fig. 3.3). This was equipped with an on-board camera with a $1''$ CMOS sensor, mounted on a three-axis gimbal. The focal length was 8.8 mm (24 mm in the 35 mm format equivalent) and Field of View (FOV) was 84° (DJI 2020b). The take-off weight of the UAV was 1.4 kg and the total cost of quadcopter and accessories was below 2,000 € (March 2018). Video A was acquired in the *h.265* codec with a frame rate of 29.97 Hz; B in the *h.264* codec at 23.97 Hz. Both were recorded with an image size of 4096×2160 px.

During the survey of 22/10/2019, additional measurements with a hydrometric impeller and RioPro 2440 ADCP were executed by FOEN. Information about the flow

Figure 3.1: Photo of the Limmat - Zürich Unterhard gauging station. On the image are marked: the flow direction (arrows); the area within which the surface velocity is computed through AIV (sold line); the rope used to control monitoring instruments such as the ACDP (dashed line).

Table 3.1: Summary of the main characteristics of the two datasets used in this study. The hydrological information about the river Limmat at the time of the surveys is taken from Swiss Federal Office for the Environment FOEN (2020).

Name	Date	Time	Discharge [m^3/s]	Water level [m a.s.l.]	View angle	Seeding
A	22/10/2019	11.09	83	400.2	Nadiral	Yes
B	10/10/2019	14.45	220	401.2	Nadiral	No

velocity near the water surface acquired by the hydrometric impeller and the measured discharge were used as reference data in the comparison against the surface velocity fields computed by AIV. The data acquired by the ACDP were not used in this study because they were more difficult to handle and the surface velocity profiles were similar to that measured near the water surface by the hydrometric impeller.

3.1.3 Spatial framework of the surveys and reference SfM models

For the surveys carried out on river Limmat, the Switzerland national reference system CH1903+/LV95 (Swisstopo 2020) was used as spatial framework to georeference all the data. The reference system was materialized with 29 Ground Control Points (GCPs) placed on natural elements along the riparian sides of the river (e.g. characteristic rocks and particular textured points on the concrete footpath). Their coordinates were acquired by Detert and co-workers on 10/10/2018 by using a GNSS Trimble R8 in RTK mode and the service *Swipos*, based on Automated GNSS Network of Switzerland (AGNES). From the video acquired during the survey with the same UAV DJI Phantom 4 Pro+ described above, 50 images were extracted. Together with 20 GCPs, those were used to produce a 3D model through SfM, named M1, by using the software Agisoft Metashape Professional (Fig. 3.4). The geometrical accuracy of M1 was estimated from

Figure 3.2: MEDEWO FILL-PAC Bio chips. The diameter of the chips were about 3 cm

Figure 3.3: The quadcopter DJI Phantom 4 Pro+ used for the two surveys on river Limmat.

Table 3.2: Geometrical accuracy of the 3D models M1 and M2, computed through SfM from the data acquired respectively on 10/10/2018 and 10/10/2019

SfM model	Acquisition date	E error [m]	N error [m]	h error [m]	RMS error [m]
M1	10/10/2018	0.06	0.05	0.1	0.12
M2	10/10/2019	0.06	0.07	0.1	0.17

the RMS of the reprojection error of 9 check points (i.e. the GCPs not used in the Bundle Block Adjustment), as listed in Tab. 3.2. Moreover, an orthophoto and a Digital Surface model (DSM), both with a Ground Sample Distance (GSD) of 0.05 m/px, were built from the 3D model M1. Those were used to georeference the videos acquired on 10/10/2019 and 22/10/2019. To this end, a set of artificial GCPs was each time extracted from the orthophoto and the DSM.

A second SfM model, called M2, was built by using the video frames of the dataset B and 16 artificial GCPs, obtained as described above. The geometrical accuracy of M2 is presented in Tab. 3.2. From M2, an additional orthophoto and a DSM, both with GSD of 0.05 m/px, were extracted and used to rectify video B only. This step was necessary due to the high water level occurred on 10/10/2019, when video B was acquired. The water, in fact, was covering the stones along the riparian sides to be used as double points and these were no longer visible in the images.

By considering the geometrical accuracy of the photogrammetric models and the GSD of the DSMs and orthophotos, it is possible to assess that the accuracy of the artificial GCPs collimated on those was in the order of magnitude of the decimetre.

3.2 River Lambro (IT)

In addition to that carried out on River Limmat, a further experiment aimed at testing an orthorectification approach based on a Structure from Motion (SfM) technique was set up on river Lambro in Italy. This procedure requires the reconstruction of a photogrammetric model of the study area first by using sequences of images acquired with a robust geometry of acquisition (i.e. long bases within the camera centres of projection, good overlap between the images etc.). The noise in proximity to the water surface must be cleaned and the model smoothed. The video frames to be used for velocimetry are oriented in the same reference system as the model and used to produce a sequence

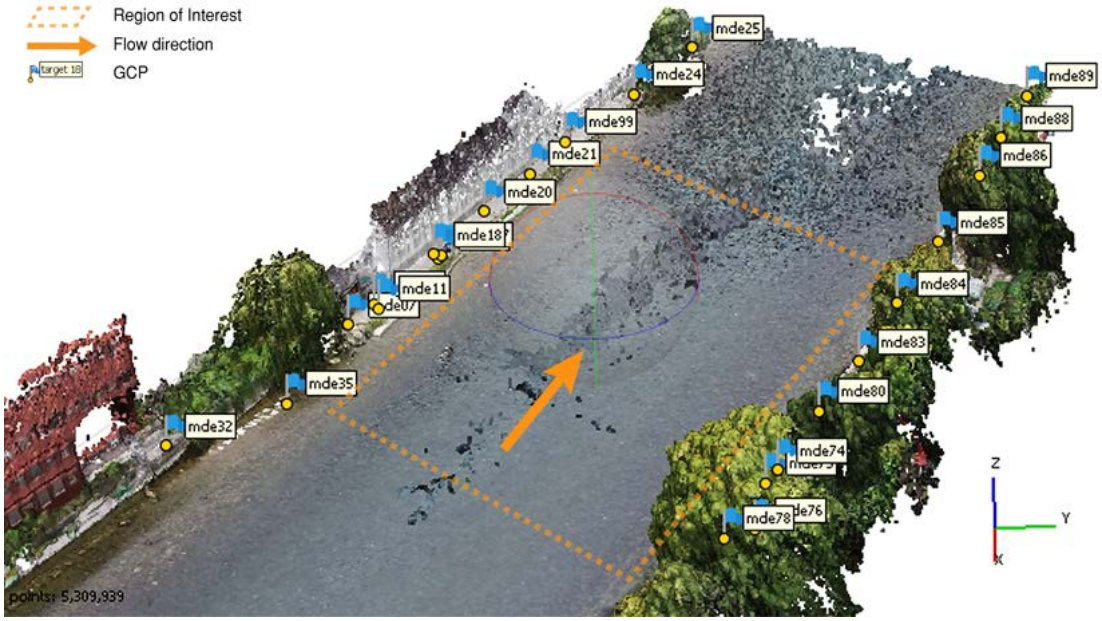


Figure 3.4: Screenshot of the SfM model M1 produced with Agisoft Metashape with the data acquired on 10/10/2018 by M. Detert and co-workers. The blue flags visible on the model are some of the markers used either as GCP or check points.

of orthophoto by solving the collinearity equations. Those can be further analysed by PIV.

3.2.1 Study area

The study area was in proximity to Diga delle Fornaci in Inverigo (Italy), displayed in Fig. 3.5. This is a small riverine dam built in 2018 to regulate the discharge of the river Lambro and reduce the flooding hazard in the Lambro hydrological basin. The dam is composed of two mobile gates: if the water level of the river increases too much due to heavy precipitations, the gates are partially closed and the area upstream of the dam is flooded. At the end of the flood event, the gates are gradually re-opened and the water is safely released. The gates are divided by a bridge pile, with its foundation inside the riverbed. The study area was chosen especially thanks to its easy accessibility.

3.2.2 AIV survey and dataset C

The survey on river Lambro was carried out on 20/02/2020. Due to a long draught period without any precipitation, the water level at the nearest gauging station Costamasnaga - Lamburgo (5 km upstream) was 0.39 m, which corresponds to a discharge of 1.2 m^3/s, according to the rating curve calibrated for river Lambro (ARPA Lombardia 2020). This is a relevant low flow condition for the river, as the water level occurred was extremely close the lowest limit of the region of validity of the rating curve (i.e. $0.38 < h$ [m] < 1.41, where h is the water level).

Despite the low flow conditions occurred during the survey, no tracers particles were used. This made it challenging to obtain surface velocities with PIV. As reported by Benetazzo, Gamba and Barbariol (2017), in fact, the water surface dynamics under low flow regime is strongly influenced by the effect of the capillary-gravity waves, propagating with their own speed and direction. Therefore, in order to obtain reliable surface velocity results, this effect should be corrected in order to discriminate the surface

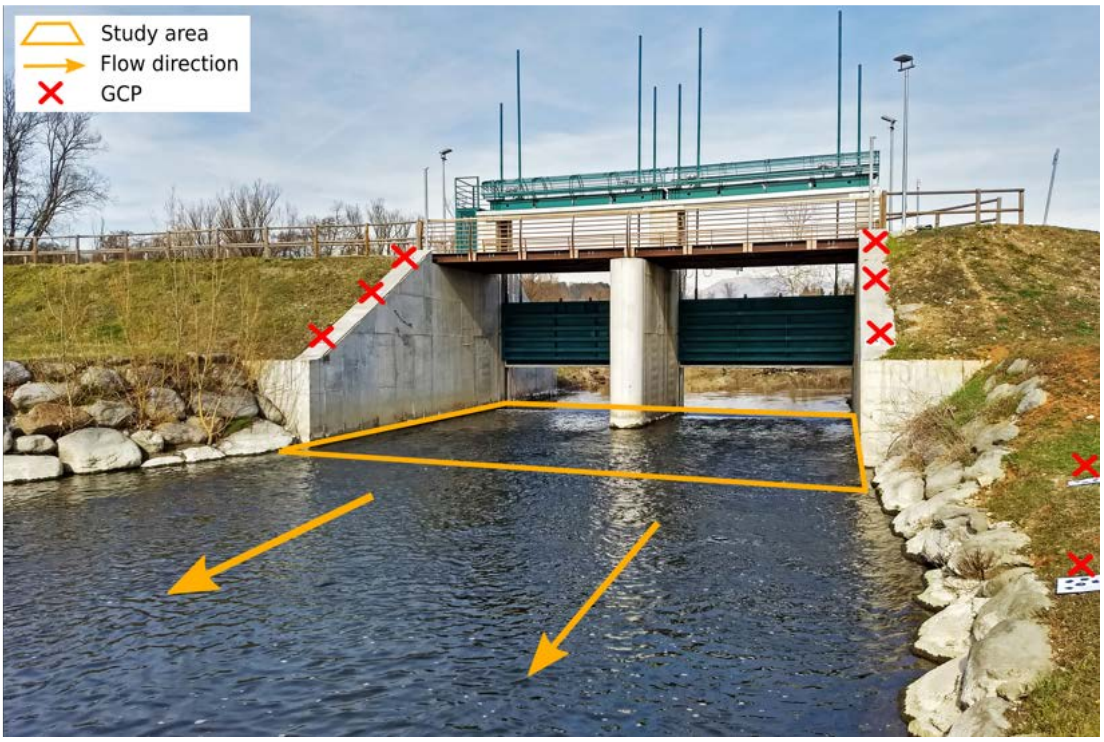


Figure 3.5: Photo of the study area on river Lambro in proximity to Diga delle Fornaci in Inverigo (IT). On the image are marked: the flow direction (arrows); the area within which the surface velocity is computed through AIV (sold line); 8 of the 16 GCPs used to georeference the survey (crosses).

structures advected by the current from the capillary-gravity waves. However, this was not done because the focus of the experiment was the SfM-based orthorectification and stabilization of the videos rather than velocimetry.

Several videos were acquired with different viewing angles by using small quadcopter UAV DJI Mavic 2 Pro, with a take-off weight of 907 g. That was equipped with an on-board camera with a $1''$ CMOS sensor, mounted on a three-axis gimbal. The focal length was 10.27 mm (28 mm in the 35 mm format equivalent) and Field of View (FOV) was 77° (DJI 2020a). All the videos were acquired in the *h.264* codec with a frame rate of 29.97 Hz and with an size of 3840×2160 px. Among them, only one video, named video C, was analysed. This was acquired at a viewing angle of $\sim$ 45° to the water surface plane.

3.2.3 Topographic survey and reference SfM model

For the topographic survey, 16 coded photogrammetric target were printed on A3 papers, glued on a rigid plastic support and fixed along the $\sim$ 30° concrete wall of the dam, both upstream and downstream, and on the flat riparian sides. These targets can be automatically detected on the images by Agisoft Metashape, matched with a numeric code and used as GCPs. The 3D coordinates of the targets were measured in a local reference system by using the MultiStation Leica MS60. The accuracy of the measured GCPs coordinates was in the order of magnitude of the millilitre. Ten of the 16 GCPs were used to orient a photogrammetric block composed of 46 nadiral and 10 oblique images. The remaining 6 targets were used as control points and their RMS reprojection error was 0.01 m. From the dense point cloud, a robust mesh model was

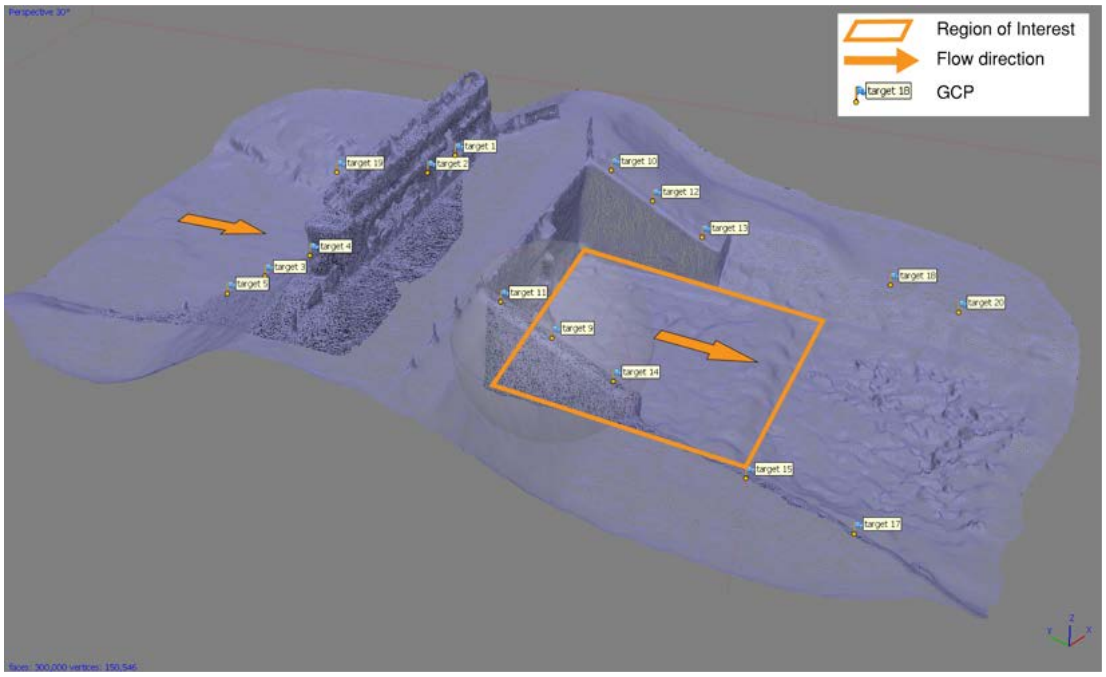


Figure 3.6: Mesh model built with Agisoft Metashape by using a combination of nadiral and oblique images. On the model are marked: the GCPs (flags with labels); the flow direction (arrows); the region of interest (solid line).

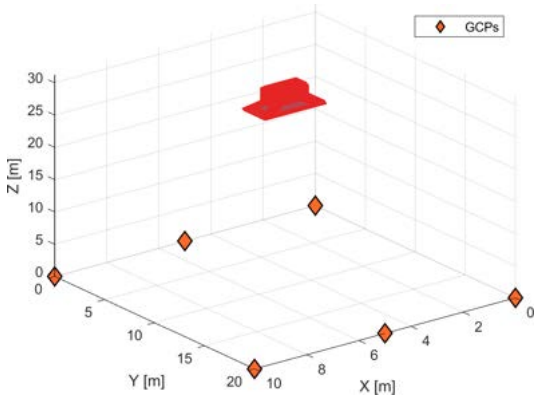
built (Fig. 3.6). This was essential in order to build a sequence of georeferenced orthophotos to be analysed with PIV and, at the same time, to correct the video shaking. From video C, in fact, it was not possible to build any photogrammetric model because the drone was hovering over the river and the basis between the projective centres of the video frames were negligible.

As visible in Fig. 3.6, some small details such as the railings above the bridge were not well reconstructed because not enough oblique images had been acquired during the survey. This may causes errors in the orthophoto in proximity to those. Nevertheless, the region of interest for AIV was the water surface plane that was well reconstructed.

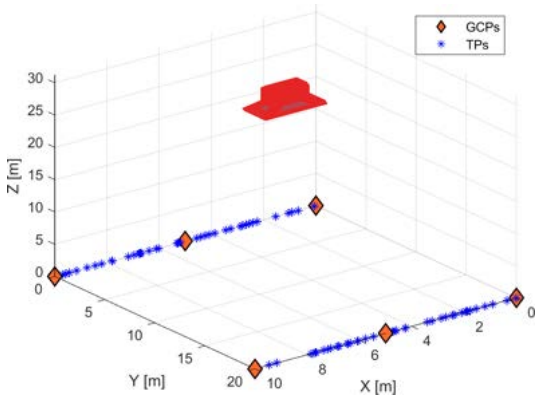
3.2.4 Simulations of the photogrammetric blocks

Before performing the experiments, simulations of the photogrammetric blocks were carried out by using CALGE, a scientific software developed by the Department of Civil and Environmental Engineering (DICA) of Politecnico di Milano (Forlani 1986). CALGE is able to adjust topographic networks and photogrammetric blocks, or combinations of those, by using the least-squares adjustment technique. The aim of the simulation was to test the feasibility of the SfM technique to estimate the External Orientation of the video frames to be used for AIV when the UAV is hovering and thus the basis between the projective centres are small. In the case of a photogrammetric block, the simulation requires as input the approximate EO of the cameras, i.e. their coordinates in the world reference system and the orientation angles. Moreover, the approximates coordinates in the world reference system of the Ground Control Points and the Tie Points (TPs, i.e. points visible in more than one image but with unknown coordinates in the world reference system) are required. In addition to that, Internal Orientation (IO) has to be provided. The output of the simulation is the covariance matrix of the collinearity equation parameters (usually called C_{xx}), which only depends

(a) Simulation RS



(b) S1



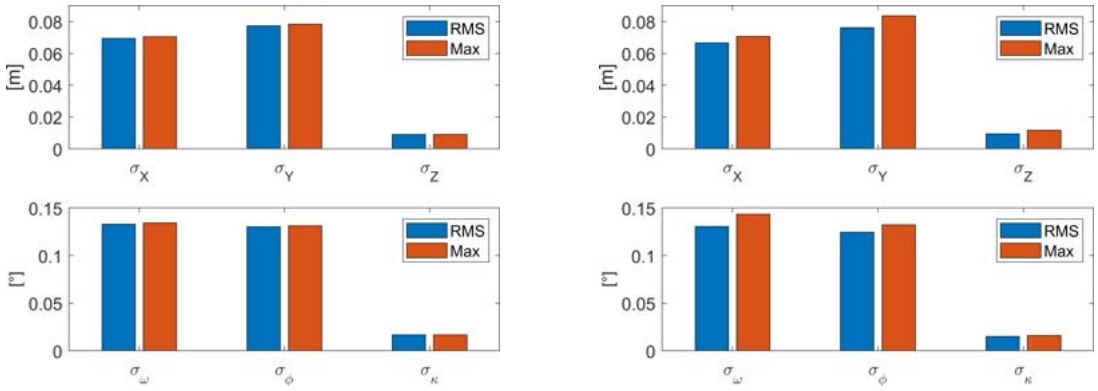
(c) S2

Figure 3.7: Geometry of the simulations: the world reference system used is parallel to the GCPs plane (a); acquisition geometry of simulation S1 with only GCPs (b); acquisition geometry of simulation S2 with GCPs and TPs (b).

on the design matrix, but not on the observations (i.e. the image coordinates of GCPs and TPs). Therefore, those are unnecessary to perform the simulation and estimate the variances of the cameras projection centres.

Two simulations, named S1 and S2, with a simplified geometry were carried out: in both of them the world reference system was define parallel to the GCPs plane, as in Fig. 3.7a. The cameras were assumed to be in the same position and to be nadirally looking, i.e. the UAV was hovering and the image plane of the camera was parallel to the GCPs plane. In order to take into account the unavoidable shaking of the drone, the approximate EO parameters of the cameras were perturbed with random noise within $\pm 0.2\,\mathrm{m}$ for the location and within $\pm 0.3°$ for the orientation angles ω, ϕ and κ. In S1, only 6 GCPs, distributed on the lateral sides of a rectangular area of interest, were assumed to be visible in 15 images (Fig. 3.7b). No TPs were used for the orientation of the block. In S2, the same 6 GCPs as before were assumed to be collimated in only two images. However, 77 TPs were placed the images (Fig. 3.7c). The two simulations reflect realistic scenarios concerning SfM-based orthorectification procedures. S1 assumes that all the 6 GCPs can be detected automatically by Metashape in all the video frames (i.e. hundreds or even thousands of images) by means of automatic recognition of the coded targets. S2, on the other hand, represents a situation in which the 6 GCPs are manually collimated in only a small subset of the images (e.g. tens of images) and the orientation of all the other video frames is performed on the basis of TPs only, automatically detected and matched on the images by using interest operators.

The results of the simulations are presented in terms of standard deviations of the camera EO parameters. These were computed as the square roots of the main diagonal



(a) S1 (b) S2

Figure 3.8: Results of the simulations: (a) S1; (b) S2. For each EO parameter, the RMS and the maximum value of the simulated standard deviations of the 15 cameras are plotted.

elements of the parameter covariance matrix in the least-square adjustment (usually called C_{xx}). In Fig. 3.8, the Root Mean Square (RMS) and the maximum values of the simulated standard deviations of the 15 cameras are plotted, grouped by the EO parameter. Both the simulations were successful and the results are similar. In S1, the RMS and the maximum values are almost identical, denoting that all the cameras were oriented with a similar accuracy. In S2, the slightly higher value of the maximum compared to the RMS is due to the 13 cameras oriented only with TPs. Nevertheless, the differences seem not to be significant, suggesting that both the approaches may be suitable for the orthorectification.

Chapter 4

Processing of videos A and B

In the following chapter, the processing of videos A and B, acquired on river Limmat, is described according to the AIV steps presented in Sec. 1.2. The phases of video stabilization and frame rectification are discussed together because they are performed at the same time in BASESURV.

4.1 Video frames extraction and undistortion

For the computation of the surface flow velocity, 35 s and 30 s were analysed respectively for video A and B. From video A (with a frame rate of 29.970 Hz) 1048 frames were extracted; from B, 718 (23.976 Hz).

With BASESURV and RIVeR the frames were corrected in order to reduce the radial and tangential distortion of the camera according to Brown's model (Brown 1971). This is performed on the base of the intrinsic camera parameters estimated with the MATLAB® Camera Calibrator app, available in the Computer Vision Toolbox™. Fudaa, on the other hand, does not provide the possibility to calibrate the camera and correct the distortions in the video frames.

4.2 Video stabilization and rectification

This step is crucial in AIV. It mainly consist in correcting the apparent movement of the images due to the shaking of the UAV (video stabilization) and estimating a geometrical transformation that links the 3D real world reference system with 2D image reference system (rectification).

In BASESURV these two steps are performed simultaneously, by estimating a homography transformation between each video frame and an orthophoto of the same area. The homography is a model for the central projection between two planes: the water surface and the image plane. Therefore, the double points used for the estimation of the transformation parameters must lay on the same plane as the water surface. In order to easily rectify the two videos, a 2-step approach was followed. Initially, the first frames of video A and B were manually rectified respectively on the orthophotos obtained from M1 and M2, by collimating manually a set of double points along the riparian sides, at water surface level. The RMS reprojection errors of the double points are summarized in Tab. 4.1: the overall geometrical error of the rectified first frame (Reference Frame, RF) can be considered again in the order of magnitude of the decimetre. The orthorectification of the videos is then performed estimating a different homography transformation between each Video Frame (VF) and the RF. In order to estimate the transformation, interesting points in each VF are searched with the KAZE

Table 4.1: Rectification error of the first frames of video A and B computed from the re-projection error of the double points. BASESURV rectification is based on homography transformation, Fudaa on the full DLT.

Dataset	Software	RMSE E [m]	RMSE N [m]	Global RMSE [m]
A	BASESURV	0.11	0.16	0.14
	Fudaa	0.05	0.03	0.05
B	BASESURV	0.17	0.19	0.18
	Fudaa	0.04	0.05	0.05

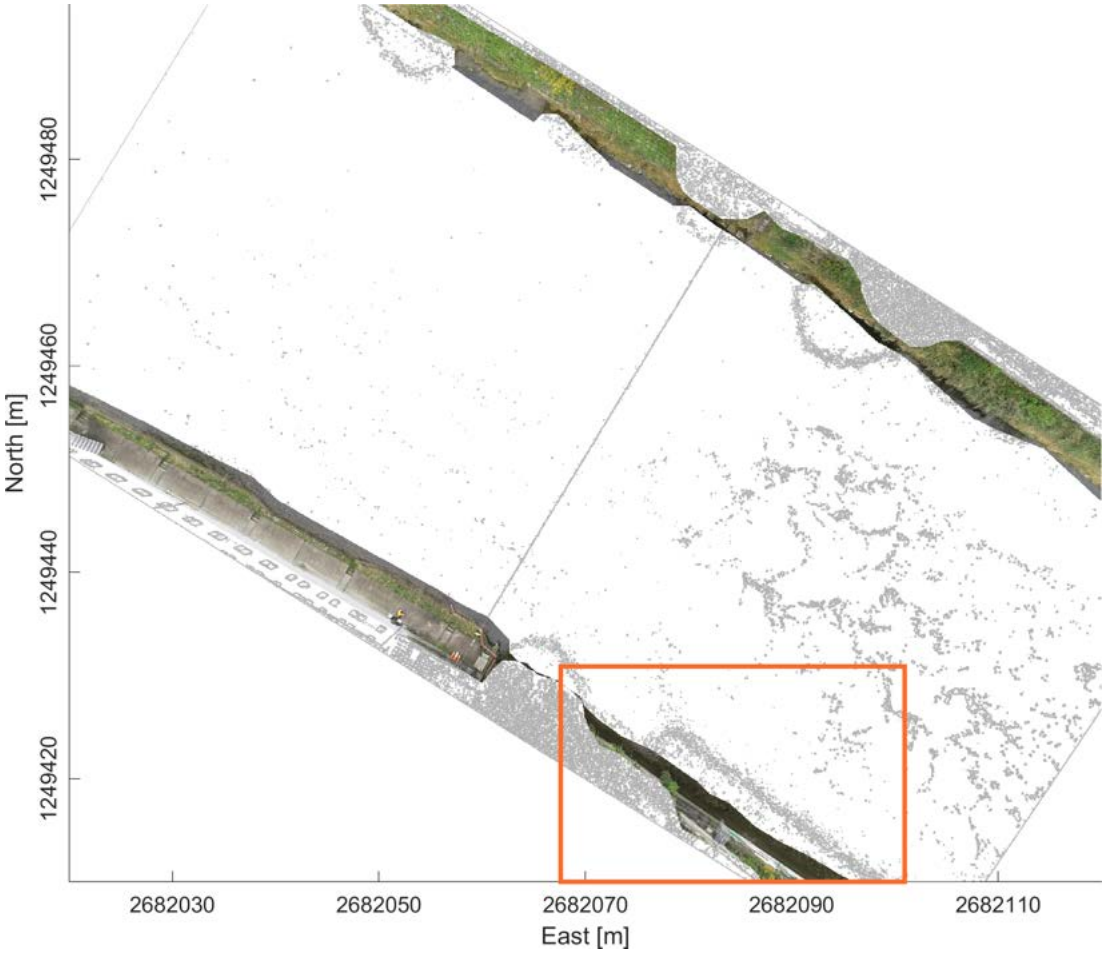


Figure 4.1: The binary mask applied on the rectified frame of the video A in the BASESURV rectification tool: only the RGB area corresponds to the visible part of the binary mask and therefore has been used in interesting points research. The south-west area in which there are no points at the water surface level along the riparian side is marked in the box.

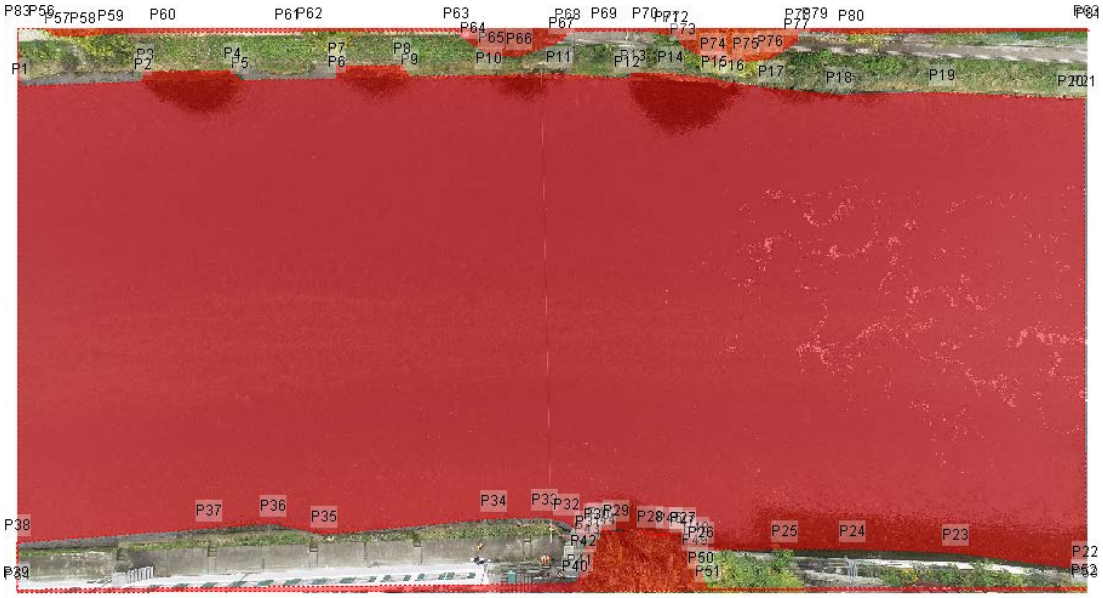
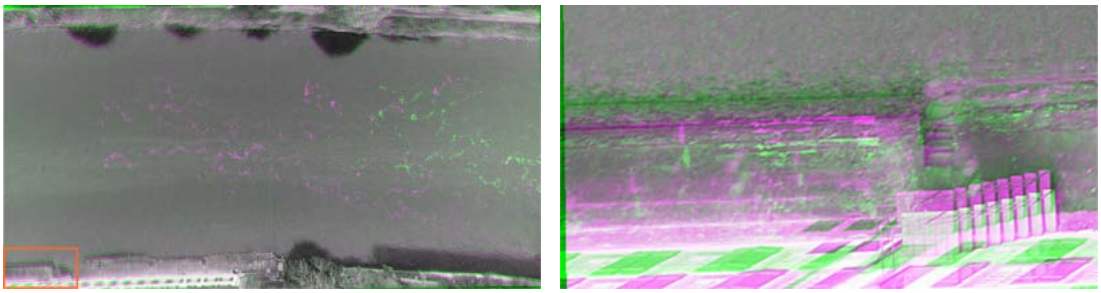


Figure 4.2: Mask applied in Fudaa on the first frame of video A. The red area, marked by the software as *Flow area*, is the one excluded by the mask.

operator (Alcantarilla, Bartoli and Davison 2012) and matched upon their descriptor with the interesting points in the RF. In order to exclude matching points on the moving areas (e.g. flow area and shadows) and objects not belonging to the water surface (e.g. trees and buildings), a binary mask is defined on the RF (Fig. 4.1). Interesting points found in the VF are rejected if they fall outside the mask in the RF, after computing the inverse transformation. Due to the presence of very small flat riparian sides with the same height as the water surface (especially in the south-west area, highlighted in Fig. 4.1), a small portion of the buildings is included in the mask to find enough matching points. This approach clearly leads to some errors in the estimation of the homography parameters and it should be avoided, if possible. However, it was necessary in order to successfully rectify the videos. Moreover, BASESURV includes a tool that smooths the coordinates time-series of the 4 corners of the rectified frames with a robust LOESS method (Cleveland 1979). For each VF, a new and final homography transformation is estimated based on smoothed coordinates of the 4 corners. This technique allows artificial errors in the rectification (e.g. due to a little number of matching points found in particular areas of the image) to be reduced.

With Fudaa the stabilization and rectification are executed separately. For the stabilization, a mask over the *flow area* is defined in the first frame (Fig. 4.2): the area within the mask is not considered in the interesting point research. The mask is then applied at the same way on all the video frames. This approach clearly leads to an issue: the mask may no longer be correct for the other frames if the camera is shaking significantly. The interesting points are detected and matched with the SURF operator (Bay et al. 2008) and they are used to estimate a projective transformation (i.e. a homography) between each video frame with the first one (personal email correspondence with M. Jodeau and J. Le Coz of 20/12/2019). For the image rectification in Fudaa, only one transformation is estimated and applied to all the frames. This implies that the video has been properly stabilized beforehand. To this end, a set of GCPs (even not laying on the same plane as the water surface) must be manually collimated on the first stabilized frame and their 3D world coordinates have to be provided. In this work, those were extracted from the orthophotos and DSMs obtained from M1 and M2. The



(a) (b)

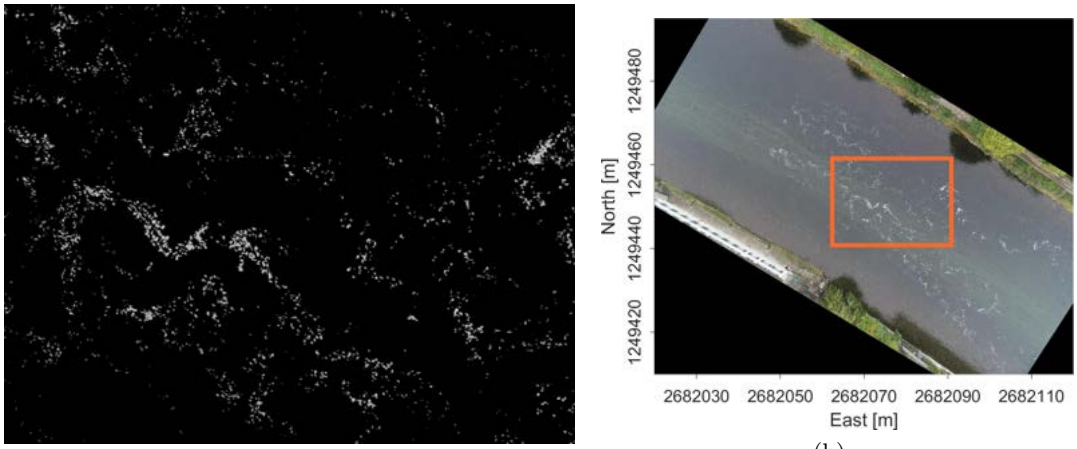
Figure 4.3: Drift error in the stabilization of video A performed with RIVeR: (a) the first and last stabilized frames are plotted in false colours, respectively in magenta and green, to enhance their differences; (b) cut-out at the location marked in (a) in which the shift between the frames is evident.

Table 4.2: Summary of the main differences in stabilization and rectification for the three AIV approaches. IP stands for Interesting Points; VF for the generic Video Frame; RF (Reference Frame) for the first, manually rectified, frames.

	Stabilization		**Rectification**	
	Transformation	IP operator	Object	Transformation
BSV	Homography (each VF with the RF)	KAZE	Images	Same as stab.
Fudaa	Homography (each VF with the first one)	SURF	Images	DLT
RIVeR	Affine (pairs of consecutive VF)	FAST	Vectors	Homography

coordinates of the GCPs are then used to estimate the parameters of a full 3D Direct Linear Transformation (DLT), that is employed for the rectification of all the frames. The reprojection errors of the GCPs are summarized in Tab. 4.1. The implementation of the full 3D DLT equations instead of a homography may be the reason for which the Fudaa RMSE values are lower than those of BASESRUV.

In RIVeR the stabilization and rectification are carried out separately. For the former, every video frame is linked to the previous one by an affine transformation, estimated on the basis of the matching points detected by the FAST operator (Rosten and Drummond 2005) (personal email correspondence with A. Patalano of 20/12/2019). This approach is not optimal because the stabilization is affected by a drift error, that is cumulating over the video (Fig. 4.3). In order to exclude the moving areas from the matching point research, a mask over the flow area has to be defined, but no other options are available in the software. Furthermore, as described in Sec. 2.3, RIVeR does not rectify all the images, but only the velocity vectors computed by PIV over the stabilized video frames. In a similar way as the frame rectification in Fudaa, vector rectification in RIVeR is performed by solving full DLT equations based on a set of GCPs, extracted from the orthophotos and DSMs obtained from M1 and M2. However, RIVeR, in contrast to BASESURV and Fudaa, does not have any tool for evaluating



(a) (b)

Figure 4.4: Example of image pre-processing: (a) Cut-out at the location marked in (b) of a video frame after the background subtraction. The white pixels are the moving objects (i.e. seeding particles) which are clearly distinguishable from the nearly-black background.

the quality of the estimated transformation. Moreover, during the tests performed, RIVeR revealed some issues using GCPs with world coordinates used (i.e. East and North in $CH1903 + /LV95$ RS). A translation of the reference system was therefore applied in order to have smaller numbers in the coordinates.

In Tab. 4.2, a summary of the differences in the stabilization and rectification techniques of the three AIV approaches is provided.

4.3 Image pre-processing

Pre-processing of the video frames is a common approach aimed at enhancing the images before PIV analysis and improving the result of the correlation. As they are both based on PIVlab, BASESURV and RIVeR adopt almost the same tools; however, this processing step is completely absent in Fudaa.

The pre-processing techniques implemented in PIVlab are detailed described in Thielicke and Stamhuis (2014). Those are the Contrast Limited Adaptive Histogram Equalization filter (CLAHE) based on Pizer et al. (1987), the intensity high-pass filter and the intensity capping method from Shavit, Lowe and Steinbuck (2007). In addition to these filters, the background subtraction can also be performed. This consists in subtracting the average image, computed as the mean of a defined number of frames, to the whole video in such a way as to enhance the moving particles only (Fig. 4.4). Instead of the intensity capping as before, BASESURV implements a different *noisy pixel intensity capping* approach: it simply consists in setting to zero all the pixels with intensity below a certain threshold decided by the user.

For both dataset A and B, only the background subtraction was used in BASESURV and RIVeR, by computing the average image from the whole stabilized video. Moreover, the noisy pixel intensity capping was applied in BASESURV: for dataset A a threshold of 10% of the highest radiometric value was chosen (i.e. pixels with greyscale intensity lower than 25.6, which are most likely not due to the seeding particles, are set to black). For video B, when no seeding were used and thus no clear signal was present, a lower threshold of 1% was applied.

4.4 PIV

Flow velocity fields were computed over a regular grid of points by PIV. For each point, an Interrogation Area (IA) on the first video frame was searched in the following one by means of a statistical correlation algorithm. Since the three AIV approaches implement the PIV technique in different ways, the workflow followed with each software package is separately described. As it is a crucial parameter in PIV, a focus on the time lag between each pair of video frames is further given.

4.4.1 BASESURV

In BASESURV, PIV was performed on a sequence of orthophotos previously rectified and merged together in a video file with the same frame rate as the raw video. For each image pair, i_1 and i_2, the cross-correlation analysis was carried out in the frequency domain, by means of Fast Fourier Transform (FFT). The discrete Fourier spectrum of the IA on i_1 was multiplied by that of the corresponding IA on i_2 in order to obtain the most probable displacement vector (Fig.4.5). A three steps multi-pass approach (see Sec. 1.1) was performed. Hence, cross-correlation was applied three times on the same area and the integer displacement estimated in the first pass was used to offset the IA in the following ones. At each pass, the size of the IA was gradually reduced by factor 2 in order to achieve a finer spatial resolution, but without losing the ability to resolve the higher velocities. Provided that, for dataset A an initial IA of 128×128 px was employed, which resulted in a final IA of 32×32 px, with a footprint on ground of 1.6×1.6 m^2. For dataset B, on the other hand, a larger IA with a final size of 64×64 px was chosen because of the lack of clear patterns on the images. Finally, the integer displacement obtained by correlation was refined with a 3 point Gaussian sub-pixel estimation technique (see Sec. 1.1).

Starting from the version 2.30 (30/09/2019), William Thielicke has implemented in PIVlab a new correlation algorithm called Multipass Window Deformation Ensemble Correlation, which may be useful to analyse steady flows with low seeding density, when a large number of images are recorded. However, at the time of this study, BASESURV was implementing an older version of PIVlab, in which the Ensemble Correlation technique was not yet available, and thus FFT correlation was used.

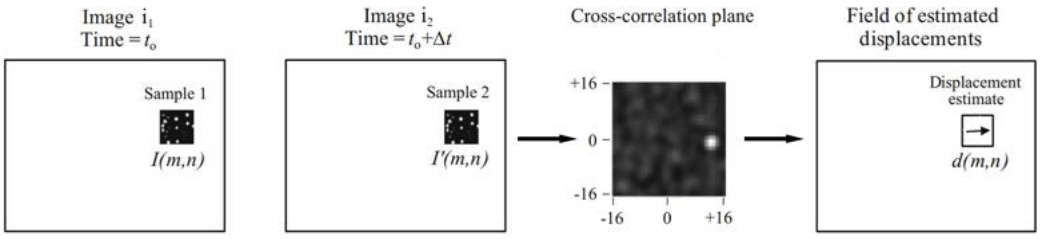


Figure 4.5: Simplified scheme of the cross-correlation algorithm implemented in BASESURV. For each grid point, the cross-correlation function between an IA of e.g. 32×32 px in image i_1 and the corresponding IA in i_2 is computed in the frequency domain. The most probable displacement vector is given by the location of the correlation peak. Images taken and adapted from Raffel, Willert, Scarano et al. (2018).

4.4.2 Fudaa-LSPIV

In Fudaa, the cross-correlation analysis was performed in the spatial domain. For every grid point a_{ij} in the first image, the correlation coefficient between the IA centred on

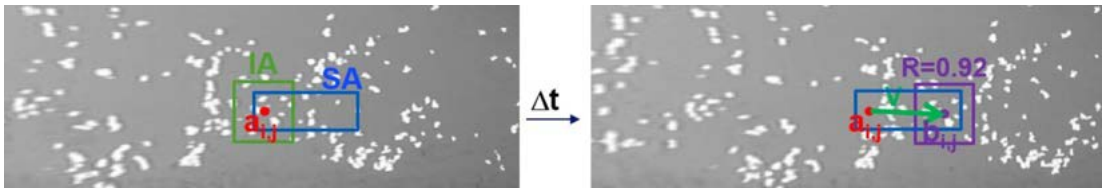


Figure 4.6: Scheme of the cross-correlation algorithm implemented in Fudaa-LSPIV for a sample grid point a_{ij}. The IA (green square) centred in a_{ij} on the first image is searched on the second image within the Search Area (blue rectangle), by maximizing the correlation coefficient $R(a_{ij}, b_{ij})$ (Jodeau et al. 2019).

it and the same IA centred on point b_{ij} in the second image was computed. The most probable displacement vector was given by the maximum of the correlation function $R(a_{ij}, b_{ij})$ (Fig. 4.6). The calculation was performed only for points b_{ij} included in a search area defined by the user to save computational time (Jodeau et al. 2019). Since Fudaa did not implement any iterative multi-pass approach, IAs comparable to the final ones of BASESURV were used: 32×32 px for dataset A; 64×64 px for dataset B. Similarly to BASESURV, the sub-pixel estimation was carried out by fitting two unidimensional Gaussian distributions on the correlation matrix. In order to have consistent results between Fudaa and BASESURV, the same grid of points were used in both the approaches. The grid was first created in BASESURV, by defining the region of interest and the size of the IA. Then, after a transformation of reference system, it was imported in Fudaa.

4.4.3 RIVeR

Since RIVeR is based on PIVlab as well as BASESURV, the algorithms implemented in those AIV software packages are similar. In RIVeR, however, PIV is performed on a sequence of non-orthorectified images. For each image pair, cross-correlation was computed in the frequency domain with FFT, by using 3 passes and a 3-point Gaussian sub-pixel estimation technique. For both dataset A and B, the same size of IAs as in BASESURV were employed. Nevertheless, the grid used in RIVeR was different from those used in BASESURV and Fudaa because it was defined on the non-orthorectified frames and then transformed afterwards.

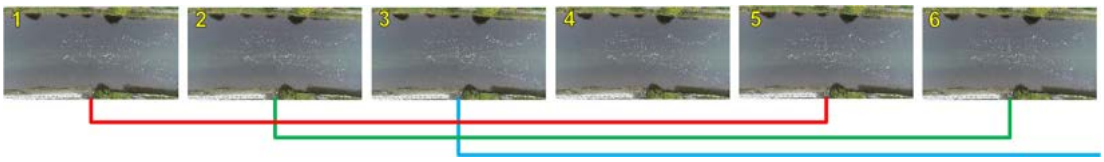
4.4.4 Time Lag

The time lag between every pair of images i_1 and i_2 is a central issue in PIV. A too short time lag, in fact, leads to a low signal-to-noise ratio in the estimated velocity vectors because the particle displacements from i_1 to i_2 may not be distinguishable from the background noise. On the other hand, if a too large time lag is chosen, the patterns (seeding particles or surface water structures) may have exceedingly changed from i_1 to i_2 and they are no longer matchable by correlation.

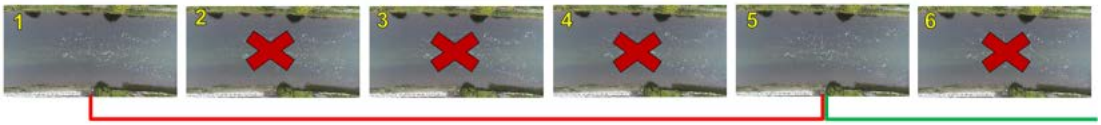
In BASESURV, the time lag can be chosen arbitrarily during the PIV phase of the processing workflow, by defining a sampling interval in terms of number of video frames (n_f). Given, for instance, an interval $n_f = 4$, the frames are processed as follows:

$$[(1, 5), (2, 6), (3, 7), ..., (N - 4, N)]$$

where N is the total number of frames in the video (Fig. 4.7a). This approach allows the whole quantity of data (i.e. video frames) to be used, having at the same time a time lag large enough to reduce the errors due to small particle displacements. In both



(a) BASESURV



(b) Fudaa and RIVeR

Figure 4.7: Scheme of PIV processing sequence in (a) BASESURV and (b) Fudaa and RIVeR if, for instance, an interval $n_f = 4$ frames is used.

Fudaa and RIVeR, if a larger time lag than the video frame rate is needed, a selection of frames must be performed beforehand during the frame extraction from the raw video. Assuming again to use $n_f = 4$, only 1 frame in every 4 is extracted from the video and thus the PIV processing sequence is

$$[(1, 5), (5, 9), (9, 13), ..., (N - 4, N)]$$

In this way, however, just a small subset of the available information is actually used (Fig. 4.7b). Moreover, if a different time lag has to be tested, all the processing workflow (including the stabilization and the rectification) must be performed again from scratch.

For video A, $n_f = 4$, corresponding to $\Delta t = 0.133\,\mathrm{s}$, was chosen: this was a fair compromise between a sufficient time lag and a large enough subsample of frames to be used for PIV in Fudaa and RIVeR (262 frames). For video B, on the other hand, it was necessary to use a frame interval $n_f = 1$ because of some problems in the video stabilization in Fudaa using larger intervals. During several tests, in fact, the stabilization process was collapsing and it was not possible to proceed in the further steps of the AIV workflow and to get a final result with Fudaa. Only one test, in which a frame interval $n_f = 1$ was used, was successful. However, a frame interval $n_f = 1$ is not an optimal choice from the PIV point of view because it leads to have a time lag of $\Delta t = 0.042\,\mathrm{s}$ which is too short for the expected velocity range in river Limmat, i.e. from $\sim$0.1 m/s up to $\sim$3 m/s. In fact, considering GSD of the images equal to 0.05 m/px and $\Delta t = 0.042\,\mathrm{s}$, the dynamic range of velocities $[U_{min}, U_{max}]$ resolvable by PIV can be estimated, according to Scarano and Riethmuller (1999), as:

$$U_{min} = \frac{l_{min}}{\Delta t} = \frac{1\,\mathrm{px} \times 0.05\,\mathrm{px\,m^{-1}}}{0.042\,\mathrm{s}} \simeq 1.2\,\mathrm{m/s}$$

$$U_{max} = \frac{c_1 W_s}{\Delta t} = \frac{0.2\ \times 64\,\mathrm{px} \times 0.05\,\mathrm{px\,m^{-1}}}{0.042\,\mathrm{s}} \simeq 15.2\,\mathrm{m/s}$$

where l_{min} [m] is the minimum displacement that can be resolved, W_s [m] is the linear size of the IA. Moreover, c_1 [/] is the ratio between l_{min} and W_s, which should not exceed values of $0.2 - 0.3$. If a 3 point Gaussian sub-pixel estimation is applied to refine the estimated displacement, l_{min} may be assumed to be in the order of 0.1 px (ibid.). Nevertheless, it is clear that the time interval $\Delta t = 0.042\,\mathrm{s}$ is not suitable for the range of velocities we are interested in and a larger one should be used. In order to have an even comparison between the three AIV approaches, PIV was performed using pair of consecutive video frames (i.e. $n_f = 1$) in all of them. However, a second test using

Table 4.3: Main PIV parameters used in BASESURV, Fudaa and RIVeR respectively for dataset A and B. Column n_f refers to interval in terms of number of frames within every pair of images used for PIV and Δt is the relative time lag. For video B, two PIV analysis have been performed with BASESURV using different time lag.

Dataset	Software	n_f [-]	Δt [s]	Frames [-]	initial IA [px]	Passes [-]	Final grid [pts^2]
A	BASESURV	4	0.133	1048	128×128	3	83×95
	Fudaa	4	0.133	262	32×32	1	83×95
	RIVeR	4	0.133	262	128×128	3	124×165
B	BASESURV	1	0.042	718	256×256	3	61×61
	Fudaa	1	0.042	718	64×64	1	61×61
	RIVeR	1	0.042	718	256×256	3	58×99
	BASESURV	12	0.501	718	256×256	3	61×61

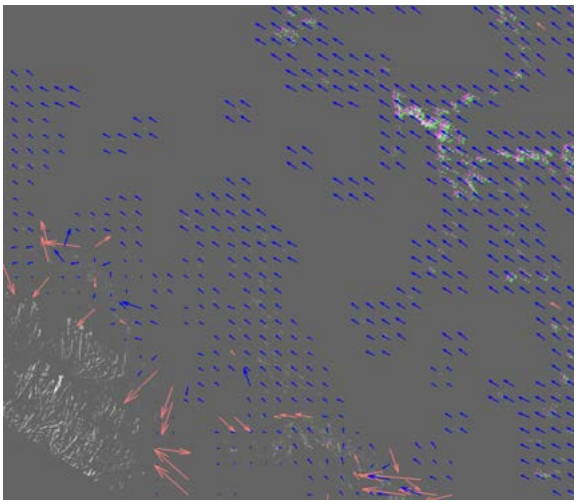
BASESURV and a larger time lag of $\Delta t = 0.501\,\text{s}$ (equal to a frame interval $n_f = 12$) was also performed in order to have a more robust PIV analysis of video B.

All the main PIV parameters used in processing the two datasets A and B are summarized in Tab. 4.3.

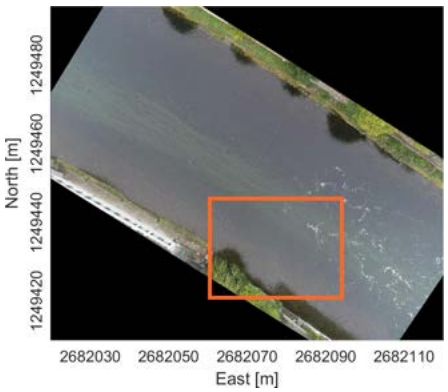
4.5 Filtering of spurious vectors

The identification and removal of the outliers in the velocity fields is a crucial step in AIV because of the usually large number of spurious vectors which can be erroneously detected in real data (e.g. due to errors in the correlation analysis, residual rectification errors, changes in the illumination condition and presence of shadows, apparent or real movement of objects not belonging to the water flow etc.). In Fig. 4.8, a cut-out of an instantaneous velocity field computed with BASESURV is presented as an example. It is possible to notice that some of the vectors are not consistent with the river flow. These must be detected and removed by the post-processing filters.

Although all of the three AIV approaches allow the spurious vectors in each instantaneous velocity field to be filtered, they implement different post-processing filtering tools. PIVlab itself (and therefore also BASESURV and RIVeR) includes filters based on lower and upper thresholds for each velocity component (Thielicke and Stamhuis 2014) and on the normalized median test (Westerweel and Scarano 2005). In addition to the filter based on thresholds in the velocity components, in Fudaa it is also possible to exclude the vectors with a low correlation index computed by PIV (Jodeau et al. 2019). All those filters are based on the spatial distribution and characteristics of instantaneous velocity vectors. However, they do not consider the temporal dimension of the process because they are applied for each time step independently. Only BASESURV implements an effective filtering tool based on a dispersion analysis of the time-series of the velocity components at each point of the grid. This is composed of a 2D multi Gaussian filter and a smoothing one. The former assumes that the variation over time of the two velocity components u and v at each point can be described with a bivariate Gaussian distribution. Therefore, in order to separate the inlier vectors from the outliers, two 2D multivariate Gaussian models (McLachlan and Peel 2000) are fitted on the sample built by considering the whole time-series of velocity vectors at each grid point and the surrounding eights together. Points belonging to the more sparse Gaussian model are considered as spurious vectors and rejected. The latter filter



(a)



(b)

Figure 4.8: (a) Cut-out of instantaneous velocity field computed by BASESURV between frames # 151 and #155 for dataset A at the location marked in (b). The light red vectors are the ones rejected by PIVlab instantaneous vector filters.

smooths the time series of the accepted velocity vectors during the previous step by applying a robust LOESS smoothing (Cleveland 1979).

The parameters of the post-processing filters of each code were tuned empirically with a trial and error approach. From a general point of view, more stringent filters were applied to dataset A in order to keep only the vectors produced by the seeding particles; for B, more spurious vectors have been accepted in order not to remove useful signals at the same time.

4.6 Computation of the time-averaged velocity fields

The instantaneous velocity fields are typically used to compute a time-averaged velocity field. An example computed from 35 s of video is shown in Fig. 4.9. For each point on the grid, a unique vector (in the two Cartesian components u and v) can be computed from the relative time series using either the mean or the median operator. The latter is known to be more robust against the outliers than the former. BASESURV employs the median operator, whilst both Fudaa and RIVeR use the mean. Despite that, Fudaa allows all the computed instantaneous velocity fields to be exported. Therefore, also the median of the time series at each point of the grid was computed in MATLAB® and compared against the average. On the other hand, with RIVeR it is possible to export just one instantaneous velocity field at the time, but not all together. This makes difficult to easily compute the median velocity field using an external software.

4.7 Velocity time series and cross-sections

In order to analyse the results obtained, the time-series of the two velocity vector components at certain grid points were extracted. The location of those are displayed in Fig. 4.10. In BASESURV and in Fudaa, the instantaneous surface velocity fields can be exported and managed completely by using MATLAB®, for instance. This made it possible to select particular grid points (based on their world coordinates) and extract the time-series at the same location for both the AIV approaches. With

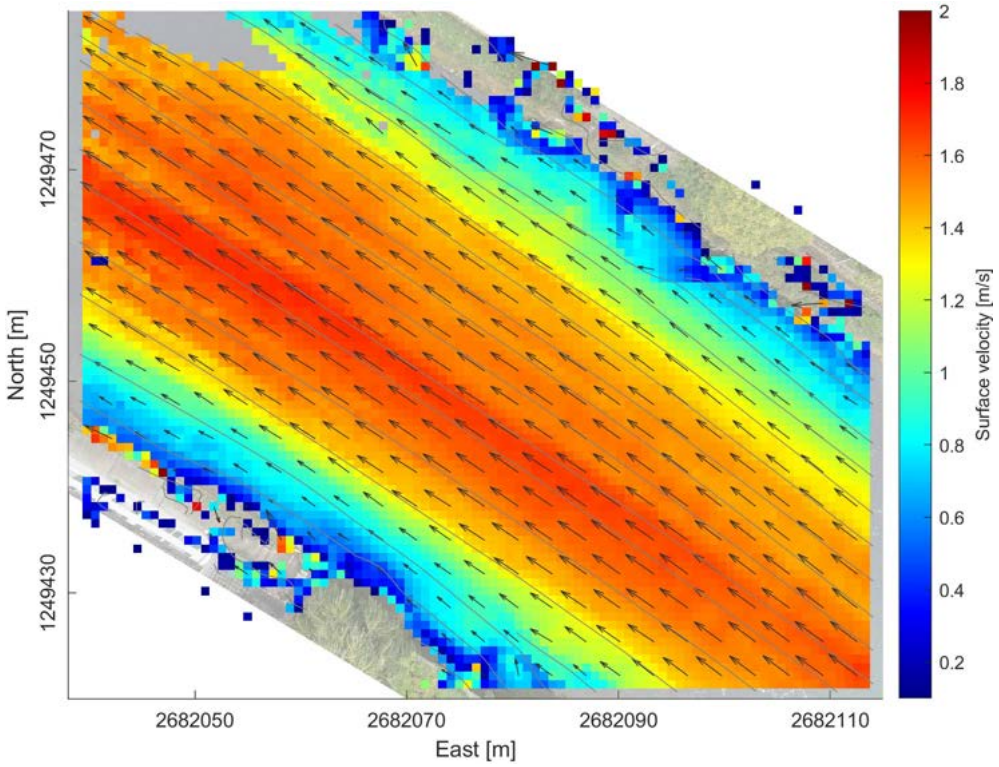
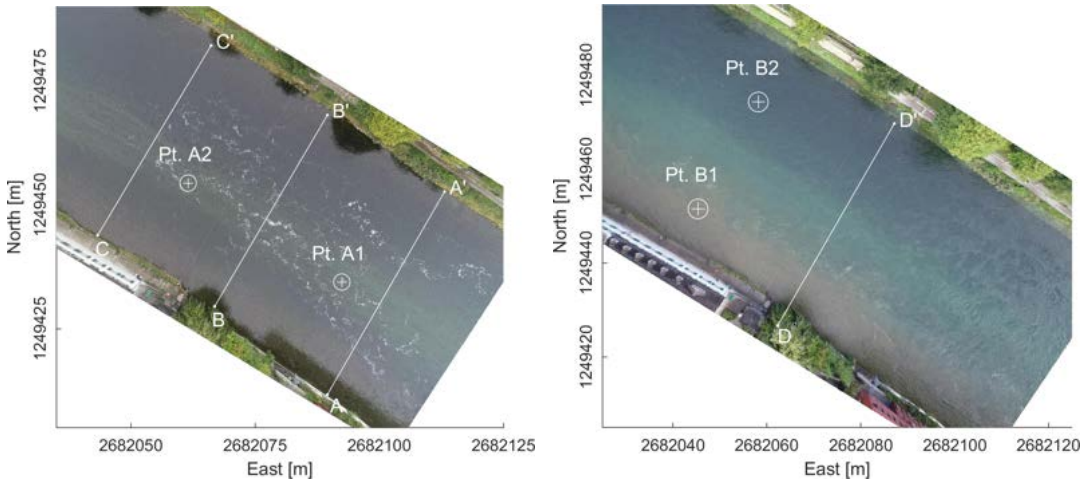


Figure 4.9: Example of time-averaged surface velocity field over a window of 35 s. The median velocity vector computed on each grid point is superimposed to its magnitude, represented with a scale of colours.



(a) Dataset A (b) Dataset B

Figure 4.10: Location of the cross-sections and the grid points at which the vector time-series is extracted, respectively for: (a) Dataset A; (b) Dataset B.

RIVeR, however, the time-series were computed inside the software itself, because of the impossibility to easily manage all the instantaneous results in MATLAB®. This was done by collimating visually grid points as much close as possible to those used for the time-series of BASESURV and Fudaa. Clearly, that was not a rigorous approach but it was the only possibility to compare the results.

From the final time-averaged velocity fields, surface velocity profiles along certain cross-sections were extracted for both dataset A and dataset B (Fig.4.10). For the former, one cross-section was computed in proximity to the FOEN gauging station (BB'), one upstream (AA') and one downstream (CC'). For video B, only one velocity profile was extracted in proximity to the gauging station (DD').

4.8 Estimation of the discharge

In order to evaluate the results obtained, from the velocity profile along the cross-section BB', for dataset A, and DD', for dataset B, the values of discharge are estimated according to Hauet et al. (2008). Those are compared to the values measured by the gauging station. The discharge is obtained by a velocity-area method: assuming that the shape of the vertical velocity profile is the same at each point i along the cross-section, the discharge is computed by numerical integration of the depth-averaged velocity (V_i) over the bathymetry. The values of V_i can be obtained from the free-surface velocity (v_i) as follows:

$$V_i = \alpha v_i \tag{4.1}$$

where α is a constant multiplicative coefficient depending on the shape of the vertical velocity profile. The values of α are taken from Le Coz, Hauet et al. (2010), who proposed the following relation:

$$\alpha = \frac{h}{h - z_0} - \frac{\beta h - z_0}{h - z_0} \left(\log \frac{\beta h}{z_0} \right)^{-1} \tag{4.2}$$

where h is the flow depth, z_0 is the roughness length of the river bed and β the maximum relative elevation of the flow below which a log velocity is observed. According to Le Coz, Hauet et al. (ibid.), possible values of β between 0.6 and 0.7 are considered.

The bathymetry profile was obtained from the survey performed by FOEN during the same day, as explained in Sec.3.1.2. As it was measured in-situ from the water surface, it can be assumed as a measure of flow depth along the section of the river. The values of free-surface velocity in proximity to the ADCP rope (v_i) were extracted from the surface velocity fields obtained with the three AIV approaches. They were resampled in the same point as the bathymetry profile with a linear interpolation. In case of missing values, the interpolation and extrapolation approaches proposed by Le Coz, Hauet et al. (ibid.) were applied.

The values of discharge estimated as before are compared to the actual ones in order to have a rough assessment of the reliability of the surface velocity fields. Nevertheless, also the actual measures of discharge, obtained from FOEN, are affected by some uncertainties and they should be taken into account in the comparison against the values estimated through the AIV technique. For dataset A, a hydrometric impeller measurement is available and its error can be assumed about the 3% of the discharge (Spreafico and Weingartner 2015), which results in ± 2.5 m^3/s. For dataset B, on the other hand, the reference value of discharge has been obtained from the water level using the rating curve for the river Limmat computed by FOEN. According to Horner et al. (2018), the uncertainties in the discharge estimation with this approach can be considered less than $\pm 10\%$, which corresponds to, at most, ± 22 m^3/s.

Chapter 5

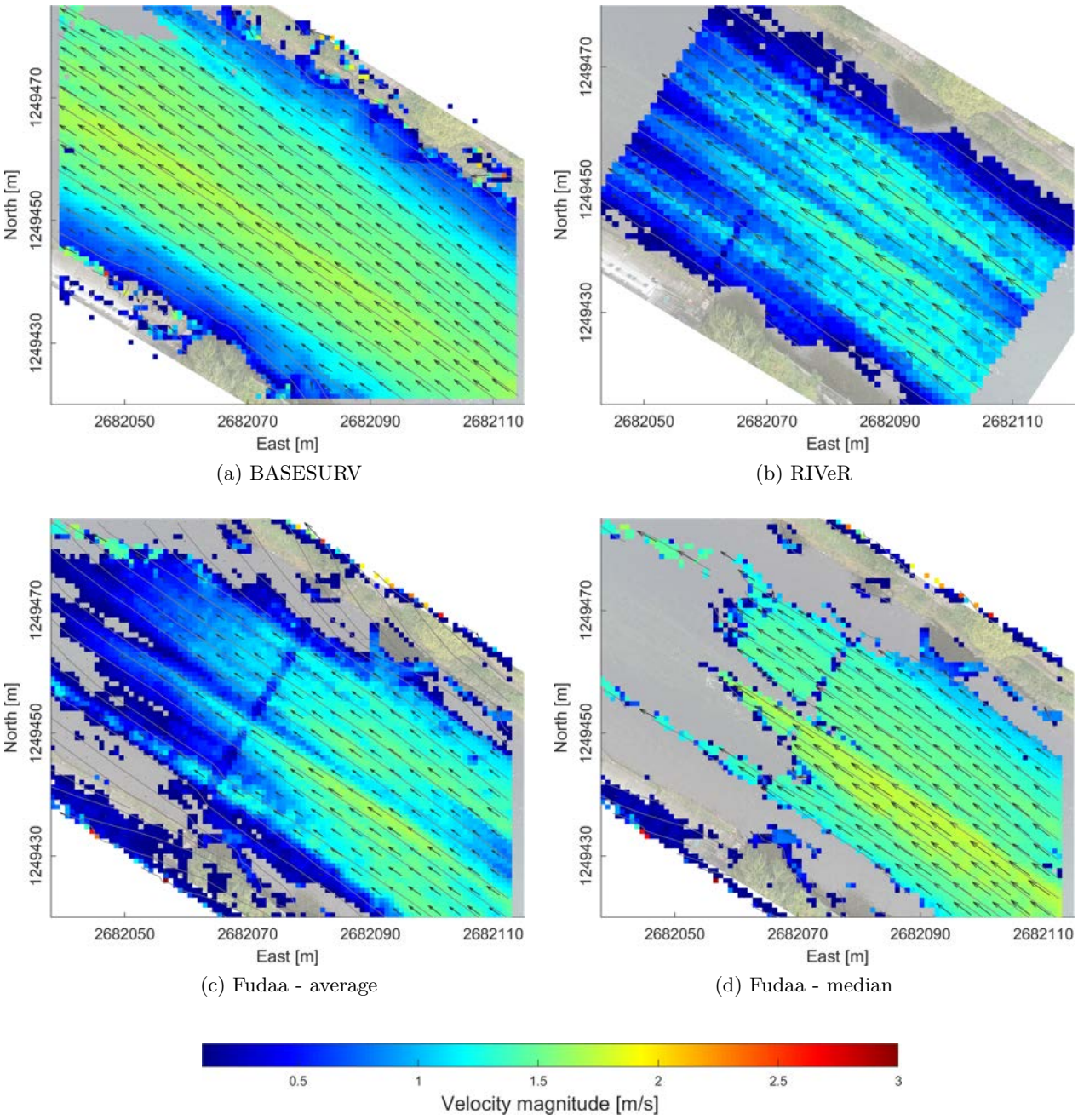
Results of the AIV approaches comparison

In this chapter, results computed by BASESURV, Fudaa-LSPIV and RIVeR from the two datasets are presented. First, the time-averaged velocity fields are plotted and the time series of the velocity vectors are analysed. Then the comparison between the velocity profile along certain cross-sections are presented. The discharge estimated from them is compared to that measured in-situ or by the gauging station.

5.1 Dataset A

The main output of the AIV process is the time-averaged surface velocity field. In Fig. 5.1 the results of the three approaches for dataset A are presented: the vectors of surface water velocity are superimposed to their magnitude, plotted with a scale of colours. As explained in Sec. 4.6, from the instantaneous results of Fudaa, both the average and the median of the vectors time series were computed.

The results obtained with the different AIV approaches are different: the most complete (i.e. with fewer missing data in the study area) and smoothest surface velocity field is that obtained with BASESURV (Fig. 5.1a). The averaged surface velocity fields obtained with Fudaa (Fig. 5.1c) and RIVeR (Fig. 5.1b) show a clear underestimation of the velocity compared to the BASESURV one, especially in the downstream portion of the Region of Interest (ROI). This effect is due to the lack of seeding particles in a significant portion of the video frames because they are not yet in the camera view (e.g. downstream to the ACDP rope, where the seeds started to arrive around frame #100 out of 262) or they have already moved away (e.g. uppermost portion of the ROI, especially in central part of the river, where the surface velocity is higher). This low-bias of the velocity can be explained looking to the time series of the two velocity components u and v at the grid points Pt. A1 and Pt. A2 (respectively upstream and downstream of the ACDP rope), plotted in Fig. 5.2. In Pt. A2 (Fig. 5.2b), Fudaa and RIVeR time-series clearly show near-zero velocity vectors up to $\sim$ 15 s (i.e. when the seeding particles have reached the point under analysis) and this causes a strong shift in the average towards lower values. In cases such as Pt. A2, the amount of near-zero vectors in the Fudaa and RIVeR time series is even greater than 50%. Therefore, the median vector falls into the low-value cluster and it is filtered out using thresholds on the minimum and maximum accepted velocity magnitude, producing a *no-data* result in the final velocity field. However, also in the time series of Pt. A1 (Fig. 5.2a), where the tracer particles are present for the whole video, it is possible to see a cluster of near-zero vectors due to an imperfectly homogeneous seeding, which is actually rather



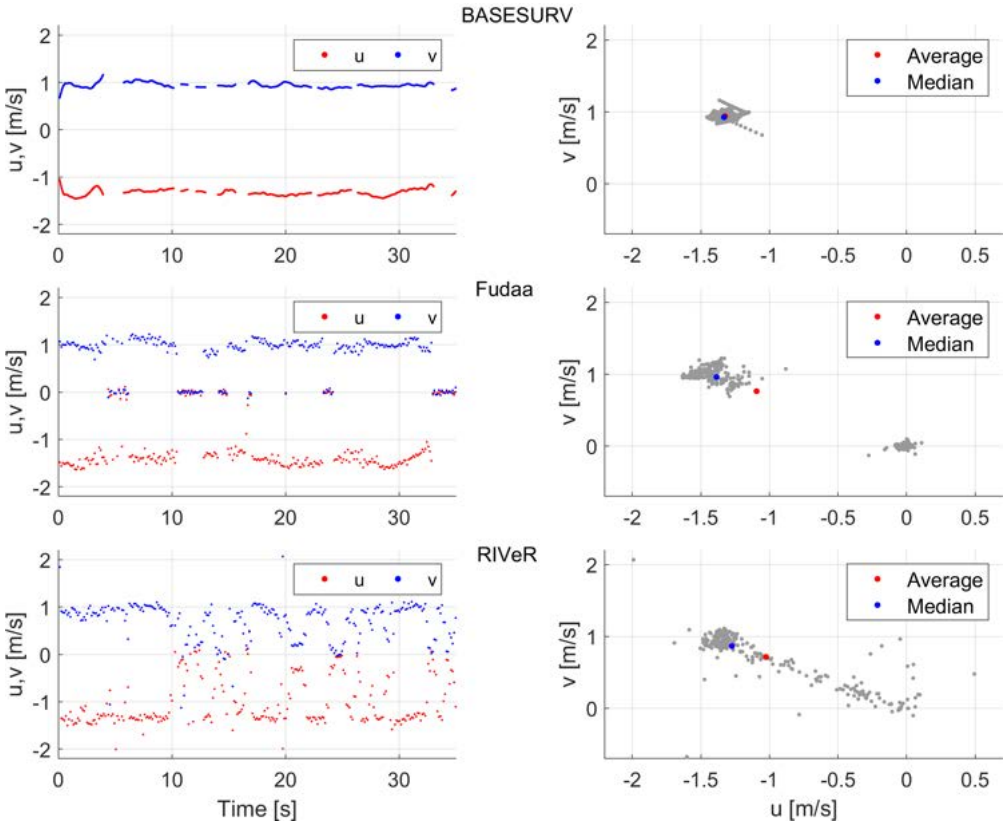
(a) BASESURV

(b) RIVeR

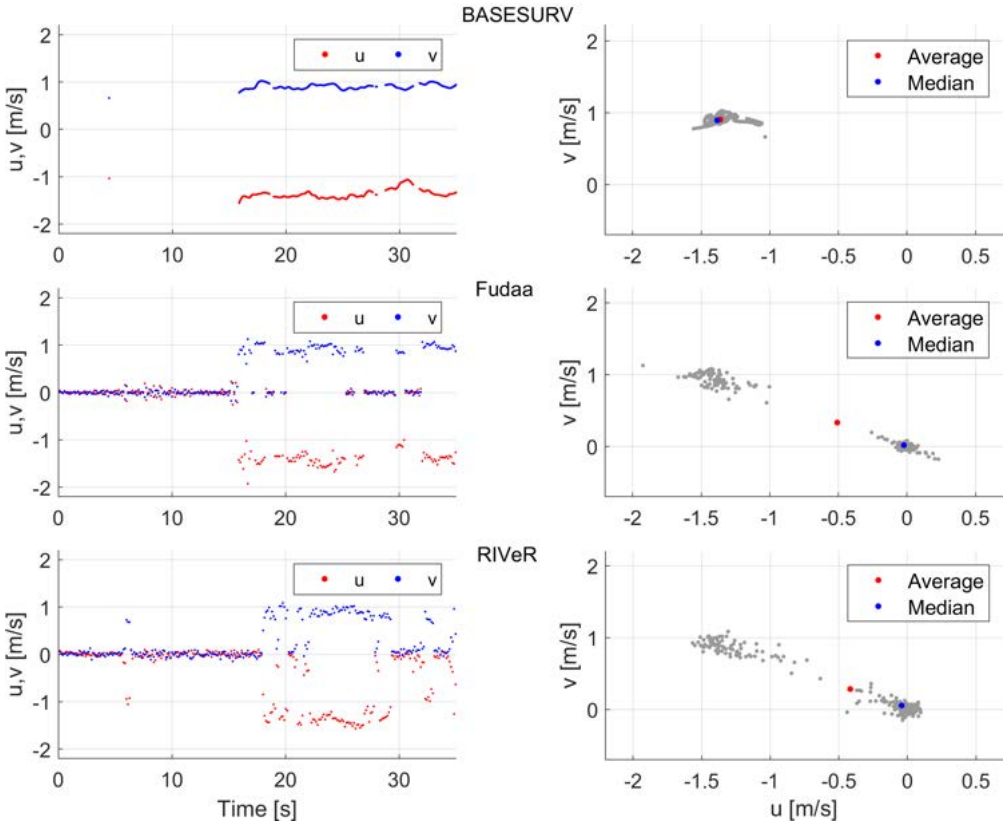
(c) Fudaa - average

(d) Fudaa - median

Figure 5.1: Time-averaged surface velocity field obtained from dataset A with the three different AIV approaches: (a) BASESURV median velocity field; (b) RIVeR average velocity field; (c) Fudaa average velocity field; (d) Fudaa median velocity field. For larger figures, see App.A

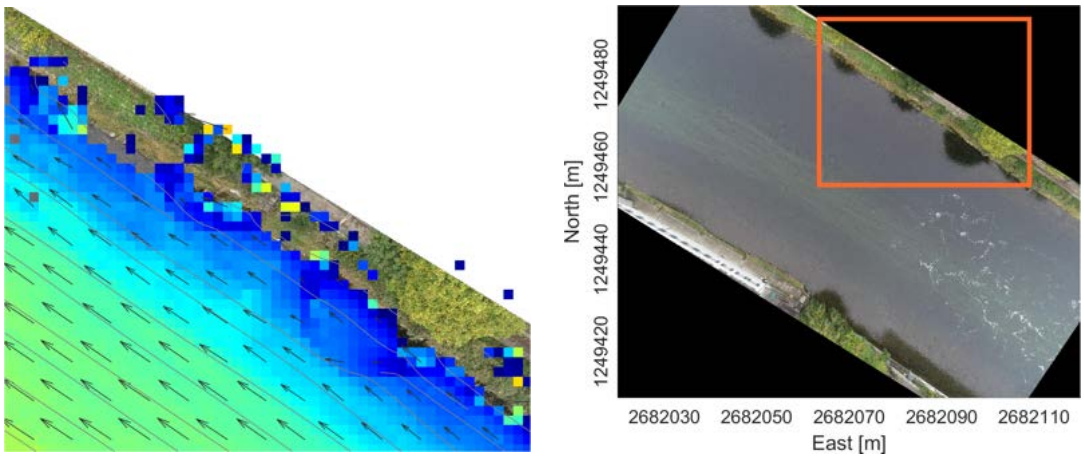


(a) Pt. A1



(b) Pt. A2

Figure 5.2: Time series and scatter plot of the two cartesian velocity components u and v at grid points as follows: (a) Pt. A1, upstream of the ACDP rope; (b) Pt. A2, downstream the ACDP rope. The location of the two points is given in Fig. 4.10a.



(a) (b)

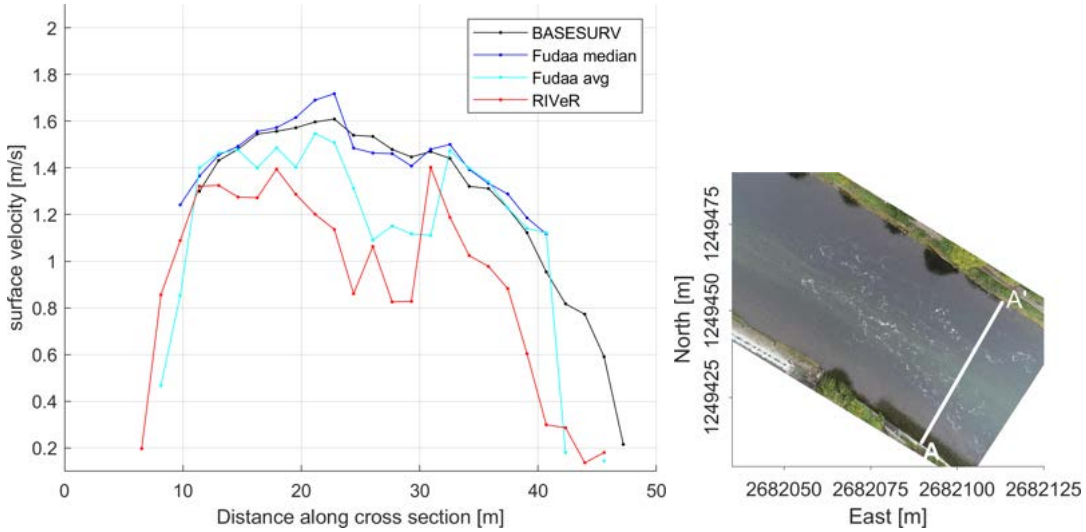
Figure 5.3: (a) Cut-out at the location marked in (b) of the BASESURV time-averaged velocity field. The noise caused by the shadows of the trees is clearly visible; (b) Location of the cut-out.

unrealistic in a riverine environment. These outliers, if not filtered out, cause a bias in the average vector whilst the median operator is more robust against them.

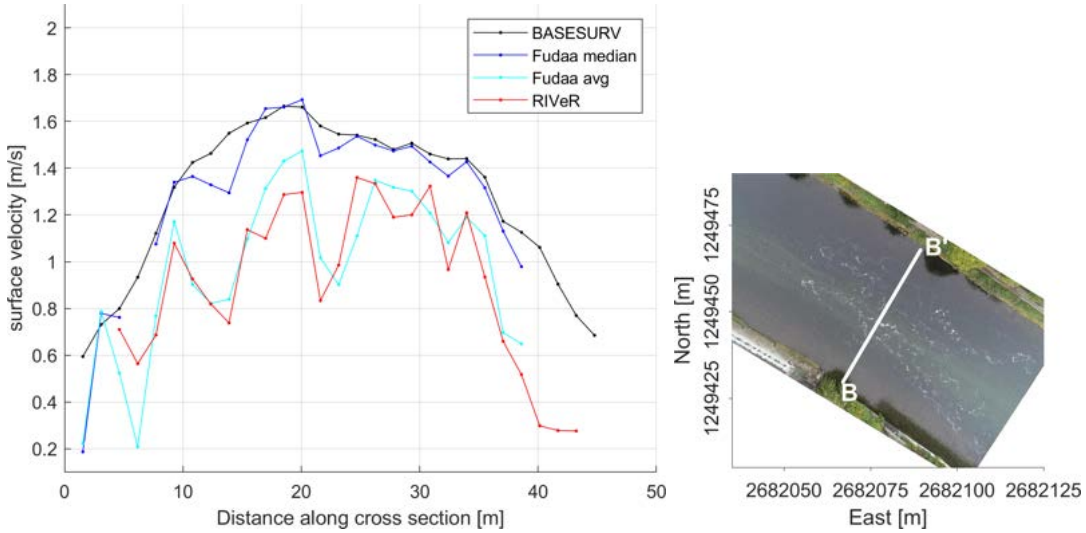
Considering again the time series in Pt. A1 and Pt. A2 obtained with BASESURV, these are more dense than those obtained with Fudaa and RIVeR due to the higher number of instantaneous results computed (see Tab. 4.3). The missing data in the BASESURV time series mostly occur when those of Fudaa and RIVeR show near-zero vectors. This highlights the effectiveness of the 2D multi Gaussian filter in separating the two clusters and retaining only the inlier velocity vectors. Moreover, the very low dispersion of the time series and its deterministic appearance is due to the RLOESS smoothing filter.

The cross-sections extracted from the final, average or median, velocity fields are plotted in Fig. 5.4. In the same way as the surface velocity field, the profiles obtained with BASESURV are the most uniform along the river: they present the highest velocity value of $\sim 1.6 - 1.7$ m/s between 18 m and 23 m from the left-hand riparian side and the lowest values between $0.6 - 0.7$ m/s. The cross-sections computed from the Fudaa and RIVeR results highlight again the underestimation of the velocity due to the low-value cluster of data in the time series: the bias increases from upstream (section AA') towards downstream (sections BB' and CC'), where the overall signal to noise ratio given by the seeding particle is lower. On the other hand, the profiles obtained from the Fudaa median velocity field along the cross-sections AA' and BB' are significantly closer to those obtained with BASESURV. Nevertheless, on the cross-section CC', where more than 50% of the time series is characterized by outliers, the median surface velocity field does not present any reliable velocity vector.

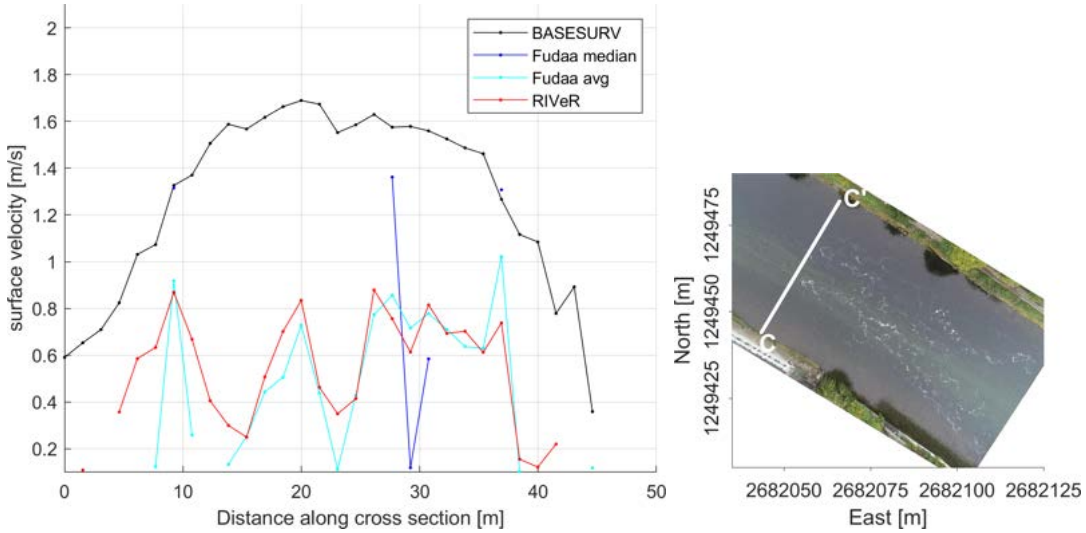
Looking back to Fig. 5.1, it is possible to see that the PIV technique suffers from the presence of shadows on the moving water surface. The results of all the three AIV approaches, in fact, are affected by noise in proximity to shadows of trees (especially along the right-hand side of the ROI, Fig. 5.3) and the large building in the south-west corner. The ACDP rope, in the middle of the study area, is found to be another source of noise. As it is $\sim$ 2 m above the water surface, during frame rectification it can be mapped in slightly different pixels, depending on the position and attitude of the camera, which is affected by the shaking of the UAV. The apparent movement of the



(a) cross-section AA'



(b) cross-section BB'



(c) cross-section CC'

Figure 5.4: Velocity profiles at the cross-sections: (a) AA'; (b) BB'; (c) CC'.

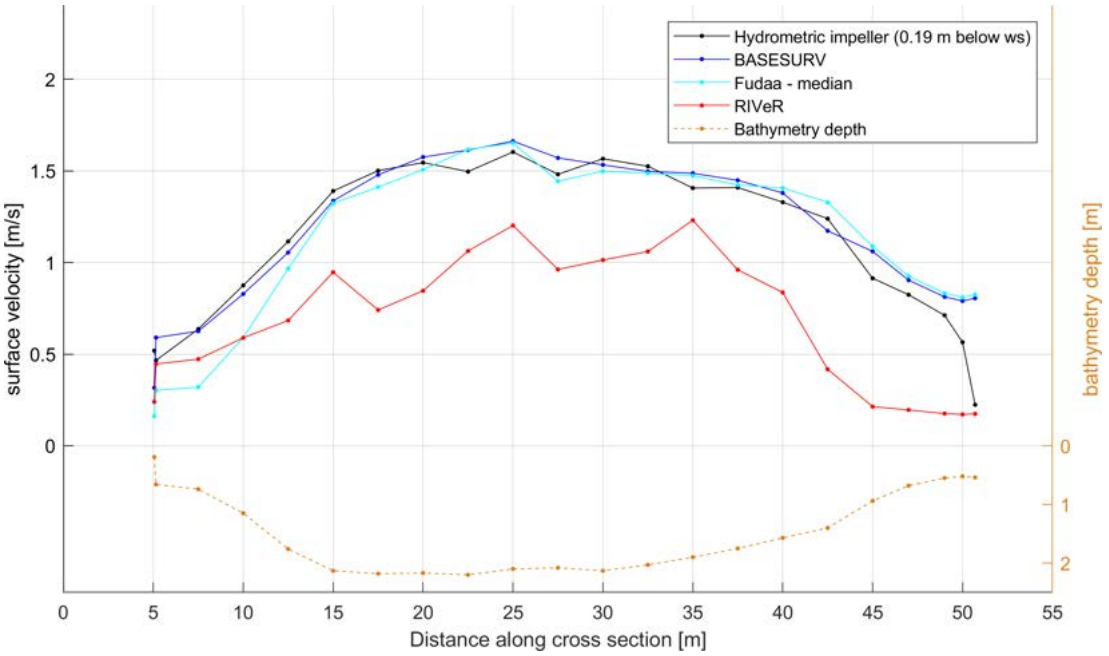


Figure 5.5: Velocity profile measured by the hydrometric impeller (0.19 m below the water surface) compared against the surface velocity profiles obtained by AIV along the cross-section BB' (its location is given in Fig. 4.10a). These have been resampled in the same points as the bathymetry depth profile and extrapolated with the approach proposed by Le Coz, Hauet et al. (2010) to fill missing data.

Table 5.1: Comparison between the discharge measured by the hydrometric impeller and those estimated by AIV. For the latter ones, the interval of possible values, depending on the parameter β as explained in Sec. 4.8, is given.

	Hydrometric impeller	BASESURV	Fudaa	RIVeR
Q [m^3/s]:	83 ± 2.5	$[79, 86]$	$[77, 84]$	$[49, 54]$

rope leads to have spurious vectors in the PIV analysis, mostly characterized by lower velocity values and different direction compared to the water flow. This effect is evident in both the results computed with Fudaa and RIVeR; only in BASESURV does this issue seem not to have a strong impact. By analysing the time series, the 2D multi Gaussian filter, in fact, is able to identify this cluster of vectors with low velocities and high scattering in the direction due to the ACDP rope and separate it from the velocity vectors produced by the movement of seeding particles.

In Fig. 5.5, the velocity profiles measured by the hydrometric impeller are compared against those obtained by the three AIV approaches along a cross-section immediately upstream to the ADCP rope (in order to avoid the noisy area in the Fudaa and RIVeR results). For Fudaa, only the median velocity field was considered. In the centre of the 45 m wide river, i.e. after 5 m from each riparian side, relative percentage errors in the estimated velocity of 4 %, 5 %, 36 % were obtained respectively with BASESURV, Fudaa and RIVeR. Near the riparian sides, i.e. within 5 m, higher errors were obtained with all the approaches: respectively, 56 %, 71 %, 47 %. Moreover, the comparison between the discharge values, estimated as explained in Sec. 4.8, against the reference hydrometric impeller measurement is presented in Tab. 5.1. The agreement between the discharge estimated from the BASESURV and Fudaa results with the one measured

in-situ can be considered as a proxy for the reliability of the surface velocity field (at least in proximity to the cross-section BB'). The underestimation of the discharge computed from the RIVeR results is caused by the low-bias in the velocity field, as already explained above.

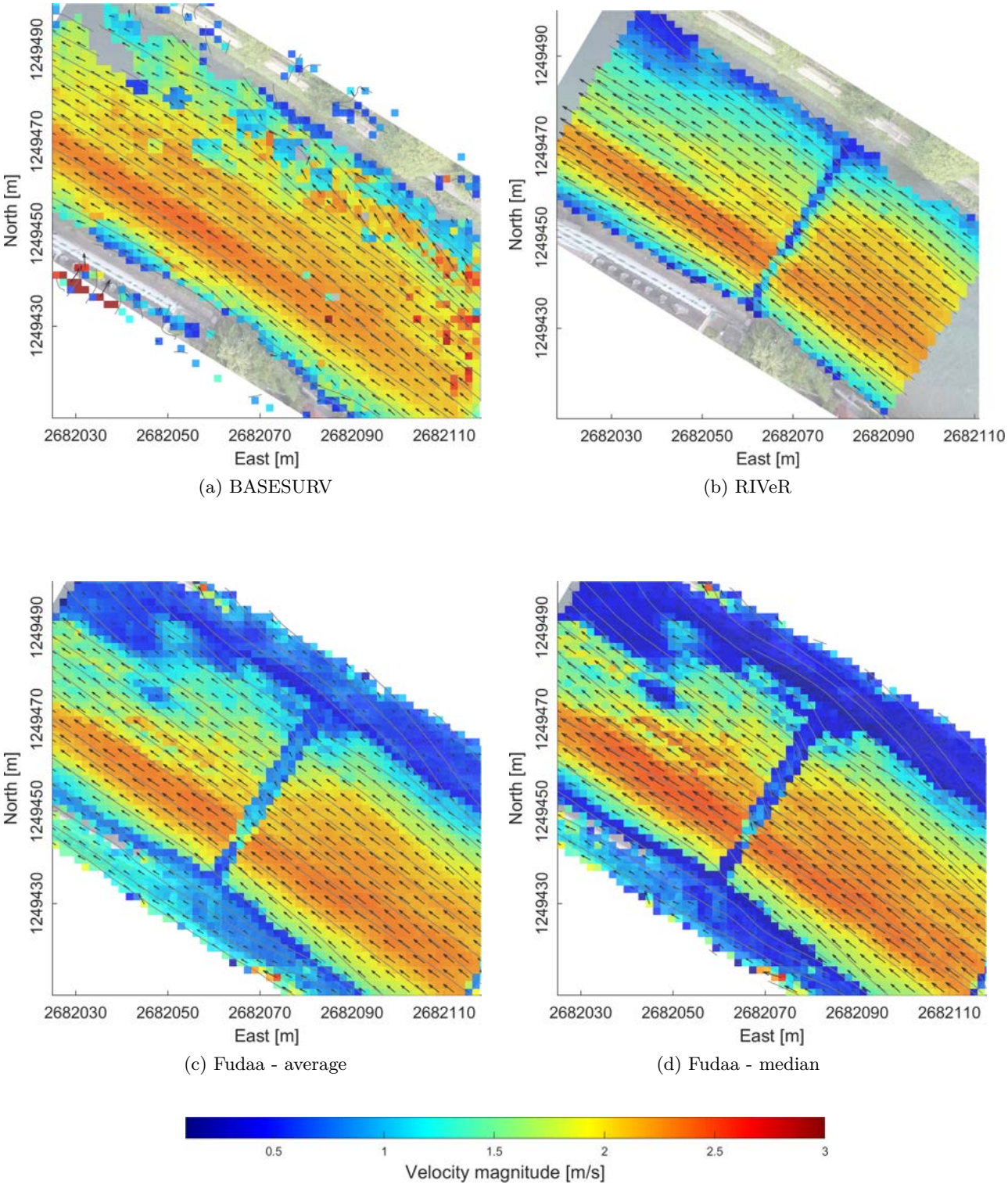
All things considered, when clear seeding particles are present, all the three software are able to obtain similar results. However, the result of Fudaa and RIVeR can be biased if there are no tracers in some portion of the ROI for a significantly long time window compared to the total length of the video.

5.2 Dataset B

Analogously to dataset A, the time-averaged surface velocity fields obtained from video B with the three AIV approaches are presented in Fig. 5.6. First, it is evident that the surface velocity of the flow is higher compared to that of dataset A: excluding the noisy areas, the velocity ranges between 1.5 m/s, near the riparian sides, and $2 - 2.5$ m/s, in the central part of the flow. All the results are similar and show a high-velocity current (about 2.5 m/s) on the left-hand side of the river Limmat. On the right-hand side, however, all the AIV approaches have difficulty in estimating the surface flow velocity and the results are strongly affected by noise. In contrast to the results of dataset A, there are no significant differences between the mean and median time-averaged velocity fields computed by Fudaa. The latter shows slightly higher velocities than the first one, but the difference is overall less then 0.1 m/s. Moreover, in the same way as for dataset A, Fudaa and RIVeR are not able to compute reliable velocity vectors in proximity to the ACDP rope.

Fig. 5.7 displays the time-series of velocity vectors at two grid points in the downstream portion of the ROI: Pt. B1 on the left side of the river and Pt. B2 on the right side. For Pt. B1 (Fig 5.7a), a rather strong signal is visible in all the time-series. However, a strong dispersion is present in those obtained with Fudaa and RIVeR (the BASESURV time-series, in fact, has already been smoothed by the RLOESS smoothing filter). This scattering is mainly due to the sub-pixel estimation which allows the most probable integer values of displacement to be refined. Errors in the sub-pixel estimation have a strong impact on the accuracy of the estimated velocity if the integer displacement is very small (i.e. the relative error is high). In Pt. B1, for instance, the estimated integer displacement is equal to just 1 px for most of the time-series. As already stated in Sec. 4.4, a higher time lag should have been used in this case to overcome this issue. Analogous considerations are possible for Pt. B2 as well (Fig 5.7b). In this case, however, the time-series is more affected by noise and it is more difficult to identify a clear signal. Fig. 5.8a shows a frame of video B: it is possible to see that on the left-hand side of the river, the water carries a large amount of suspended solids whilst on the right-hand side it is cleaner. This suspended sediment comes from river Sihl, which flows into river Limmat just $\sim$ 200 m before the Zürich Unterhard gauging station (Fig. 5.8b). These suspended solids work like natural seeding particles, improving the cross-correlation in PIV. On the right-hand side of flow and especially in the downstream portion of the ROI, there are fewer patterns on the water surface that can be detected by PIV. Therefore, the signal-to-noise ratio in these areas is lower. Probably, this, together with the small time lag used, is the cause of the strong noise visible in the time-averaged velocity fields and in the time-series of vectors obtained with all three AIV approaches.

Moreover, in the results of Fudaa (both in the average and in the median surface velocity fields), it is possible to see that velocity vectors with a magnitude between



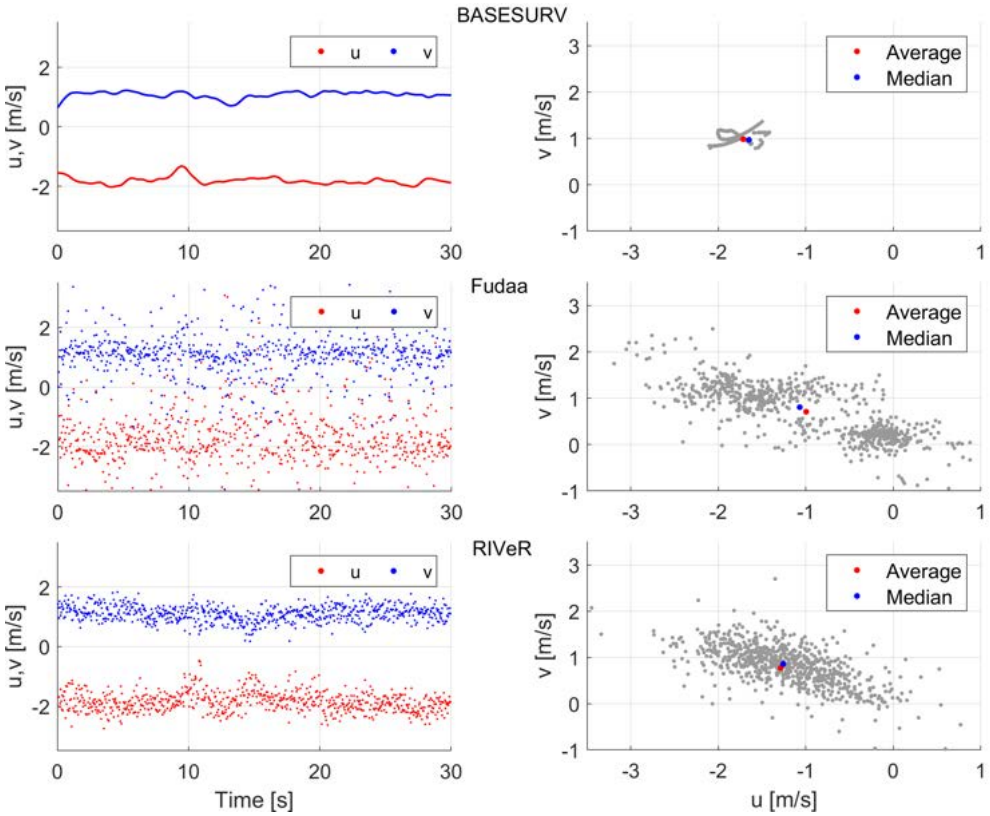
(a) BASESURV

(b) RIVeR

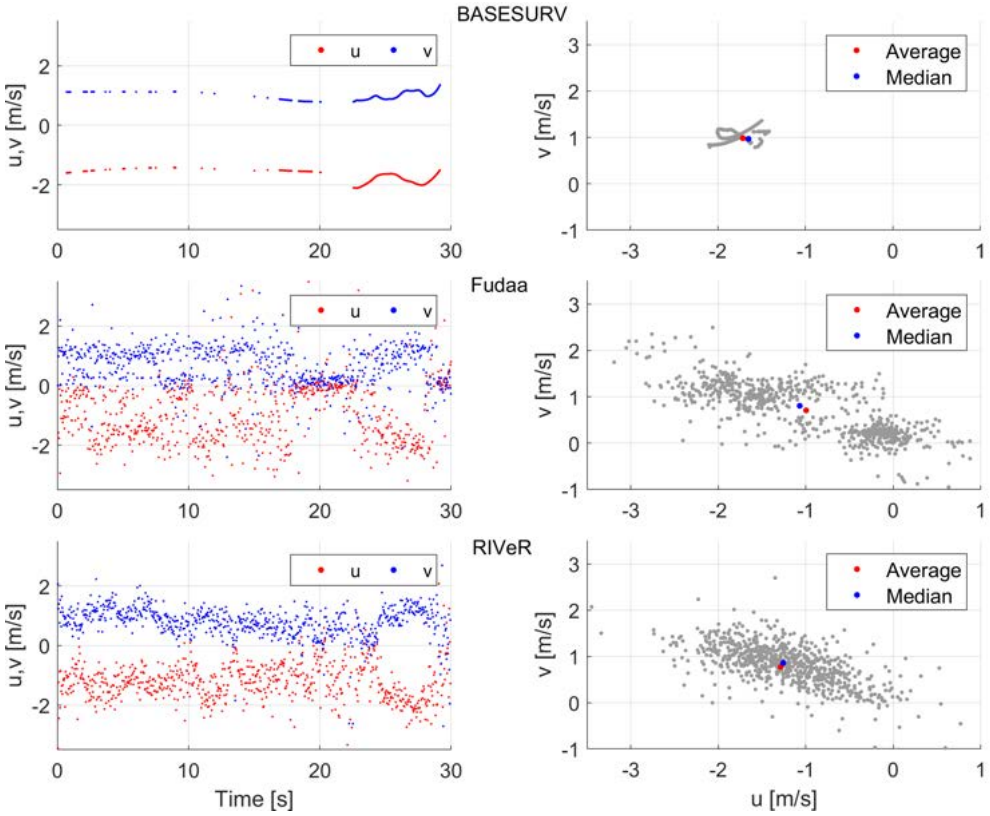
(c) Fudaa - average

(d) Fudaa - median

Figure 5.6: Time-averaged surface velocity field obtained from dataset B with the three different AIV approaches: (a) BASESURV median velocity field; (b) RIVeR average velocity field; (c) Fudaa average velocity field; (d) Fudaa median velocity field. All the results are computed using a time lag of 0.042 s. For larger figures, see App.A

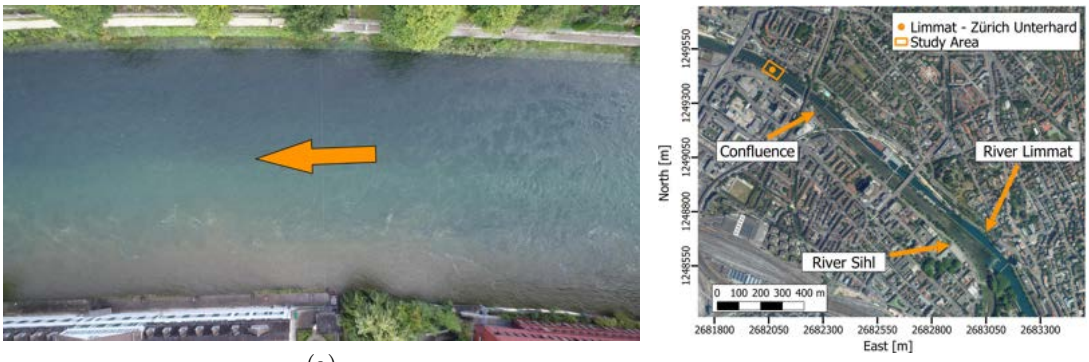


(a) Pt. B1



(b) Pt. B2

Figure 5.7: Time series and scatter plot of the two cartesian velocity components u and v at grid points as follows: (a) Pt. B1 on the left-hand side; (b) Pt. B2 on the right-hand side. The location of the two points is given in Fig. 4.10b.



(a) (b)

Figure 5.8: (a) Sample frame of video B. The orange arrow shows the flow direction. The water coming from river Shil, carrying lot of suspended sediments, can be clearly distinguished. (b) Orthophoto of the city of Zürich showing the confluence of river Sihl into river Limmat, just ∼200 m upstream of the Zürich Unterhard gauging station.

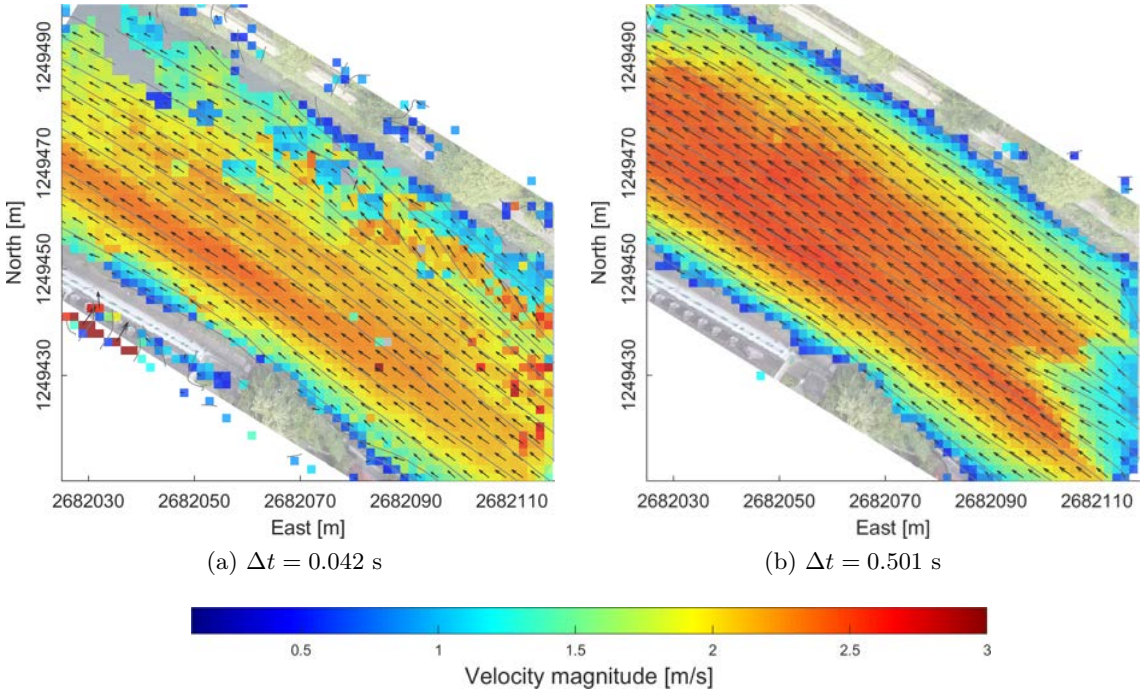
Table 5.2: Comparison between the discharge values estimated by AIV against those estimated with the rating curve of river Limmat. For the first ones, the interval of possible values, depending on the parameter β as explained in Sec. 4.8, is given.

	Rating curve	BASESURV	Fudaa	RIVeR	BASESURV (time lag 0.5 s)
Q [m^3/s]:	220 ± 22	$[170, 184]$	$[176, 192]$	$[174, 190]$	$[210, 228]$

0.3 m/s and 1 m/s are computed over the stable areas along the riparian sides. These bands of spurious velocity are clearly due to an imperfect stabilization of the video, which should be definitely improved for a reliable AIV analysis.

As explained in Sec. 4.4, video B was processed a second time using BASESURV and a longer time lag, more suitable for the range of velocities under investigation. Fig. 5.9 illustrates the comparison between the surface velocity field obtained using a time lag $\Delta t = 0.042$ s and $\Delta t = 0.501$ s. The result computed with the latter is significantly smoother over the region of interest with higher surface velocities and it does not present noise in the left-hand side of the river. Therefore, it is a further evidence that a longer time lag would be necessary. This outcome, however, reveals that it is potentially possible to obtain reasonable results with AIV even when no seeding particles are used, but only identifying characteristic patterns due to water structures.

Furthermore, the velocity profiles along cross section DD' were extracted from the time-averaged surface velocity fields computed with the three AIV software packages by using $\Delta t = 0.042$ s and with BASESURV by using $\Delta t = 0.501$ s (Fig. 5.10). From those, the discharge was estimated in an analogous way as for video A. Tab. 5.2 presents the comparison between the discharge estimated by AIV that obtained from the water level using the rating curve of the river Limmat (Swiss Federal Office for the Environment FOEN 2020). Both the velocity profiles and the discharge obtained by the three software packages with the short time lag are similar, but underestimated compared to those obtained with BASESURV with the longer time lag. Only that is comparable to the discharge estimated with the rating curve.



(a) $\Delta t = 0.042$ s (b) $\Delta t = 0.501$ s

Figure 5.9: Time-averaged surface velocity field computed with BASESURV using different time lag in PIV: (a) $\Delta t = 0.042\,\mathrm{s}$; (b) $\Delta t = 0.501\,\mathrm{s}$.

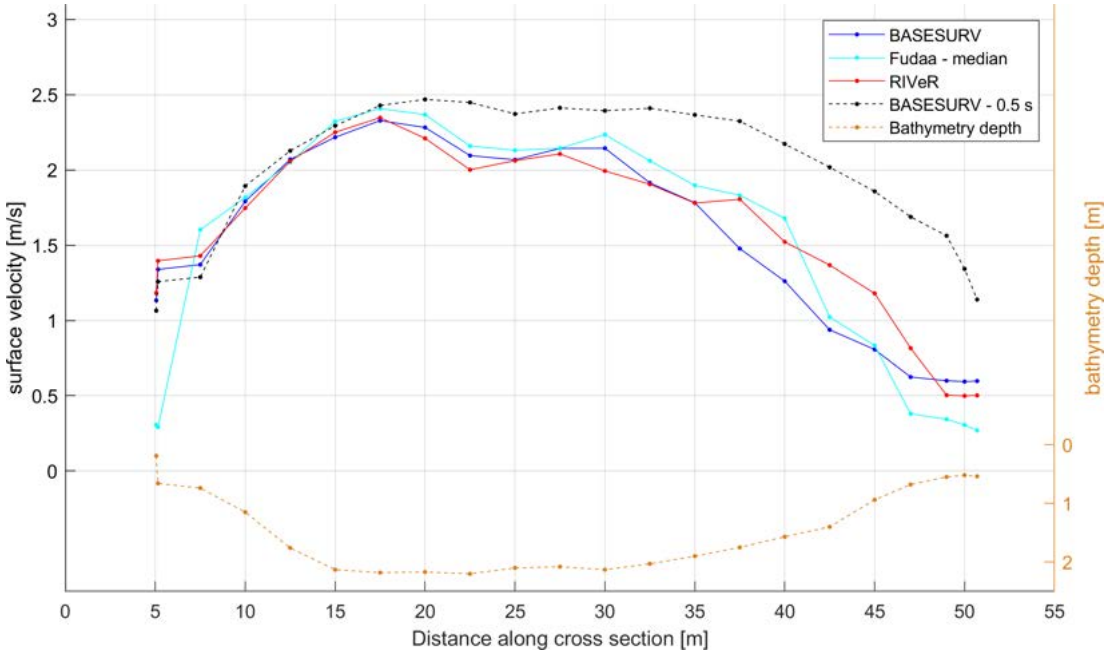


Figure 5.10: Comparison of the velocity profiles along the cross section DD' obtained by AIV (the location is given in Fig. 4.10b). These have been resampled in the same points as the bathymetry depth profile and extrapolated with the approach proposed by Le Coz, Hauet et al. (2010) in order to fill missing data. Both the velocity profiles obtained with BASESURV using the two different time lags are plotted.

Chapter 6

Discussion

In this chapter, the performance of the three AIV approaches are discussed in the light of the results obtained in Chapter 5. Tab. 6.1 presents a summary of the computational time needed by each software package to process video A. The values are normalized by the number of frames used in the computation in order to make the comparison even. BASESURV, in fact, is able to use all the video frames, independently of the time lag chosen for PIV, while Fudaa and RIVeR may use just a sub-sample of them. All the processes were carried out by a mid-range office workstation (CPU Intel® Xenon® E3-1245 v6 @ 3.7 GHz; RAM 16 GB; GPU Intel® HD Graphics P630).

The main strengths and limitations of the three AIV approaches experienced in this work are listed, grouped by the processing phase, as follows:

1. Image undistortion:

    - In contrast to BASESURV and RIVeR, Fudaa does not provide any tool to correct the internal distortions of the camera. In this study case, image undistortion was not crucial, but it may assume more importance when a wide-angle objective or a fish-eye camera is used.
    - In RIVeR, video frame undistortion requires extremely long time to be performed, even more than the whole video stabilization. Even if the results seem to be correct, there may be some problems with the code or it may need to be optimized.

2. Stabilization and rectification:

    - The approach proposed by BASESURV to stabilize and rectify the frames is probably the most accurate and robust, also from the theoretical point of view. Assuming the water surface as a plane parallel to the one used to produce the orthophoto, the homography transformation hypotheses are satisfied if the double points are selected on the same plane. Under these conditions, only an orthophoto in which flat riparian sides are visible and a mask to exclude *out-of-plane* objects are needed to perform the rectification of the video frames (see Sec. 4.2). If those are available, the process is completely autonomous. On the other hand, this approach is the most demanding in terms of computational time (almost twice that of Fudaa, if the stabilization and rectification are summed together) and there is room for improvement in optimizing the code.
    - Video stabilization in Fudaa and RIVeR presents some critical aspects from the theoretical point of view, especially with strong shaking videos. In Fudaa, for the same reason discussed above, the *out-of-plane* objects should

Table 6.1: Comparison of the time needed by the three AIV approaches to process video A, normalized by the number of frames involved in the computation.

Software	Phase	Time [min/frame]	Time tot. [min/frame]
BASESURV	Rectification [1]	0.65	0.73
	PIV [2]	0.06	
	Post-processing filters [3]	0.02	
Fudaa	Video stabilization	0.08	0.42
	Frame rectification	0.23	
	PIV	0.11	
RIVeR	Frame undistortion	0.06	0.16
	Video stabilization	0.04	
	PIV [4]	0.06	
	Vector rectification	0.004	

[1] Rectification includes frame undistortion, video stabilization and frame rectification.
[2] PIV includes also the image pre-processing (background subtraction and noisy pixel intensity capping).
[3] That refers to the time-series based post-processing filters (2D multi Gaussian filter and RLOESS smoothing).
[4] Only for the comparison of the computational time, PIV in RIVeR was performed using the rectified frames from BASESURV, in order to have the same grid for all the AIV approaches. It includes also image pre-processing (background subtraction).

be excluded in the research of matching points, when a projective transformation is used. Moreover, the stabilization approach of Fudaa seems to be less robust than the others. With video B, indeed, some problems occurred during stabilization taking 1 frame in every 4 or 8. Only by using all the video frames, the process was successful. Even in this case, however, non-optimal stabilization was achieved and several spurious vectors due to a bad alignment of the frames were found. The causes of this problem are not completely clear and they should be investigated further. In the case of RIVeR, the stabilization is based on the estimation of affine transformations between pairs of consecutive frames. It is the fastest approach (it takes about half the time of the stabilization in Fudaa) and, likely, the easiest to implement from the developers' point of view. However, it is probably too approximate and results in drift residual errors which may produce spurious velocity vectors in PIV.

- Rectification of the frames in Fudaa and vectors in RIVeR is straight forward and easy to perform. Nevertheless, a minimum number of GCPs (or distances between them) are required to be measured in the field and collimated on the first video frame. This is time-consuming, both during the survey and in the processing phase. Moreover, RIVeR does not implement any tool for the accuracy assessment of the rectification, which could be very useful for the final user.
- The difference in the rectification approaches of BASESURV and Fudaa compared to that of RIVeR (i.e. either rectifying the images or the velocity vector) is a central issue. Rectifying all the images is found to be a more robust technique, but certainly more demanding from the computational point of view. Following this approach, PIV is applied directly on orthophotos with a known pixel dimension in the world reference system. That makes it easier to manage the results of PIV and to obtain a rough assessment of the effects of the rectification error on the velocity vectors (e.g. by applying PIV over a stable area). Moreover, the application of state-of-the-art computer vision techniques is possible. For instance, the feasibility of an automatic detection of GCPs in the images (e.g. photogrammetric targets placed on the ground) and matching them with their 3D coordinates may be investigated in the future. That would overcome the limits of BASESURV due to the application of a 2D homography transformation for the image rectification such as the need for a flat surface with the same height as the water level. On the other hand, rectifying only the vectors allows for a significant computational time reduction. This may be particularly useful if the transformation between the image reference system and the world reference system is known with high accuracy and it does not change during the video, as with a fixed camera. In this case, the process of vector rectification could be performed automatically and very fast, allowing, for instance, a real-time monitoring system to be implemented. However, it would be interesting to study in more depth the grade of impact of the propagation of the error in the estimated homography transformation over the velocity vectors, especially if the video is not perfectly stabilized and only one transformation is estimated.

3. Image pre-processing:

    - While BASESURV and RIVeR provide several image pre-processing tools

(see Sec. 4.3), this processing step is completely absent in Fudaa. It would be useful to implement at least the background subtraction in the future developments of the software.

4. PIV:

   - Both BASESURV and RIVeR are based on PIVlab and therefore both share the same characteristics: the correlation analysis is carried out in the frequency domain using FFT and a multi-pass iterative procedure is implemented. Fudaa performs the correlation in the spatial domain by computing the correlation matrix in one pass (see Sec. 4.4). The first approach is theoretically the most effective: it allows the dynamic range of velocity that can be resolved to be expanded (especially rising the upper limit), but without loss of spatial resolution. However, this aspect seems not to be related to any significant difference in the results obtained in this work, but it may be relevant in other circumstances.
   - As the correlation analysis is performed in the frequency domain through FFT, in BASESURV and RIVeR the PIV computation is almost twice faster than in Fudaa.
   - The ability of BASESURV to use all the video frames in PIV computation, regardless of the time lag chosen, is found to be a great advantage, both in terms of results (no loss of useful data) and user experience of the software. In this way, in fact, all the video frames are rectified and the time lag is chosen afterwards during PIV. Therefore, it is not necessary to perform the whole process from the beginning (i.e. extracting again the video frames using a different step) if a different time lag has to be tested.
   - In both Fudaa and RIVeR there is the possibility of removing some grid points from the computation (e.g. creating a mask inside the ROI or deleting them manually). This allows the computational time in PIV to be reduced, when some grid points fall in stable areas or outside the region of interest in the images (e.g. when the river direction is not parallel to one of the image axes). It would be useful to implement this also in BASESURV.

5. Post-processing filters:

   - All three approaches implement some filters on each instantaneous velocity field, independently one from the others. Only BASESURV implements filters based on the time series of vectors at each grid point (see Sec. 4.5). These are found to be very effective, as presented in Sec. 5.1. However, it is clearly challenging to develop smart filters based on the time-series and further research on this aspect may be carried out in the future to improve the existing techniques.
   - In Fudaa, it would be useful to implement a tool to plot the time-series of the velocity vectors as well as the u-v scatter plot at certain grid points, such as those presented in Chapter 5. That would help the user to analyse the results and to calibrate the thresholds in the velocity components in order to reject spurious vectors.
   - In both Fudaa and PIVlab (and therefore also in RIVeR), a further possible improvement in filtering the instantaneous vectors may be the possibility to define an elliptical region of acceptance in the u-v plane, instead of a rectangular one (i.e. by defining thresholds in the u and v velocity components).

6. Time-averaged velocity field:

    - As seen in Sec. 5.1, computing the median vector from the time series at each grid point provides a more robust estimation of the time-averaged velocity against the outliers. Implementing the median operator in place of or in addition to the average in both Fudaa and RIVeR would be a positive improvement and, at the same time, probably not extremely demanding from the programmer's point of view.

Several aspects have not been considered in this work and they may be investigated further in future studies:

- An accuracy assessment of estimated velocity: due to the fact that several factors (e.g. misalignment of the images during stabilization, errors in rectification as well as in the identifying the peak of the correlation, sub-pixel estimation etc.) contribute to the errors of the final velocity vectors, it is challenging to provide an accuracy of the estimated velocity. However, this would be necessary for both scientific and professional applications of AIV and it should be investigated in future studies.

- Use of oblique viewing angles: during the two surveys carried out on river Limmat, two oblique angle videos were also acquired. However, they were not carefully processed and analysed due to lack of time.

- Possibility not to use GCPs for rectification: if the position and the attitude of the camera is known with sufficient accuracy (e.g. mounting an RTK GNSS antenna on the UAV and an IMU on the camera), it should be theoretically possible to apply an homography transformation to the video frames directly, without the need for GCPs. This would be extremely useful to save time both during the survey in the field and in post-processing. Moreover, this may enable dangerous or hardly accessible areas to be surveyed just by flying over them.

- Investigation of other image-based velocimetry techniques: in this work only PIV has been considered; however other techniques such as Particle Tracking Velocimetry (PTV) (Adrian 1991), Space-Time Image Velocimetry (STIV) (Fujita, Watanabe and Tsubaki 2007), optical flow are available and may be considered as well.

Overall, despite the difficulties faced during this work, all the codes are found to be effective in performing AIV and enable a characterization of the time-averaged surface velocity field of the river, without being in contact with the water. However, the results achievable with each AIV approach may be improved if the processing parameters are chosen specifically for it, regardless of the constraints imposed by the comparison in order to be consistent.

Chapter 7

Studies on SfM-based orthorectification using video C

According to the goal of the experiment carried out on river Lambro, the focus of this chapter will be on video orthorectification by using a SfM technique. An example of raw video frame and the relative orthophoto produced from it is shown in Fig. 7.1.

7.1 Orthophotos generation

The requirements for the generation of an orthophoto by using a SfM technique are the External Orientation (EO) of (at least) one camera and a DSM or a 3D model of the same area. If each frame in the video is used to build an orthophoto georeferenced in the same reference system and with the same pixel size in world units, the video shaking is corrected at the same time. To this end, video C was processed by employing Agisoft Metashape as follows:

a) The frames were extracted from the video and imported in Metashape.

b) Values of sensor pixel dimension and focal length, in world unit, were provided as initial values for the Internal Orientation (IO) of the camera.

c) The reference system was defined according to the local reference system of the GCPs, whose coordinates were acquired during the field survey with the MultiStation MS60.

d) A mask over the flow area was defined on each video frame in order to not accept any TPs on the flowing water.

e) The EO of the cameras was estimated on the basis of GCPs collimated on the video frames and on TPs, detected and matched by Metashape with interesting operators. When codec targets had been employed, the GCPs might be automatically detected on the images. This step was crucial because the geometrical accuracy of each orthophoto depends on the accuracy of the EO parameters (i.e. location and orientation angles) estimated for each camera.

f) The IO parameters were optimized by solving again the bundle block adjustment (self-calibration).

g) A 3D mesh model, with the same reference system as the GCPs, was loaded in Metashape.

(a) Raw frame (b) Orthorectified frame

Figure 7.1: Example of (a) raw video frame and (b) orthorectified frame. The railing is still visible in the orthophoto because it was not properly reconstructed in the mesh model due to a lack of oblique photos (see Sec. 3.2.3).

h) An orthophoto was built from each video frame and exported.

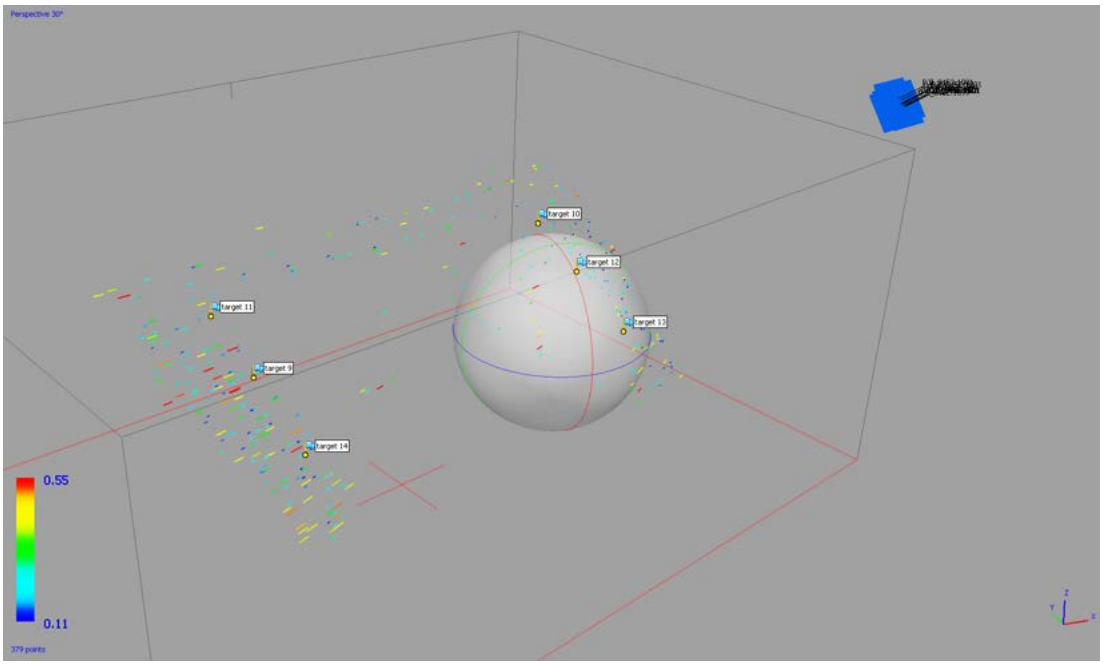
i) By using e.g. MATLAB®, the series of orthophotos were cropped within the same ROI and merged together in a new video file.

All the computation were carried out on a ASUS N552VX laptop with CPU Intel® Core™ i7-6700HQ @ 2.6 GHz, RAM 16 GB, GPU NVIDIA GeForce GTX 950M with 4 GB-DDR3 of dedicated memory.

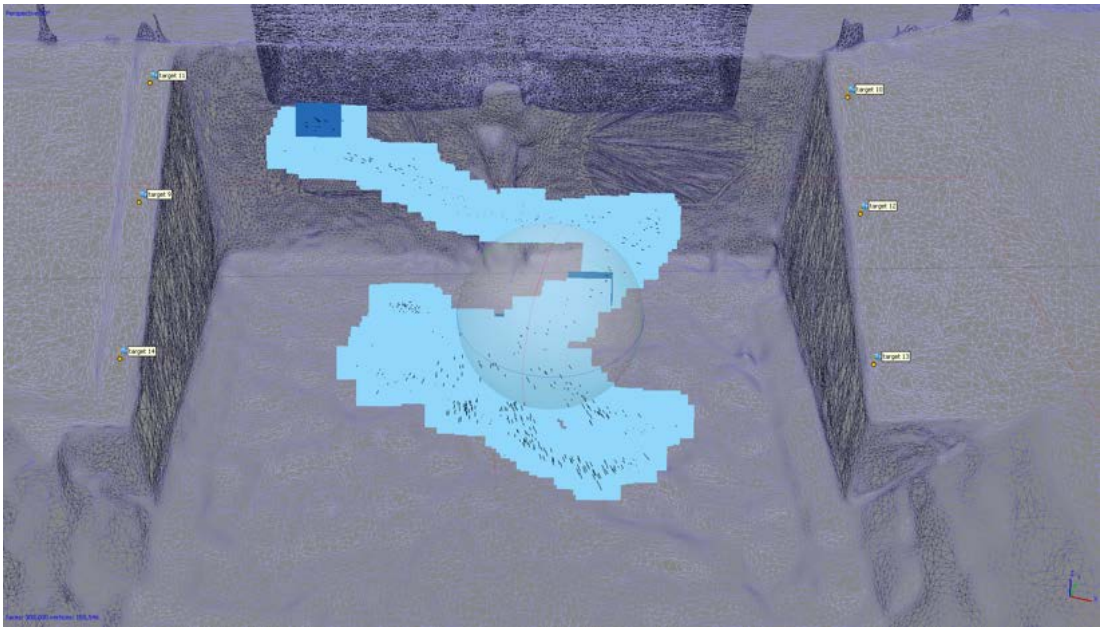
7.2 Orthorectification procedure C1 and C2

In agreement with the two simulations described in Sec. 3.2.4, video C was processed twice with two different procedures. In the first, named C1, the automatic detection of codec targets was used in order to identify the position of the GCPs in all the video frames. This set-up corresponds to the simplified simulation S1. However, it is clear that this is an optimal condition and it can hardly be achieved in real AIV application. In case of larger rivers for which higher flight height is required or of more complex environments, the coded target may not be recognized in all the video frames or may not be detected at all. Therefore, a second procedure, named C2, was carried out in a analogous way as the simulation S2: the GCPs were manually collimated only in a subset of video frames, while the others were oriented on the basis of TPs only.

Concerning case C1, thanks to the low flight height kept by the UAV and to the viewing angle almost parallel to the GCPs plane, all the 6 codec targets were correctly detected by Metashape in every video frame. Therefore, the number of interesting points and TPs to be detected by Metashape in subset of 20 images was limited to 1000 points per image (instead of the default values of 40 000 interesting points and 10 000 TPs). The ill-conditioned geometry of the photogrammetric problem, derived from the extremely small bases between the cameras, caused large variances in the estimated coordinates (Fig. 7.2a). If the base is zero, in fact, the perspective rays passing though the homologous points in two images intersect at infinity. However, the result of simulation S1 reveals that the camera EO can be determined with enough accuracy, even if the geometry of the reconstructed model is not reliable (see Sec. 3.2.4). Subsequently, the remaining 879 frames were oriented with the tool *Align Selected Cameras*, implemented in Metashape to realign a subset of incorrectly positioned cameras, provided

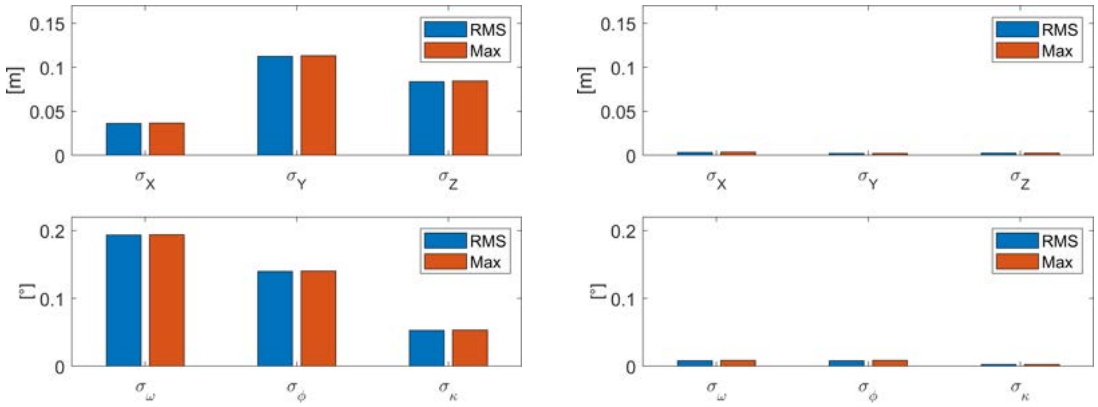


(a) Sparse cloud



(b) Cameras EO

Figure 7.2: Sfm products computed with the C1 approach. (a) Sparse cloud generated by ∼400 TPs by solving the bundle block adjustment with 20 video frames. The colorbar refers to the RMS of standard deviations of the TPs coordinates in the world reference system. TPs with the largest standard deviations were previously removed. (b) Perspective view of the estimated position and orientation of the 899 images with respect to the mesh model. The image is distorted by a perspective effect in order to enhance the different location of the cameras. For a scaled plot of the camera projective centres, one may refer to Fig. 7.4.



(a) C1 (b) C2

Figure 7.3: For each EO parameter, the RMS and the maximum value of the standard deviations of the 899 cameras, estimated by Metashape, are plotted. The coordinates are expressed in the local reference system defined by the GCPs, which is different than that used for the simulations S1 and S2.

that at least 4 GCPs per image are collimated. Hence, each of the 879 video frames was independently oriented on the basis of the 6 visible GCPs. This approach allowed for a significantly short computational time: about ∼30 min for estimating the EO of 899 frames plus ∼10 min for the optimization of the IO parameters. The estimated position and orientation of the 899 images are shown in Fig. 7.2b. Fig. 7.3a displays the RMS and the maximum value of standard deviation of the 899 cameras for each EO parameter. Those are comparable to the values simulated in S1 (see Sec. 3.2.4). The similar values of RMS and maximum of standard deviation for all the EO parameters denotes that all the cameras were oriented with a similar accuracy.

With the procedure C2, the 6 GCPs were manually collimated on a subset of 20 video frames (taking 1 in every 50 frames). In order to have consistent results, the IO estimated in C1 was exported and loaded again in C2, as fixed constrain. Those were first oriented by using the Metashape default values for the matching point limits (*high accuracy*, 40 000 interesting points and 10 000 TPs). In the others 879 frames, TPs only were detected with interesting operators. The two-steps procedure followed allows the tie points detected on the first subset of images to be kept in the second step and to be matched with the interesting points found in the remaining images, limiting the computational time to ∼3 h plus ∼20 min for the optimization. Nonetheless, the alignment in C2 is still 5 times slower than that of the approach C1 (∼40 min). Fig. 7.3b presents the RMS and the maximum values of standard deviation for each EO parameter estimated with C2. These are one order of magnitude smaller than the values obtained in C1 and than the simulated ones (see Sec. 3.2.4). This is probably due to a misestimation of the covariance matrix of the parameters $C_{\hat{x}\hat{x}}$ in the least-square adjustment. In fact, it depends on the *a-posteriori* scaling coefficient $\hat{\sigma_0}^2$ as follows:

$$C_{\hat{x}\hat{x}} = \hat{\sigma_0}^2 N^{-1}; \qquad \hat{\sigma_0}^2 = \frac{\underline{\hat{v}}^T Q^{-1} \underline{\hat{v}}}{n-m}$$

where N is the normal matrix, which depends on the acquisition geometry only, $\underline{\hat{v}}$ is the vector of the residuals and Q is the cofactor matrix. The denominator $n-m$ expresses the redundancy of the adjustment, as n is the number of equations and m of unknowns.

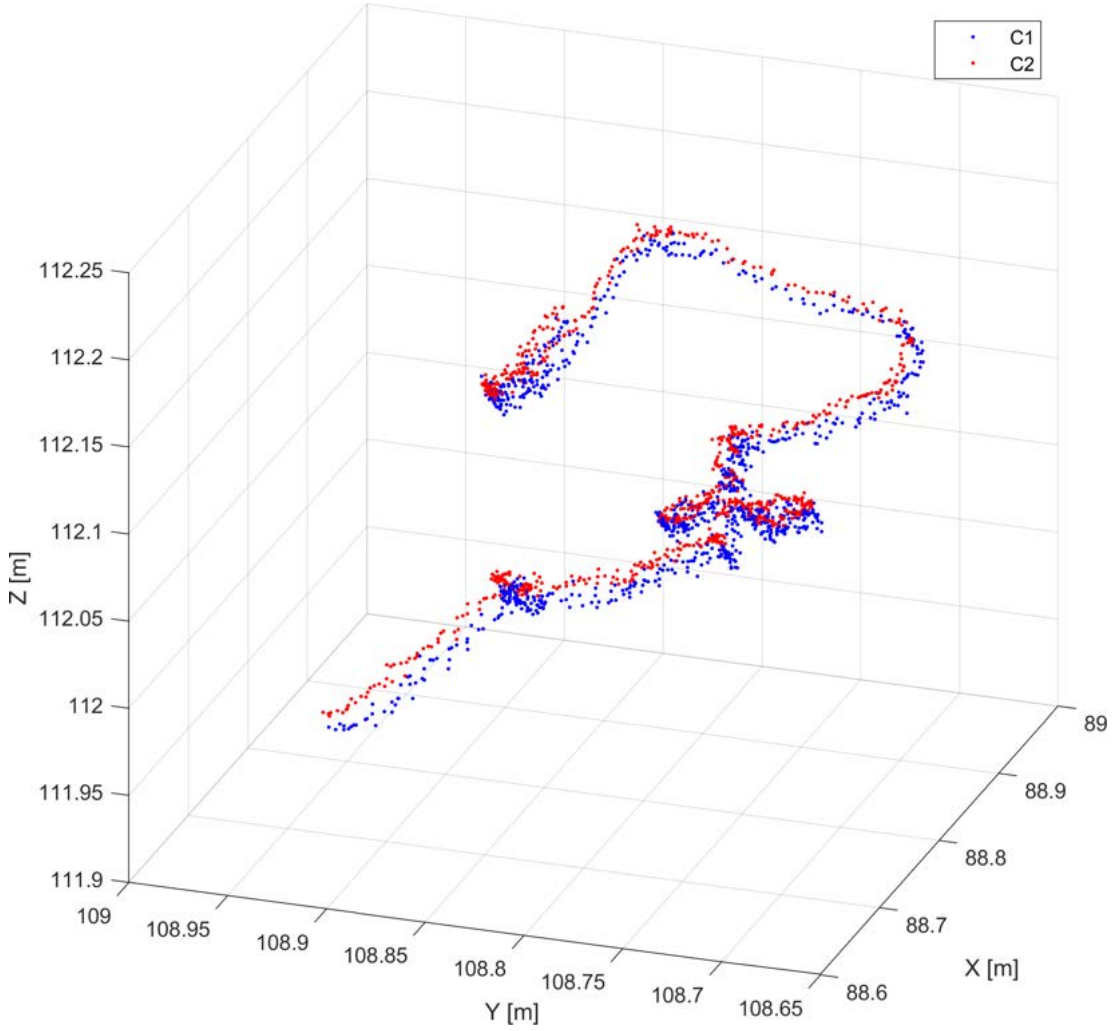


Figure 7.4: Centres of projection of the cameras estimated in C1 (blue) and C2 (red).

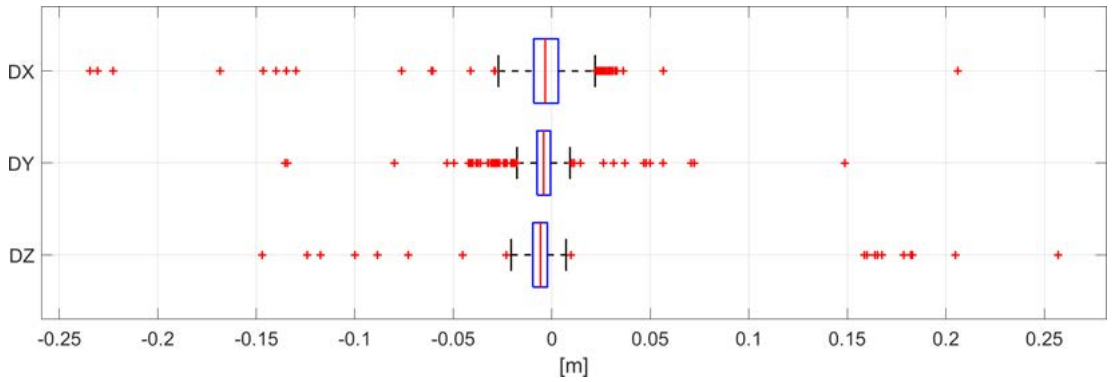


Figure 7.5: Boxplots of the differences DX, DY, DZ of the coordinates of the projection centres of each camera estimated with the two approaches, computed as $\xi_{C1} - \xi_{C2}$, where ξ is a generic coordinate. The red line marks the median of the differences in each coordinate. The left and right edges of the box (blue lines) indicate respectively the 25^{th} and 75^{th} percentiles. The whiskers extend to the most extreme data points not considered outliers. The outliers are individually plotted using the '+' symbol.

In the estimation of $\hat{\sigma_0}^2$, the residuals $\underline{\hat{v}}$ are always very small, in both the cases C1 and C2, because of the ill-conditioned geometry of the problem due to the small baselines between the projective centres. In C1, however, each image was oriented on the basis of 6 GCPs automatically detected on it and only a small number of TPs (just ∼400) was identified. On the contrary, in C2 ∼9000 TPs were used to orient the 899 video frames. The high number of TPs, all with small and homogeneous residuals, leads to an underestimation of $\hat{\sigma_0}^2$ and, hence, of the covariance matrix $C_{\hat{x}\hat{x}}$. Consequently, the standard deviations of the EO parameters estimated by Metashape in C2 are not reliable.

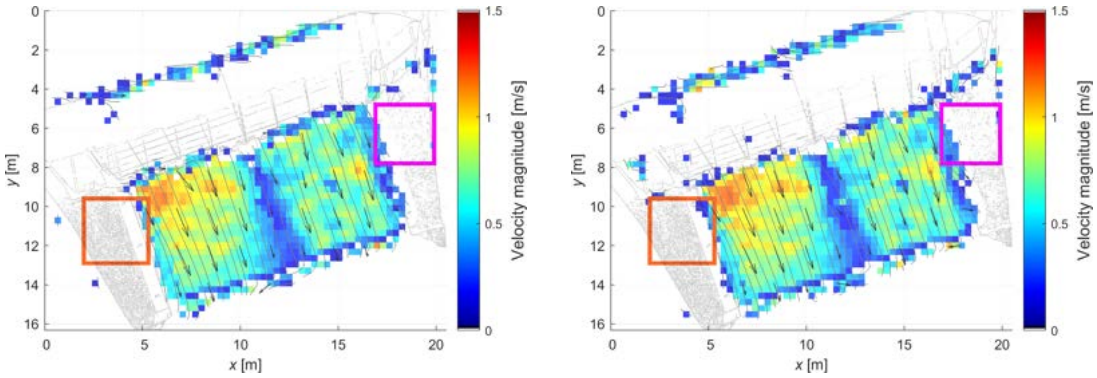
The positions of the camera centres of projections estimated in C1 and C2 are plotted in Fig. 7.4. A little shift between the two group of aligned cameras is visible. However, the results are consistent. In order to quantify the shift, the coordinates of each camera estimated with the approaches C2 were subtracted to those computed with C1. The boxplots of Fig. 7.5 reveal that the medians of the differences are close to zero, with absolute values in the order magnitude of the millimetre. The 25^{th} and 75^{th} percentiles and the whiskers of the boxplots differ from the medians less than 0.05 m, underling that the relative orientation of the cameras in two blocks is consistent. This indicates that both the approaches C1 and C2 may be suitable to estimate the EO of the cameras, despite the non reliable variances of the camera EO parameters obtained with C2.

7.3 Orthorectification accuracy assessment

Finally, two sequences of 899 orthophotos with a GSD of 0.01 m/px were built with both the approaches C1 and C2 and merged again in a video file. From visual inspection, both the rectified videos do not present any significant shakes and the results appear to be satisfactory for an AIV application. Therefore, BASESURV was employed to perform PIV on the rectified videos and the results, in terms of time-averaged surface velocity field, are displayed in Fig. 7.6. No significant differences are appreciable in the median velocity fields computed with C1 and C2.

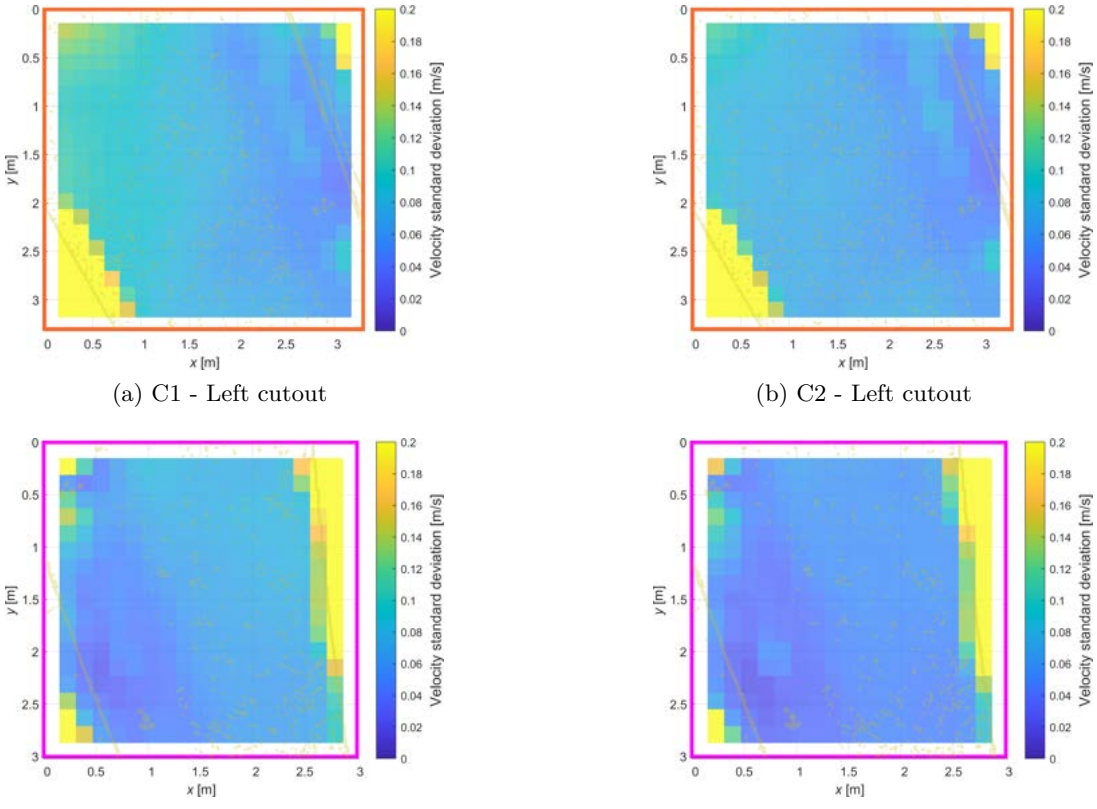
In order to evaluate quantitatively the effect of the orthorectification errors, PIV was applied over non-moving areas, where the time-series of the velocity was supposed to be always zero. Two cutouts were considered, respectively on the Left hand (orange bounding box) and Right hand (magenta bounding box) side of the region of interest (Fig. 7.6). The time-averaged velocity at each grid point within the two cutouts was found to be close to zero for both the approaches C1 and C2, suggesting the orthorectification was not affected by any systematic error. Moreover, Fig. 7.7 displays the standard deviation of the velocity time-series at each grid point. In both the Left-hand and Right-hand cutouts, the values of standard deviations obtained with the approach C1 and C2 are comparable and everywhere less than 0.15 m/s. The residual rectification error so estimated can be considered as a background noise for the velocities estimated by PIV. This may be acceptable for the velocities of the main currents of river Lambro, below the two gates of the Fornacetta dam and respectively on the left and on the right of the bridge pile. However, an error in orthorectification in the order of 0.15 m/s, may be too large to resolve the instantaneous velocities in proximity to the recirculation vortexes immediately downstream of the pile, where the velocities are lower (Fig.7.6).

Overall, both the SfM-based approaches tested in this study for video rectification are promising. They allow video stabilization and rectification to be performed at the same time by estimating one transformation between the world reference system



(a) C1 (b) C2

Figure 7.6: Time-averaged surface velocity field computed with the approaches: (a) C1; (b) C2. The two cutout areas, in which PIV is employed for the orthorectification error assessment (see Fig. 7.7), are marked with coloured bounding boxes.



(a) C1 - Left cutout (b) C2 - Left cutout

(c) C1 - Right cutout (d) C2 - Right cutout

Figure 7.7: Results of PIV computation over the Left and Right hand cutout for both the approaches C1 and C2. The standard deviation of the velocity vector time-series at each grid point is plotted with a scale of colour. In order from top-right up to bottom left: (a) Left hand cutout computed with the approach C1; (b) Left hand cutout computed with the approach C2; (c) Right hand cutout computed with the approach C1; (d) Right hand cutout computed with the approach C2.

and each video frame based on the collinearity equations. They also overcome the orthorectification issue in BASESURV, which requires all the GCPs to lay on the same plane as the water surface. However, this technique involves the use of a commercial software such as Agisoft Metashape, which may not be available for a low-cost AIV application.

The possibility to automatically detect the GCPs in every video frame and to match them with their world coordinates might be the most interesting step forward compared to the works of Detert, Huber and Weitbrecht (2016) and Detert, Johnson and Weitbrecht 2017. This allows for a significant computational time reduction. Moreover, the automatic detection of the targets may also be performed without using Metashape, but implementing recognition of the same target in consecutive video frames e.g. by using a template matching technique such as that implemented in the tool ImGRAFT (Messerli and Grinsted 2015). To this end, specific software may be developed e.g. in MATLAB® with the Computer Vision Toolbox™ or in Python/C++ with OpenCV.

Chapter 8

Summary and conclusions

The goal of this work was to carry out a comparison and evaluation of the three AIV approaches BASESURV, Fudaa-LSPIV and RIVeR. For the evaluation, two nadiral videos, called video A and video B, were acquired on river Limmat at the gauging station Zürich Unterhard (Switzerland), using a low-cost UAV DJI Phantom 4 Pro. Video A was taken on 22/10/2019 under stable flow conditions, with discharge comparable to the daily average for October, and the river was seeded with biodegradable corn starch chips. Video B was acquired was acquired on 10/10/2019, during a 1-year return period flood and no tracer particles were used. To validate the results obtained from video A, the surface velocity profile along a cross-section was measured on the same day with a hydrometric impeller by the Swiss Federal Office for the Environment (FOEN). In the centre of the 45 m wide river, i.e. after 5 m from each riparian side, relative percentage errors in the estimated velocity of $4\%, 5\%$ and 36% were obtained respectively with BASESURV, Fudaa-LSPIV (considering the median surface velocity field) and RIVeR. Near the riparian sides, i.e. within 5 m, higher errors were obtained with all the approaches: respectively, 56%, 71%, 47%. The surface velocity fields obtained from the two videos were used to estimate the river discharge, together with the bathymetry and water level information provided by FOEN. That was then compared to the one measured by the gauging station to have a further rough assessment of the estimated surface velocity.

According to the results obtained from the two datasets, BASESURV is found to be the most accurate, complete and flexible AIV approach. It is the only software that implements smart and effective post-processing filters based on the time-series of vectors at each grid point. Moreover, it is the only one performing the video rectification by estimating one different geometric transformation for each video frame, even if it is limited to a 2D-to-2D projective transformation. However, neither a GUI nor a documentation are yet available. This implies that the user must go to the code in depth and know the programming language in order to handle the software. It is also the most demanding software in terms of computational time. Therefore, BASESURV may be the best approach for research purposes.

Fudaa-LSPIV is a well developed AIV software application, with a complete and user friendly GUI and good documentation. The whole AIV process in Fudaa-LSPIV is almost 1.7 times faster than that of BASESURV. The workflow is straight forward and it can be fully performed inside the software itself, from the video stabilization up to the time-averaged velocity field computation and, if needed, the discharge estimation. However, the source code is closed and some useful tools are missing, such as image pre-processing or time-based post-processing filters. Fudaa-LSPIV may be the best alternative for professional applications.

Also for RIVeR, a GUI developed in MATLAB® and software documentation are available, even if it does not provide almost any information about the algorithms implemented. However, the source code remains closed. The main strength of RIVeR is to perform the rectification of the velocity vectors only. This is found to be very effective in the reduction of computational time, especially when video stabilization is not needed, as for a fixed camera. The processing workflow of RIVeR, in fact, is almost 6 times faster than that of BASESURV, if the video stabilization is not performed, and even 10 times if also frame undistortion is not applied. Therefore, RIVeR may be suitable for fast processing as well as for real time monitoring and alarm systems.

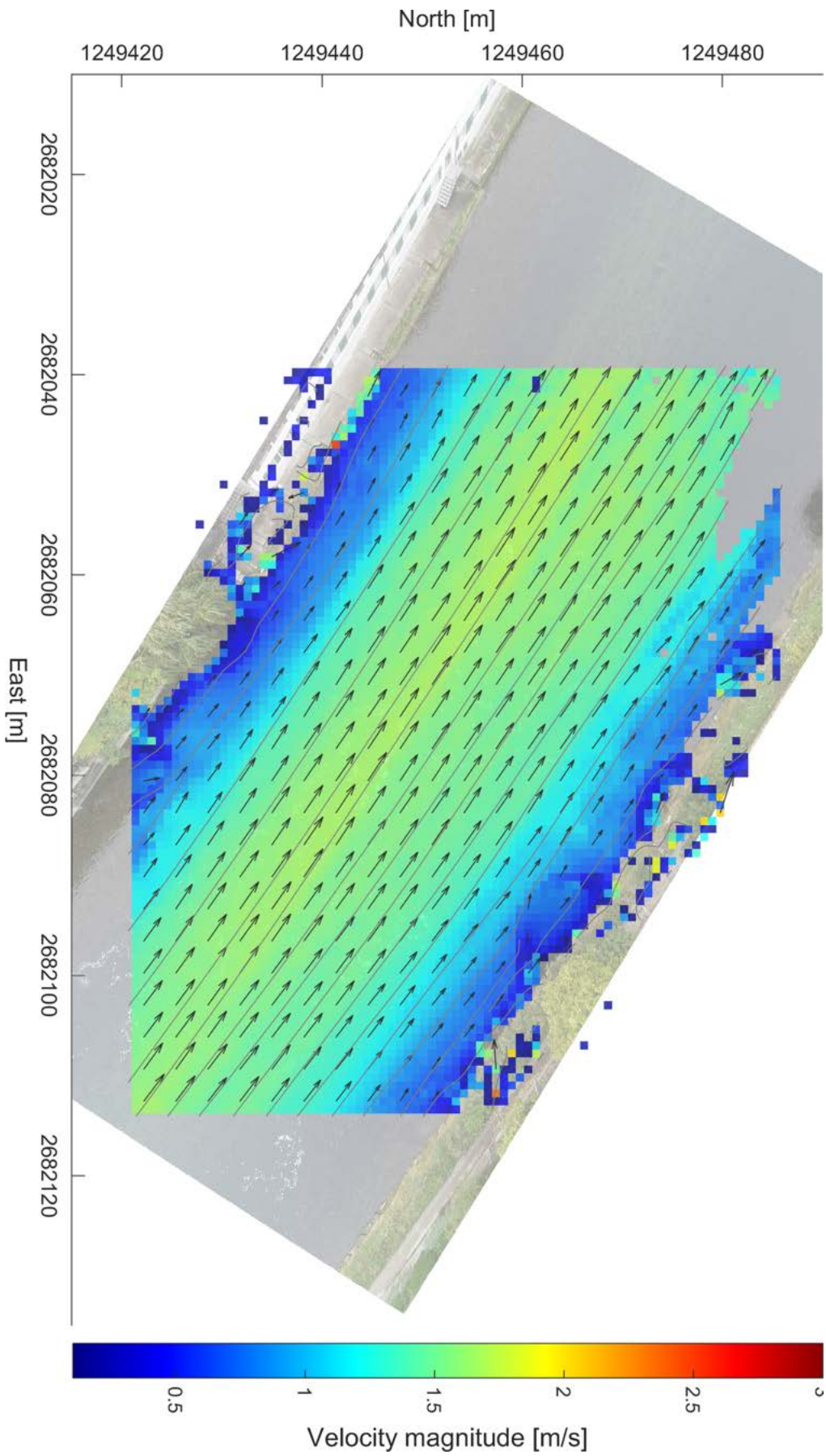
Overall, all the AIV codes are found to be effective in performing AIV in riverine environments. Despite the difficulties faced during this work, each enables a characterization of the time-averaged surface velocity field without being in contact with the water, but analysing videos recorded by a camera mounted on low-cost UAVs.

In addition to the comparison of the AIV approaches, a further experiment aimed at testing an orthorectification approach based on a Structure from Motion technique was set up on river Lambro (Italy). This procedure requires the reconstruction of a 3D model of the study area first. For each video frame to be used for velocimetry, the external orientation of the camera is estimated and an orthophoto is built by solving the collinearity equations. The sequences of orthophotos may be then analysed by PIV. This technique is found to be promising because it allows the rectification to be performed by estimating one 3D-to-2D geometric transformation for each video frame, overcoming a limit of BASESURV. Moreover, the automatic detection of the GCPs in all the video frames may open to the development of new orthorectification techniques, more efficient and accurate than those implemented in the considered AIV approaches.

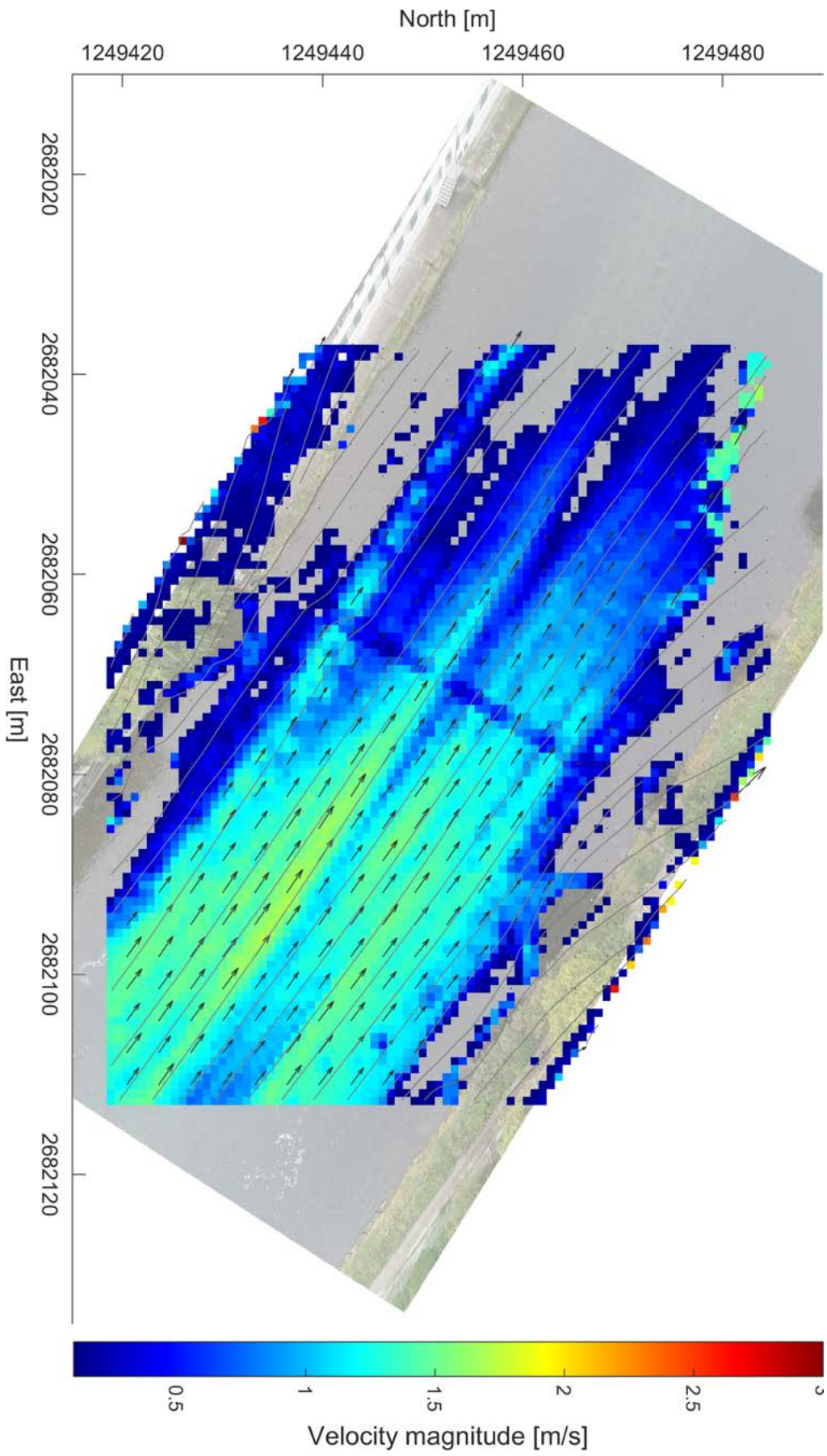
Future developments in AIV may involve a systematic investigation of the accuracy of this technique in such a way as to build a framework to provide an estimation of the error related the computed surface velocity. This would expand the potential applications of AIV in both professional and scientific sectors. Moreover, the possibility to perform AIV analysis without neither GCPs nor seeding particles is an interesting outlook that may be explored in further studies. This would allow dangerous or hardly accessible areas to be surveyed, just by flying over them with UAVs. Resources and time may be also saved, both in the field and during the post-processing phase.

Appendix A

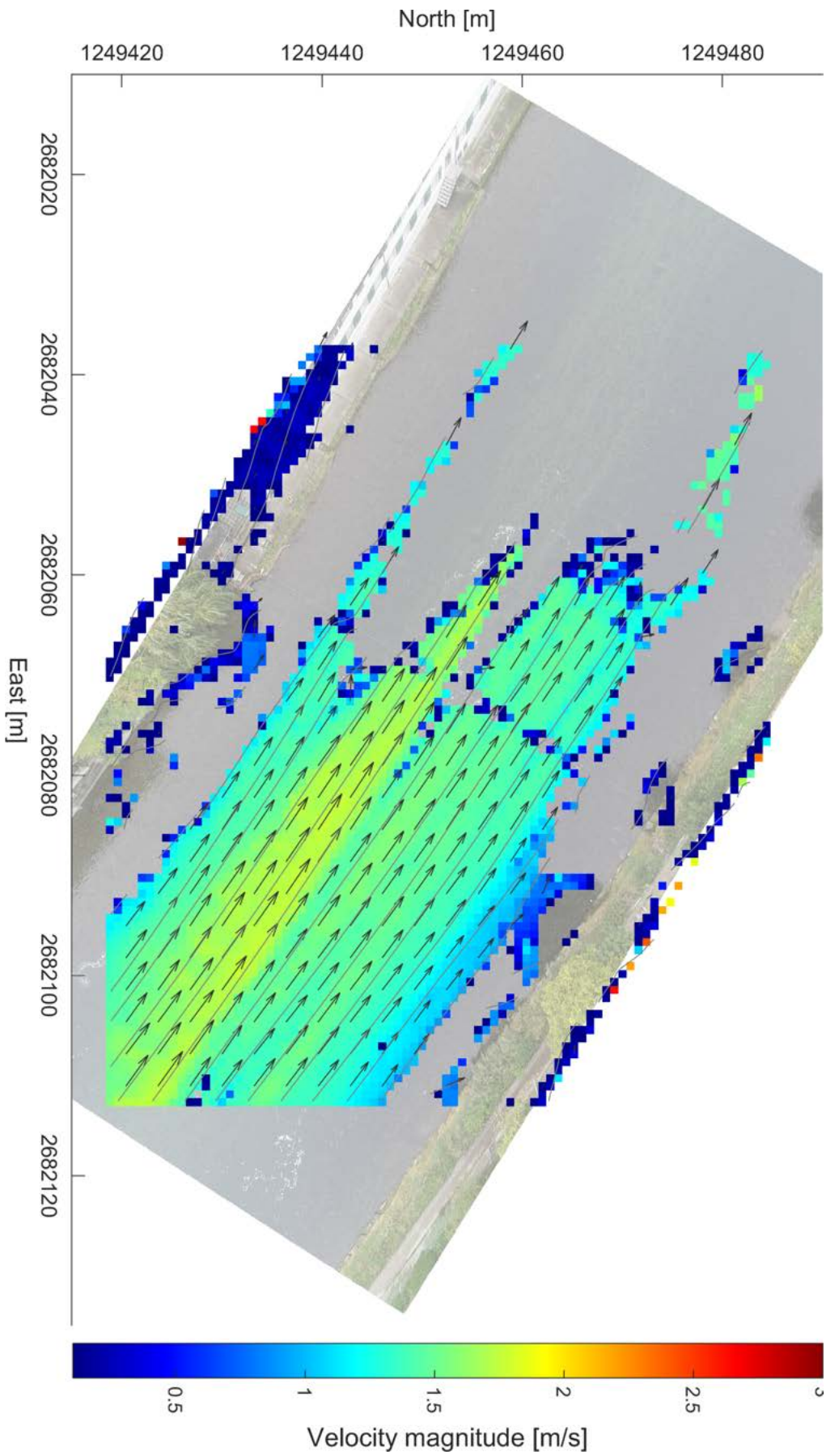
Time-averaged surface velocity fields



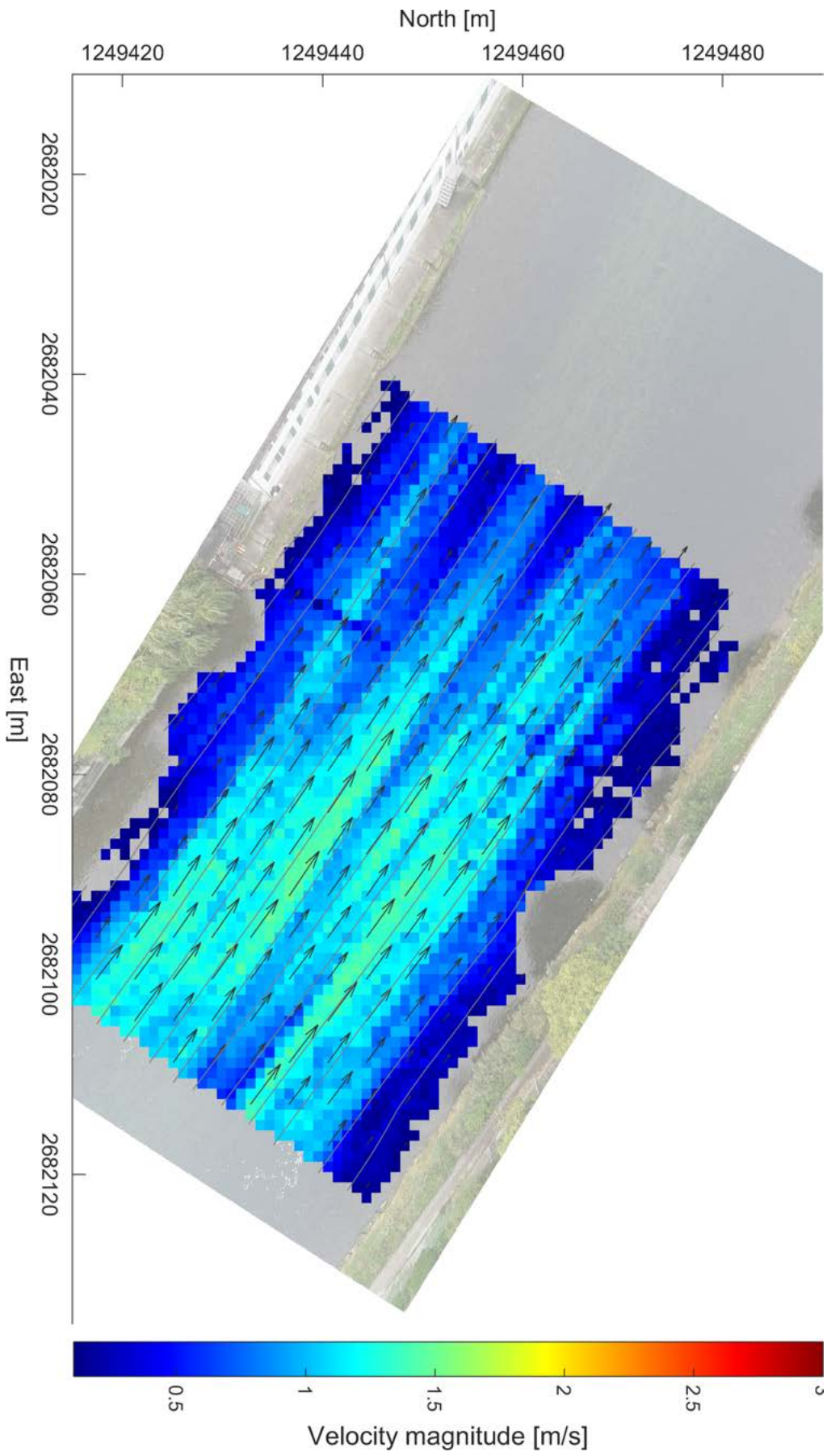
Video A: BASESURV median surface velocity field.



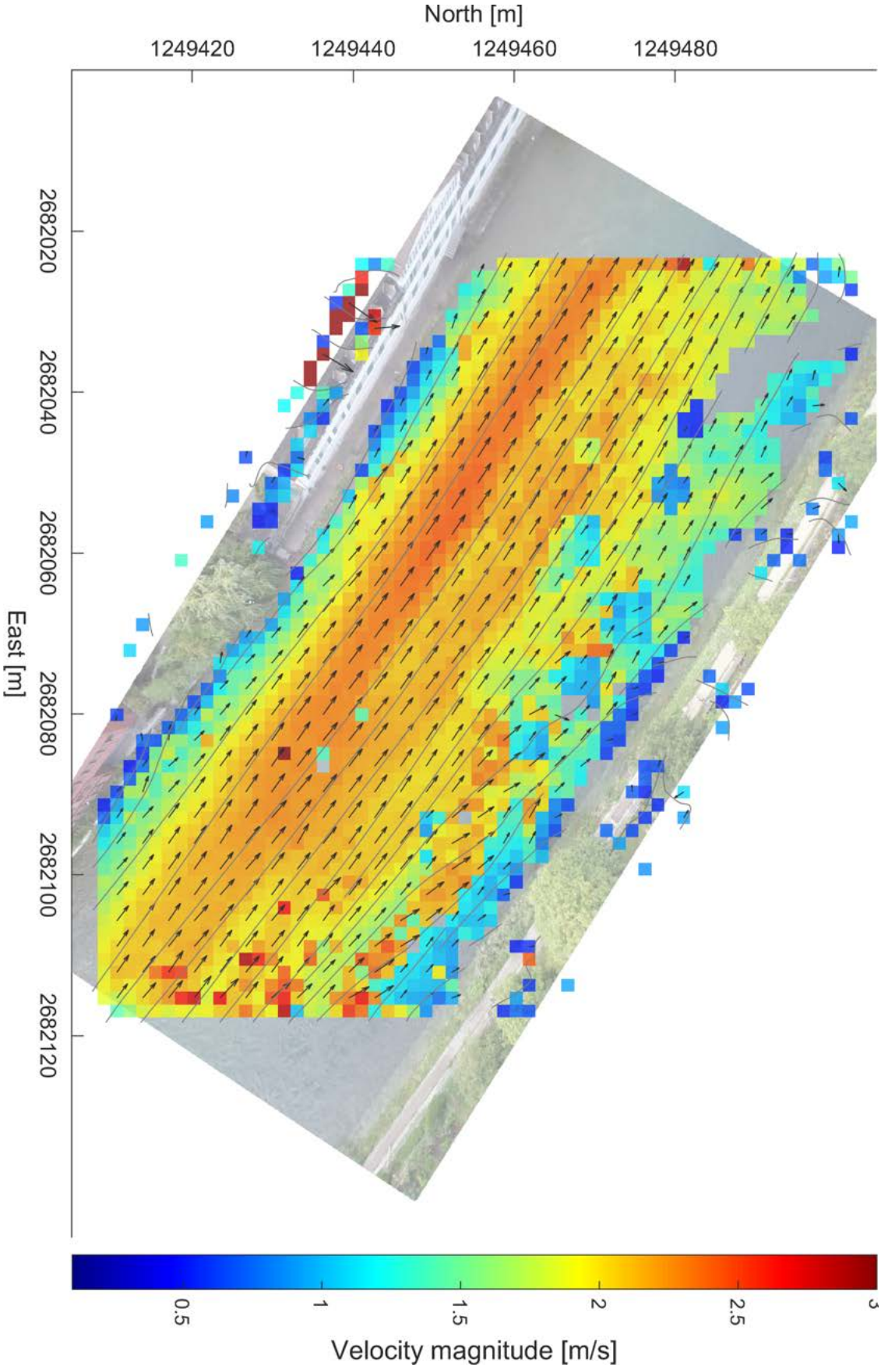
Video A: Fudaa-LSPIV average surface velocity field.



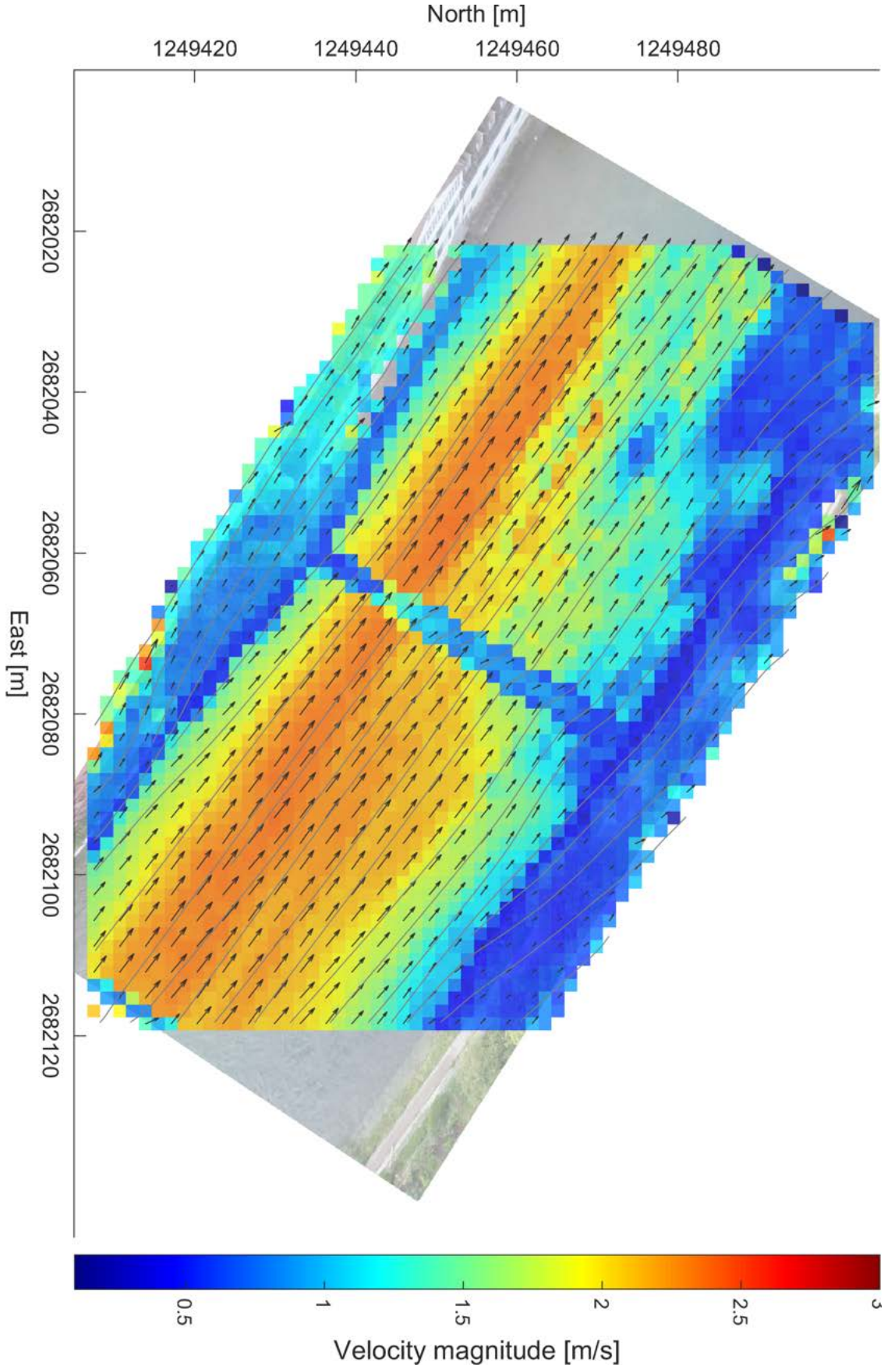
Video A: Fudaa-LSPIV median surface velocity field.



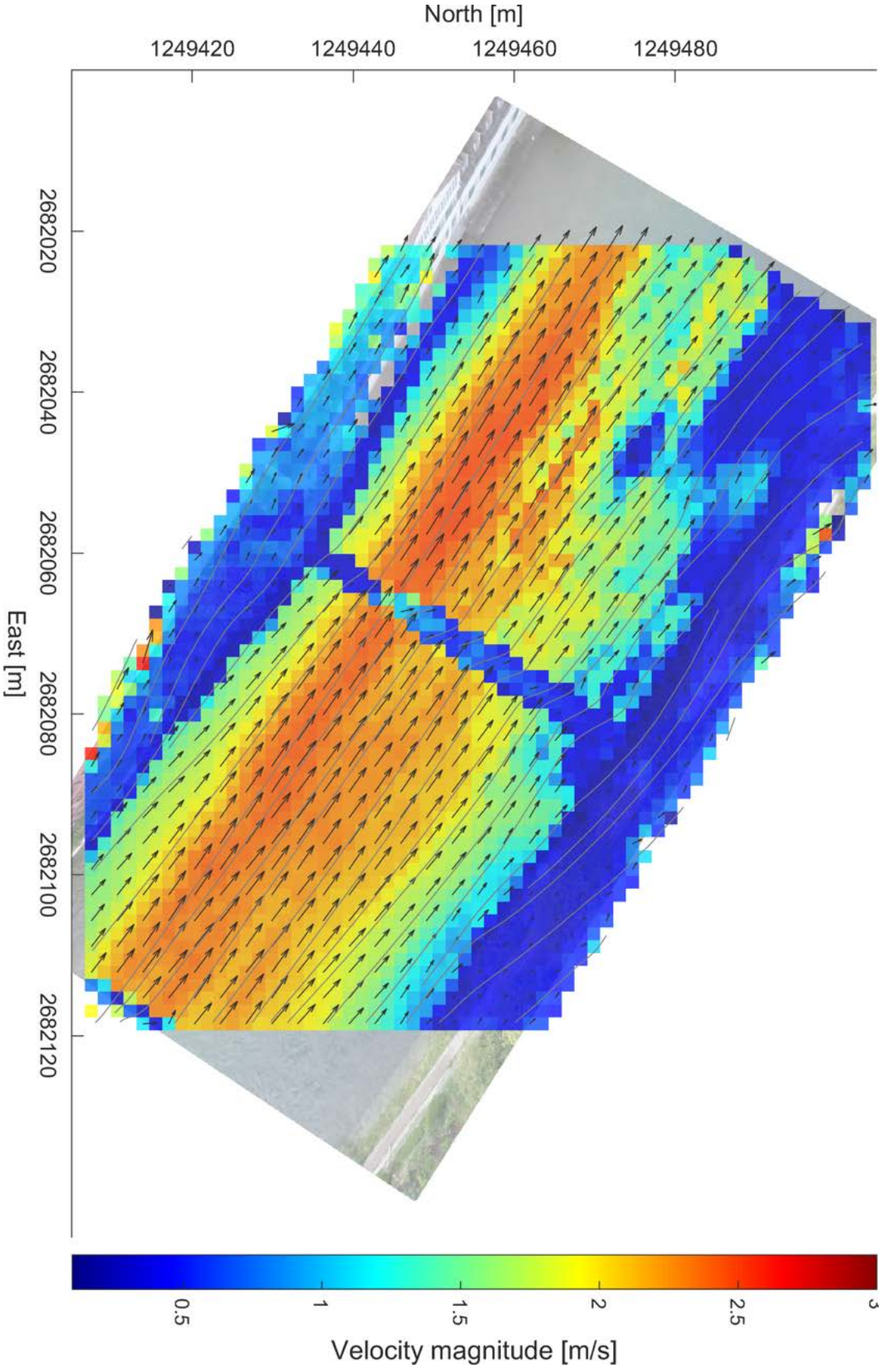
Video A: RIVeR average surface velocity field.



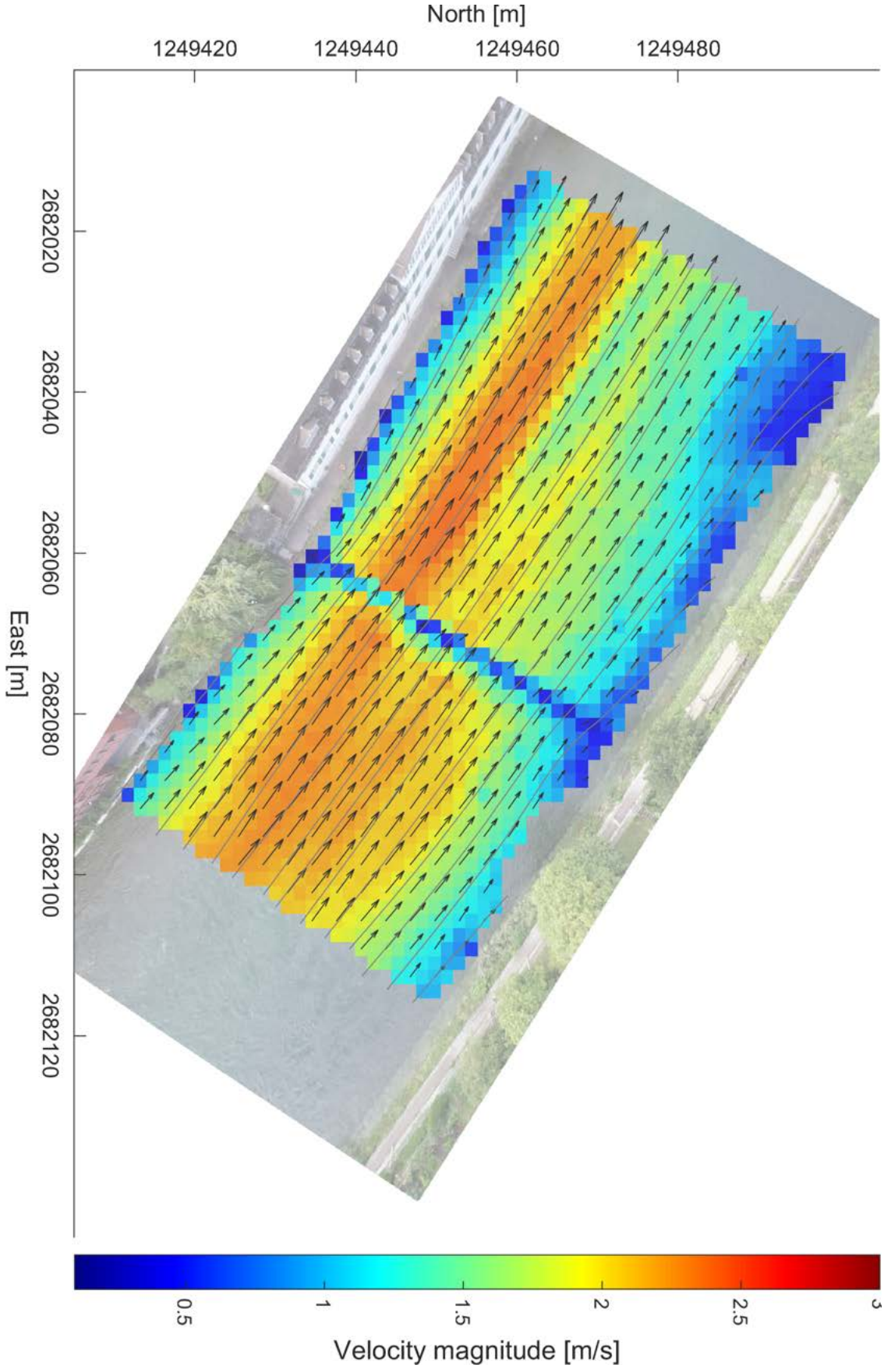
Video B: BASESURV median surface velocity field, PIV time interval $\Delta t = 0.042$ s.



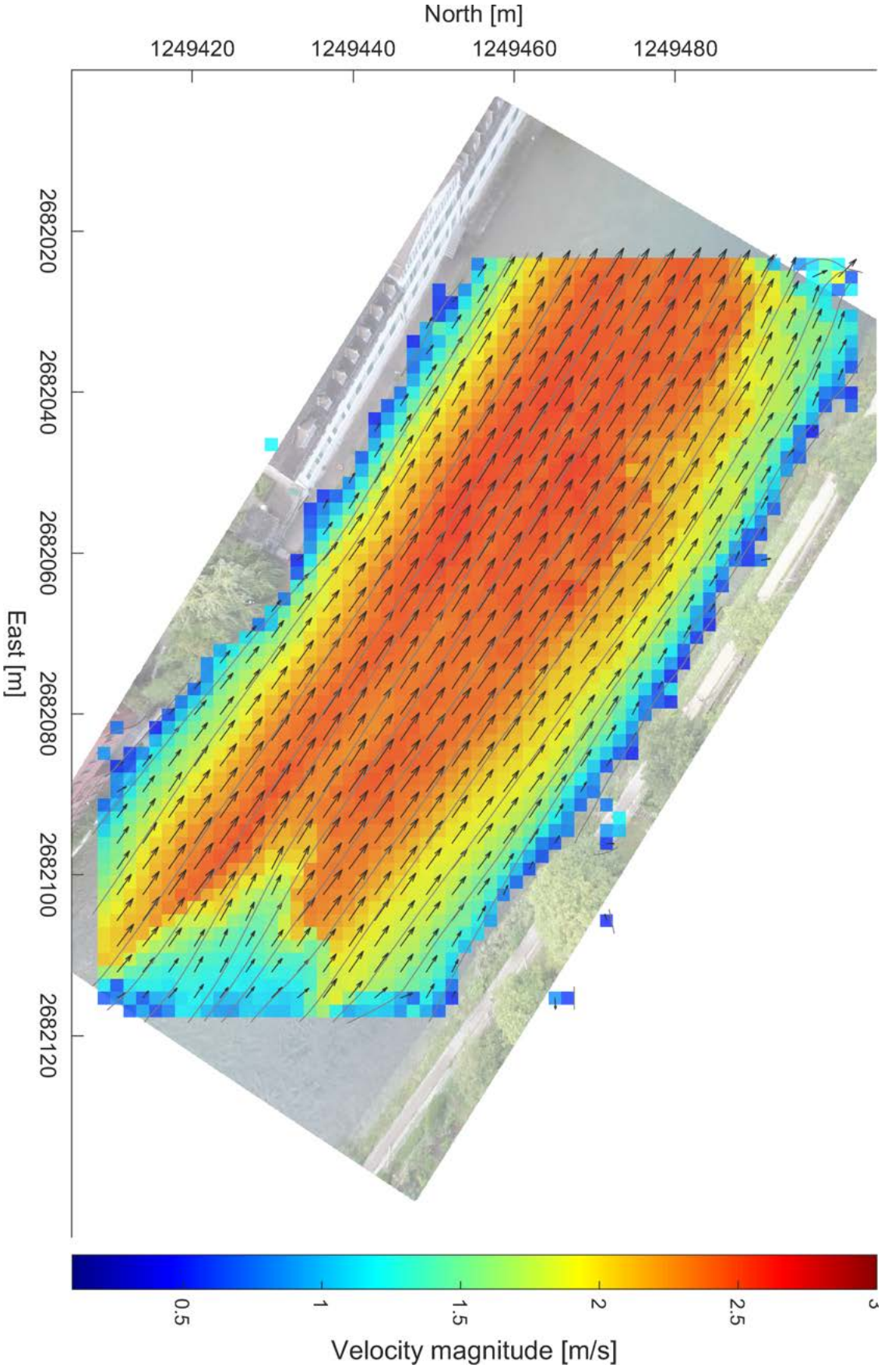
Video B: Fudaa-LSPIV average surface velocity field, PIV time interval $\Delta t = 0.042$ s.



Video B: Fudaa-LSPIV median surface velocity field, PIV time interval $\Delta t = 0.042$ s.



Video B: RIVeR average surface velocity field, PIV time interval $\Delta t = 0.042$ s.



Video B: BASESURV median surface velocity field, PIV time interval $\Delta t = 0.501$ s.

Appendix B

Comments from the Fudaa-LSPIV developers team

The report is well written, and the software comparison is meticulous, so the conclusions are very interesting for all of us. Thanks for taking the time of identifying the potential developments for Fudaa-LSPIV, it will help our future works.

Jodeau Magali

The report looks very good. Thanks for sharing it.

Jerome Le Coz & Alexandre Hauet

Appendix C

Comments from the RIVeR developers team

I found the study very interesting and objective. I think many of the future improvements and new features in the next version of RIVeR will be based on your conclusions. It would be interesting to go further and analyze the videos from the oblique views that you mentioned and also with different approaches like STIV or KTV-IV. Thank you for considering RIVeR in your study.

Antoine Patalano

Appendix D

Comments from the BASESURV developer

*Mr. Francesco Ioli (*03.09.1995) was writing his Master's thesis as an exchange-student at our institute from September 2019 to January 2020. In collaboration with Politecnico di Milano (Italy, Prof. L. Pinto) I was the supervisor to his Master´s thesis on "Evaluation of Airborne Image Velocimetry (AIV) Approaches".*

He provided a comprehensive overview over available AIV approaches and software tools and evaluated them very carefully by own measurement data. Mr. Francesco Ioli's work was a major contribution in the research field of river bed surveying. He achieved more than is usually expected from students during a Master's thesis and he was able to work with innovative high-end technology. He used photogrammetric 3D evaluation methods, which were taken by a lightweight camera mounted to a low-cost drone. Mr. Francesco Ioli's thesis demonstrated the practicability of river's surface velocity surveys without expensive software and hardware.

Mr. Francesco Ioli worked very efficiently, his work attitude was excellent and it was always a scientific-engineering pleasure to work with him. He handled the hardware (camera and drone) as well as the software (mainly MATLAB and Agisoft Metashape) very well. After a very brief instruction he worked with it quite professionally. Furthermore, he independently got in touch with scientists from France, Norway and Argentina to exchange and compare research results.

As the consequence of his great scientific work he contributed to a publication with partial results of his Master´s thesis at the "ISPRS conference" (June 2020, Nice, France).

I wish Mr. Francesco Ioli all the best for his future and I give him the best possible recommendations for doing a PhD, being convinced that he will succeed.

Martin Detert
VAW, ETH Zurich

Bibliography

Adrian, R.J. (1991). 'Particle-Imaging Techniques for Experimental Fluid Mechanics'. In: *Annual Review of Fluid Mechanics* 23.1, pp. 261–304. DOI: 10.1146/annurev.fl.23.010191.001401.

— (2005). 'Twenty years of particle image velocimetry'. In: *Experiments in Fluids* 39.2, pp. 159–169. DOI: 10.1007/s00348-005-0991-7.

Alcantarilla, P.F., A. Bartoli and A.J. Davison (2012). 'KAZE features'. In: *Lecture Notes in Computer Science*. Vol. 7577. 6, pp. 214–227. DOI: 10.1007/978-3-642-33783-3_16.

ARPA Lombardia (2020). *Hydrological data*. URL: http://idro.arpalombardia.it/ (visited on 05/03/2020).

Bay, H. et al. (2008). 'Speeded-Up Robust Features (SURF)'. In: *Computer Vision and Image Understanding* 110.3, pp. 346–359. DOI: 10.1016/j.cviu.2007.09.014.

Benetazzo, A., M. Gamba and F. Barbariol (2017). 'Unseeded large scale PIV measurements corrected for the capillary-gravity wave dynamics'. In: *Rendiconti Lincei* 28.2, pp. 393–404. DOI: 10.1007/s12210-017-0606-2.

Brown, D (1971). 'Close-Range Camera Calibration'. In: *Photogrammetric Engineering* 37, pp. 855–866.

Cleveland, W.S. (1979). 'Robust Locally Weighted Regression and Smoothing Scatterplots'. In: *Journal of the American Statistical Association* 74.368, pp. 829–836. DOI: 10.1080/01621459.1979.10481038.

Detert, M., L. Cao and I. Albayrak (2019). 'Airborne Image Velocimetry Measurements at the Hydropower Plant Schiffmühle on Limmat River, Switzerland'. In: *Proceedings of the 2nd International Symposium and Exhibition on Hydro-Environment Sensors and Software, HydroSenSoft 2019*, pp. 211–217. DOI: 10.3929/ethz-b-000341626.

Detert, M., F. Huber and V. Weitbrecht (2016). 'Unmanned aerial vehicle-based surface PIV experiments at Surb Creek'. In: *River Flow 2016*, pp. 563–568. DOI: 10.1201/9781315644479-90.

Detert, M., E.D. Johnson and V. Weitbrecht (2017). 'Proof-of-concept for low-cost and non-contact synoptic airborne river flow measurements'. In: *International Journal of Remote Sensing* 38.8-10, pp. 2780–2807. DOI: 10.1080/01431161.2017.1294782.

Detert, M. and V. Weitbrecht (2015). 'A low-cost airborne velocimetry system: proof of concept'. In: *Journal of Hydraulic Research* 53.4, pp. 532–539. DOI: 10.1080/00221686.2015.1054322.

DJI (2020a). *Mavic 2 Pro specifications*. URL: https://www.dji.com/mavic-2/info#specs (visited on 05/03/2020).

— (2020b). *Phantom 4 Pro specifications*. URL: https://www.dji.com/phantom-4-pro/info#specs (visited on 31/01/2020).

Forlani, G. (1986). 'Sperimentazione del nuovo programma CALGE dell'ITM'. In: *Bollettino della Società italiana di topografia e fotogrammetria* 2, pp. 63–72.

Fujita, I. and T. Hino (2003). 'Unseeded and seeded PIV measurements of river flows videotaped from a helicopter'. In: *Journal of Visualization* 6.3, pp. 245–252. DOI: 10.1007/BF03181465.

Fujita, I. and Y. Kunita (2011). 'Application of aerial LSPIV to the 2002 flood of the Yodo River using a helicopter mounted high density video camera'. In: *Journal of Hydro-environment Research* 5.4, pp. 323–331. DOI: 10.1016/j.jher.2011.05.003.

Fujita, I., Y. Notoya and M. Shimono (2015). 'Development of UAV-based river surface velocity measurement by STIV based on high-accurate image stabilization techniques'. In: *E-proceedings of the 36th IAHR World Congress*. Vol. 28.

Fujita, I., H. Watanabe and R. Tsubaki (2007). 'Development of a non-intrusive and efficient flow monitoring technique: The space-time image velocimetry (STIV)'. In: *International Journal of River Basin Management* 5.2, pp. 105–114. DOI: 10.1080/15715124.2007.9635310.

Hauet, A. et al. (2008). 'Experimental system for real-time discharge estimation using an image-based method'. In: *Journal of Hydrologic Engineering* 13.2, pp. 105–110. DOI: 10.1061/(ASCE)1084-0699(2008)13:2(105).

Horner, I. et al. (2018). 'Impact of stage measurement errors on streamflow uncertainty'. In: *Water Resources Research* 54, pp. 1952–1976. DOI: 10.1002/2017WR022039.

Jodeau, M. et al. (2019). *Fudaa-LSPIV Version 1.7.1 User Manual*. URL: https://forge.irstea.fr/attachments/download/2865/User%20Manual%20Fudaa-LSPIV%201.7.pdf.

Keane, R. and R.J. Adrian (1992). 'Theory of cross-correlation analysis of PIV images'. In: *Applied Scientific Research* 49.3, pp. 191–215. DOI: 10.1007/bf00384623.

Le Coz, J., A. Hauet et al. (2010). 'Performance of image-based velocimetry (LSPIV) applied to flash-flood discharge measurements in Mediterranean rivers'. In: *Journal of Hydrology* 394.1-2, pp. 42–52. DOI: 10.1016/j.jhydrol.2010.05.049.

Le Coz, J., M. Jodeau et al. (2014). 'Image-based velocity and discharge measurements in field and laboratory river engineering studies using the free Fudaa-LSPIV software'. In: *River Flow 2014*, pp. 1961–1967. DOI: 10.1201/b17133-262.

McLachlan, G.J. and D. Peel (2000). *Finite mixture models*. Wiley Series in Probability and Statistics. Wiley. DOI: 10.1002/0471721182.

Messerli, A. and A. Grinsted (2015). 'Image georectification and feature tracking toolbox: ImGRAFT'. In: *Geoscientific Instrumentation, Methods and Data Systems* 4.1, pp. 23–34. DOI: 10.5194/gi-4-23-2015.

Muste, M., I. Fujita and A. Hauet (2008). 'Large-scale particle image velocimetry for measurements in riverine environments'. In: *Water Resources Research* 44.4. DOI: 10.1029/2008wr006950.

Nobach, H., N. Damaschke and C. Tropea (2005). 'High-precision sub-pixel interpolation in particle image velocimetry image processing'. In: 39.2, pp. 299–304. DOI: 10.1007/s00348-005-0999-z.

Nobach, H. and M. Honkanen (2005). 'Two-dimensional Gaussian regression for subpixel displacement estimation in particle image velocimetry or particle position estimation in particle tracking velocimetry'. In: *Experiments in Fluids* 38.4, pp. 511–515. DOI: 10.1007/s00348-005-0942-3.

Patalano, A., C. García and A. Rodriguez (2017). 'Rectification of Image Velocity Results (RIVeR): A simple and user-friendly toolbox for large scale water surface Particle Image Velocimetry (PIV) and Particle Tracking Velocimetry (PTV)'. In: *Computers and Geosciences* 109, pp. 323–330. DOI: 10.1016/j.cageo.2017.07.009.

Pizer, S.M. et al. (1987). 'Adaptive histogram equalization and its variations'. In: *Computer Vision, Graphics and Image Processing* 39.3, pp. 355–368. DOI: 10.1016/s0734-189x(87)80186-x.

Raffel, M., C. Willert and J. Kompenhans (1998). *Particle image velocimetry: a practical guide*. 1st ed. 1998. Springer International Publishing.

Raffel, M., C. Willert, F. Scarano et al. (2018). *Particle image velocimetry: a practical guide*. 3rd ed. 2018. Springer International Publishing. ISBN: 978-3-319-68852-7.

Rosten, E. and T. Drummond (2005). 'Fusing points and lines for high performance tracking'. In: *Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1*, pp. 1508–1515. DOI: 10.1109/iccv.2005.104.

Scarano, F. and M.L. Riethmuller (1999). 'Iterative multigrid approach in PIV image processing with discrete window offset'. In: *Experiments in Fluids* 26.6, pp. 513–523. DOI: 10.1007/s003480050318.

Shavit, U., R.J. Lowe and J.V. Steinbuck (2007). 'Intensity Capping: a simple method to improve cross-correlation PIV results'. In: *Experiments in Fluids* 42.2, pp. 225–240. DOI: 10.1007/s00348-006-0233-7.

Spreafico, M. and R. Weingartner (2015). *The hydrology in Switzerland. Selected aspects and results*. Federal Office for the Environment FOEN, p. 139. URL: https://www.bafu.admin.ch/bafu/en/home/topics/water/water--publications/publications-water/the-hydrology-in-switzerland.html.

Swiss Federal Office for the Environment FOEN (2020). *Limmat - Zürich, Unterhard 2099*. URL: https://www.hydrodaten.admin.ch/en/2099.html (visited on 31/01/2020).

Swisstopo (2020). *CH1903+: Official Swiss reference system*. URL: https://www.swisstopo.admin.ch/en/knowledge-facts/surveying-geodesy/reference-systems/switzerland.html (visited on 31/01/2020).

Thielicke, W. and E.J. Stamhuis (2014). 'PIVlab - Towards User-friendly, Affordable and Accurate Digital Particle Image Velocimetry in MATLAB'. In: *Journal of Open Research Software* 2.1. DOI: http://dx.doi.org/10.5334/jors.bl.

Westerweel, J. and F. Scarano (2005). 'Universal outlier detection for PIV data'. In: *Experiments in Fluids* 39.6, pp. 1096–1100. DOI: 10.1007/s00348-005-0016-6.