POLITECNICO DI MILANO
DEPARTMENT OF ELECTRONIC, INFORMATION AND BIOENGINEERING
DOCTORAL PROGRAMME IN INFORMATION TECHNOLOGY

REMOTE BIOMETRIC SIGNAL PROCESSING BASED ON DEEP LEARNING USING SPAD CAMERAS

Doctoral Dissertation of:
Marco Brando Mario Paracchini

Supervisor:
Prof. Marco Marcon

Co-supervisor:
Prof. Federica Villa

Tutor:
Prof. Maurizio Magarini

The Chair of the Doctoral Program:
Prof. Barbara Pernici

2020 – XXXIII

Acknowledgments

Innanzitutto vorrei ringraziare il professor Marco Marcon che non solo mi ha guidato e seguito in questi tre anni di dottorato ma che mi supporta da più di sei anni, cioè da quando mi affidai a lui per la mia tesi magistrale. Gli sono grato sia a livello accademico per avermi dato la possibilità di lavorare a molti progetti interessanti, sia a livello umano per la gentilezza, i consigli e il supporto che mi ha sempre offerto.

Ringrazio anche la professoressa Federica Villa per avermi dato la possibilità di intraprendere il percorso di dottorato. La ringrazio anche per avermi accolto per tre anni nel suo ufficio, condividendo con me la sua esperienza e la sua simpatia.

In aggiunta vorrei ringraziare anche la professoressa Binaghi e il professor Milani per gli utili consigli forniti al fine di migliorare il presente lavoro. Inoltre ringrazio tutte le persone dentro e fuori l'università che mi hanno accompagnato e aiutato in questo percorso.

Sono anche grato a tutta la mia famiglia, a partire dai miei genitori, che mi hanno sempre supportato nel mio percorso universitario.

Il ringraziamento più grande va a Rossella per aver trascorso con me tutti i momenti, dai più alti ai più bassi, che si sono susseguiti in questi ultimi tre anni. Un ulteriore ringraziamento va sempre a lei per aver coronato i momenti di inizio e fine del mio percorso di dottorato con la nascita di due bambini meravigliosi, Leonardo (13/11/2017) ed Andrea (12/10/2020). In particolare ringrazio Leonardo per essere un bambino eccezionale e per darmi sempre un motivo per essere orgoglioso.

Acknowledgments will be in Italian since all the people I want to thank primarily speaks this language.

Abstract

REMOTE PhotoPlethysmoGraphy (rPPG) allows the extraction of cardiac information just by analyzing a video stream of a person face. In this work the adoption of Single-Photon Avalanche Diode (SPAD) cameras for rPPG applications is investigated in order to exploit the higher sensitivity of the SPAD sensors. In particular, a rPPG application in an automotive environment is proposed in order to monitor, in a non invasive fashion, the driver's health state and potentially avoid accidents caused by acute illness states. In order to compensate for the SPAD camera's low spatial resolution, a novel facial skin segmentation method, based on a deep learning architecture, is proposed. This method is able to precisely associate a skin label to each pixel of a given image depicting a face even when working with low resolution grayscale face images (64x32 pixel) and is able to work in presence of general environment condition regarding illumination, facial expressions, object occlusions and regardless of the gender, age and ethnicity of the subject. Moreover, some metrics were developed in order to monitor the dependability of the heart rate estimation and detect situations where an optical solution, such as rPPG, could fail. Finally, a rPPG application has been developed able to run in real time on a small ARM device equipped on a car. After receiving data from the SPAD camera, it is able to extract the heart signal and analyze it in order to constantly monitor the driver's health condition.

List of Figures

List of Tables

List of Abbreviations

HR	Heart Rate
HRV	Heart Rate Variability
bpm	beats per minute
PPG	Photopletysmography
rPPG	Remote Photopletysmography
ECG	ElectroCardioGram
SPAD	Single-Photon Avalanche Diode
LED	Light Emitting Diode
CCD	Charge Coupled Device
CPS	Cyber-Physical System
CPU	Central Processing Unit
GPU	Graphics Processing Unit
ROI	Region Of Interest
RGB	Red, Blue, Green
DL	Deep Learning
ML	Machine Learning

CNN	Convolutional Neural Network
ReLU	Rectified Linear Unit
DFT	Discrete Fourier Transform
PCA	Principal Component Analysis
ROC	Receiver Operating Characteristic
SVD	Singular Value Decomposition

Contents

CHAPTER 1

Introduction

Being able to constantly check, in real time and without any contact, the health condition of a person could have a significant impact in many different situations. Possible applications include fitness assessments [103], medical diagnosis [103] and driver monitoring [130]. The act of extracting biomedical information analysing a video stream is called remote PhotoPlethysmoGraphy (rPPG) or imaging PhotoPlethysmoGraphy (iPPG) [103]. This is an evolution of contact PPG, a technique introduced in early 20th century that is nowadays commercially and clinically implemented in order to monitor the cardiac activity. The basic concept of PPG is placing a light emitter and a light receiver in contact of the subject skin and analysing the light intensity variation in order to estimate information about the cardiac activity. This is possible since the light intensity fluctuations that could be observed with a PPG device are caused by the periodic passage of blood in the vessel underneath the skin which changes how the light is reflected and transmitted by the subject's skin. On the other hand, rPPG aims at conducting the same analysis in a remote way without making any physical contact with the subject. As stated above, the benefit of using rPPG are numerous in many different situations and in particular this could have a significant impact in the automotive industry. The possibility of con-

stantly check the driver's heart state could be extremely useful especially if this could be achieved without distracting or disturbing the driver. A computational unit equipped on the car that is able to extract a rPPG signal and analyze it in real time in order to consistently monitor the driver's cardiac activity could be pivotal in many situations. For example, these data could be used to enable particular features of the vehicle, such as autonomous driving, that could take control of the vehicle itself and avoid car accidents by simply safely parking the car in case of detected driver sickness or altered emotional state. With the evolution of smart cars this could become an important on board safety feature. In addition to that, all the acquired biometric parameters could also be transmitted to a cloud based system in order to constantly monitor the health conditions and the emotional state of the driver, for example for automatically activating health services or live remote assistance in case of necessity. Moreover the biometric data could also be exploitable for other purposes, such as for example, by insurance company in order to check the driver's health state in case of car accidents and/or automotive companies, in order to evaluate driving comfort and conditions. For all these motivations, many automotive companies are researching on rPPG and the task of developing an rPPG automotive system was an important part of project DEIS. This was a H2020 project that ran from 2017 to 2020 which has the purpose to develop methods in order to asses the dependability of many Cyber-Physical Systems. On the described automotive related task Politecnico di Milano jointly worked with General Motors and Ideas & Motion.

1.1 Problem statement and objectives

The main goal of this work is to develop a rPPG system able to estimate numerous biomedical measurements in real time and in a dependable fashion. Moreover, this work explores the possibility of adopting a SPAD (i.e. Single-Photon Avalanche Diode) array camera instead of traditional RGB camera, as done in the majority of publications in the rPPG field [103], [93]. SPAD cameras are capable to detect even a single photon [13], have extremely high frame rate [14] and have proved their usefulness in a very large range of applications [15], such as 3D optical ranging (LIDAR) [15], Positron Emission Tomography (PET) [5] and many others. In rPPG applications SPAD's high precision could be useful in measuring accurately the fluctuations in the light intensity reflected by the skin produced by the

http://www.deis-project.eu/
https://www.gm.com/
https://www.ideasandmotion.com/

blood flow. On the other hand, the main drawback of using a SPAD sensor is their low spatial resolution due to technical limitations. In order to overcame this problem and use as much spatial information as possible, an *ad-hoc* deep learning based method is proposed. This is one of the first work in which Deep Learning methods are used in this field, being the adoption of this kind of techniques very recent in rPPG, the first publications started in 2019 [12], [57], [95]. Moreover, all the other rPPG methods based on deep learning completely substitute the classical signal processing techniques with data driven ones using end-to-end networks. On one hand, the use of an end-to-end deep learning model has proven to achieve state of the art results on many computer vision tasks such as image segmentation, object detection, and many others [31]. On the other hand, this kind of methods required a massive amount of training data in order to learn how to extract heart related information directly from video frames and no prior domain knowledge is incorporated. This make the performance of these methods tightly linked to the training dataset and potentially unable to generalize in different setting conditions. Moreover, the complete substitution of classical signal processing techniques developed using solid theoretical backgrounds (signal filtering, Fourier transform, etc.) with data driven ones could lead to non-optimal solutions. For the best of our knowledge no prior work has been done in trying to combine traditional and deep learning based signal processing in this field. Lastly, in all the considered studies the cameras used are traditional RGB cameras. The main aim of this study is to validate the effectiveness of performing rPPG using SPAD camera, in particular in low illumination conditions, coupled with a deep learning technique in order to compensate for the low spatial resolution of Single-Photon cameras. Adopting a SPAD camera could also be beneficial in the use of the propose rPPG system in uncontrolled environments in which there could be sudden light variations (for example, if this technology is used in order to monitor a driver, this could happen in tunnel or in presence of car light reflexes). In this kind of scenarios, the best strategy in order to remove this high frequency noise is oversampling and SPAD cameras are the best one in this field [14]. Finally, since the rPPG estimation of biomedical measures is related to optical signals that could be affected or masked by noise some dependability evaluation metrics are also proposed.

1.2 Thesis overview and contributions

The rest of this work is organized in chapter tackling problems, such as pulse signal extraction using SPAD camera, skin segmentation on low res-

olution grayscale images, which are preparatory and necessary in order to develop a SPAD based rPPG system that will be described in the last chapter. In particular, the rest of the work is organized as follow:

CHAPTER 2 proposes an extensive overview on the PPG and rPPG state of the art describing the two techniques and analysing the different between them. Moreover a particular focus is directed to deep learning methods in general and their particular use in the rPPG field critically describing also the possible drawbacks hidden behind using such methods.

CHAPTER 3 describes SPAD cameras highlighting their usefulness in rPPG applications. In particular the scope of this chapter is to investigate the possibility of using SPAD cameras in rPPG application and evaluate their performances in respect to RGB cameras. In order to achieve that, a set of experiments have been conducted on still subjects acquired in controlled conditions in order to:

- Select the best wavelength for performing rPPG with SPAD camera.
- Compare the rPPG estimations obtained using SPAD camera and the ones that could be obtained using traditional RGB cameras.

Five different metrics are also introduced in order to evaluate the experimental results.

CHAPTER 4 has the scope to propose an automatic method with the aim of solving the task of detecting skin pixels in grayscale low resolution face images, as the one obtained using a SPAD array camera. Since the facial skin detection problem is very specific, very few data are available for this specific problem. For this reason, a complex transfer learning approach is described in which an *ad-hoc* developed Convolutional Neural Network (CNN) model was trained starting from a colorization problem.

CHAPTER 5 introduces a modification of the CNN described in the previous chapter. In particular this new version is able to solve the problem of detecting skin pixels in grayscale low resolution face images efficiently and in real-time even when run on hardware with limited computing capabilities. The Convolutional Neural Network model described in Chapter 4 is optimized and modified in order to adapt to the

computational requirement. Also in this case, a transfer learning approach is adopted in order to exploit the knowledge already present in the first network.

CHAPTER 6 has the goal to introduce a rPPG system that, making use of a SPAD camera and an single-board ARM computer, is able to estimate biometric parameters, such as Heart Rate, in real time and in a dependable way. A rPPG pipeline is proposed making use of the SPAD camera, described in Chapter 3, the deep learning based method for facial skin segmentation, described in Chapter 5, and traditional signal processing techniques in order to estimate biometric parameters.

CHAPTER 7 draws the conclusions of the presented work, highlighting its major contributions and the possible paths for future works that could be carry on inside this field.

CHAPTER 2

Related Works

Contact photoplethysmography (PPG) is a simple technique that traces back to the 1930s [36]. Using this approach blood volume changes related to the pulsating nature of circulatory systems [107] are measured using light. In more recent years, starting from 2008, it was demonstrated [114] that PPG could be performed remotely (i.e. rPPG) using ambient light as the optical source and, since then, many studies focused on the extraction of heart rate using cameras were published [24, 27, 46, 82, 93, 94, 111, 115]. Some surveys on the state of the art of this field could be found in [103], [71], [122] and [34]. More recent works [12], [95], [57] explored the possibility of using deep learning techniques in rPPG applications. In the rest of this chapter an overview on PPG and rPPG systems will be described with a focus on the adoption of deep learning techniques in this particular field.

2.1 Photoplethysmography

The term "plethysmography" is commonly used in the medical field and describes the action of registering and measuring ("grapho") the increase ("pletysmos") of volume in an organ or living body [51]. In 1930s [36] the idea of using light ("photo") for measuring this volume changes was

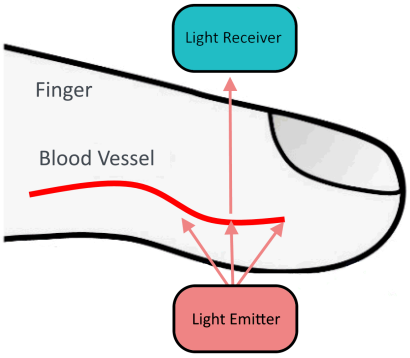

Figure 2.1: *PPG using transmitted light*

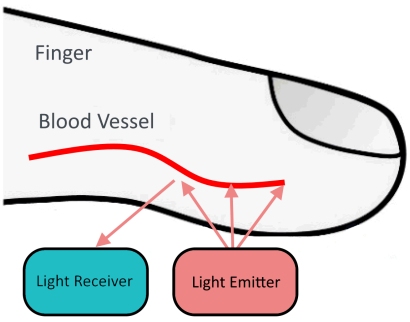

Figure 2.2: *PPG using reflected light*

firstly introduced giving birth to the term "PhotoPlethysmoGraphy" (PPG). Although the term PPG does not specify what kind of volume variation is observed, nowadays PPG is strongly connected to the study of blood volume changes in blood vessels [17]. The basic form of PPG requires only a few electrical components: a light source, used to illuminate the skin, and a light detector, needed to measure the small light variation produced by the blood flowing in the vessels [17]. PPG could be performed exploiting reflected light or transmitted light, as shown in Fig. 2.1 and Fig. 2.2 respectively. In the first case, the tissue to analyse, a finger in the example shown in Fig. 2.1, is placed in between the light source and light detector; the light rays are propagated through the subject's finger and the light intensity recorded by the detector is related to the amount of blood in the vessels in each sampling time. On the other hand, PPG could be performed in reflection mode placing the light receiver and light source side by side, as in Fig. 2.2. The lights penetrate the first layer of skin (the penetration depth depends on the light wavelength [8]) and reaches the blood vessels, a portion of it is reflected back impacting on the light sensor. Also in this case, the intensity of the light received by the sensor is correlated to the amount of blood in the vessel in each sampling time. Although the use of transmitted light could lead to relatively good signal, the measurement site may be limited since the PPG system must be placed in body locations in which transmitted light can be detected [109]. Fingertip and earlobe are the preferred monitoring positions since a sufficient amount of transmitted light could be detected; however, these sites have limited blood perfusion [109] and moreover fingertip sensor interferes with daily activates. On the other hand, systems using reflected light could be adopted in various body positions. However, reflection-mode PPG is affected by motion artifacts and

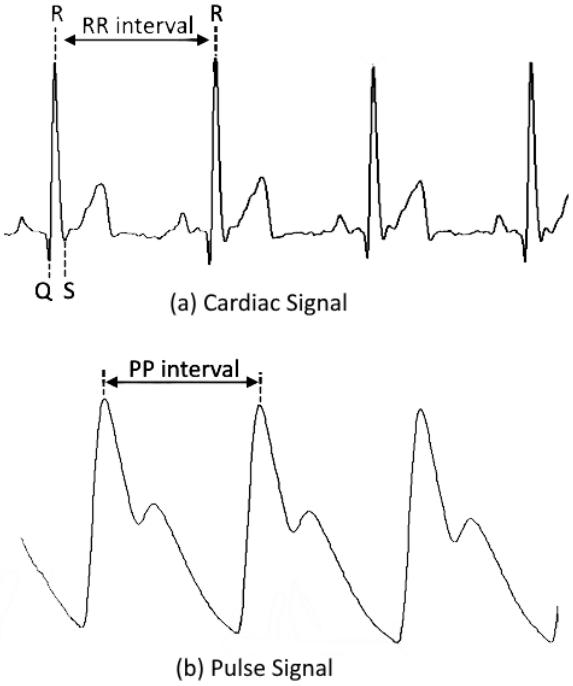

Figure 2.3: *Differences between electrical cardiac signal and PPG pulse signal.*

pressure disturbances [109].

The pulse signal extracted using PPG consists of two different components: the "AC" component, relative to the pulsative nature of the signal, and the "DC", which is relative to the average blood volume in the tissue [6]. Although the DC component, which more precisely is quasi-DC since it varies slowly over time, carries important information on the respiration [3], vasomotor activity [43] and vasoconstrictor waves [6], Traube-Hering-Mayer (THM) waves [6] and also thermoregulation [100], the majority of PPG systems focused more on the study of the AC component. For this reason, PPG is commonly used nowadays in order to monitor the heart activity and in particular the cardiac cycle. The latter is defined as the sequence of events that takes place between two consecutive heartbeats [108], [80]. The cardiac cycle is composed by two consecutive phases: the ventricular diastole, i.e. the relaxation phase, and the ventricular systole, i.e. the contraction stage [80]. During the first phase the blood pressure in the vessels decrease whilst after the contraction the blood is pumped outside of the heart thus being distributed in the body through the vessel increasing their pressure [80]. There are many other methods and de-

vices that could perform heart monitoring like Photoplethysmography [80], for example Electrocardiogram (ECG). On the other hand, PPG is based on the analysis of pulse signal, which represents the variation of the light intensity reflected (or transmitted, depending on the sensor's position) from the skin due to the transition of blood in vessels. Although the pulse signal is different from the electric one generated by the heart activity, due to their own natures, the two are strongly related. Obviously, as can be observed from Fig. 2.3, due to a mechanical delay of approximately 200 ms [123] which also depends on the body part used to extract the pulse signal, they are not synchronized but, on the other hand, they show the same trend since the pressure wave frequency correspond to the heart beating. Therefore, by analysing the pulse signal it is possible to retrieve the Heart Rate (HR). Further analysis on pulse signal could lead to Heart Rate Variability (HRV) estimation. In particular, the tachogram, which is a chart reporting time on the abscissa and time interval between two consecutive R waves on the ordinates [67], could be retrieved from the pulse signal. Moreover, the tachogram representation in the frequency domain presents two different main components that are commonly called Low Frequency component (LF) and High Frequency component (HF) and the ratio between this two quantities is a measure of the simpatho-vagal balance, or rather gives a quantitative information about the functioning and the activation of the autonomic nervous system [67]. Finally the peak of the HF component in a normal subject at rest corresponds to the respiration frequency [67]. For these reasons, performing a spectral analysis of the tachogram could lead to the following information: Heart Rate, LF/HF balance and Respiration Rate.

Studies in the field of PPG started in 1936 with the first experiments of two separate research groups which developed similar instrumentation in order to monitor the blood volume changes in rabbit ear after drugs administration [6], [17]. In the following year, Hertzman and his team, published the first paper describing the adoption of a PPG technique to a human patient in order to monitor the blood volume changes in their fingers [36]. In this paper reflected light was used and in the following year the same team was able to validate the clinical utility of PPG comparing estimations obtained using it with ones gathered by monitoring simultaneously the same patient adopting mechanical plethysmography. An example of one of the early PPG system, proposed in [37], is depicted in Fig. 2.4. Although in the following years (1940 [38]) the same research team was able to develop an electronic system able to split AC and DC components, this technology was abandoned for a long period shortly after. The main reason behind it

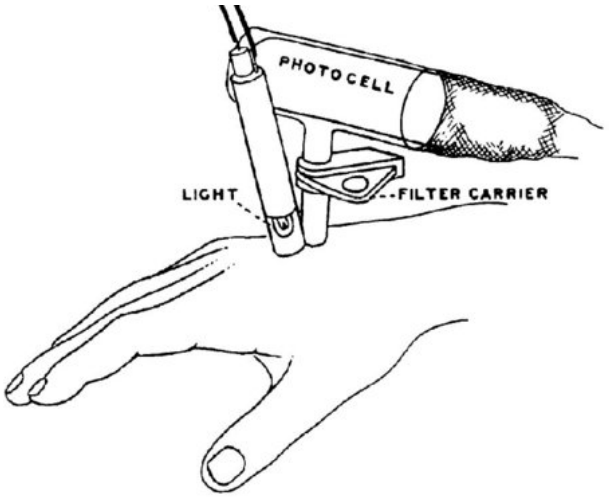

Figure 2.4: *One of the first PPG system proposed in 1938 [37].*

was the limited advancement in lighting technology [6]. As a matter of fact, the power battery light used in these preliminary studies had a very wide spectrum and was not optimized for this specific task. Moreover, using this kind of torch, constant light intensity could not be guaranteed [6]. In more recent years, the developments in semiconductor technology, i.e. light emitting diodes (LED), photodiodes and phototransistors, coupled with the need of non-invasive and low-cost cardiovascular monitor devices helped re-establishing PPG [6]. Nowadays PPG is used in both clinical and commercial devices. In particular, in recent years, the wearable devices market is on the rise and among the different categories on the wearable technology market, pervasive health monitoring applications are ranked the fastest growing segments due to the overwhelming need to monitor chronic diseases and aging populations [29]. Not only this kind of devices are able to provide input to fitness tracking applications but also monitor important physiological parameters, such as Heart Rate, Heart Rate Variability, glucose measures, blood pressure readings and many others. The development of this kind of wearable devices started approximately in 2001 with the creation of a smart PPG ring [89] and continuous to this day with ear, forehead and wristband devices [29]. One of the most recent smart watch devices is depicted in Fig. 2.5 where in the back PPG light emitters and sensors could be noticed.

PPG is not the only method existing in order to estimate heart related information. The other conventional method used nowadays in clinical operations is ECG [58]. This technology is considered to be one of the oldest

Figure 2.5: *Rear view of a modern commercial wearable device (smartwatch) equipped with a PPG system.*

diagnostic tools still used in medicine today with first recordings dating back as early as 1903 [58]. ECG uses conductive electrodes attached to the patient's body in a predefined and standardised configuration in order to detect and record the difference in the electric potential between different electrodes generated by the electric activity of the cardiac muscles [58]. Although fixed-on-body electrodes are reliable and give good signal quality, there are several disadvantages in using this method. The main drawbacks are related to the direct contact of sensors and patient skin. This method could be perceived as uneasy or annoying. Moreover could not be adopted in many situations (infants, patients with skin allergy and so on). Electrodes misplacement could also cause faulty measurements [58]. Modern alternative heart activity monitoring methods include HR from speech [74], thermal imaging [19], optical vibrocardiography [81], Doppler radar [113] and capacitively coupled ECG [87]. Although all of these methods are remote (i.e. they do not require contact with the subject) some of them are unreliable, require expensive hardware and/or expose the subject to microwave/ultrasound radiation [58]. The other remote method commonly used to estimate information about the cardiac activity is called remote PPG and its the main focus of this work. Its principles and development will be described in the following section.

2.2 Remote photoplethysmography

There are a number of terms used in the literature to describe this class of approaches. The most common ones are: remote PPG (rPPG), non-contact PPG (ncPPG), imaging PPG (iPPG) and PPG imaging (PPGi/PPGI) [71]. For consistency the term rPPG will be used exclusively in the rest of this work in order to refer to this approach. The basic idea behind this class of methods is to perform PPG without contact with the subject, thus increasing the distance between subject-sensor and subject-light source. In a typical setting the subject face is acquired by the camera and using reflected light the fluctuations of the light intensity received related to the subject cardiovascular system are measured. The time varying amount of blood in the subject vessels causes the light modulation [122]. This cardiovascular related modulation in the reflected intensity light is experimentally observed and more than a theoretical reason have been proposed [122]. The first one is the conventional theory behind contact PPG so that the intensity light variation is a direct measurement of the periodically changing vessels' cross-sections [78]. The second theory is based on the assumption that visible light is not able to penetrate down to pulsating arteries [122]. According to this theory [101], the deformation of the larger arteries, caused by the blood volume changes, causes a cyclic deformation of the skin tissue above them [52]. Finally a third theory, which is not alternative to the first two but complementary, links the pulsating nature of the observed signal to ballistocardiographic effects [122]. Both local and global movements (for example, respectively, tilting due to small arteries and head movement due to the aorta's blood injection) contribute to the creation of the pulse signal [16], [79]. The discussion on the theoretical nature of the pulse signal is still open.

In 2008 one of the first rPPG work was published [114]. This publications show that a video captured with a common RGB camera is enough to obtain a plethysmographic signal whence measuring HR and respiration rate. In the following years many publications in this field emerged. Typical setups for validating rPPG involve the use of a low-cost RGB camera, and devices used to obtained ground truth values for HR, HRV and respiration rate, such as ECG. In these experiments the face of the subject is recorded. Typically, the subject is asked to stand still in front of the camera. Different subject positions have been explored in different works: in [86] and in [24] for example, the subjects were asked to sit still in front of the camera at a distance of 1-2 meters. In [46] a two step approach was adopted; firstly the subjects were asked to rest lying horizontally then they were asked to stand

up, thus seeing the modulation of the sympatho-vagal balance. In [82] the differences between the supine position and the sitting position was investigated. In the majority of these setups the distance kept between the subject and the camera varies from 30 cm to 1 meter and the illumination used is commonly ambient light, with exception like in [24] in which a professional studio illumination was used and in [82] in which a low amplitude fluctuation lamp was used. Results in the extraction of parameters of interest were compared with signals extracted by different devices. Particular attention is paid to the choice of the camera and the camera settings like frame rate and resolution. The choice of the camera is critical as long as it is a single device used to extract all the biological signals mentioned above. Cameras used in literature are commonly CCD (Charge Coupled Device) cameras: some studies use webcam integrated in laptop [86, 112], while others record video using compact-cameras or giga-Ethernet-cameras [46]. The resolution of these devices varies around 640x480 pixels, but is enough to extract the signal. All these cameras are RGB cameras so the output is composed by 3 channels, red, green and blue, and the depth resolution is 8-bit per channel. Particular attention is commonly paid to acquisition frequency but different works provide different values: in [86] 15 fps were used, while others acquire at 20 to 120 fps. Very few works studied the relationship between heart rate estimation accuracy and frame rate. Experimental results reported in [10] showed that the effects of lowering the acquisition frame rates to 60 and 30 fps does not introduced observable differences in heart rate estimation accuracy, while others [70] noticed that in other related tasks, like estimating heart rate variability, this could have a remarkable impact. Once the setup has been established the aim is to extract the biological signals from the video and compare them to signals coming from the other devices.

Several algorithms were developed in order to extract heart rate, heart rate variability and respiration rate. As described in Sec. 2.1, the waveform of the signal extracted from a camera is completely different from the one extracted using an ECG, but there is a strong correlation in the frequencies and in the relative time position of particular features like peaks or zero-crossing. Usually the video records the face of the subject and a Region Of Interest (ROI) is selected in each frame. There are several ways to determine this ROI. The easiest one considers a region by manually choosing the pixels of the image corresponding to the skin of the subject [93, 94]. In this kind of choice typical ROIs are rectangles and the selected portion of the face are forehead and cheeks. Other approaches are based on face recognition and tracking [4, 86]. In these cases two different methodologies

were developed: the first one adopt a face detection algorithm and the forehead region is obtained with fixed proportion while novel works focuses on the detection of skin pixels [4, 27]. Once the ROI is selected, the signal extracted is given by the mean of the value of all the pixels in the region.

The output given by the camera is divided in 3 channels: red, green and blue. This is due to the Bayer filter placed on the sensors of the camera. Different approaches have been proposed: some consider single channel, while other consider a combination of the different channels. The easiest method explained in literature takes into account the information embedded only in the green channel. For example, in [82] 3 different signals coming from the R,G and B channels were considered. Results reported in [82] show that the cardiac signal is present in all the 3 signals, with the G component having the highest amplitude, so the G component was chosen in the rest of the study in order to obtain the cardiac information. Also the study conducted in [94] arrived to the conclusion that G channel contains enough information recommending the use of the single G channel in order to reduce computational costs and to implement online analysis. A completely different approach was used in [24] in which all the channels are considered in order to remove movement artifacts; the results of this study show that the pulsatility of the signal varies in respect to the wavelenght of the light considered. In particular, it exhibits its maximum in green and the in minimum red, so a ratio of normalized green and red would make a motion robust pulse signal. Further development lead to the usage of all the three channels and in particular their differences in order to obtain a chrominance signal. This returned good signal and robustness to motion artifacts. A similar approach was followed in [46] in which the chrominance model was adopted coupled with zero-phase component analysis (ZCA). A combination between chrominance model and independent component analysis (ICA) on the three channels was also adopted in order to extract a robust signal. ICA was also used in [86]. ICA and ZCA works similarly: given independent signals these algorithms detect all the components that are present in all channels and clearly separates them. In these studies the inputs are the three channel and these algorithms are capable to reject motion and noise components and to extract he pulse wave. Once the channel or the combination of channels is selected the extraction of HR, HRV and Respiration Rate (RR) can be developed.

After extracting the signal, the majority of the proposed methods apply some data filtering techniques in order to remove noises due to electronic interference and quick movements. Since these filters usually are band-pass filters, also slow components are removed from the signal; these compo-

nents are due to slow movements of the subject during acquisition, but also to slow and small variations in illumination. The signal searched has frequency components between 0.4 and 2 Hz so the band-pass filter showed in most of the studies has passband between 0.4 and 4 Hz to avoid introduction of processing artifacts. Most recent publication developed an adaptive bandpass filters, which dynamically changes the cut-off frequencies based on previously estimated HR [72].

Once the signal has been denoised and filtered, the rest of the analysis is dedicated to find the signal frequency components. In order to achieve this result a Discrete Fourier Transform (DFT) is performed, leading to find the peak corresponding to HR. Further analysis lead to the extraction of HRV, but, in order to obtain this information, it is necessary to detect the peaks in the previously elaborated signal. In order to perform this analysis, a recording period of at least 5 minutes and a subject at rest with paced breathing are required. In [82] this setup was used in order to assess the practicability and the feasibility of a non-contact rPPG method based on video recording of the human face for analyzing HRV, comparing the results for RR data and HRV parameters with those obtained simultaneously using a validated standard heart rate device.

All the studies compared the obtained results with a signal extracted using medical devices such as ECG or contact PPG sensors. Tasli et al. [111] underlined that after detrending and filtering operations on the video signal, the error on the estimation of HR is around 3% and stated that the main limitation of his method is observed under poor lighting conditions. De Hann et al. [24] demonstrated that rPPG provided pulse rate in 92% good agreement with a contact PPG sensor. De Hann et al. encountered problems in the choice of illumination and in motion artifacts that ruined the signal. For what concern the HRV calculation, Moreno et al. [82] explained that the main encountered problems were facial movements and illumination changes.

A standard rPPG pipeline is depicted in Fig. 2.6. As can be observed the rPPG pipeline could be divided in two consecutive steps: the signal extraction part, which focus on the processing of the video stream in order to obtain the pulse signal, and the signal processing one which analyses the pulse signal in order to obtained the biometric parameters estimation.

2.3 Deep learning

The term "Deep Learning" (DL) relates to a class of Machine Learning (ML) algorithms that raised in popularity (also on general media) in recent

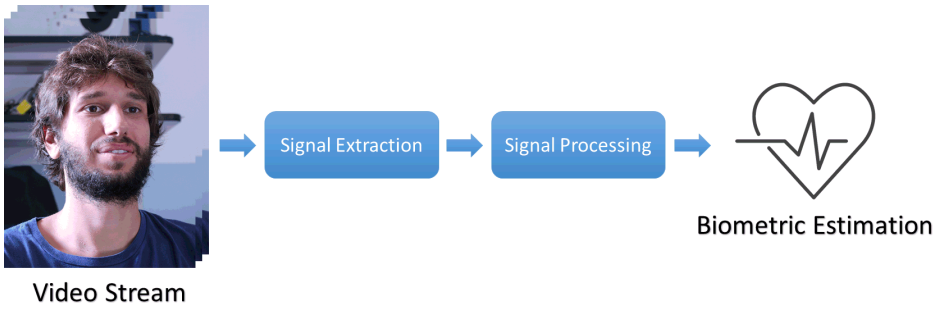

Figure 2.6: *Example of a traditional rPPG pipeline.*

years. Since 2012 [60] DL based methods where able to outperform classical ML algorithms in almost any field they were applied [31], from image classification [60], speech recognition [118], autonomous vehicles [119], drug discovery [20], medical diagnosis [2] and many others [31], [127].

The problem of Artificial Intelligence (AI) is tightly connected to the history of information since could be traced back to 1842 [73], many years before the first working physical computer was ever created. Artificial Intelligence is a broad term that contain, but is not restricted to, ML which is the problem of solving tasks without, or limiting, the use of hard-coded knowledge and instead extracting significant pattern directly from row data [31]. Extracting complex patterns from raw data, that are typically affected by a plethora of different factors, such as noise, intrinsic and extrinsic variability, large dimensionality, etc., is generally a hard task. For this reasons the first proposed class of ML algorithms, which in many application fields were the state of the art until very recent years [106], uses a mix of learned and hard coded knowledge. In particular, expert knowledge was transferred into the algorithm by adopting an engineered feature extraction step that was able to treat the raw data by preprocising them in order to extract a-priori relevant information. This operation, called feature extraction, made the learning part of traditional ML algorithms much easier. On the other hand, DL based methods directly work on raw data essentially merging the feature extraction and feature analysis stages into a single trainable end-to-end phase [31]. This has the benefit of not relying at all on external knowledge and letting the algorithm select (learn) the best data representation directly analysing the data. Clearly this choice comes with costs that directly affect the complexity of the model (number of parameters to be trained), the sample size of required data, the computational complexity of the training process (both hardware and software) and many

others.

While deep learning is a term that gained popularity only in recent years, the idea behind it could traced back to 1940s [31]. As matter of fact the terms DL, Artificial Neural Networks (ANN), Cybernetics, Connectionism and some others are just the re-branding of the same idea during its 80 years of history. It all started with the theoretical development of biological learning [69] and the first proposed models that tried to mimic the human brain functionalities by modelling a biological neuron [31] using linear models. In the 60s some algorithms were proposed in order to train this kind of linear models on real data, one example is the Perceptron [92]. In the 70s and 80s, again drawing inspiration from the brain biology, nonlinear functions were added to these models in order to increase their representation power [31]. During the same years scientists begun to realise that the problem of realising accurate biological neural model would be much harder to realise. In particular, since it is impossible to monitor a large number of neuron during their activity such models could not be validated. Moreover, starting from the 80s it was proven numerous times that simpler model could outperform more biological oriented ones in many tasks. For all these reasons, nowadays deep learning methods do not have the ambition of reproducing exactly biological neural structures, which are currently studied by neuroscience, but simply uses the latter to gather inspirations. A typical example of this is the introduction of Convolutional Neural Networks (CNN) [62] which was inspired by mammalian visual system. During the 90s the backpropagation algorithm was introduce, which is still to this days the predominant method used to train a neural network method. In the years 2000s the creation and development of bigdata and powerful GPU greatly help the evolution of neural networks into deep learning models. In particular, the creation of large datasets such as ImageNet [25] and CIFAR-10 [59] both created in 2009 and containing 3.2 million and 6 thousand samples respectively, made possible the development of deep model architectures with increasing number of parameters. For example, LeNet-5 [62], proposed in 1998, was composed by 60k parameters, AlexNet [60], from 2012, had 60M parameters while VGG-16 [102], proposed in 2015 reached 138M parameters. On the other hand, the training step of such massive models could not be accomplished without the parallel development of computationally more powerful GPUs. Thanks to these development neural network models were able to grow in size and depth increasing their representation power in order to be able to solve more and more tasks with higher and higher precision [31]. Nowadays deep learning based methods are the state of the art in a plethora of different applications, especially in the computer vision

field, and an increasing number of companies, such as Facebook, Google, Microsoft, Baidu and Apple are investing more and more in this field [31].

2.3.1 The use of deep learning in rPPG

While machine learning techniques are widely used in contact PPG applications [28], very recent works [12, 57, 95] explored the opportunity of using deep learning methods also in rPPG applications.

In [12], published in 2019, authors propose the adoption of an end-to-end deep learning method to be applied directly on video stream returning an HR estimation. The authors adopted a Neural Network based on 3D convolutional layers [49]. These are convolutional layers which works on 3D data, 2 dimensions being related to space (i.e. image) plus 1 for time. The complete architecture is composed by a single 3D convolutional layer composed by 32 kernels of 58x20x20. A 3D max pooling follows the convolutional layer. Rectified linear unit (ReLU) is employed as an activation function. An additional dropout operation has been introduced to regularize the CNN. The final output of the CNN part is then flattened and passed to a fully connected module with a hidden layer that includes 512 neurons. The hidden layer is connected to the 76 output neurons: 75 for the pulse rates (in a range between 55 to 240 bpm at regular intervals of 2.5 bpm) plus an extra "No PPG" class for cases in which an estimation could not be performed. The activation functions for the first and second (output) dense layers are, respectively, ReLU and softmax functions. As for the convolutional layer, a dropout operation is implemented to improve regularization. Only a very limited number of datasets that comprise high-quality and uncompressed facial recordings with reference physiological measurements (e.g., heart rate from ECG or contact PPG) are currently available. Many of them contains compressed videos (MANHOB-HCI [105], COHFACE dataset [39]) while the sampling size of other is unfeasible for deep learning training (UBFC-RPPG [11]). For this reason the authors of [12] heavily relay on synthetic data, generated try to mimicking the statistics of real PPG signals.

The work presented in [57], also published in 2019, proposed another deep learning based rPPG method with the aim of estimating average HR. In this case a set of signals, which are time series of red, green, and blue color components averaged over certain regions in the facial area (cheeks, forehead, nose, etc.), were used as the network inputs. A traditional convolutional neural network followed by two fully connected layers were used. In particular, the size of input data sample size is (18x64), where the 1st di-

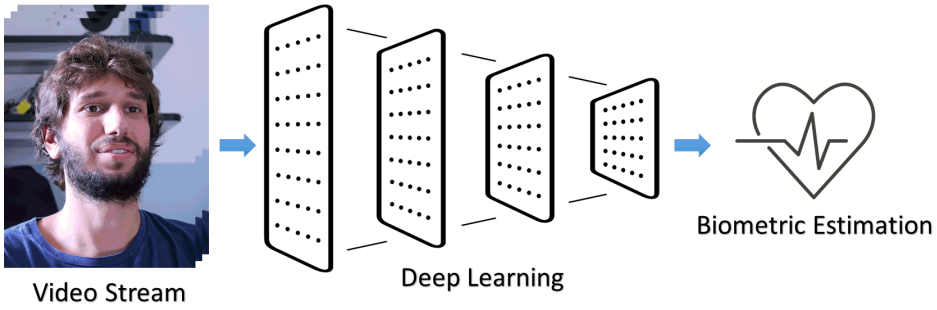

Figure 2.7: *Example of a fully deep learning based rPPG pipeline.*

mension is for color signal channels, 3 three channels signal extracted in 6 locations, the 2nd is for discrete time. Being the input bidimensional, a standard convolution network could be applied as if it were a single-channel image. Due to relatively large kernels and, therefore, quick reducing of temporal information through the convolutional layers of the network, pooling layers were not adopted in order to avoid double reducing. The architecture contains five 2D convolution layers followed by two fully connected layers, using ReLU activations after each layer. Batch Normalization and dropout layers were also adopted. Multiple outputs of this network correspond to different possible HR values inside the 40-125 range, with constant step. This method was trained and tested with proprietary video sequences acquired using 3 different cameras.

Finally, the authors of [95], published in 2020, propose the use of two consecutive deep learning modules in order to estimate average HR directly from a video stream. The two methods are called Front-End (FE) and Back-End (BE). FE has the purpose of improving the interpretability of subtle color changes of facial videos, while BE estimates HR from output of FE. FE is also divided in two consecutive steps, one able to select relevant ROI and one used to extract the pulse signal. The first one is the adoption of a state of the art neural network for object detection [64] trained to detect relevant ROI inside a face image. A simple convolution network is then used to extract and enhance the quality of the pulse signal. Two refiner networks, were also adopted in order to asses the quality of intermediate and final output of FE. These two refiners are adversarially learned to understand the distribution of high quality RoIs and extracted color signals from RoIs. BE is obtained with three-fully connected layers that estimate the average HR from the signal received from FE. The MANHOB-HCI [105] coupled with a proprietary dataset were used for training and testing. Real

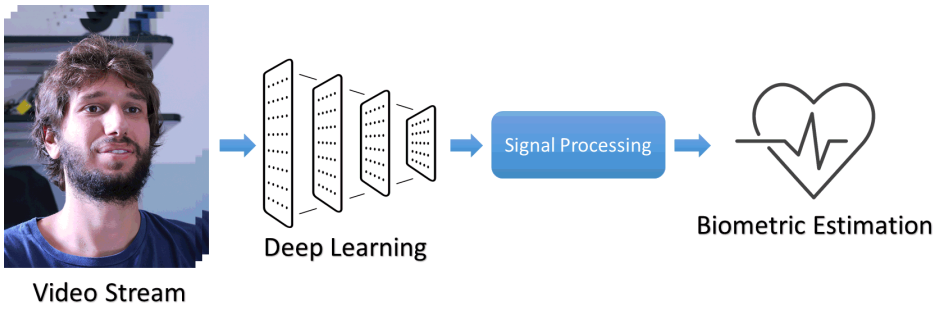

Figure 2.8: *Example of an rPPG pipeline mixing deep learning signal extraction and classical signal processing.*

time performances were achieved using a GeForce GTX 1080 GPU.

All these works completely substitute the classical signal processing techniques with deep learning ones using an end-to-end network, as in [12] and [57], or by using two consecutive neural networks, as in [95]. A representation of this kind of methods is depicted in Fig. 2.7. On one hand, the use of an end-to-end deep learning model has proven to achieve state of the art results on many computer vision tasks such as image segmentation, object detection, and many others. On the other hand, this kind of methods required a massive amount of training data in order to learn how to extract heart related information directly from video frames and no prior domain knowledge is incorporated. This make the performance of this kind of methods tightly linked to the training dataset and potentially unable to generalize in different setting conditions. A possible solution to the scarcity of training data could be the adoption of transfer learning techniques but, due to the peculiarity of the rPPG task, extremely few datasets exist for similar problems. Moreover, the complete substitution of classical signal processing techniques developed using a solid theoretical background (signal filtering, Fourier transform, etc.) with data driven ones could lead to non-optimal solutions. Moreover even than some of this work claim to achieve realtime performances as [95] they require powerful GPU. For the best of our knowledge no prior work has been done in trying to combine traditional and deep learning based signal processing in this field. An example of the rPPG pipeline proposed in this work is depicted in Fig. 2.8. Lastly, in all the considered studies the cameras used are traditional RGB cameras.

CHAPTER 3

Performing rPPG using SPAD cameras

SCOPE & AIMS: The scope of this chapter is to investigate the possibility of using SPAD cameras in rPPG application and evaluate their performances in respect to RGB cameras.

METHODS: Two experiments have been conducted on still subjects acquired in controlled conditions in order to select the best wavelength for performing rPPG with SPAD camera and compare its estimations with the one that could be obtained using traditional RGB cameras. Five different metrics have been introduced in order to evaluate the experiment results.

RESULTS: The best performance are achieved using 550 nm light but reasonable results are also achieved using near infrared light (850 nm). SPAD cameras are able to achieve comparable results in respect to RGB cameras in heart rate estimation and slightly superior accuracy in estimation of the tachogram and respiration rate.

PUBLICATIONS: The main part of this chapter was published as a journal paper [133] and a conference paper [130].

3.1 Problem description

In rPPG applications SPAD's high precision could be vary useful in accurately measure the intensity variations of the light reflected by the skin, caused by the blood flowing underneath it. The work presented in this chapter has the aim of exploring the possibility of performing rPPG using a SPAD camera to compute HR, HRV and RR. In order to evaluate the performances of SPAD camera in a rPPG task two experiments are described in this chapter. The first of the two experiments has the aim of comparing the rPPG estimation obtained with a SPAD camera using light with different wavelength. In order to find the optimal optical wavelength, different optical filters were used in order to find out which wavelength results in containing the highest information related to pulse wave. In particular, ten different optical filters starting from 400 nm, blue light, up to 850 nm, infra-red light, with 50 nm steps, were used for this comparison. This wavelengths range was chosen in order to match the spectral range of the SPAD camera. The main goal of this experiment is selecting the light component that simultaneously is able to penetrate the first layer of skin and carries the most amount of information and is efficiently detected by the SPAD camera. The second experiment on the other hand, is performed in order to compare the rPPG results that could be achieved with a SPAD camera in respect to the one obtainable using a traditional camera. For both experiments measurements are performed on a sat still subject in front of the camera with the artificial illumination directed on its face. The values of the pixels inside a manually obtained ROI are averaged resulting in a pulse wave which represents the raw signal that is processed in order to estimate HR, HRV and RR. In order to evaluate the results of both these experiments five parameters were considered: single beat detection, heart rate estimation, tachogram estimation, LF/HF estimation and respiration rate estimation. Moreover, in order to perform and validate biometric measurements with a SPAD camera and compare it to estimation that could be obtained from a traditional RGB camera, a portable ECG device was used for reference.

The rest of this chapter is organized as follow: in Sec. 3.2 all the devices used in order to collect the experimental data are described, including the SPAD camera. Subsequently, in Sec. 3.3 the two experiments are described, including the setup and the evaluation metrics definition. Following this, in Sec. 3.4 all the signal processing techniques used in order to obtain from the acquired raw data the evaluation of each metric are described. Moreover, in Sec. 3.5 the results obtained on the experimental collected data are reported.

Finally, in Sec. 3.6, the conclusions of this chapter are drawn.

3.2 Materials

In this section all the devices uses in order to collect the experimental data described in Sec. 3.3.1 and Sec. 3.3.2 are described.

3.2.1 SPAD camera

Due to their design SPADs are photodetectors able to reveal even a single photon [121]. Considering a biased p-n junction, on which a bias voltage greater than the breakdown voltage is applied $V_A > V_B$. A single incident photon impacting in the depletion layer causes the creation of pair of electron and hole. The electric field is so high that each of this electric charges, that would be normally eliminated by recombination, are accelerated so much that instead of recombining could impact against an atom causing ionization (i.e. the creation of another pair of electron-hole). In this positively-looped process each charge creates even more electrical charges generating a a self-sustaining avalanche [121]. Due to this process using a SPAD a single incident photon impacting in the depletion layer is able to trigger a macroscopic current (in the milliamp region); this means that from a single-photon event a digital output is obtained. After the photon has been seen, the avalanche is then stopped in order to avoid unnecessary power dissipation due to the avalanche itself, and to rearm the SPAD making it able to see another photon; in order to achieve these targets a dedicated hardware is implemented. A simple solution is the use of a properly sized resistor in series with the SPAD: after a photon-event the parasitic capacitance of the photodetector is discharged, the resistor has the task to reload the capacitor in the way to rearm the SPAD. This gives a simple but slower way to obtain our proposals. A better solution is the use of an Active-Quenching Circuit (AQC) [23] that provides a faster quenching in a smaller area respect to the previously mentioned solution. When a photon triggers the avalanche, the AQC powers down the SPAD in order to rearm it, this gives an holdoff time during which the photodetector is completely blind. This period can be made adjustable, short holdoff periods (in the order of 20 ns) are required in applications where high photon fluxes are present at the cost of high afterpulsing [7] (i.e. the retriggering of the SPAD due to a trapped charges of the previous avalanche), instead much longer holdoff periods are required when weak signals are present or when afterpulsing can heavily affect the measurement and so its reliability.

Figure 3.1: *SPC3 SPAD camera commercialized by Micro Photon Devices (MPD) for single-photon counting applications.*

Counting, timing and other applications

A simple use of the SPAD is counting photons: a counter is added at the output of the photodetector and each photon increases the counter value by one. This behavior can be compared to a conventional pixel of a camera that integrate the light signal over the time. The advantaged of using SPAD in this way is the single-photon resolution of the integrated signal. This technology is currently applied in the observation of fluorescence [30], spectroscopy [76], night vision [97], driver assistance [116] and other fields. A more advanced use is the Time-Of-Flight (TOF) measurements, in which a pulsed illuminator provides a pulsed signal to the target. Due to the presence of background light, in order to have a reliable measure, many repetitions are needed in such a way as to realize a histogram containing the arrival time of the photons. Once the peak has been found the TOF measure is concluded. TOF measurements done in this way can be used to map 3D places in dark conditions or to realize LIDAR [66] (Light Detection and Ranging) systems in not heavily illuminated conditions or with a proper Field of view (FOV) and proper optical filters.

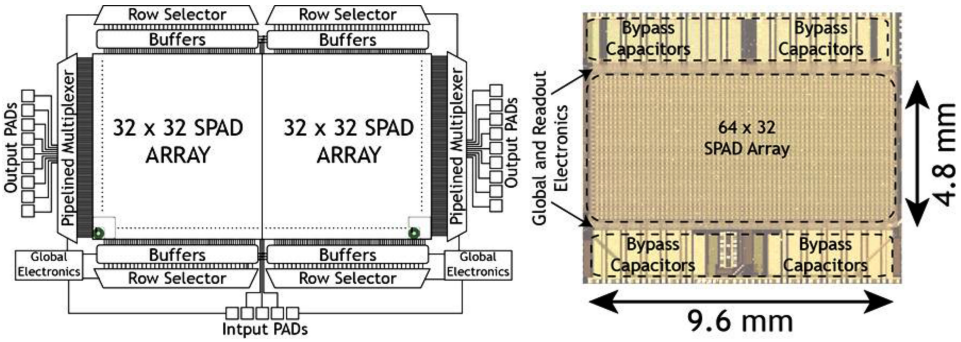

Figure 3.2: *Block diagram (left) and micrograph (right) of the 64 x 32 SPAD array chip in an high-voltage CMOS 0.35 μm technology.*

SPC3 SPAD Camera

The camera used in this project is based on a SPAD array developed by Politecnico di Milano [14]. The whole camera has been developed and commercialized by Micro Photon Devices (MPD) and belongs to the SPC3 SPAD camera series . In Fig. 3.1, a picture of the SPC3 camera is shown. As reported in the diagram in Fig. 3.2, the matrix is composed by 32×64 pixels, each pixel produces an unsigned 9-bit integer output and contains a 30 μm SPAD, the AQC, counters and the memories. The camera, connected through USB 3.0 interface, can be used in counting mode and it is capable to reach 96 kframe/s, which, for the purpose of this project, is more than enough. To recover part of the efficiency lost due to the low, 3%, fill factor of the pixel (because of to the presences of electronics in the pixel) the matrix has been equipped of microlenses that provide a partial enhancement of performances (80% equivalent fill-factor for parallel light beams). This camera has the maximum Photon Detection Efficiency (PDE) of about 50% at around 400 nm, the readout is completely parallel for all the 2048 pixels of the matrix that makes it possible to realize a global shutter. Another important metric for SPAD cameras is the detector intrinsic noise, called Dark Counting Rate (DCR). Dark counts are the triggering events that are not associated to photons but related to other kind of generations (as the thermal one), this parameter affects the signal to noise ratio in low signal regimes. In this project the camera has been used at 100 fps, considering that the SPC3 SPAD has the dark count around 100 cps (counts per seconds), DCR is completely negligible.

An FPGA is used to readout the camera, to sum consecutive frames in

`http://www.micro-photon-devices.com`
`http://www.micro-photon-devices.com/Products/Photon-Counters/SPC3`

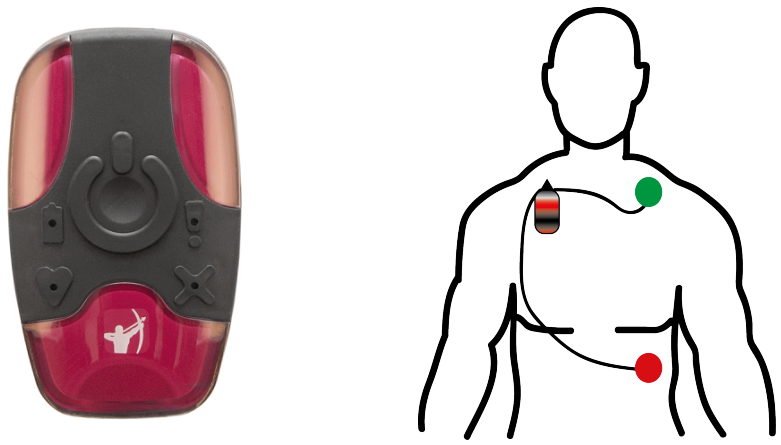

Figure 3.3: *eMotion Faros 180° on the right and electrodes positioning on the left*

order to reduce the final frame-rate increasing the counts depth up to 16-bits and to transfer the data to PC through the USB 3.0 interface. In order to acquire at 100 fps the camera is set to continuous acquisition mode; in this mode a start command is given externally (from the computational unit) and the frames are acquired and stored in the FPGA internal memory, used as a buffer. Each frame is obtained summing in the FPGA the results of 500 acquisitions each obtained with an exposure time of 20 μs in order to collect all the incident photons and at the same time avoid saturation issues and increase the dynamic range of the internal counters.

3.2.2 Other materials

In this section a small description for each devices adopted for the experiments described in Sec. 3.3.1 and Sec. 3.3.2 is reported.

ECG measuring device

A contact ECG measuring device is used in our experiments in order to obtain a reference ground-truth measure of the cardiac activity. The adopted device is the eMotion Faros 180°, IP54 depicted in the left part of Fig. 3.3. This device acquires data from three surface electrodes, placed as shown in the right part of Fig. 3.3, thus giving three ECG traces, one for each derivation. It is equipped also whit highly sensitive triaxial accelerometers, collecting information on movements in all directions. This functionality was

https://www.blindsight.de/product-page/emotion-faros-180-sensor

Figure 3.4: *Basler acA1920-48gc GigE RGB camera*

exploited during the data acquisition phase to increase the synchronicity between the different sensors by hitting it at the beginning of the acquisition in order to obtain a clear peak in accelerator data and consider this point as the start of the acquisition. This ECG measuring devices is programmable by a user interface where acquisition frequency of both accelerometers and electrocardiogram can be selected; in this study a frequency of 250 Hz for ECG and 400 Hz for accelerometers has been chosen.

RGB camera

The adoption of a state of the art RGB camera is critical in comparatively evaluating the performance of a SPAD camera for rPPG applications. After a careful study a Basler GigE RGB model *acA1920-48gc* was chosen. This small camera, shown in Fig. 3.4, can reach up to 50 fps with global shutter and a resolution of 1920×1200 px. The camera sensor is built in CMOS technology with a pixel depth of 10 bits. One of the main advantages of this camera is its easiness of controlling it via software using the Ethernet cable to connect it with the PC; as a matter of fact frame rate, exposure time and synchronization are fully programmable. The lens mount is C-mount, as the SPC3 SPAD camera used, thus making easy to share the same lenses and optical filters, making more fair the comparison between the two. The sensor dimension is quite similar to the SPAD one being 9.2×5.8 mm with a single pixel size of 4.8×4.8 μm.

Respiration measuring device

The study presented in this chapter has also the purpose to investigate the possibility of obtaining respiration information using a SPAD camera. In

urlhttps://www.baslerweb.com/en/products/cameras/area-scan-cameras/ace/aca1920-48gc/

Figure 3.5: *Respiration measuring device.*

order to evaluate this capability ground-truth respiration information are needed. This are obtained using a device developed by electronic laboratory of Polimi shown in Fig. 3.5. This device is mainly composed by a thermistor that must be positioned under the nostril able to measure temperature changes during normal breath. An amplification circuit is adopted in order to obtain a stable signal and the acquired data are collected using an Arduino interfaced with a PC via Matlab. The thermistor used is an Negative Temperature Coefficient (NTC) resistor with a resistance at room temperature of 10 kΩ. This kind of resistors decrease rapidly their resistance when temperature is increasing, thus, if positioned under the nostril, the collected data clearly show a breathing wave. The main frequency component of this wave is the respiration rate. To obtain a more stable signal, a low pass filter is implemented by means of a capacitance before the amplifier and by implementing an integrative feedback, acting as both low pass filter and amplifier. The output of the device is connected to an Arduino that converts analog values into digital in order to be processed.

3.3 Methods

In this section two experiments will be described conducted with the aim of selecting the best light wavelength component in order to perform rPPG using a SPAD camera and compare the use of this kind of camera and traditional ones for this specific task. Moreover, all the signal processing techniques used will be described in the following section.

3.3.1 Exp. 1 - Wavelength selection

The first experiment tackles the problem of determining which illuminant wavelength is optimal in performing rPPG using the SPAD camera. For the sake of finding the optimal optical wavelength, different optical filters were used in order to find out which wavelength results in containing the highest information related to pulse wave. In particular physical optical filters were put in front of the lens so just the selected light component would be captured by the sensor. Ten different optical filters starting from 400 nm, blue light, up to 850 nm, infra-red light, with 50 nm steps, were used for this comparison. This wavelengths range was chosen in order to match the spectral range of the SPAD camera. Each one of these optical filters implements a bandpass filters centered around each specific wavelength with a Full Width at Half Maximum (FWHM) of 40 nm. In this first setup five subjects had been recorded using all filters, each cardiac activity was also monitored using a portable ECG recorder (Faros 180). Recording sessions were always taken in resting conditions, i.e. subjects seated and facing the camera avoiding head movement, and each acquisition lasted for 10 minutes. In order to obtain a wide spectrum in the light source, different kinds of illuminants were considered and tested: finally an incandescent lamp was chosen, which emission spectrum is shown in Fig. 3.6. Acquisition frequencies were set at 100 Hz and 250 Hz for the SPAD camera and the Faros ECG respectively.

3.3.2 Exp. 2 - SPAD and RGB cameras comparison

After selecting the best illumination wavelength another experiment was set up in order to compare the accuracy in rPPG applications of the SPAD camera versus a traditional RGB camera. To achieve this goal a Basler GigE RGB camera was employed. In particular, the model of the chosen camera is *acA1920-48gc* which is a microcamera that can reach up to 50 fps with

http://ecg.biomation.com/faros.htm
https://www.baslerweb.com/en/products/cameras/area-scan-cameras/ace/aca1920-40gc/

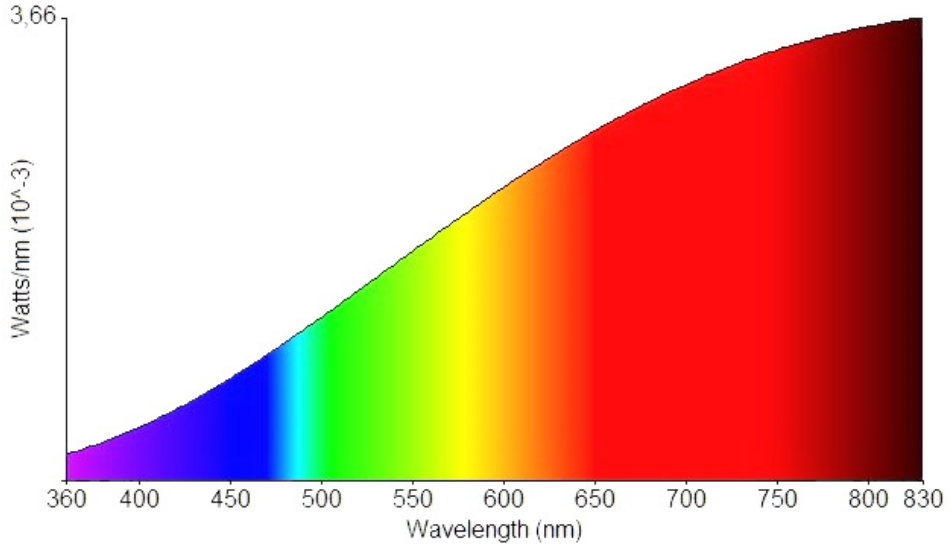

Figure 3.6: *Incandescent lamp emission spectrum.*

global shutter and a resolution of 1920×1200 pixels with a CMOS sensor. Sensor dimensions are 9.2 mm x 5.8 mm with pixel size of $4.8\mu m \times 4.8\mu m$. In order to perform the comparison between RGB and SPAD cameras, three subjects had been recorded using both cameras and the Faros portable ECG for 10 minutes each at resting conditions. SPAD and RGB cameras were put very close to each other (side by side) at an approximate distance of 50 cm from the subject's face. Lenses were chosen in order to record the entirety of each subject face from both cameras. In this way, the same frontal view could be obtained from both acquisition devices. The 550 nm optical filter was mounted on the SPAD camera since it produces the best results in the wavelength selection experiment, as will be described in Sec. 3.5.1. The same incandescent lamp as the former experiment was used also in this case. Acquisition frequencies were set at 100 Hz, 50 Hz and 250 Hz for the SPAD and RGB cameras and the Faros ECG respectively. For each acquisition, the two cameras were synchronized via software.

3.3.3 Evaluation metrics

In order to quantitatively evaluate the results of experiments described in the previous subsections, 5 different parameters are introduced and considered. For each one of them, a brief description and definition is given in the following paragraphs. A complete description of the signal processing methods adopted to evaluate these metrics are described in Sec. 3.4.

Single beat detection The first parameter considered is the accuracy in the single beat detection, which represents the capability of the acquired signal to produce an average wave shape recognizable as an heart beat (qualitative evaluation) and with a small standard deviation (quantitative evaluation). Exploiting a reference groundthrout ECG tack, all the time position of the QRS complexes were determined using the Pan-Tompkins algorithm [84]. A segmentation of the pulse signal is then obtained in which each element represents a signal portion relative to a specific heart beat. Therefore, after resampling each segmented heart beat wave in order to have the same amount of sampling points, each pulse wave was normalized using L^2 norm. A complete description of the signal processing involved in this metric evaluation is given in Sec. 3.4.2

Heart rate The second metric chosen is the computed HR estimation. The average HR error is defined as the absolute difference between the average HR estimation obtained from the SPAD signal and the one obtained from the ECG trace (considered as ground truth). In Sec. 3.4.3 a description is reported on how the HR is estimated from the pulse signal.

Tachogram The third considered figure of merit is the tachogram estimation error. The tachogram estimation error is calculated using the Root Mean Squared Error (RMSE) between the tachogram estimated with the SPAD signal and the one obtained with the ECG groundthruth. The processing steps performed for the tachogram estimation are described in Sec. 3.4.4.

LF/HF The spectrum of the thacogram presents two different main components that are commonly called Low Frequency component (LF) and High Frequency component (HF). The ratio between this two quantities is a measure of the sympatho-vagal balance [98]. These two components are defined as the integral of the spectrum in the following ranges of frequency: LF from 0.04 to 0.15 Hz, while HF from 0.15 to 0.4 Hz. The forth considered metric is the LF/HF estimation error and it is calculated as the percent error between the LF/HF ratio obtained staring from SPAD rPPG signal and the ECG track respectively. A complete description of the signal processing involved in this metric evaluation is given in Sec. 3.4.5

Respiration Rate The HF component of the tachogram is also know as respiratory band [98] and in particular, the peak of the HF component in a normal subject at rest condition correspond to the respiration frequency [18]. The last metric introduced in order to chose the best illumination wavelength is the respiration rate estimation error calculated as the absolute error between the respiration rate obtained with the SPAD signal and the one form the ECG expressed as breaths per minute. The processing

steps performed in order to estimate the respiration rate are described in Sec. 3.4.6.

3.4 Signal processing

In this section all the signal processing steps adopted in order to evaluate the experimental metric introduced in the previous section are described.

3.4.1 Signal preparation

All experiments are conducted with still subjects so the optical signals (SPAD and RGB camera) are extracted by selecting manually a Region Of Interest (ROI) area of the subject forehead. The pixel intensity inside these regions have been averaged in order to create the signals. Both the signal extracted by the SPAD and RGB cameras were firstly filtered with a Butterworth bandpass filter with bandwidth between 0.4 Hz and 4 Hz, and then averaged in windows of 5 samples, thus the resulting signal had an effective sampling frequency of 20 Hz. All the signals were then resampled in order to have the same number of samples and the same time references.

3.4.2 Signal segmentation

The following section describes the method applied in order to identify each heart beat in the signal extracted by the SPAD camera. For each signal, after applying the preprocessing step described in Sec. 3.4.1, the first 30 seconds and the last 30 seconds were removed. This operation was necessary in order to remove movement artifacts at the beginning and at the end of the acquisition, and consider only the steady state of the recorded subject. The aim of this step is to segment each signal into all the heart beat sections present. This is done in order to quantitatively and qualitatively evaluate the average pulse wave shape.

In order to perform the heart beat segmentation the ECG signal was used. In particular, the ECG track was used in order to detect regions of the signal between two consecutive QRS complexes so a complete beat was present. In order to do so, the following procedure was implemented: a window between 120 and 180 seconds in both ECG and pulse wave signals was considered. Inside this window the sample position of all the QRS complexes in the ECG track were determined using the Pan-Tompkins algorithm [84] while in the pulse wave, the position of the maxima was determined by searching for signal peaks. Subsequently, the difference between the positions of all the detected QRS complex and the nearest maxima in the

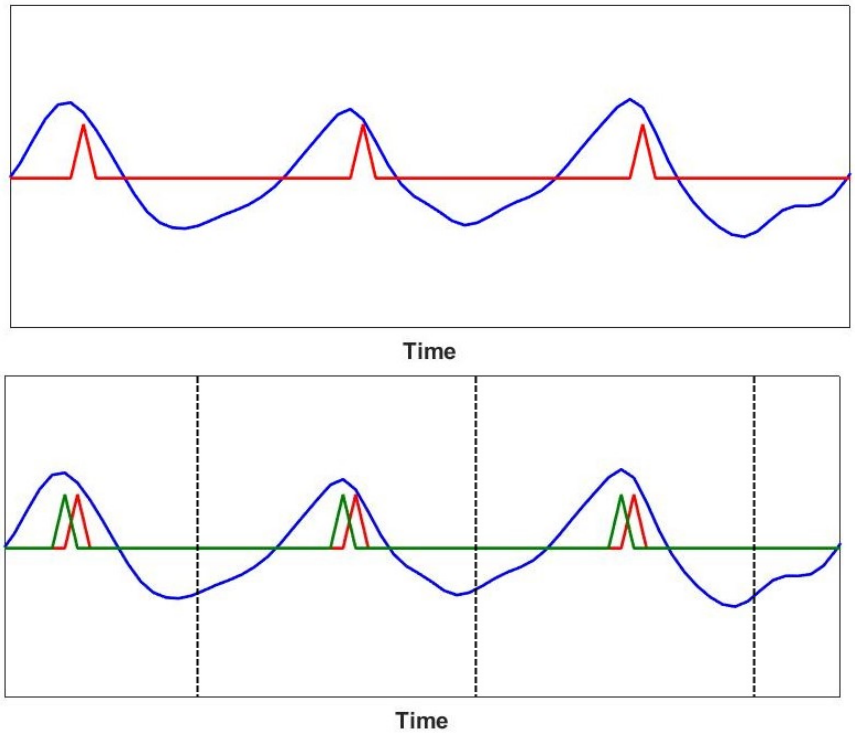

Figure 3.7: *Heart beat segmentation algorithm example. Top picture: original pulse wave and QRS complex time position. Bottom picture: Synchronization alignment between pulse wave maxima and QRS complex time position. Black dashed lines represent estimated segmentation time. Blue: rPPG signal. Red, green: QRS complex time positions.*

pulse wave was computed and averaged, in order to find the mean distance between the same beats in the two signals. This average allow to translate the pulse wave and obtain a better synchronization of the two signals, and also it allowed to consider the distance between two consecutive QRS complex, that depends on the heart rate (top Fig. 3.7). Once the signals were perfectly matched a shift of half average beat was applied. Then the whole pulse signal and ECG track were segmented in pieces corresponding to the time interval between two consecutive QRS complex.

As shown in Fig. 3.7, rPPG signal was segmented between two black dashed lines which time position was estimated as described above. This segmentation allow us to evaluate the characteristic shape of each beat of the pulse wave. In particular all the beat shapes extracted from the same signal were normalized using the L^2 norm and resampled, if necessary, in

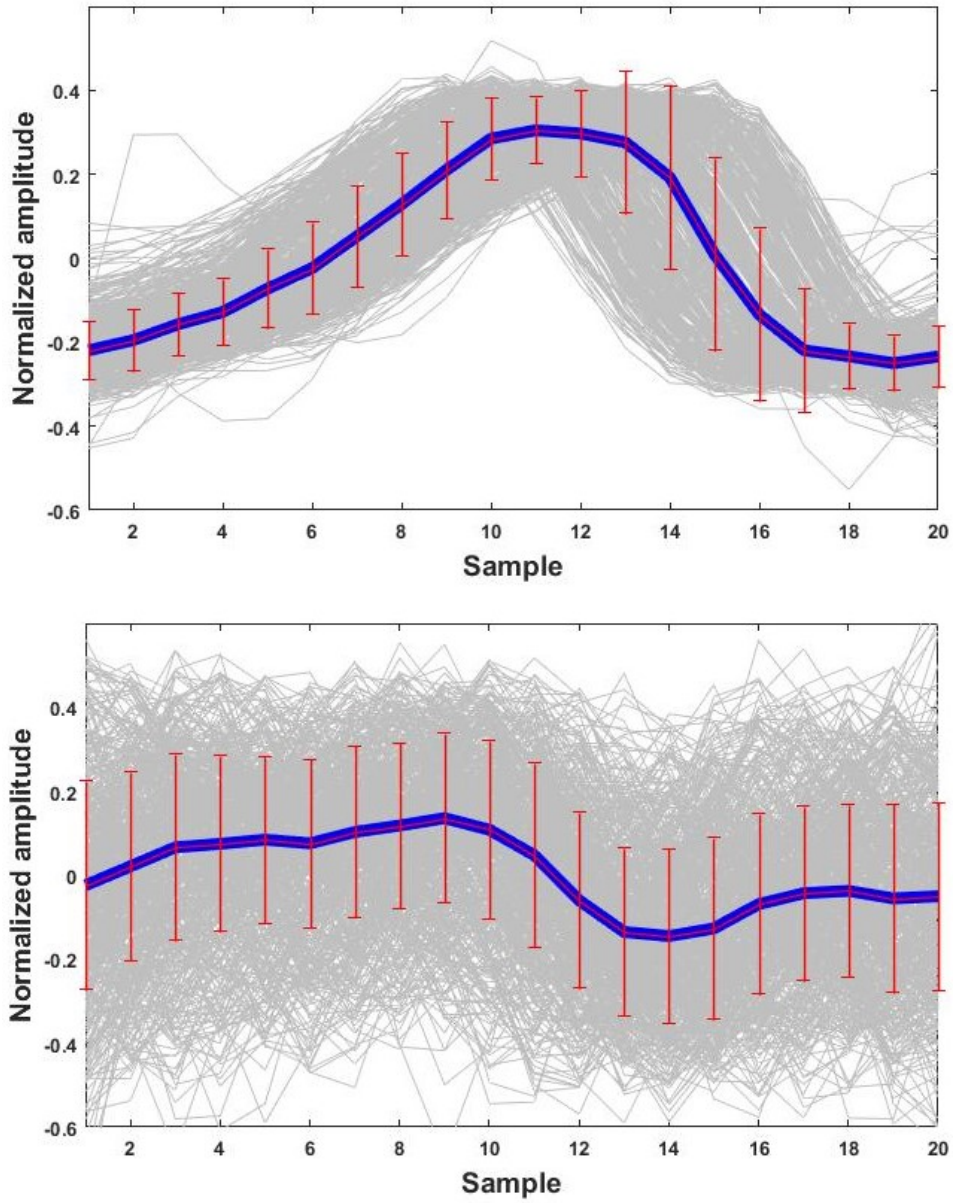

Figure 3.8: *Gray: all the segmented beats in the pulse wave. Blue: average pulse wave beat. Red: standard deviation of all the beats in the pulse wave signal. Upper panel shows an example of optimal beat shape, while lower panel shows a signal with very few information about heart activity.*

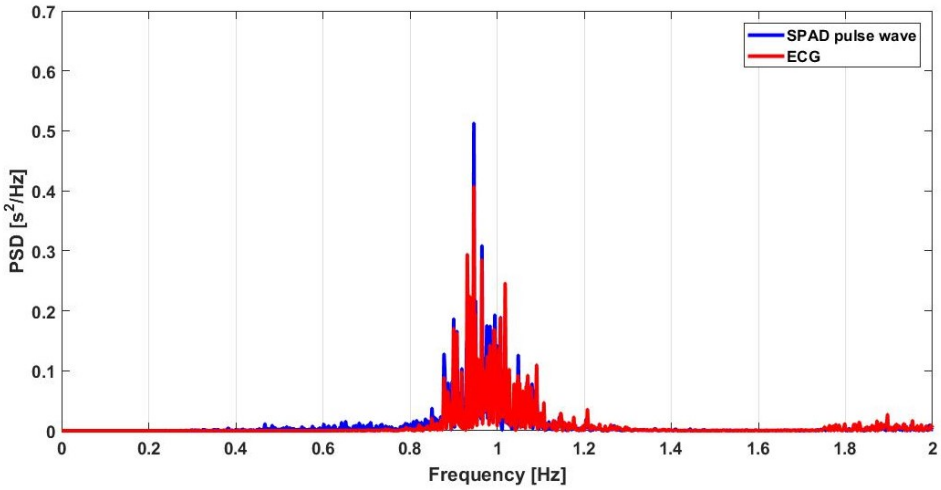

Figure 3.9: *Red: power spectral density of ECG signal; Blue: power spectral density of pulse wave extracted from SPAD camera.*

order to obtained comparable signal pieces. This normalized signal segments were than averaged, using point-wise median, in order to estimate an average beat shape. The difference between each beat and the median was performed. Considering the distribution of the calculated errors, the beats giving an error out of the 90^{th} percentile of the error distribution were eliminated and not considered in the definition of the shape of the beat, since they were probably due to motion artifacts.

In before applying the average, uninformative signal segments (i.e. which standard deviation from the median beat were higher than a threshold) were discarded. In Fig. 3.8, the segmented heart beat pulse wave, from two different acquisitions, are plotted (the blue lines represent the mean pulse wave shape). The upper panel shows an optimal result in which the average beat is clearly related to the heart activity and the point-wise standard deviation is small. On the other hand, the lower panel shows a signal in which a pulse shape is not recognizable.

3.4.3 Average heart rate estimation

The second metric taken into consideration in the evaluation of the experiment performed is the heart rate accuracy. To achieve this result, after the bandpass filtering of the camera signal and the high pass filtering, over 0.4 Hz, of the ECG track needed to remove slow trends of the signal due mainly to movement artifacts, a Fast Fourier Transform (FFT) was performed. The

HR estimation was then performed by selecting the frequency corresponding to the maximum of the signal in the frequency domain. The ideal result would show a perfect matching between the heart rate calculated from ECG and the one calculated from the pulse wave. An example of result is illustrated in Fig. 3.9. As it can be seen in the figure, the maximum frequency component of both pulse wave and ECG track is the same and in particular is at 0.95 Hz, resulting in 57 beats per minute (bpm), physiological value for a resting subject. The accuracy in the determination of heart rate is calculated as the absolute difference in the number of beats per minute, between the rPPG estimation and the ground-truth value obtained from the ECG track.

3.4.4 Tachogram estimation

The third metric introduced in the experiments evaluation is the error in the tachogram estimation. For this task the original sampling frequency of the rPPG signal, 100 Hz, was considered in order to have an improved time resolution. As explained in Sec. 3.1, heart rate variability is important for many reasons: the mean value of the tachogram coincide with the heart rate, its frequency components contains information on respiration rate and, above all, information on the health of the autonomic nervous system. This kind of information is commonly extracted from an ECG, due to the typical shape of the QRS complex that make it easy to detect with high accuracy each heart beat and its temporal position. The aim of the presented method was to calculate a tachogram starting from a pulse wave, mechanical phenomenon, that resulted as similar as possible to the tachogram calculated from an ECG, electrical phenomenon. Moreover, it must be taken into account that the present study aims to extract a tachogram starting from a remote-photoplethysmography, without contact with the subject, thus adding movement and environmental artifacts. The tachogram estimation is tightly connected to finding all the pulse maxima inside the signal. All the processing steps described in the next paragraphs aims at detecting all maxima that represent each heart beat in the pulse wave.

Once the signal was filtered and processed, a first round of maxima detection was performed. In particular, the signal was scanned in search of local maxima imposing a minimum temporal distances between consecutive maxima and considering as peaks only local maxima that exceed a certain treashold. The temporal distance threshold was set to 75% of the inverse of estimated heart rate while the minimum peak height was set as

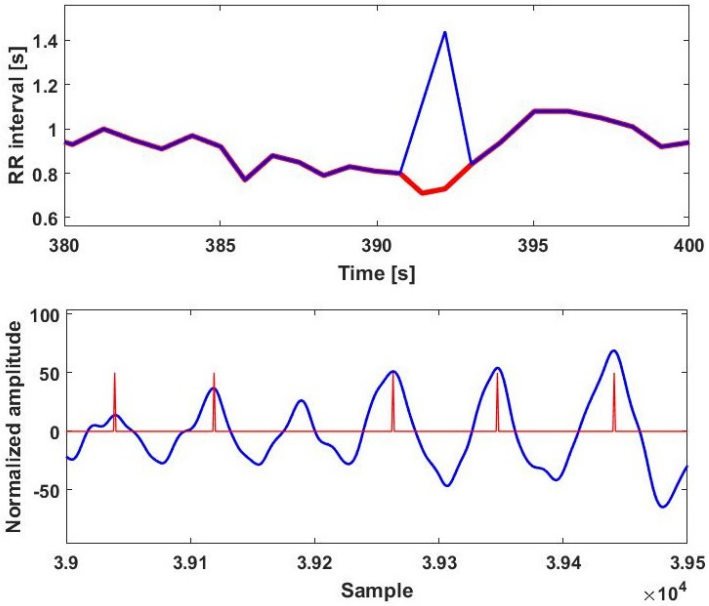

Figure 3.10: *Example of a first round of maxima detection. A missed beat could be noticed.*

50% of the maximum peak in respect to the signal baseline. The temporal difference between consecutive detected local maxima position represent a first estimation of the RR-intervals. Once the maxima had been found, the average of RR intervals was calculated. This operation gave information about the average distance between two consecutive maxima. The average was used to adjust the temporal threshold and perform a second round of searching the maxima on all the pulse wave with the constrain that there must be a maximum inside a window as wide as the calculated RR average. Fig. 3.10 shows an example of the results obtained adopting the first round of maxima detection. In this case a beat is missed, and Fig. 3.11 shows the effects of the second round, detecting the missed beat and correcting the tachogram. As a result of this first part of the method a raw tachogram was calculated.

After applying the steps described above, the obtained tachogram could presented two main situation of error: in one case it could occur that a beat was missed and the tachogram presents a large peak with respect to the baseline; in the other, it could also happen a small peak was mistakenly considered as a maximum, resulting in a peak under the baseline, and the consequently compensation error due to the method searching for the following

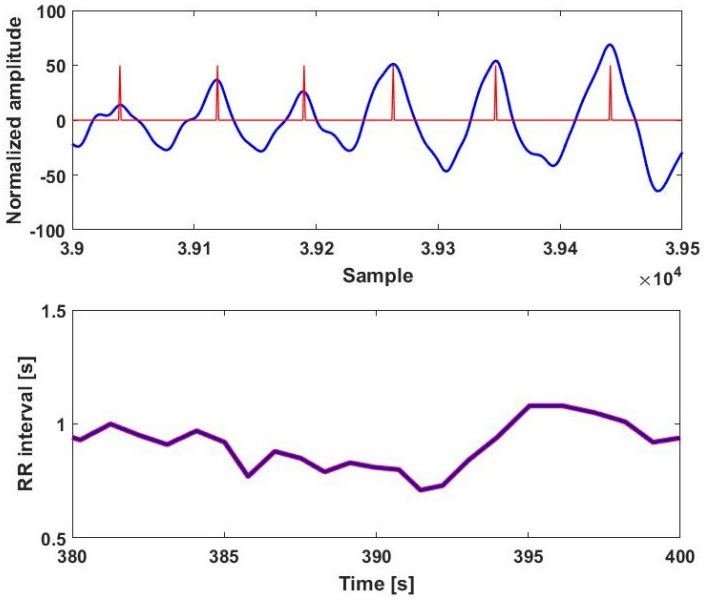

Figure 3.11: *Second round of maxima detection. The missed beat is correctly detected.*

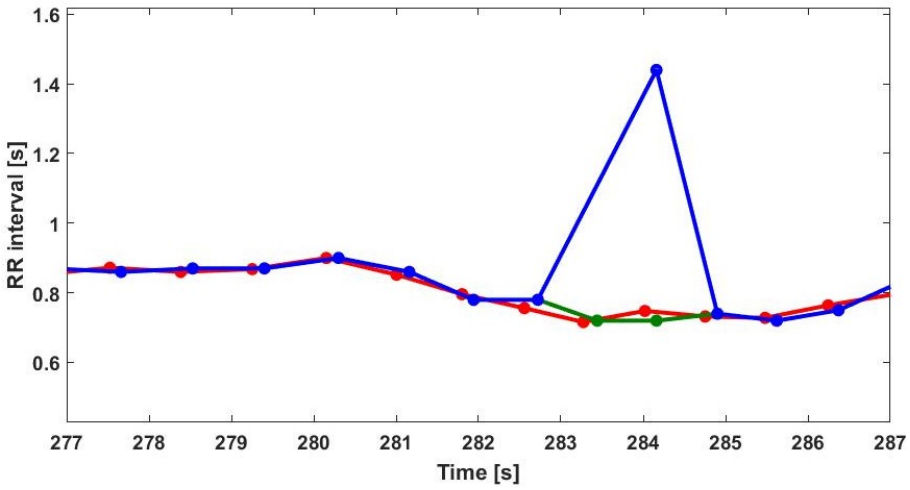

Figure 3.12: *Example of the effect on the tachogram of two peaks mistakenly detected as one. Blue: rPPG tachogram before refinement application. Red: ECG tachogram. Green: rPPG tachogram after refinement application.*

maximum after the previously calculated average RR interval skipping the real maximum, resulting in a high peak over the baseline. Examples of

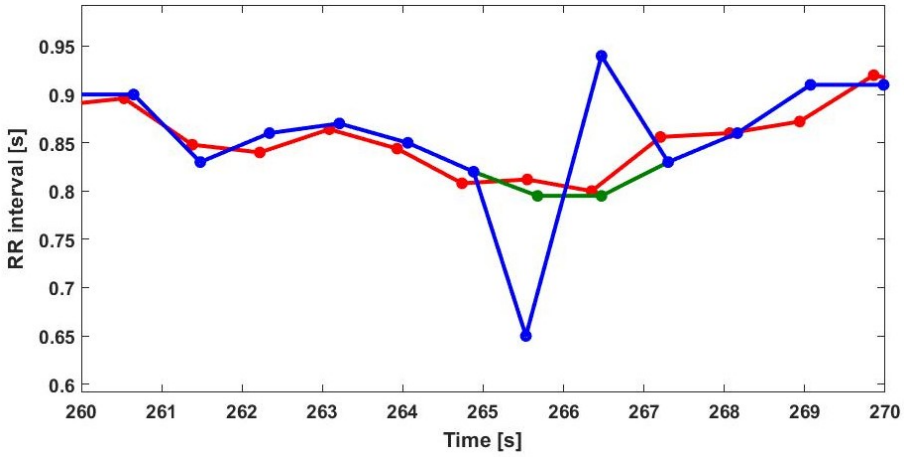

Figure 3.13: *Example of the effect on the tachogram of incorrect peak detection and following compensation error. Blue: rPPG tachogram before refinement application. Red: ECG tachogram. Green: rPPG tachogram after refinement application.*

these two situations are showed in Fig. 3.12 and Fig. 3.13. In particular Fig. 3.12 shows the former situation, a non-detected beat resulting in a high peak in the tachogram, while Fig. 3.13 shows the latter situation, where two consecutive peaks were mistakenly detected returning an unexpected minimum followed by a large maximum.

In order to detect one of the two described situations, the estimated tachogram is scanned and for each tachogram point i a local average between the four previous consecutive RR intervals was calculated (eq. 3.1).

$$RR_{avg} = \frac{1}{4}\sum_{k=2}^{5} RR(i-k) \tag{3.1}$$

If the considered tachogram value, $RR(i)$, is greater than the previous one, $RR(i-1)$, multiplied by a fixed threshold, T_r, it means one of the two possible situation described is present (eq. 3.2).

$$RR(i) > RR(i-1) \times T_l \tag{3.2}$$

If this value is also higher than the local average multiplied for another threshold (eq. 3.3), a single beat was missed.

$$RR(i) > RR_{avg} \times T_h \tag{3.3}$$

In this case this element is split in two equal values, in order to maintain the time position of all the following tachogram points. Otherwise if this

value is not significantly higher than the average (eq. 3.4), the second case happened, so a small RR was detected followed by an extremely high RR interval.

$$RR(i) \leq RR(i-1) \times T_h \tag{3.4}$$

In this case the current element is summed with the previous one, the results was divided by two and this new value is assigned to replace the value of both of them.

$$\begin{gathered} RR_{new} = \frac{RR(i) + RR(i-1)}{2} \\ RR(i-1) = RR_{new} \qquad RR(i) = RR_{new} \end{gathered} \tag{3.5}$$

Finally, in order to evaluate the error between the tachogram calculated starting from the ECG and the one extracted from the camera, both tachograms were resampled with a higher sampling rate. The optimal result obtained with these operations is shown in Fig. 3.14, where the tachogram extracted from the signal of the remote photoplethysmography is almost the same as the one calculated by the ECG track. Quality of the algorithm is based on the root mean square error calculated between the tachogram extracted by the rPPG and the ECG.

$$RMSE(RR_{rPPG}, RR_{ECG}) = \sqrt{\frac{\sum_{i=1}^{n}(RR_{rPPG}(i) - RR_{ECG}(i))^2}{n}}$$

3.4.5 LF/HF estimation

As already illustrated in Sec. 3.1, the estimation of the sympathovagal balance, given by the ratio of the low frequency (LF) and the high frequency (HF) components of the tachogram, is particularly important in many application. These two components, which give information on the activation status of orthosympathetic and parasympathetic nervous system, could change heavily passing from a resting to a standing condition. Studies demonstrated that tachograms extracted from ECG and PPG are almost the same in resting condition, while present great differences in standing condition. These great differences affect mainly the LF components, while the HF components present almost the same values. For this reason, as described in Sec. 3.3.1 and Sec. 3.3.2, the chosen setup for both the experiments considered the subject in resting condition, thus a comparison between the spectrum of the rPPG tachogram and the ECG tachogram is achievable.

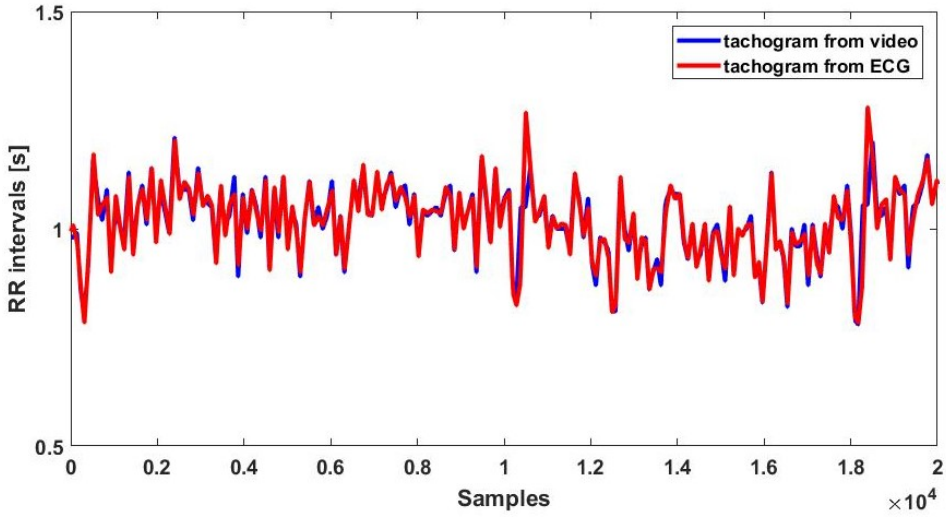

Figure 3.14: *Red: tachogram extracted from the ECG track; Blu: tachogram extracted from SPAD camera video.*

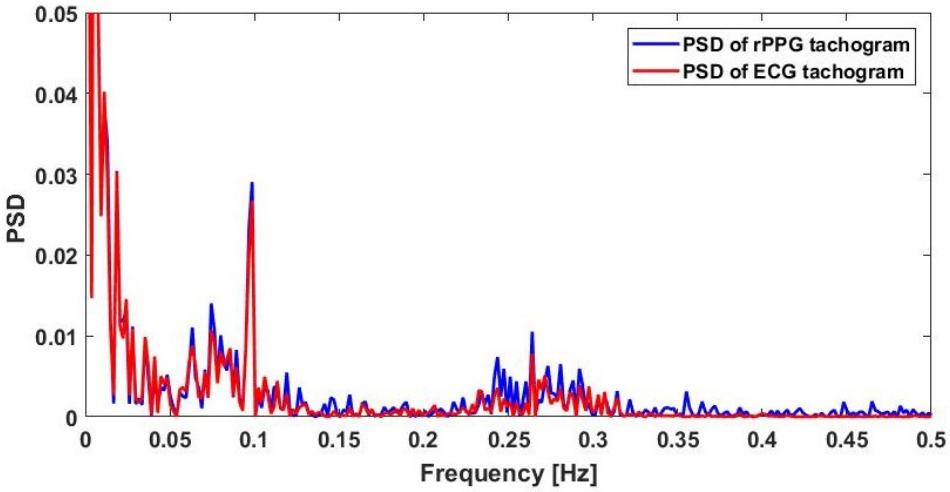

Figure 3.15: *Red: PSD calculated from ECG tachogram; Blue: PSD calculated from rPPG tachogram.*

In order to obtain HF and LF components a simple FFT was performed on both ECG and rPPG tachogram. LF is computed as the area under the curve between 0.04 Hz and 0.15 Hz, while HF is calculated as the integral between 0.15 Hz and 0.4 Hz. The error is calculated as the differ-

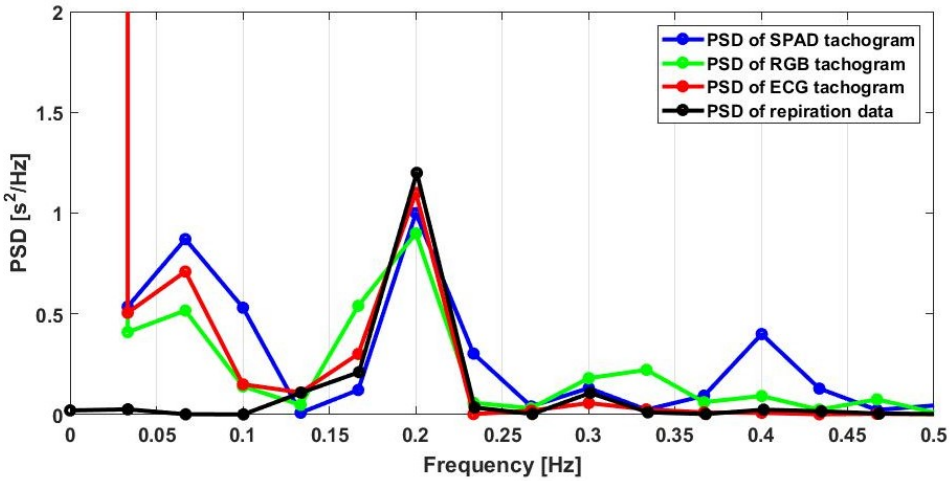

Figure 3.16: *Red: PSD calculated from ECG tachogram; Blue: PSD calculated from SPAD tachogram; Green: PSD calculated from ECG tachogram; Black: PSD calculated from respiration data.*

ence between the LF and HF values obtained staring from ECG track and SPAD rPPG signal respectively. An example of optimal result is shown in Fig. 3.15, where the overlapping between the blue and red line is clearly visible.

3.4.6 Respiration rate estimation

The last metric considered is the error in the respiration rate estimation. This estimation was achieved by performing a FFT on the tachograms and selecting the frequency that corresponds to the spectrum maximum. Since the respiration frequency varies considerably in a long period, analysing the spectrum of a tachogram obtained on a long period (e.g. ten minutes) would result in a PSD with different peaks, each one related to the respiration frequency observed in different periods of the acquisition. Therefore, the recorder signals were cut in windows of one minute so that the respiration rate observed could be considered constant and a comparison was made between all the one-minute window. An example of optimal result is shown in Fig. 3.16.

Table 3.1: *Standard deviations for the single beats detection for each acquisition.*

	Wavelength [nm]									
	400	**450**	**500**	**550**	**600**	**650**	**700**	**750**	**800**	**850**
Sbj 1	0.13	0.15	0.22	0.09	0.13	0.22	0.17	0.22	0.13	0.13
Sbj 2	0.13	0.11	0.12	0.10	0.14	0.15	0.16	0.15	0.14	0.12
Sbj 3	0.14	0.13	0.10	0.11	0.12	0.18	0.21	0.17	0.16	0.19
Sbj 4	0.15	0.12	0.13	0.12	0.15	0.21	0.20	0.20	0.14	0.13
Sbj 5	0.15	0.12	0.12	0.09	0.12	0.15	0.16	0.14	0.11	0.10
Avg.	0.14	0.13	0.12	**0.11**	0.13	0.18	0.18	0.18	0.14	0.13

3.5 Evaluation results

In this section results obtained by performing the two experiments described in Sec. 3.3 are reported.

3.5.1 Exp. 1 - Wavelength selection

Heart beat estimation

Since the evaluation of the first metric has a qualitative component, the average beat shapes obtained for three subjects are shown in Fig. 3.17 , Fig. 3.18 Fig. 3.19, Fig. 3.20, and Fig. 3.21, , respectively for subject 1, 2, 3, 4 and 5. In each figure each subplot is relative at the results obtained with each filter at different wavelengths. In particular, each beat shape is reported in gray and the blue line represents the point-wise median. The red intervals represent the standard deviation for each sampling point. As can be observed qualitatively for some wavelength the beat shape is not recognizable (e.g. 650 nm) while for other the pulse wave is clearly visible (500 nm, 550 nm and 850 nm). From a quantitative point of view, standard deviations for all the subjects and all the wavelengths are reported in Tab. 3.1. As can be observed the 550 nm wavelength is generally able to produce more precise results.

Heart rate

Tab. 3.2 reports the result in the average heart rate estimation. As can be observed estimations obtained using 500 nm and 550 nm achieve the best results since the mean absolute error is less then 2 bpms for both wavelengths. Particular attention should be paid to the results obtained while

Remark, in some tables, reported N.A. values mean that the original signal carried so little information about the pulse wave that the analysis could not be performed.

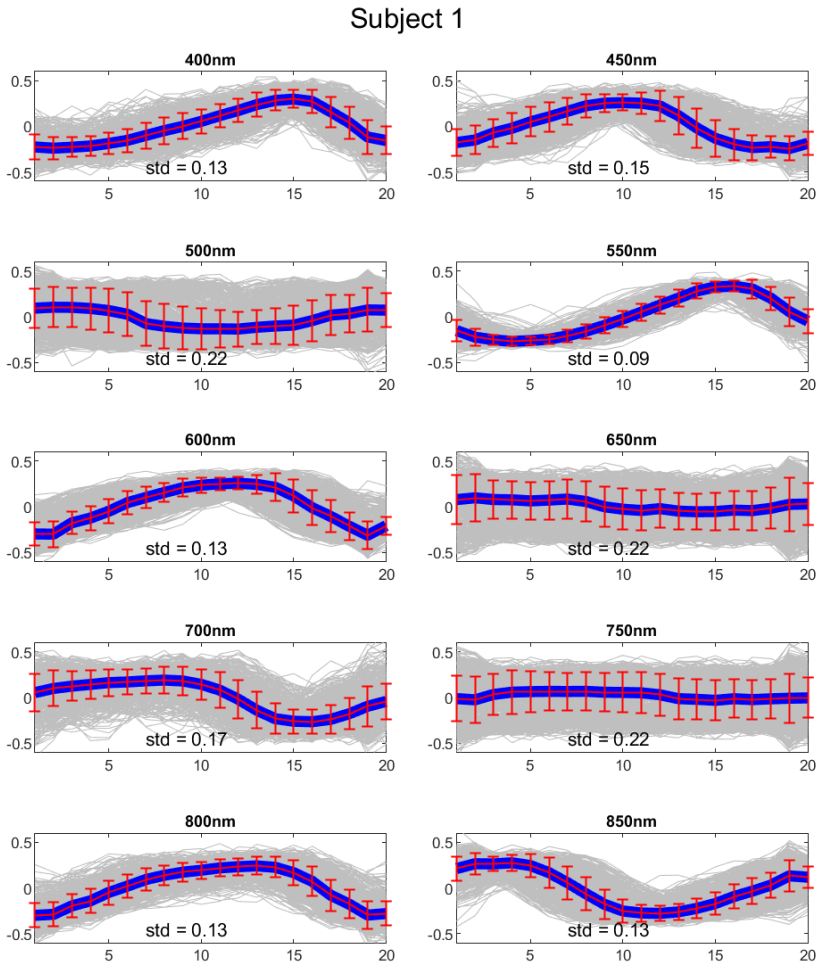

Figure 3.17: *Average beat shape for subject 1 for each one of the wavelengths. Each beat shape is reported in gray and the blue lines represent the average. The red intervals represent the standard deviation for each sampling point. Values on the y axes refer to normalized amplitudes.*

using 850 nm light; excluding some outlier results, using this wavelength good results could achieve and this could be useful in situations in which an active visible illumination could not be possible to use.

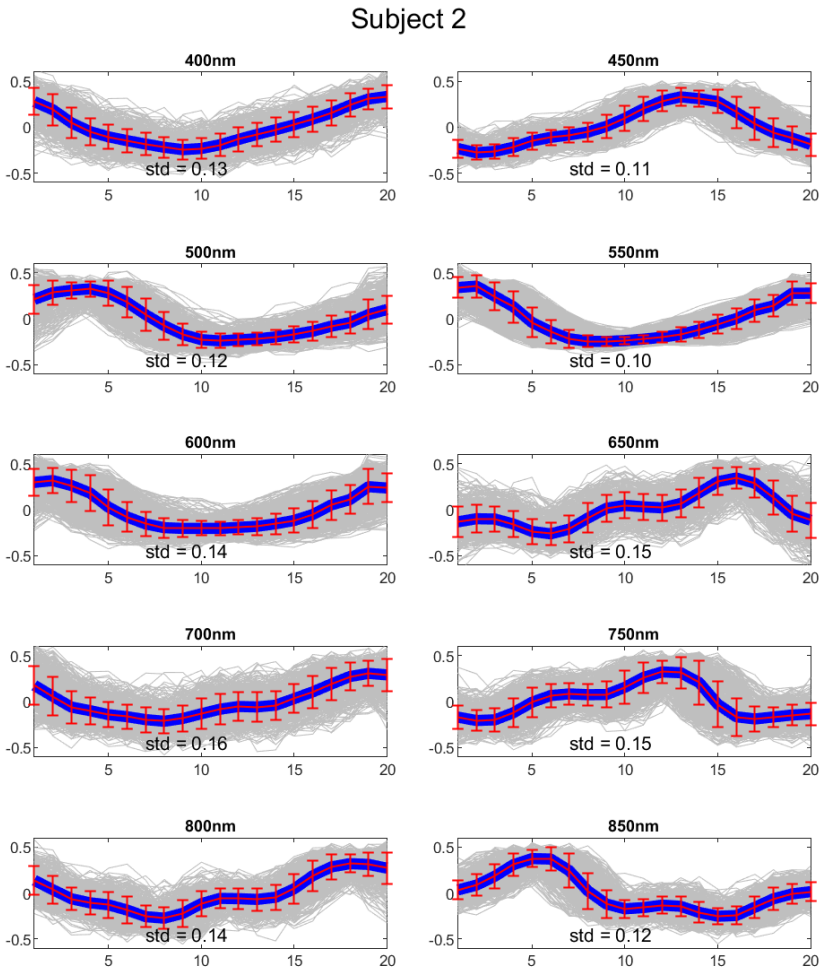

Figure 3.18: *Average beat shape for subject 2 for each one of the wavelengths. Each beat shape is reported in gray and the blue lines represent the average. The red intervals represent the standard deviation for each sampling point. Values on the y axes refer to normalized amplitudes.*

Tachogram

In Fig. 3.22 the tachogram extracted from the SPAD camera (blue lines) and from the portable ECG device (red lines) are reported for all the acquisitions of one of the experiment subjects. As can be observed, all the estimated

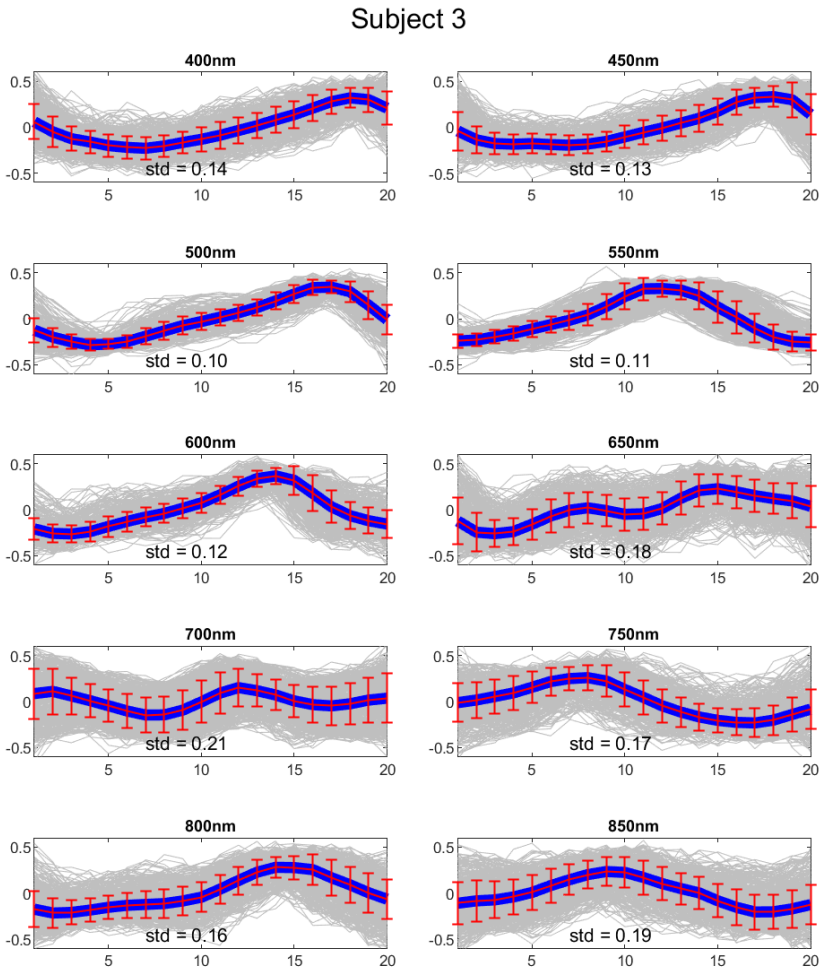

Figure 3.19: *Average beat shape for subject 3 for each one of the wavelengths. Each beat shape is reported in gray and the blue lines represent the average. The red intervals represent the standard deviation for each sampling point. Values on the y axes refer to normalized amplitudes.*

SPAD tachograms correctly have the ground truth line as the mean value. In particular, the one obtained with the filter at 550 nm wavelength is the one with the lowest fluctuations. In Tab. 3.3 the complete RMSEs between the estimated curves and the ground truth ones are reported. In case of

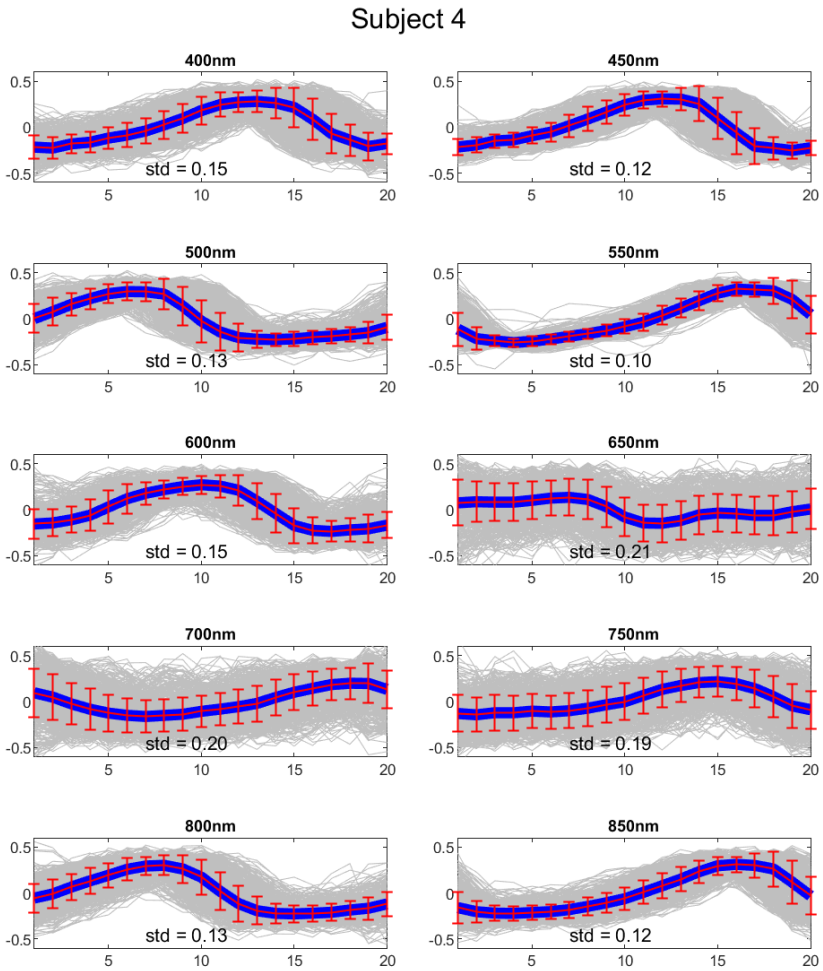

Figure 3.20: *Average beat shape for subject 4 for each one of the wavelengths. Each beat shape is reported in gray and the blue lines represent the average. The red intervals represent the standard deviation for each sampling point. Values on the y axes refer to normalized amplitudes.*

subject 4 and 5, in the acquisition at 700 nm the algorithm couldn't detect enough beats due to the noisiness of the signal, thus it was impossible to extract the tachogram. As can be observed the lowest errors are reached while using the 550 nm wavelength filter.

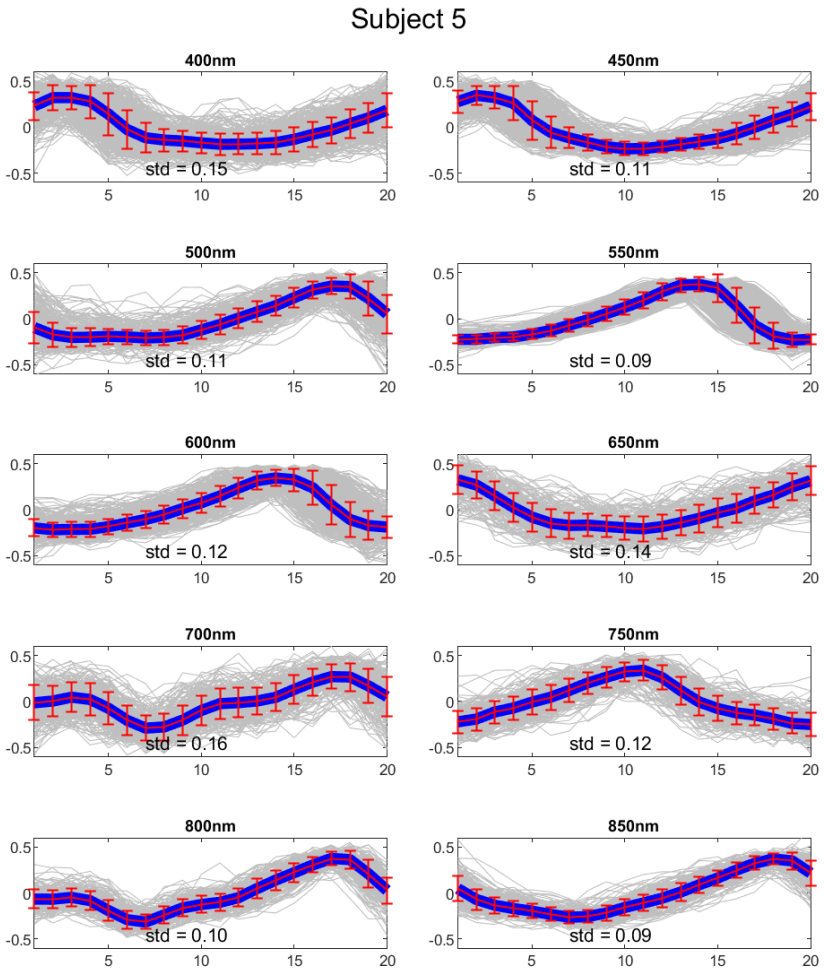

Figure 3.21: *Average beat shape for subject 5 for each one of the wavelengths. Each beat shape is reported in gray and the blue lines represent the average. The red intervals represent the standard deviation for each sampling point. Values on the y axes refer to normalized amplitudes.*

LF/HF

Furthermore in Tab. 3.4 the HF/LF ratio percent errors are reported. As can be observed from Fig. 3.23, performing rPPG using the SPAD camera, and the tachogram estimation techniques described in Sec. 3.4.5, could retrieve

Table 3.2: *Errors calculated as absolute differences between heart rate obtained from ECG track and SPAD pulse signals.*

	Wavelength [nm]									
[bpm]	**400**	**450**	**500**	**550**	**600**	**650**	**700**	**750**	**800**	**850**
Sbj 1	0	2	4	1	3	7	4	2	1	0
Sbj 2	27	41	0	0	0	9	32	22	50	24
Sbj 3	18	1	3	2	14	15	12	17	9	18
Sbj 4	0	0	0	1	2	12	N.A.	6	1	0
Sbj 5	4	6	3	0	3	1	N.A.	1	0	0
Avg.	9.8	10.0	2.0	**0.8**	4.4	8.8	16.0	9.6	12.2	8.4

Table 3.3: *Errors calculated as mean square error (MSE) between tachogram obtained from ECG track and SPAD pulse signals.*

	Wavelength [nm]									
[s]	**400**	**450**	**500**	**550**	**600**	**650**	**700**	**750**	**800**	**850**
Sbj 1	0.10	0.04	0.08	0.03	0.07	0.11	0.10	0.07	0.06	0.03
Sbj 2	0.05	0.07	0.03	0.07	0.23	0.22	0.23	0.15	0.09	0.06
Sbj 3	0.09	0.07	0.06	0.03	0.08	0.17	0.17	0.15	0.10	0.28
Sbj 4	0.07	0.05	0.05	0.03	0.07	0.21	N.A.	0.18	0.11	0.07
Sbj 5	0.12	0.10	0.14	0.02	0.08	0.30	N.A.	0.12	0.05	0.05
Avg.	0.09	0.07	0.07	**0.04**	0.11	0.20	0.17	0.13	0.08	0.10

Table 3.4: *HF/LF RMSE between the SPAD estimation and the ECG ground truth one.*

	Wavelength [nm]									
	400	**450**	**500**	**550**	**600**	**650**	**700**	**750**	**800**	**850**
Sbj 1	1.6	1.5	1.1	1.3	2.1	1.6	1.7	1.2	1.3	1.5
Sbj 2	0.3	0.4	1.2	0.5	0.9	0.9	0.5	0.6	0.3	0.3
Sbj 3	1.0	1.7	1.4	1.3	1.8	4.2	1.2	2.8	2.5	3.3
Sbj 4	1.3	1.2	1.8	0.8	1.1	1.9	N.A.	1.5	1.8	1.7
Sbj 5	1.2	0.8	2.5	0.1	0.5	1.4	N.A.	0.8	0.6	0.6
Avg.	1.1	1.1	1.6	**0.8**	1.3	2.0	1.1	1.4	1.3	1.5

some information on a relatively hard task as remotely obtaining information about the simpatho-vagal balance. In particular, the best results are achieved at the 550 nm wavelength with a average RMSE of 0.8, which is a state of the art result as reported in [32].

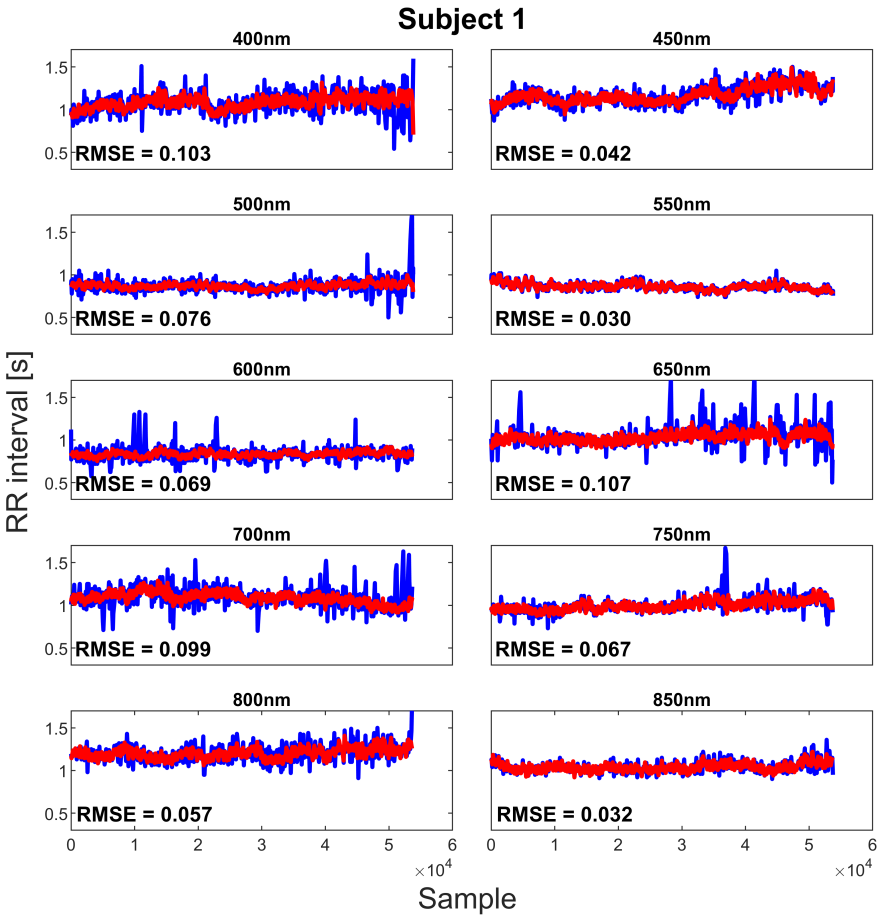

Figure 3.22: *Tachogram estimation results obtained for subject 1 for all the tested wavelengths. Blue: tachogram extracted by pulse wave; Red: tachogram calculated using ECG track.*

Respiration rate

Lastly in Tab. 3.5 the respiration rate errors are reported. As can be observed, the respiration rate could be estimated with a high accuracy using all the different wavelengths reaching the best results while using the 550 nm optical filter.

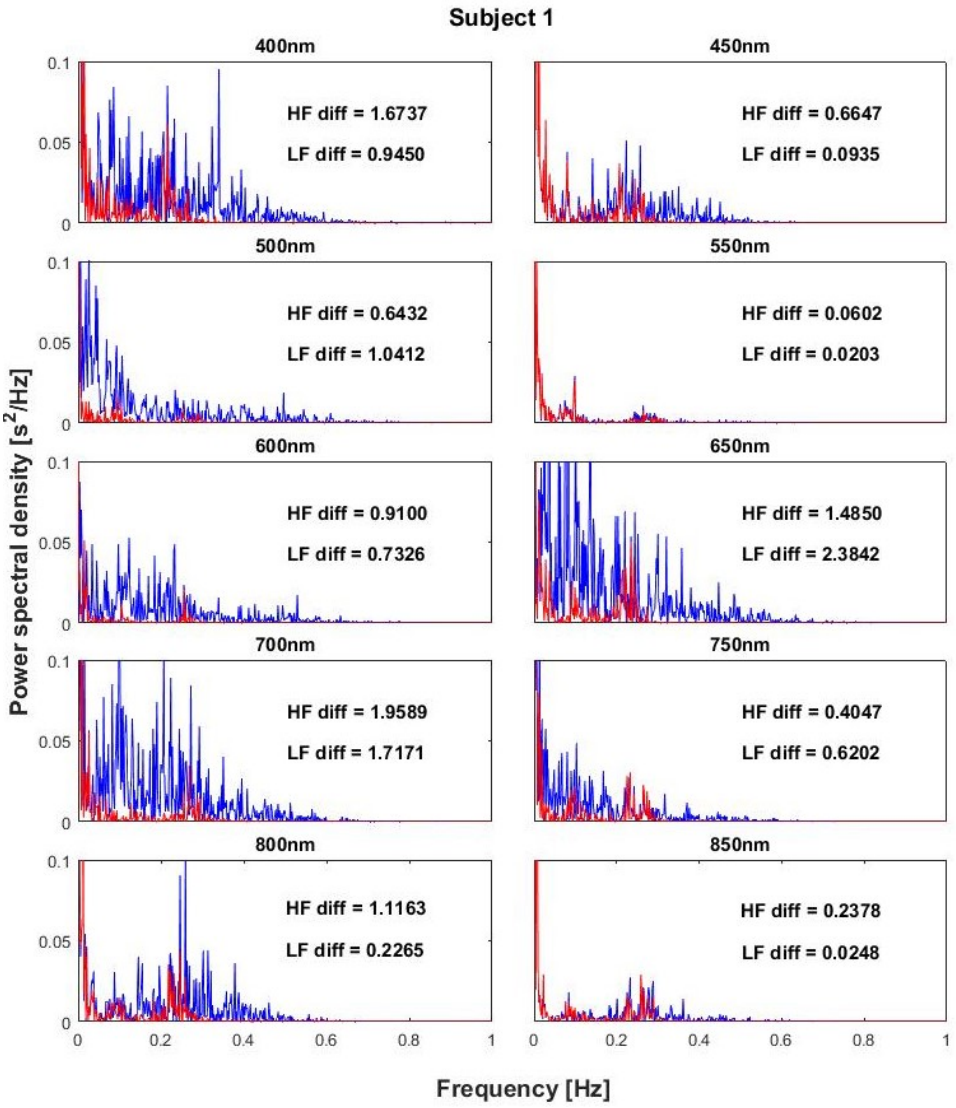

Figure 3.23: *Tachogram spectrum estimation results obtained for subject 1 for all the tested wavelengths. Blue: tachogram spectrum extracted by pulse wave; Red: tachogram spectrum calculated using ECG track.*

3.5.2 Exp. 2 - SPAD and RGB cameras comparison

Heart rate

As reported in Sec. 3.3.2, in order to evaluate the accuracy in determination of heart rate expressed in beats per minute, the maximum of the power

Table 3.5: *Respiration rate errors between the SPAD estimation and the ECG ground truth one.*

	Wavelength [nm]									
[bpm]	**400**	**450**	**500**	**550**	**600**	**650**	**700**	**750**	**800**	**850**
Sbj 1	0.4	0.3	0.4	0.2	0.4	0.7	0.5	0.4	0.3	0.3
Sbj 2	0.4	0.2	0.4	0.0	0.2	0.3	0.4	0.1	0.3	0.1
Sbj 3	0.3	0.5	0.1	0.1	0.7	0.4	0.7	0.8	0.6	0.5
Sbj 4	0.3	0.1	0.3	0.5	0.4	0.5	N.A.	0.6	0.6	0.4
Sbj 5	0.4	0.7	0.4	0.3	0.2	0.7	N.A.	0.2	0.6	1.1
Avg.	0.36	0.36	0.32	**0.22**	0.38	0.52	0.53	0.42	0.48	0.48

Table 3.6: *Average errors in determination of heart rate in one-minute windows.*

Error [bpm]	**SPAD**	**RGB**
Sbj 1	0.0	0.0
Sbj 2	0.2	0.2
Sbj 3	0.2	0.2
Avg.	0.1	0.1

spectral density of each device was calculated for each one-minute window and the difference between the maxima of each camera and the ECG was calculated. As the first and the last minute of the acquisition were removed in order to avoid motion artifacts and consider the subjects under static conditions, 8 one-minute windows were considered. The errors, calculated in each windows as the absolute difference in heart rate determination between the devices for each subject, were averaged in order to find the average error in detection of heart rate. An example of one-minute windows spectrum is reported in Fig. 3.24. Heart rate estimation results are shown in Tab. 3.6. The table shows that the developed setup and signal processing allow a high accuracy in the determination of the heart rate, showing an average error lower than 0.2 bpm. It is important to notice that the resolution is limited to 1 bpm due to the temporal length of the considered window (1 min).

Tachogram

The tachogram metric is obtained by computing the root mean square error between the tachogram obtained from the SPAD and the one from ECG, and between the tachogram of the RGB camera and the ECG.

The deviations calculated over the entire tachograms are reported in Tab. 3.7. From that table, for two subjects (Sbj1 and Sbj2) results are equiv-

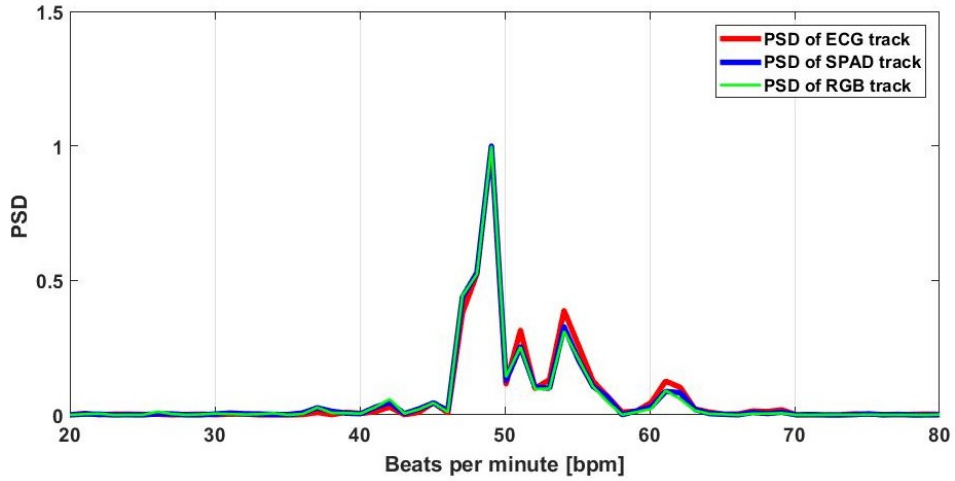

Figure 3.24: *Example of heart rate estimation in a one-minute window using RGB and SPAD camera. It could be noticed that both cameras are able to estimate a HR of 49 bpm in this window, that exactly match the heart rate calculated with the ECG track.*

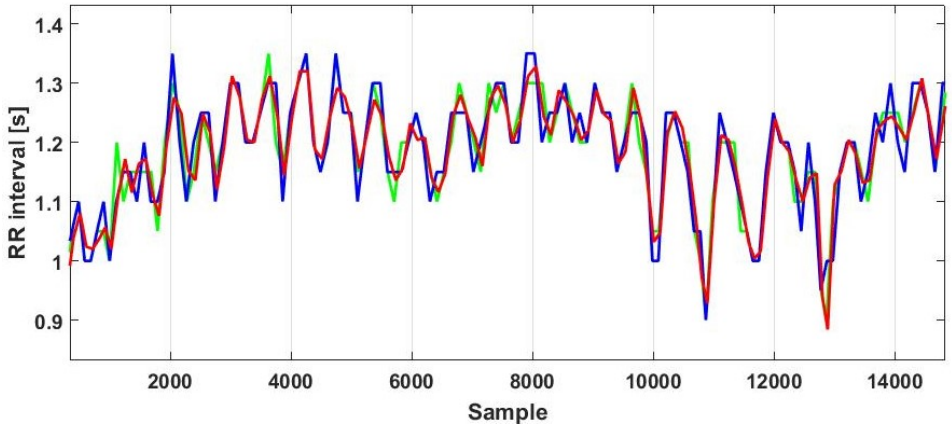

Figure 3.25: *Tachogram estimation obtained for subject 1 using signal extracted by SPAD camera (blue), RGB camera (green) and Faros 180 (red).*

alent in terms of accuracy, while for the third subject the SPAD camera returns better results, because of a beat missing in the RGB tachogram estimation. The tachogram estimation for subject 1, using signal extracted by SPAD camera (blue), RGB camera (green) and Faros 180 (red), is shown in Fig. 3.25

Table 3.7: *Mean square error between tachogram extracted from cameras and ECG.*

RMSE [ms]	SPAD	RGB
Sbj 1	2	2
Sbj 2	0.8	0.7
Sbj 3	0.6	6
Avg.	**1.1**	2.9

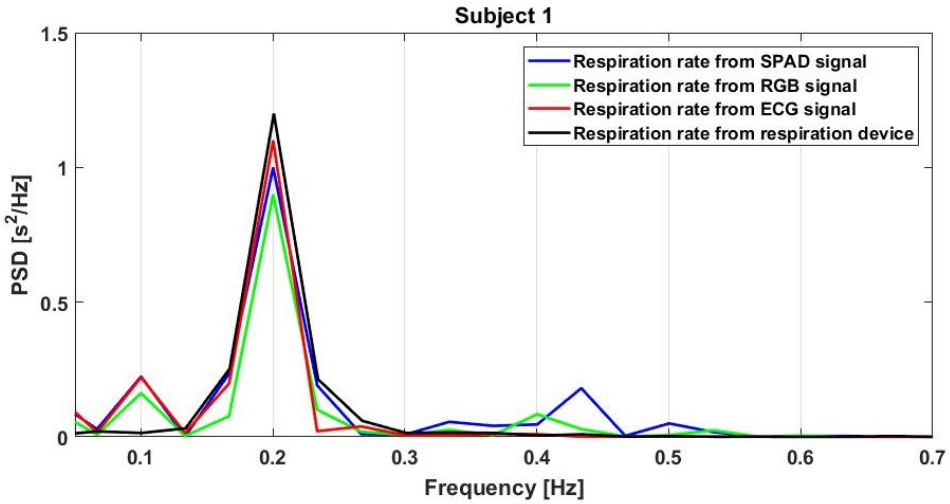

Figure 3.26: *Example of respiration rate calculation, computed as the FFT of the tachogram of the signals from the three devices compared with the FFT of the signal measured with the respiration measurement device.*

Respiration rate

Finally the respiration rate measurement accuracy was calculated. Respiration rate was obtained by performing an FFT on the previously calculated tachogram. As breath is a non-autonomic action, the breathing frequency can vary significantly during the 10 minutes acquisition For this reason it was decided to divide the tachogram in windows of thirty seconds, in order to see less variations in the frequency an better focus on the accuracy of one clear respiration rate. Graphical results are shown in Fig. 3.26. Tab. 3.8 shows the RMSE in the respiration rate estimation obtained with this experiment, by using the SPAD camera and traditional RGB camera. A slight improvement could be observed by performing rPPG using a SPAD camera.

RMSE [breath in 10 min]	SPAD	RGB
Sbj 1	1.0	0.6
Sbj 2	1.1	1.8
Sbj 3	0.5	0.7
Avg.	**0.9**	1

Table 3.8: *Average errors in respiration rate calculation. Errors calculated as mean square errors between the measurements taken with the breathing sensor and the three devices.*

3.6 Discussion and conclusions

In this chapter the possibility of performing rPPG using a SPAD camera to compute HR, HRV and RR had been investigated. The working principle and reason behind the use of SPAD cameras had been discussed in Sec. 3.1. In this work two experiments have been set up, performing measurements on a subject sat still in front of the camera with the artificial illumination directed on its face. The values of the pixels inside a manually obtained ROI were averaged resulting in a pulse wave. This was the starting signal that was processed in order to estimate HR, HRV and RR. In order to evaluate SPAD based rPPG five parameters were considered: single beat detection, heart rate estimation, tachogram estimation, LF/HF estimation and respiration rate estimation. In order to perform and validate biometric measurements with a SPAD camera and compare it to estimation that could be obtained from a traditional RGB camera, a portable ECG device was used for reference. One of the two experiments conducted (experimental setup described in Sec. 3.3.1) had the aim of comparing the SPAD rPPG performance using light with different wavelength. As can be observed from results reported in Sec. 3.5.1, 550 nm light (i.e. green light) is able to achieve the better results. Many parameters influence this result, in particular the most significant are light penetration depth in the tissues [8], absorption coefficient of the oxygenated hemoglobin [126], SPAD efficiency [13] and illumination power. Light below 500 nm is mostly reflected by stratum corneum, which is the most external skin layer, which being not reached by blood does not contain any information on pulse wave. Concerning light between 600 nm and 750 nm, the absorptivity of oxygenated hemoglobin is very low, thus reducing the modulation in rPPG signal. Therefore, only wavelengths between 500 nm an 600 nm and between 750 nm and 900 nm are able to carry useful signal. As a matter of fact, as shown from the results reported in Sec. 3.3.1, the best performance are achieved using 550

nm light but reasonable results are also achieved using near infrared light (750 nm to 850 nm). This is promising results since many scenarios could be imagined in which the use of non-visible light could be preferred (e.g. in the automotive field an rPPG system could be used in order to monitor the health state of the driver).

The second experiment (described in Sec. 3.3.2) was conducted in order to compare the rPPG SPAD based performance with the one obtainable using traditional RGB cameras. As can be observed in a normal light scenario, as reported in Sec. 3.5.2, SPAD cameras are able to achieve comparable results in respect to RGB cameras in heart rate estimation and slightly superior accuracy in estimation of the tachogram and respiration rate.

CHAPTER 4

Skin Detection on SPAD Camera

SCOPE & AIMS: The scope of this chapter is to propose an automatic method with the aim of solving the task of detecting skin pixels in grayscale low resolution face images, as the one obtained using a SPAD array camera.

METHODS: Since the facial skin detection problem is very specific, very few data are available for this specific problem. For this reason a transfer learning approach was adopted in the training phase. In particular, a Convolutional Neural Network model was trained starting from a method originally proposed to solve a colorization problem.

RESULTS: A new dataset is proposed and made publicly available in order to tackle the skin detection problem. A novel model was trained in a transfer learning framework. Quantitative and qualitative results show the proposed method adequately solves the skin detection problem.

PUBLICATIONS: The main part of this chapter was published as a journal paper [132].

4.1 Problem description

Skin detection is an important preliminary task in a wide range of image processing problems and in particular remote PhotoPlethysmoGraphy. Many rPPG applications [93] estimate the face regions in which to extract the signal using a combination of classical face detection methods, such as [117], and fixed proportions in order to select specific parts of the face, e.g. typically the forehead. This procedure is not optimal since the skin in preselected parts of the face could not be visible due to occlusions of hair, wearable objects or other elements. Furthermore skin segmentation based on a predefined template suffers from errors of the face detection phase and/or due to intrinsic variance of face shapes. Moreover due to the high variability of the subject pose, motion blur, age, ethnicity, hair, facial hair, wearable objects, etc., the first step of a rPPG application (i.e. selecting the face region in which to extract the signal) is not trivial and errors in this step could heavily compromise the final hearth rate estimation. The majority of rPPG applications [93] utilize a standard RGB camera, based on CMOS or CCD technologies, in order to acquire the video stream. The goal of this chapter is to propose a skin detection algorithm able also to work when applied to images acquired using SPAD cameras. The high precision of SPAD cameras is useful in measure accurately the skin intensity fluctuations produced by the blood flow. On the other hand, due to the complexity of the SPAD sensor, this kind of cameras has a very small spatial resolution, 64x32 in [14], and produces grayscale intensity image, since the low spatial resolution does not allow the use of Bayer filters.

In this chapter we propose an automatic method, based on deep learning, with the aim of solving the task of detecting skin pixels in face images. Furthermore, for the aforementioned reasons, the proposed method is designed to work with low resolution grayscale images such the one obtained using a SPAD array camera [14]. The rest of the chapter is organized as follows: in Sec. 4.1.1 a brief state of the art review on skin detection is reported highlighting the peculiarity of the problem addressed in this work; in Sec. 4.2 the proposed method is described while in Sec. 4.2.3 the training procedure exploiting transfer learning is illustrated; qualitative and quantitative results are shown in Sec. 4.3 and finally in Sec. 4.4 the contributions of this work are highlighted.

4.1.1 State of the art

The skin detection problem is usually tackled using color information and exploiting the fact that skin-tone colors share some common properties de-

fined in particular color spaces [54]. After applying the optimal color space transformation it is possible to define rules to discriminate between skin pixels and other materials. Since these kinds of methods are based on color information, they obviously require color images (RGB) to be applied to. As stated in Sec. 4.1, due to the choice of developing a method able to work with SPAD camera output (grayscale), this class of methods could not be applied in this specific problem. Moreover, they have no way to discriminate between the face and other body parts and this could be a problem in rPPG in which, due to the blood flow dynamic in the body, different body parts could carry different information (i.e. time-shifted signal). An extensive review of color based skin segmentation methods could be find in [50]. Some skin detection methods able to work with grayscale images exist, e.g. [96], but they achieve good results only working with high resolution images since they learn local texture characteristics.

Another problem, related to the one described in Sec. 4.1, is face parsing or face segmentation, which is the problem to analyze an input image of a face and densely segment it in different regions corresponding to different face parts and the background [125]. This is performed by labeling pixels in a dense fashion, i.e. to each pixel a label is assigned. In recent years, many deep learning methods have been proposed in order to solve this kind of problems, e.g. [65], [83] and [125], exploiting the promising results achieved by neural network based methods in semantic segmentation [33]. Even though this problem is very similar to the one tackled in this paper (e.g. this last could be view as a simplified segmentation problem with just two classes, i.e. skin and other) some differences exist in the definition of the two problems. In fact, in face parsing methods, wearable objects such as glasses and sunglasses, or facial hair are not separated from the face region in which they are present, making this kind of methods not suitable for the skin detection problem. Moreover, methods such the ones proposed in [125] and [65] work on high resolution color images. To the best of our knowledge no other method specifically designed to solve the skin detection problem on low resolution grayscale images exists in the state of the art.

4.2 Methods

As described in Sec. 4.1.1, deep learning based methods represent the state of the art for segmentation problem and they usually require a massive amount of data. On the other hand, due to the uniqueness of the skin detection problem, the amount of data available is very limited. For this reason

a transfer learning [85] procedure was adopted. In particular, a colorization network [9] was adapted for the skin detection task. The main reason behind the choice of exploiting a colorization method as a starting point for the proposed network is the empirical observation that a method of this kind applied to a grayscale image that depicts a face correctly colorize each skin pixel with the proper skin color. This means that the network must have learned a way to discriminate between skin pixels and pixels that depicts other objects. Furthermore, both the skin detection problem described in Sec. 4.1, and the colorization problem share the same kind of input, i.e. grayscale image. Moreover collecting training data in order to train a colorization network is trivial and the problem could be seen as a self supervised one. The driving idea is to propose slight changes to the colorization network in order to be able to transfer as much knowledge as possible from the colorization task to the skin segmentation one and then use a fine tuning approach.

4.2.1 Colorization network

As stated in Sec. 4.2 due to unavailability of labeled data, the approach chosen in order to create a skin detection method was modifying, adapting and retraining a preexisting network. The chosen network is a colorization network presented in [9]. The purpose of this network is to assign colors to grayscale input images. In particular, given an input image, of size $H \times W$, represented in CIE L*a*b* color space [90], whose only the luminance (L*) component is known, $\mathbf{X_L} \in \mathbb{R}^{H \times W \times 1}$, the colorization method is used in order to generate the remaining channels and obtaining $\tilde{\mathbf{X}} \in \mathbb{R}^{H \times W \times 3}$, the corresponding color image [9]. Especially, the colorization neural network is used in order to approximate the colorization function $\mathcal{F}$:

$$\mathcal{F} : \mathbf{X_L} \rightarrow (\tilde{\mathbf{X}}_\mathbf{a}, \tilde{\mathbf{X}}_\mathbf{b}) \tag{4.1}$$

Where $\tilde{\mathbf{X}}_\mathbf{a}$ and $\tilde{\mathbf{X}}_\mathbf{b}$ are respectively the a* and b* channels of $\tilde{\mathbf{X}}$.

Network architecture

The author of [9], following the work presented in [44], propose to use a CNN composed by two different branches in order to reconstruct the a*b* starting from L*. In this work [9], the auxiliary branch is obtained using Inception-ResNet-v2 network (referred as Inception in the rest of the chapter) [125], a widely used CNN architecture, in order to analyze the input image and retrieve relevant information [9]. The complete network architecture could be found in Fig. 4.1. As can be observed from this figure, the

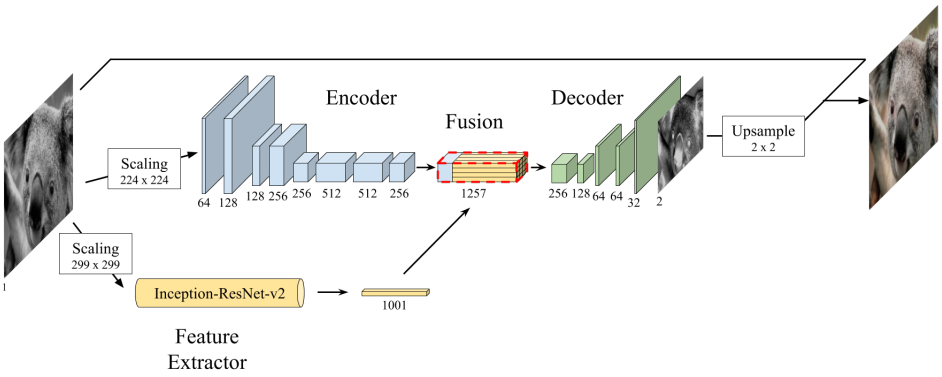

Figure 4.1: *An overview of the deep learning based colorization method combining CNN and Inception-ResNet-v2 proposed in [9].*

grayscale input image is analyzed jointly by a costume cascade of convolutional layers (i.e. Encoder) that extract low and mid-level features and by the additional parallel branch represented by the Inception network which extract high-level features. After vectorizing the Inception output, all the features extracted in parallel are then merged in the fusion layer. Lastly the decoder part of the network analyses the merged feature in order to produce the desired output (estimated a* and b* components). Inputs are scaled in order to obtain values in the $[-1, 1]$ range for avoiding convergence problems in the learning phase.

The main branch (i.e. Encoder-Decoder) follows the classical implementation of an autoencoder. In particular in the encoder stage, 8 convoulutaionl layers with 3×3 kernels are applied on the $H \times W$ input, using a stride of 2 in the first, third and fifth layers in order to obtain a $H/8 \times W/8 \times 512$ feature representation. In the parallel branch the Inception network (without the top classification layers) is applied in order to extract a $1001 \times 1 \times 1$ representation of the input image containing high level semantic information of the image such as "underwater", "indoor scene", etc. [9]. In the fusion layers the results coming form the two parallel branches are merged into a single layer by replication and concatenation. In particular, the feature vector obtained form the Inception branch is replicated $(HW/8)^2$ times and attaches it to the feature volume outputted by the encoder along the depth axis, following the approach introduced and described in [45]. By replicating the feature vector and concatenating it several times the semantic information is uniformly distributed among all spatial regions of the image. An additional convolution layer is then applied

Table 4.1: *Colorization network architecture [9]. Blue layers are used for transfer learning.*

Encoder		**Decoder**	
Layer	**Kernels**	**Layer**	**Kernel**
Conv. (str. 2x2)	64x3x3	Conv.	128x3x3
Conv.	128x3x3	Upsamp. 2x2	
Conv. (str. 2x2)	128x3x3	Conv.	64x3x3
Conv.	256x3x3	Conv.	64x3x3
Conv. (str. 2x2)	256x3x3	Upsamp. 2x2	
Conv.	512x3x3	Conv.	32x3x3
Conv.	512x3x3	Conv.	2x3x3
Conv.	256x3x3	Upsamp. 2x2	
Fusion			
Conv.	256x1x1		

on the fused feature vector in order to obtain a $H/8 \times W/8 \times 256$ output. Lastly, following a classical autoencoder architecture [31], a decoder stage is applied to the obtained $H/8 \times W/8 \times 256$ layer alternating convolutional layers and up-sampling in order to obtained the desired $H \times W \times 2$ output containing the input color information. A complete representation of the main branch network topology is described in Tab. 4.1.

Training procedure

The training procedure described in [9] is straight forward since the network illustrated above is then trained minimizing the mean square error between the groundtruth color values (i.e. a*b* components) and the ones estimated by the network over all the pixels in a given training image. The chosen minimization algorithm is Adam Optimizer [56] with an initial learning rate set to $\eta = 0.001$. The colorization network has been trained on a subset of ImageNet [25] containing approximately 60'000.

Results

This approach produces high quality colorized images, the authors performed a survey on 41 users asking to guess if a set of images were colored correctly, resulting in some recolored images classified as real up to 80% of the time and in general 45.87% of users miss-classified one or more recolored images as original [9].

4.2.2 Skin detection network architecture

As described in Sec. 4.1 the goal of this chapter is to introduce a method that is able given a grayscale image to label each pixel of it as depicting skin or not. In order to do so a deep learning method is introduced with the purpose of approximating a function called $\mathcal{F}$ defined as:

$$\mathcal{F} : \mathbf{X} \rightarrow \mathbf{Y} \tag{4.2}$$

Where $\mathbf{X}$ is the input image and $\mathbf{Y}$ is the method output having the same dimension of the input (number of rows and columns) and containing real values in the real interval $[0, 1]$ which for each pixels represent the estimated probability of having skin in that particular location of the input image. In particular, the described method is introduces in order to solve the *relaxed* skin detection problem, which expresses the problem of assigning a probability value to each pixel instead of a binary output. The needed binary output is then obtained by applying a fixed threshold to the continuous output.

As outlined above, the colorization network presented in [9] is based on a convolutional autoencoder with an auxiliary parallel branch. This additional branch is used to extract a vectorized high level representation of the image semantic. This vector is then merged to the encoded representation of the main branch before performing the decoding part. In particular, this operation is performed to help the colorization method better understand the scene depicted in the input image, in order to colorize more precisely a large variety of objects and scenes. On the other hand, this auxiliary branch is totally unnecessary in the case that the input images are *a-priori* known to contain just a single human face. Even if its role was crucial in the [9] approach, in the proposed skin detection network this additional branch was completely removed, providing us with a suitable architecture. Another major difference between the proposed network topology and the one proposed in [9] resides in the output layer dimension. In particular, for each given grayscale input image, the original colorization network outputs a two channels image relative to the a* and b* channels of the L*a*b* [90] color representation of the image. On the other hand, as described above, the proposed network needs to output a single channel image with each pixel value $\hat{y}_{ij} \in [0, 1]$. This is achieved substituting the last activation function with a sigmoid function. In particular, for each pixel of the output image, its value represents the probability attributed by the network of the input image having a skin pixel in that particular location. As reported in Fig 4.2, the encoding part of the network is composed by 8 convolutional layers, with 3x3 kernels and ReLu activation functions, and 3 max pooling

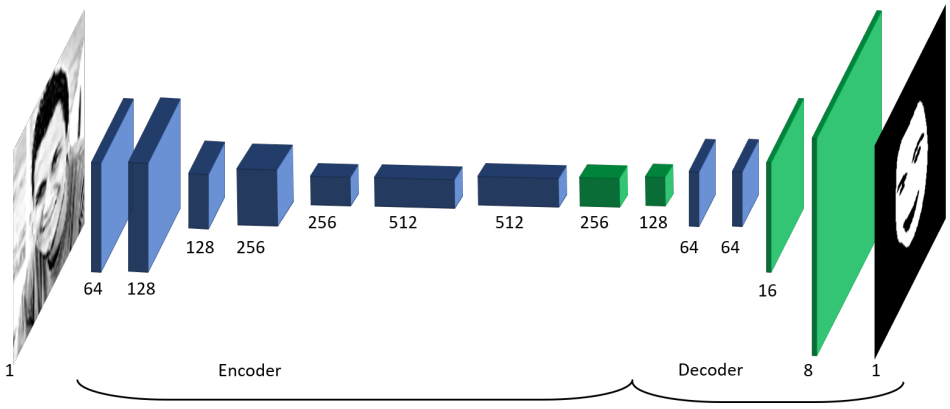

Figure 4.2: *Proposed network topology. The green layers and the last one are trained from scratch while for the blue ones the knowledge is transferred from the colorization network. The number under each layer indicate the dimension of its output (number of filters).*

layers in order to reduce the spatial dimension in the last encoding layer to 1/8 of the original input dimension. On the other hand, the decoding part is composed by 6 layers with 3x3 kernels and ReLu activation functions (except the last one, which is a sigmoid function in order to output values in $\in [0, 1]$) coupled with upconvolutional layers to increase back the spatial dimension to the input one.

As can be observed by comparing the architecture of the colorization network proposed in [9], reported in Fig. 4.1 with the skin detection one represented in Fig. 4.2, the two methods share a large portion of their structure. In particular, the encoder topology is identical with the exception of the removal of the last hidden layer due to the less complexity of the faced task. As said above, the feature extractor additional branch, represented by the Inception network was removed and consequently the fusion layer. The decoder part is similar but simplify both for the simplicity of the skin detection task in respect to the colorization one, and for the uselessness in the transfer learning framework since this layers, i.e. the final ones, are the ones closer related to each task. In Fig 4.2 the layers colored in blue are the ones trained propagating the colorization network knowledge while the other ones are trained from scratch. The central ones (with output depth 256 and 128) need to be trained with no prior information due to the removal of the colorization fusion layer. The next two have input and output shapes as in [9] so their weights value is propagated. Finally, the last ones, since are introduced to solve the skin detection problem, are randomly initialized.

4.2.3 Training procedure

As stated in Sec. 4.2, the training procedure, adopted to estimate the optimal network parameters, is based on a transfer learning approach. In particular, using the general definition given in [63], transfer learning is the act of exploiting the knowledge acquired for solving a particular learning task $\mathcal{T}_s$ in a particular domain $\mathcal{D}_s$, called source, while trying a to solve a different task $\mathcal{T}_t$ in a different domain $\mathcal{D}_t$, called target, with $\mathcal{T}_s \neq \mathcal{T}_t$ and $\mathcal{D}_s \neq \mathcal{D}_t$. In our case, the source problem is represented by the colorization task while the skin detection represents the target. In order to fatherly increase the probability of successfully implement transfer learning the source task was slightly modify in order to move closer to the target one. In particular instead of solving the colorization problem for any kind of input images only images depicting human faces had been chosen. This implies that, the colorization network described in Sec. 4.2.1 instead of being trained on a generic image dataset, such as Imagenet [25] as done in [9], needs to be trained on an appropriately chosen dataset. The same loss function and optimization algorithm as the ones described in the original work were used, i.e. mean square error, between the ground truth color image and the one reconstructed by the network, and Adam Optimizer [56] respectively; further information on the colorization network training are reported in Sec. 4.2.1.

The dataset Labeled Face in the Wild (LFW) [42] was chosen in order to perform the colorization training step. This is a public benchmark for numerous computer visions tasks related to human faces. Although this dataset does not contain an equal number of samples for each ethnic subgroups it was considered appropriate especially for the pretraining process. It contains more than 13,000 images of faces collected from the web [42]. All the images in this dataset are obtained in an uncontrolled environment with variations in lighting, pose and in presence of occlusions so the names "Faces in the wild". Some example of images contained in LFW are reported in Fig. 4.3. In order to increase the number of samples the horizontal flipped version of each image was added, doubling the sample size reaching over 26,000 images (in this case horizontal flipping is a safe techniques given the symmetry of a human face). Even if this number is less then the number of training samples used in [9] considering that the task is more limited it is enough to train the face colorization network.

In order to perform the colorization network training step, the colored images were used as the desired ground truth output while a grayscale representation of them were used as the network input. The model was trained exploiting the implementation described in the original paper [9] which is

Figure 4.3: *Samples of images extracted from the Labeled Face in the Wild [42] dataset and used to train the colorization network.*

implemented using synergistically Keras [22] and Tensorflow [1]. After performing the training of the face colorization method, the shared layers between the two networks (the ones colored in blue in Fig 4.2) were frozen (i.e. set to not trainable) and their weights value set to the corresponding one obtained from the colorization training step described above. The other ones were randomly initialized. The network was trained using Keras [22], with Tensorflow [1] as backend, with the Adam Optimization algorithm [56] and a learning rate of 0.0005. The loss function and the datasets used are described in Sec. 4.2.3 and Sec. 4.2.4 respectively. After a sufficient amount of epochs, 50, a fine tuning step was finally performed in which all the layers were trained on the whole dataset and using the same training conditions, for an additional 100 epochs on the same training set. The following list recap the adopted transfer learning training procedure.

Transfer learning training procedure

1. The colorization architecture described in Sec. 4.2.1 was trained using a set of more than 26,000 face images.

2. The initial training of the skin detection method described in Sec. 4.2.2

was performed on data described in Sec. 4.2.4 keeping frozen the values of layers weight shared with the face colorization network.

3. The complete skin detection network was trained with a smaller learning rate in order to fine tune the network performances.

Loss function

The loss function is a key component in the training phase. In particular different choice of it could steer the optimized method to a particular direction selecting some characteristics instead of others. On the other hand, the mean square error could be sufficient in order to train the network to perform the skin segmentation task. Furthermore, considering the main motivation that drives the building of this network (i.e. the rPPG application described in Sec. 4.1), false negative and false positive errors should not equally weighted in the computation of the loss function. In particular, in order to estimate the heart rate of a subject, it is not strictly necessary to consider all visible skin pixels whilst, on the other hand, labeling as skin a pixel depicting other tissues or materials could have an important negative impact on the final estimation. For this reason, given a predicted mask $\hat{y}$ obtained applying the proposed network to an input grayscale image x having a ground truth mask y with elements $y_{ij} \in \{0, 1\}$, we define the loss function as:

$$E(\hat{y}, y) = \sum_{ij} (y_{ij} - \hat{y}_{ij})^2 (\alpha \cdot y_{ij} + (1 - \alpha)(1 - y_{ij})) \tag{4.3}$$

Where $\alpha \in [0, 1]$ is a parameter introduced in order to make E asymmetric. We choose a value for α smaller than 0.5, e.g. 0.4, in order to penalize false positive errors (i.e. $\hat{y}_{ij} = 1$ with $y_{ij} = 0$).

4.2.4 Dataset creation

To the best of our knowledge, there was no dataset available specifically created for the purpose of solving the facial skin segmentation problem. Some skin detection dataset exists, e.g. [110], but they feature images with multiple people and annotations with other body parts. This made them related to a task substantially different in respect to the problem defined in Sec. 4.1, which made them not usable for this particular problem. Moreover the number of images in this dataset is extremely low, e.g. 78 images are present in [110], and insufficient to train deep learning methods. For this reason, we choose to adapt two already existing datasets, proposed for other purpose, i.e. MUCT [77] and Helen [124], consisting of RGB face images

annotated with landmark locations, in order to produce facial grayscale images associated with skin masks. In particular, both datasets provide diversity in lighting, pose, age and ethnicity of the subjects. Moreover, the ones appertaining to the MUCT dataset are acquired in a controlled environment whilst the Helen ones are captured in the wild. A more detailed description on the processing performed on the two dataset is described in the following sections (Sec. 4.2.4 and Sec. 4.2.4 for the MUCT and Helen datasets respectively).

MUCT dataset

As described in [77], the MUCT dataset consists of 3755 images (each one with a resolution of 640x480 pixels) captured from 276 subjects. Each image depicts a single face with a homogeneous blue background and it is associated with the pixel coordinates of 76 manually annotated facial landmarks. This dataset includes students, parents attending graduation ceremonies, high school teachers attending a conference, and employees of the University Of Cape Town university such as cleaners and security personnel [77]. A wide range of subjects was photographed, with approximately equal numbers of males and females, and a cross section of ages and races [77]. During the photo acquisition, in order to increase the dataset variety, five different camera views and three different lighting sets were used. It's one of the most used benchmark dataset for facial landmark detection [77], [131], especially with images taken in controlled conditions. The landmarks provided are relative to the lower face contour, eyes, eyebrows, nose and mouth. Starting from these landmark positions, for each image, a mask is produced considering a filled polygon shape with corners given by the jaw/chin contour points and the eyebrows upper contour. The definition of these landmarks is identical as the 68 points used in XM2VTS [75], plus 4 extra points around each eye [77]. The position of landmarks obscured by hair or glasses was estimated by the human landmarker while those that were obscured, in three-quarter view, behind the nose or side of the face were marked as unavailable [77]. Finally, for each image, the position of all landmarks were carefully checked by a third party [77].

Adapting MUCT to the skin problem

In Fig. 4.4 all the major steps performed in order to adapt the MUCT dataset to the facial skin detection task are reported. In particular in Fig. 4.4 (a) the original data in the MUCT dataset are represented (image coupled with 76 facial landmarks). In Fig. 4.4 (b) the first step is represented in which, ex-

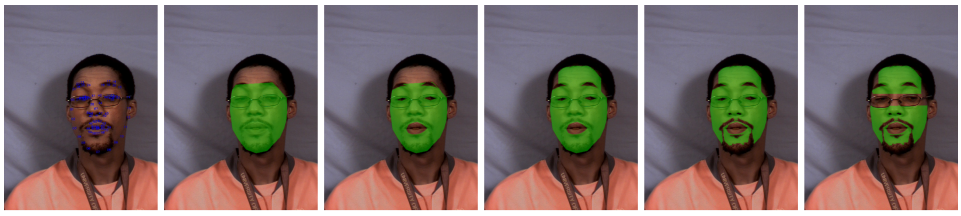

Figure 4.4: *Steps involved in the adaptation of the MUCT dataset. (a) Original data in MUCT (image and landmarks). (b) Face region. (c) Eyes, eyebrows and mouth removal. (d) Forehead addition. (e) Facial Hair removal. (f) Glasses removal. Phases (e) and (f) are executed only on male and people wearing glasses respectively.*

ploiting the labeled coordinate of each of the 67 landmarks, all the facial region is selected. Subsequently in Fig. 4.4 (c) the position of the eyes', eyebrows' and mouth's contours are easily estimated, using the corresponding landmarks position. These regions are then consequently removed from the mask being not related to facial skin. Unfortunately, as in the majority of facial landmark datasets (e.g. PUT [53] and BioID [48]), no upper face contour annotation is provided in this dataset (skin/hair contour). In order to extend the obtained skin masks to the forehead region the second step reported in Fig. 4.4 is performed. In particular, a color similarity method has been used, exploiting the RGB channels information. It is important to notice that the color information is indeed available in this preprocessing step involved in the creation of the dataset but is not available in the network training step. In particular, a rectangular region above the eyebrows is considered; each pixel in that region is then clusterized in 3 different sets using a K-means algorithm and adopting the Euclidean distance in the RGB space. In other words, only the pixels belonging to cluster $\mathcal{S}$ are added to the skin mask, where $\mathcal{S}$ is defined as:

$$\mathcal{S} = \underset{i=1,2,3}{\arg\min} \left\| \mathbf{C}_s - \mathbf{C}_i \right\|_2 \tag{4.4}$$

Where $\mathbf{C}_s \in \mathbb{R}^3$ is the average RGB color value in the skin region found in the first step and $\mathbf{C}_i \in \mathbb{R}^3$ are the centroids of the K-means clusterization for each of the three clusters, $i = 1, 2, 3$. This operation is performed so the pixel belonging to the hair or other occluding objects are rejected. The results of this step is depicted in Fig. 4.4 (d). This method, being automatic and based on color similarity, inevitably introduces some errors in the pixel labeling and produces worse results compared to manual anno-

tation, which is unfortunately unavailable due to the large dataset sample size. The next operation described in Fig. 4.4 (e) regards the removal of beard and facial hair from the skin region. Since in the original dataset facial hair is not labeled, in order to remove it from the mask a similar approach to the one adapted for the forehead region is performed considering the lower part of the face region. This operation is performed only on male subjects since the gender labels are available in the MUCT dataset. Lastly, in this dataset, a binary information on the presence of glasses is provided although their position inside the image is not available. In order to remove the glasses region from the mask the third stage in depicted in Fig. 4.4 (f) is performed. In particular, two rectangles of fixed size and centered around the eyes are subtracted from the mask. All the images in the MUCT dataset are processed with the described method and for each one of them the corresponding skin mask is created.

Helen dataset

The Helen dataset [61] features 2330 high quality, real world photographs of a large variety of people. These dataset was introduced in 2002 gathering images from Flickr using specific general search terms, such as "portrait", "boy", etc. [61]. Using an automatic face detection method high resolution images were generated centered around each found face. The faces with insufficient pixel resolution were discarded. The kept images were cropped around each face obtaining more than 2000 pictures with varying resolution resolution (from less than 1 Mpixel up to 12 Mpixel). The Helen dataset is composed by 2000 training images and 330 additional testing images which do not include any subject from the training dataset [104]. Multiple annotations using this dataset were proposed over the years [61], [124] using different densely annotated facial landmarks configurations. Moreover it has been used for face parsing works [104] in which an accurate face segmentation annotation for different part of the face has been provided. In this particular, the authors of [104] automatically generated ground truth eye, eyebrow, nose, inside mouth, upper lip, and lower lip segments using similar techniques to the ones described in Sec. 4.2.4. Since this automatic segmentation methods could lead to some labeling errors, in order to provide more fair results, the authors of [104] manually annotated a subset of the 300 test images.

Adapting Helen to the skin problem

The masks needed for the skin segmentation problem are simply built combining different segmentation regions given in [104]. Unfortunately, in the Helen dataset many images feature more than one visible face while just one face is annotated in each image. Training our skin detection method on this data could compromise its performances due to a not consistent annotation. In order to avoid this problem a simple state of the art face detector algorithm [117] is run on each image included in the dataset. Since even in presence of multiple faces the ground-truth annotation is always related to just one of faces a new image was created cropping the original images in a region centered around the annotated face. This step, being performed automatically, introduces inevitably some errors. Lastly, also in this dataset, a facial hair annotation is unavailable and the same method used for the MUCT dataset was implemented in order to remove beard regions from the masks.

Complete dataset

The complete dataset is built merging the two datasets obtained as described above resulting in roughly 6000 grayscale face images (converted from the original RGB images) each associated with a skin labeling mask. Moreover, in order to better approximate the test conditions (images coming from low spatial resolution devices, such as SPAD cameras) the grayscale images were downsampled to 64x64 adding black border if necessary. The training/testing data split was obtained selecting 100 images (50 for each original dataset, randomly selected from MUCT and selected in the same way as in [124] for Helen) for building the testing set. In order to ensure fair skin detection results, all the images belonging to the test set were checked manually and the annotations were corrected if needed. Subsequently, a horizontal flipped version of each training image is added in order to perform data augmentation. Finally, a validation set is created randomly selecting the 10% of the training set. Some examples of samples drawn from the final dataset are reported in Fig. 4.5. For example, the first image in the first row was originally in the MUCT dataset while the second one comes from the Helen one; the ground truth skin mask is superimposed in pink.

The complete dataset is available for download at the link: `https://github.com/marcobrando/Deep-Skin-Detection-on-Low-Resolution-Grayscale-Images`

Figure 4.5: *Example of some images in the created dataset for facial skin detection. The skin masks are superimposed in pink.*

Figure 4.6: *Some examples of results obtained with the face colorization network. The first raw represents the grayscale input, the second is the image colorized by the network, the third is the groundtruth color image.*

4.3 Results

The proposed method was trained as described in Sec. 4.2.3 using the training set described in Sec. 4.2.4. In this section some results are reported highlighting the necessity for the transfer learning approach and the accuracy of the obtained method, in Sec. 4.3.2 and Sec. 4.3.3 respectively.

4.3.1 Colorization results

As discussed in Sec. 4.2.3, the required first step in the skin detection network training procedure is training the face colorization. After the face colorization network, described in Sec. 4.2.2, was trained following the procedure described in Sec. 4.2.3, the colorization model was run on a small grayscale facial image testing set, in order to validate qualitatively its behavior. Some of the obtained results are shown in Fig. 4.6. In particular, the first raw represents the grayscale test input images, in the second the images colorized by the network are depicted while in the third one the groundtruth color images are reported. It can be observed that the model is able to produce realistic color output especially in the face region, which is compliant with the goal set in Sec. 4.2.3. These results demonstrate that the face color auxiliary network is correctly able, not only to detect human faces inside an image but only to discern, with sufficient precision, different

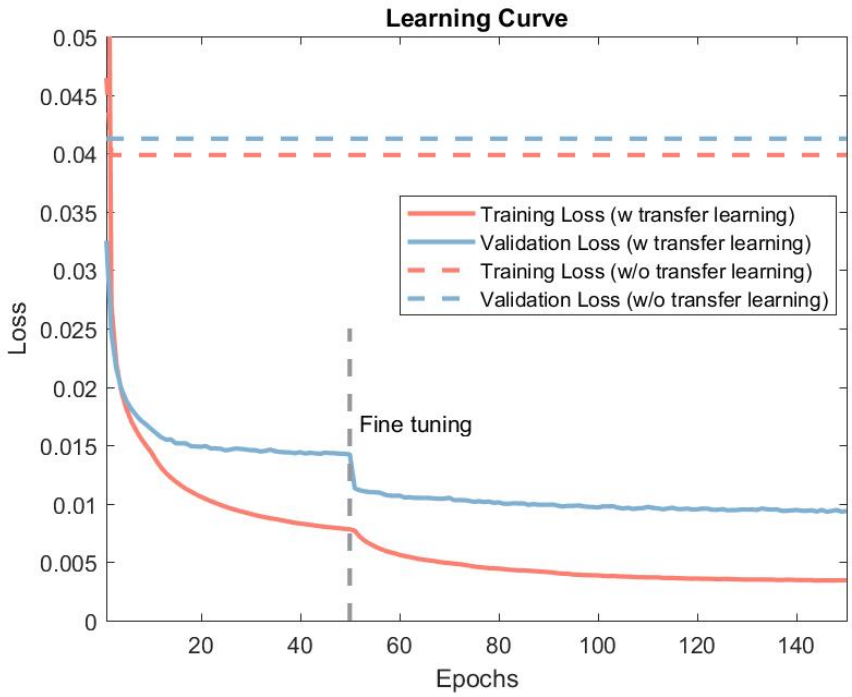

Figure 4.7: *Loss values during the training. Red lines represent loss values in each epoch on the training set while blue ones are obtained on the validation one. Dashed lines are the related to training directly on the skin detection problem with random initialization.*

facial parts (such as hairs, eyebrows, mouths, etc.). In almost all the results reported in Fig. 4.6 the cloths and background color are incorrect since they are assigned randomly as could be expected due to the randomness of these elements in uncontrolled conditions. These results are also qualitatively similar to the ones reported in [9]. No further tests and evaluations where performed on this step since it represents just the initial step of the training process and the qualitative results obtained are considered sufficient.

4.3.2 Training with transfer learning

The skin detection network's learning curves, obtained following the training procedure described in Sec. 4.2.3, are reported in Fig. 4.7. In particular, red curves are related to the loss error calculated on the whole training set while blue ones are obtained on the validation set described in Sec. 4.2.4. Both values were calculated at the end of each training epoch. As described in Sec. 4.2.3, following a transfer learning approach, in the first

part of the training (first 50 epochs) the majority of the layers are kept frozen, as described also in Sec. 4.2.2, preserving the weights value inherited from the colorization network, trained in a preliminary step as described in Sec. 4.2.3. This allow the network to quickly adapt to the skin detection problem as can be seen in the steep drop in the first part of the solid red and blue curves reported in Fig. 4.7.

On the other hand, since the colorization and skin detection problems are related but obviously different, an additional fine tuning step is necessary in order to further specialize the network to solve the specific skin detection problem. The effect of the fine tuning is clearly visible in Fig. 4.7, in which both the solid curves have a sharp decay after the dashed vertical gray line (fine tuning begin point). The importance of the transfer learning approach could be also observed in Fig. 4.7, in which the red and blue dashed lines represents respectively the training and the validation loss obtained without using the colorization network wights as the initialization. In this case the optimization almost immediately (in just a few epochs) collapses to the trivial solution of producing a masks with just zero values. The zero output solution is reached due to the asymmetry introduced in the loss function, as reported in Sec. 4.2.3. Once the model reaches this local minim point the training step could not be recovered and the network is not able to converge to other more interesting solutions. This trivial result is obtained in all the several training runs executed, regardless of the random initialization and hyperparameter settings. As can be observed from Fig. 4.7, a two steps approach, is able to drive the model training to a non trivial solution reaching a more adequate minimum point of the loss function surface. Lastly, since both the solid lines (validation and training loss) show a stable decay, no overfitting problems could be observed from the loss and validations curves.

4.3.3 Skin detection accuracy

After the training process described in Sec. 4.2.3 was completed, the obtained network was run on the test set described in Sec. 4.2.4, obtaining results able to evaluate the model both qualitatively and quantitatively. These results are shown in the following subsections.

Quantitative results

The proposed method was tested on the 100 images test set described in Sec. 4.2.4 resulting in a test loss value of 0.012 obtained between the output masks (values $\in [0, 1]$) and the ground truth ones (values $\in \{0, 1\}$). This

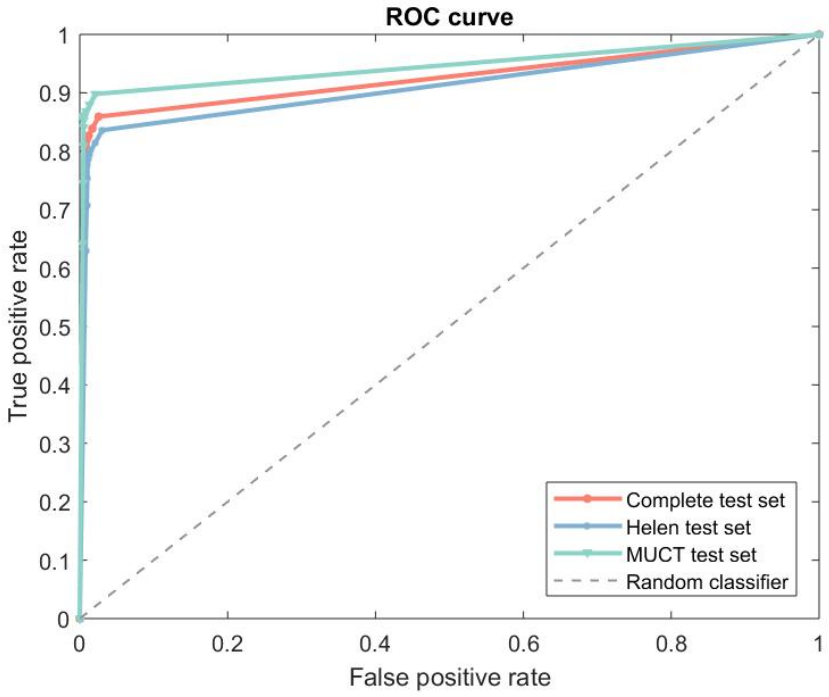

Figure 4.8: *Skin classification ROC curves obtained with the proposed method on the complete test set (red), MUCT test subset (green) and Helen test subset (blue).*

particular value is close the the validation error obtained in the last epoch of the training and reported in Fig. 4.7, i.e. last value in the solid blue line. This shows that the validation set was chosen in a correct way and represents a correct approximation of the statics of the test set.

ROC curves related to the per pixel skin classification task are reported in Fig. 4.8. Results on test images originally belonging to the MUCT (green line) and Helen dataset (blue line) are represented separately. This curves where obtained by varying the classification threshold (between 0 and 1) used to binarize the model output. In particular, points in position (0,0) where obtained using a threshold value equal to 0, points in (1,1) using a threshold value equal to 1. As can be observed, since the green line is always above the blue one, the proposed skin detection method achieved the best results on the test images originally belonging to the MUCT dataset. This is due to the less variability of the data presented in this dataset, i.e. controlled conditions while acquiring images, as described in Sec. 4.2.4. Considering the complete test set curve (red line), the best work point have

Table 4.2: *Comparison between the proposed method and [83] based on intersection over union and F-score results obtained on MUCT, Helen and complete test set. The second line show results obtained combining [83] and ground-truth masks in order to exclude eyes, eyebrows and mouth regions.*

	IOU			F-score		
Method	MUCT	Helen	Complete	MUCT	Helen	Complete
[83]	70	56	63	82	71	76
[83] + GT	-	62	-	-	76	-
Proposed method	**78**	**69**	**73**	**87**	**81**	**84**

a true positive rate (i.e. recall) of 89.8% with just 3.0% of false positive rate (i.e. fallout). This is achieved thanks to the asymmetrical loss function defined in Sec. 4.2.3, which penalizes explicitly false positive results.

As explained in Sec. 4.1.1, other methods for facial skin detection on grayscale low resolution images are rare or no existing. However a quantitative comparison between the proposed method and facial segmentation ones could be made. In particular we selected the state of the art facial segmentation method proposed in [83] since as ours it can work with occluded faces and grayscale input images. Due to the difference in the skin detection and face segmentation problem definition the masks estimated by the method proposed and [83] are different by design. In particular the face segmentation method proposed in [83] produces masks that contain the eyebrow, eye and mouth regions. In order to fairly compare the two methods we tested also the accuracy of [83] combined with ground-truth information of these regions. In particular we removed from the mask obtained from [83] the ground-truth mask of the unwanted regions. This could be done since the positions of those regions were available since they were used in order to generate the skin dataset (Sec. 4.2.4). The operation of combining [83] and groundtruth information for eyebrow, eye and mouth regions assumes a perfect estimation of them by [83]. On the other hand, in this comparison no groundtruth information was used in order to enhance the accuracy of the proposed skin detection method. We compared the three methods (the one proposed by us and the one in [83] not using or using ground-truth information) adopting the Intersection Over Union (IOU) and F-score metrics. In particular the F-score, also known as F_1 score or F-measure, is defined as the harmonic mean between precision and recall which is also equal to:

$$F_1 = \frac{t_p}{t_p + \frac{1}{2} * (f_p + f_n)} \tag{4.5}$$

Where t_p,f_p,f_n are the true positive, false positive and false negative respectively. On the other hand IOU is a commonly used metric in object detection tasks and it is defined as:

$$IOU = \frac{A_I}{A_U} \tag{4.6}$$

Where A_I, A_U are respectively the measures of the area of intersection and union between the ground truth mask and the one estimated by the considered method; the value 0.78 which correspond to 200 in a 8-bit unsigned integer representation, was used in order to binarize the proposed method results. The results obtained are summarized in Tab. 4.2; as can be observed the proposed method outperforms [83] even when using the ground-truth information for the eyes, eyebrows and mouth regions. The proposed method produces more accurate results achieving a IOU of 73% and an F-score of 84% on the complete test set.

Qualitative results

Some qualitative results with various images originally belonging to the test set are shown in Fig. 4.9 and Fig. 4.10, where the returned skin mask is superimposed to the input image using a pink color. Fig. 4.9 reports some results on images originally belonging to the Helen dataset while Fig. 4.10 shows other results using images initially in the MUCT dataset. As can be observed, the proposed skin detection method is able to produce qualitatively good results even in presence of non frontal faces (e.g. last row in Fig. 4.10), in-plane rotation (e.g. fifth row second column in Fig. 4.9), different head shapes and sizes (e.g. second row last column vs first row third column both in Fig. 4.9), expressions (e.g. forth row first column in Fig. 4.9), hair occlusions (e.g. second row first column in Fig. 4.9), glasses (e.g. second row third column in Fig. 4.10) and other wearable objects (e.g. forth row third column in Fig. 4.9). The beard is not always properly rejected, especially if it has an intensity similar to the subject skin (e.g. in the second row forth column in Fig. 4.9 beard is removed while second row third column in Fig. 4.10 it is not). This is probably due to some errors introduced in the automatic beard removal step in the training dataset described in Sec. 4.2.4.

Having a small generalization error [47], is one of the most import feature of supervised learning methods and it is often overlooked [47]. In particular, many deep learning methods show very accurate results when tested on data relatively similar to ones in the training set but fail to maintain the same performances on new data. For this reason, in Fig. 4.11 some masks

Figure 4.9: *Some qualitative results obtained using images in the test set originally belonging to the Helen dataset.*

obtained with the proposed method are superimposed to some input images acquired by the SPAD array camera. These results are particularly promising since their origin is very different to ones of the images that composed

Figure 4.10: *Some qualitative results obtained using images in the test set originally belonging to the MUCT dataset.*

the training and testing dataset, they are even acquired with a different technology. As can be seen in Fig. 4.11, the network is able to generalize and produce good quality results even on images acquired in different condi-

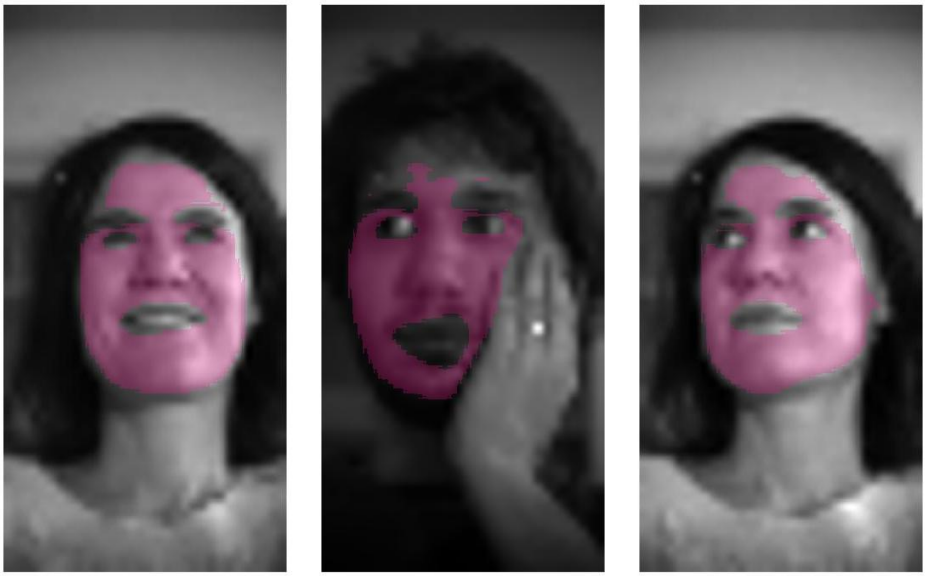

Figure 4.11: *Qualitative results on three face images acquired by the SPAD camera.*

tions compared to the training dataset, and even in presence of different expressions, poses and heavy occlusions. This also demonstrate that overfitting was avoided during training since pour accuracy on new data is often caused by that.

4.3.4 Real time performance

We evaluated the time performance of our method executing it on the test set described in Sec. 4.2.4 achieving an execution time of 6.6 milliseconds for each image corresponding to 152 fps. We obtain this result with a Tensorflow [1] implementation of the network and executing it on a Nvidia Titan Xp® GPU.

4.3.5 Hidden layer output visualization

In Fig. 4.12 a visualization of the knowledge acquired by the network is reported. In particular, Fig. 4.12 visualizes the output of the decoder's second hidden layer when the network is run on an image acquired by the SPAD camera, the same picture on the left in Fig. 4.11. This particular hidden layer has been chosen due to its relationship with high level features. Although some more sophisticated visualization techniques exist [88], simply

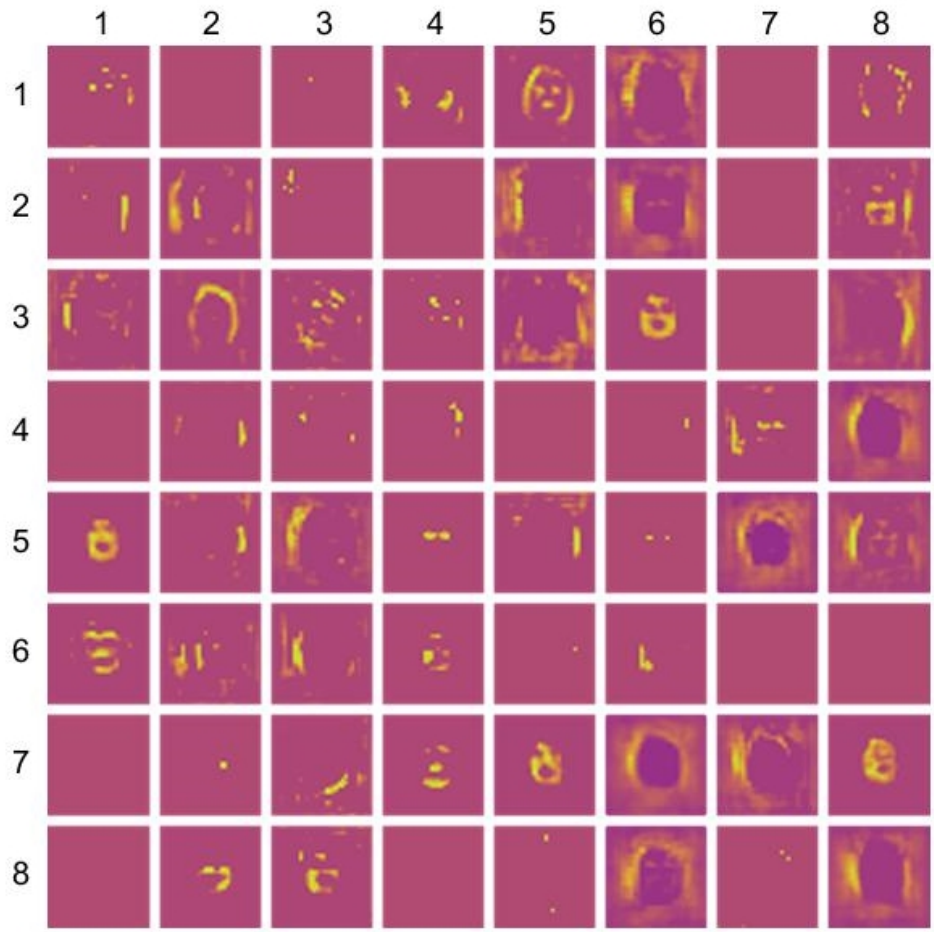

Figure 4.12: *Visual representation of the activations of the second hidden layer in the skin detection decoder stage when tested on a face image acquired by the SPAD camera.*

visualizing each filter output for a particular hidden layer can give information on what kind of feature are extracted and used in that specific layer. As can be observed, after the training is completed, some filters of this layer specialized in detecting same particular facial features relevant for the skin detection problem, e.g. eyes (forth and sixth columns on fifth row), the hair (second column on third row), the background (sixth column on first and second rows) the face contour (fifth column on first row) and finally the skin (sixth column on third row, first column on fifth row and fifth column on seventh row). The information produced by this layer appear to be redundant increasing the robustness of this method.

4.4 Discussion and conclusions

In this chapter, we presented a Deep Learning based method proposed in order to solve the facial skin detection problem on low-resolution grayscale images, motivated by the use of SPAD cameras in a rPPG application (as described in Sec. 4.1). The low spatial resolution (64x64 pixels) coupled with the unavailability of color information (grayscale images) made this task particularly ambitious. Analyzing the state of the art of similar problems, in Sec. 4.1.1, we showed the peculiarity of the proposed task and how, to the best of our knowledge, the method described in this work is the first being proposed specifically to solve the specific task of facial skin detection.

Given the similarity between this problem and a semantic segmentation one, and the good accuracy achieved by neural network methods in this latter field, a Deep Learning based method was chosen. On the other hand, these kind of methods need massive amount of data to be trained on. Since the facial skin detection problem, tackled in his chapter, is very specific unfortunately only a limited amount of data are available for this specific problem. For this reason a transfer learning approach was adopted in the training phase. In particular, the proposed network architecture was chosen in order to have the majority of layers in common with a convolutional neural network proposed to solve the grayscale images colorization problem [9]. These apparently different problems are in reality tightly linked as a colorization method, in order to work on face images, must (implicitly) solve the skin detection problem, since it needs this information in order to color in a correct way pixels depicting skin regions. On the other hand, since the skin detection problem is only a small sub-task in respect to the colorization one, the proposed network was significantly simplified, as shown in Sec. 4.2.2. Further information about the similarities between the skin detection and colorization problems are described in Sec. 4.2.2.

As discussed in Sec. 4.2.3, in order to exploit the maximum amount of knowledge possible gathered from the colorization problem, a three step transfer learning strategy was adopted. Firstly the colorization method was trained on a large dataset of unlabeled face images. This was done in order to drive the preliminary method into the specific domain of face image analysis. The proposed skin detection network was subsequently trained starting from the colorization network weights and minimizing an asymmetric loss function, described in Sec. 4.2.3, on a novel constructed dataset. This was done in two consecutive steps in order to train new and already trained layers at two different speed. In Sec. 4.2.4 the training dataset, containing

more than 6000 labeled training images and 200 labeled test images, was described, detailing also all the operations performed in order to adapt the two existing and freely available datasets (MUCT [77] and Helen [124]) to the specific skin detection problem.

Lastly in Sec. 4.3.2 the proposed training procedure has been justified showing that, without using it, it would be not be possible to train the proposed network with the few data available. In particular Fig. 4.7 shows that the adapted training procedure is able to avoid overfitting since the validation error does not increase over the training epochs. Moreover, Fig. 4.11 demonstrate that evaluation of the model on new images produces qualitatively good results, reiterating the absence of overfitting in the training process. In addition, in Sec. 4.3.3 some quantitative results were reported providing accuracy evaluation for the proposed skin detection method and showing comparisons with a state of the art face segmentation method. In particular the proposed method is able to outperform [83] in the specific task of facial skin detection on low resolution grayscale images, even when GT information where integrated to [83]. Moreover, in Sec. 4.3, many skin detection outputs were shown for both images acquired in similar conditions with respect to the ones used to built the training set and for images completely independent from the training set, acquired with the SPAD camera. Both these results show how the proposed method is able to achieve quantitative and qualitative good results in the skin detection problem even in presence of different poses, ages, expressions, ethnicity, wearable objects and other occlusions.

In evaluating the proposed method on multiple test images, it could be notice that some labelling errors still occurs especially in presence of beard, glasses and other occlusions. This is due to the fact that the training dataset was automatically annotated using color similarity to label this kind of occlusions and this inevitably introduces some errors. Moreover, the proposed solution could also benefit from some more advance deep learning architectures, such as U-Net [91], exploiting the good starting point achieved by the presented method. Another idea for future development is to improve the skin detection accuracy considering more than just one frame since, in many applications including rPPG, a stream of frame is available. In this case a Recurrent Neural Network (RNN) [31], such as LSTM [40] and GRU [21], could integrate the current CNN architecture achieving the required result.

The main part of this chapter was published as a journal article in Volume 131 on March 2020 (pages 322-328) of Pattern Recognition Letters [132]. Moreover, the complete facial skin dataset created by the author

of this work and the trained skin detection model are available at the following link `https://github.com/marcobrando/Deep-Skin-Detection-on-Low-Resolution-Grayscale-Images`.

CHAPTER 5

Fast skin detection on SPAD camera images

SCOPE & AIMS: The scope of this chapter is to propose an automatic method able to solve the problem of detecting skin pixels in grayscale low resolution face images efficiently and in real-time even when run on hardware with limited computing capabilities.

METHODS: The Convolutional Neural Network model described in Chapter 4 was optimized and modified in order to adapt to the computational requirement. A transfer learning approach was adopted in order to exploit the knowledge already present in the original network.

RESULTS: Quantitative and qualitative results show the proposed method adequately is able to outperform the accuracy of the method introduced in Chapter 4. Execution time measurements also show that the proposed method is able to run in real-time on a small single-board computer.

5.1 Problem description

As described in Chapter 1, the main contribution of this work is to explore the possibility of performing rPPG using SPAD camera with the ultimate goal to have a compact system able to monitor in real time the health condition of the driver in an automotive scenario.

As reported in Chapter 4, skin detection is an important preliminary task in rPPG applications, especially when the imaging device used is very limited in spatial resolution, as SPAD cameras are. The Deep Learning based method described in Chapter 4 is able to achieve good qualitative and quantitative results as show in Sec. 4.3.3. Moreover, as reported in Sec. 4.3.4, this method is able to achieve real-time performances when tested on a Nvidia Titan Xp® GPU using a Tensorflow implementation. These kinds of GPU are widely used for computer vision applications and are beginning to be implemented also in many different industrial area such as for example automotive (Mercedes, Tesla, Toyota and many others are producing cars equipped with them or will do so in the near future). Although these devices are powerful they are also expensive and there exists scenarios in which this high computational power is simply not available. Moreover, it needs to be taken into consideration the fact that self-driving cars need to solve a plethora of different visual related problems: simultaneous localization and mapping (i.e. SLAM) problems, pedestrian detection, road signal recognition and so on. In addition to that all the other non-visual related problems constantly need to be addressed (such as actuators management, energy consumption, processing signals coming from all different sensors, etc.). For these reasons, it could be useful to have a method able to solve the skin detection problem that could achieve real time performances even when running on much more affordable and less powerful hardware. The main goal of this chapter is to propose and describe a much faster method able to analyze in real time frames coming from the SPAD camera working with this limited hardware. Moreover the new method must be connected to the one described in Chapter 4 in order to exploit the knowledge acquired by that particular model and must not comprise to much the accuracy in favor of computational speed.

The rest of the chapter is organized as follows: in Sec. 5.1.1 a brief description of the hardware used in order to implement the skin detection method is given; in Sec. 5.2 the proposed method is described explaining the main component used in Sec. 5.2.1 and its architecture Sec. 5.2.2; in Sec. 5.2.3 the training procedure that exploit transfer learning is illustrated;

https://www.nvidia.com/en-us/self-driving-cars/partners/

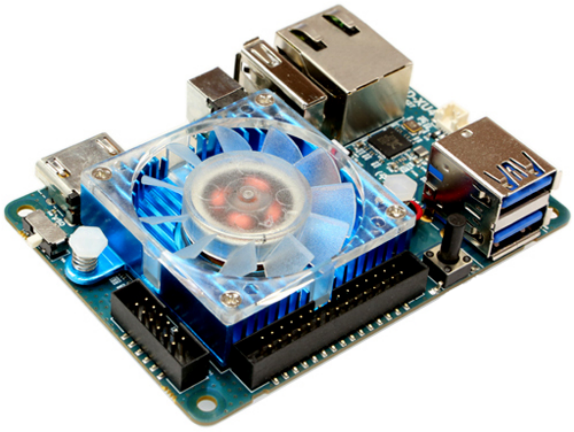

Figure 5.1: *Hardkernel Odroid-XU4 board*

qualitative and quantitative results are shown in Sec. 5.3.1; in Sec. 5.3.2 the real time performances of the propose method are shown and finally in Sec. 5.4 the contribution of this work are highlighted.

5.1.1 Materials

In order to evaluate the time performances of the proposed skin detection method a Hardkernel Odroid-XU4, depicted in Fig. 5.1, was used. This is a small but powerful single-board computer equipped with a Samsung Exynos5422 Cortex™-A15 2Ghz and Cortex™-A7 Octa core CPU, Mali-T628 MP6 GPU and 2Gbyte of RAM. These kind of single board-computer are commonly used in many other Computer Vision works such as [134], [128] and [129].

5.2 Methods

In this section the main strategy for reducing the number of parameters of the network, thus increasing its speed, is described. The complete network architecture is also described along with the training procedure.

5.2.1 Depthwise separable convolution layers

In order to reduce the network complexity (i.e. the number of parameters), depthwise separable convolution layers have been adopted. This kind of layer was firstly introduced in [41] and can drastically reduce the number

https://wiki.odroid.com/odroid-xu4/odroid-xu4

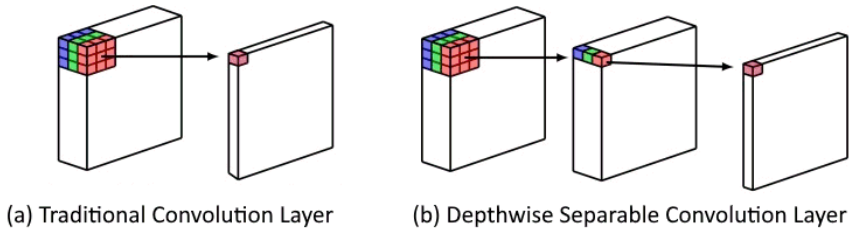

Figure 5.2: *Differences between depthwise separable convolution layers and traditional convolution ones.*

of parameters by substituting each traditional full convolution layer with a depthwise convolution followed by 1×1 convolution called a pointwise convolution. While a standard convolution in a single step both filters and combines inputs into new outputs, the depthwise separable convolution splits this into two consecutive steps, a separate layer for filtering (i.e. depthwise convolution) and a separate layer for combining (i.e. pointwise convolution). The differences between this two different kind of convolution layers are highlighted in Fig. 5.2.

Let $\mathbf{F}$ be a $D_F \times D_F \times M$ feature map (assumed spatially square for simplicity) which is the input of a traditional convolution layer having as output a feature map $\mathbf{G}$ with the same spatial dimension as $\mathbf{F}$ (i.e. stride equal to 1 for simplicity) and N channels, i.e. the dimension of $\mathbf{G}$ is $D_I \times D_I \times N$. The size of the traditional convolution layer's kernel $\mathbf{K}$ is $D_K \times D_K \times M \times N$, where D_K is the spatial dimension of the kernel itself (assumed also to be square for simplicity) while M and N are defined previously. Following the definition of convolution, each element of the output feature map $\mathbf{G}$ is then obtained as:

$$\mathbf{G}_{k,l,n} = \sum_{i,j,m} \mathbf{K}_{i,j,m,n} \cdot \mathbf{F}_{k+i-1,l+j-1,m} \tag{5.1}$$

So the computational cost of a traditional convolution layer is:

$$\mathcal{C}_\mathcal{C} = D_K \cdot D_K \cdot M \cdot N \cdot D_F \cdot D_F \tag{5.2}$$

Where both the number of input and output channels, N and M respectively, the kernel and feature map spatial dimensions, $D_K \times D_K$ and $D_F \times D_F$ respectively, appear as multiplicative factors. On the other hand, by adopting a depthwise separable convolution, the operation of filtering and combination between channels are splits in two consecutive steps. The first step, depthwise convolution with one filter per input channel, can be

represented by the following equation:

$$\hat{\mathbf{G}}_{k,l,m} = \sum_{i,j} \hat{\mathbf{K}}_{i,j,m} \cdot \mathbf{F}_{k+i-1,l+j-1,m} \tag{5.3}$$

where $\hat{\mathbf{K}}$ is the depthwise convolution kernel with size $D_K \times D_K \times M$ and it is utilized on F so the m_th filter in $\hat{\mathbf{K}}$ is applied to the m_{th} channel of F to produce the m_{th} channel of the filtered output feature map $\hat{\mathbf{G}}$. The computational cost of a depthwise convolution layer is:

$$\mathcal{C}_{\mathcal{D}} = D_K \cdot D_K \cdot M \cdot D_F \cdot D_F \tag{5.4}$$

It could be notice that in respect to traditional convolution layer, depthwise convolution layer are much faster since the multiplicative term N is present in $\mathcal{C}_{\mathcal{C}}$ but not in $\mathcal{C}_{\mathcal{D}}$. On the other hand, while using when using a traditional convolution layer the output of each channel is determined by all the other channels, while adopting depthwise convolution the output of each channel is independent from the other ones. For this reason, the second step is applied which is just a linear combination the output (pointwise convolution). The computational cost of the pointwise convolution layer is just:

$$\mathcal{C}_{\mathcal{P}} = D_F \cdot D_F \cdot M \cdot N \tag{5.5}$$

So the total computational cost of a depthwise separable convolution layers is obtained summing $\mathcal{C}_{\mathcal{D}}$ and $\mathcal{C}_{\mathcal{P}}$ and is equal to:

$$\mathcal{C}_{\mathcal{S}} = \mathcal{C}_{\mathcal{D}} + \mathcal{C}_{\mathcal{P}} = D_K \cdot D_K \cdot M \cdot D_F \cdot D_F + D_F \cdot D_F \cdot M \cdot N \tag{5.6}$$

So the cost reduction of substituting a traditional convolution layer with a depthwise separable convolution one is calculated as:

$$\begin{aligned} \frac{\mathcal{C}_{\mathcal{S}}}{\mathcal{C}_{\mathcal{C}}} &= \frac{D_K \cdot D_K \cdot M \cdot D_F \cdot D_F + D_F \cdot D_F \cdot M \cdot N}{D_K \cdot D_K \cdot M \cdot N \cdot D_F \cdot D_F} \\ &= \frac{1}{N} + \frac{1}{D_K^2} \end{aligned} \tag{5.7}$$

Obtaining a huge speed and computational improvement especially when N and D_K are large values.

Obviously being depthwise separable convolution cheaper, in term of number of parameters, in respect to traditional convolution they are also less powerful. On the other hand results reported in [41], and many other recent works such as [26] and [55], have proven that network that uses this kind of layers can achieve good accuracy results in many different applications and scenarios.

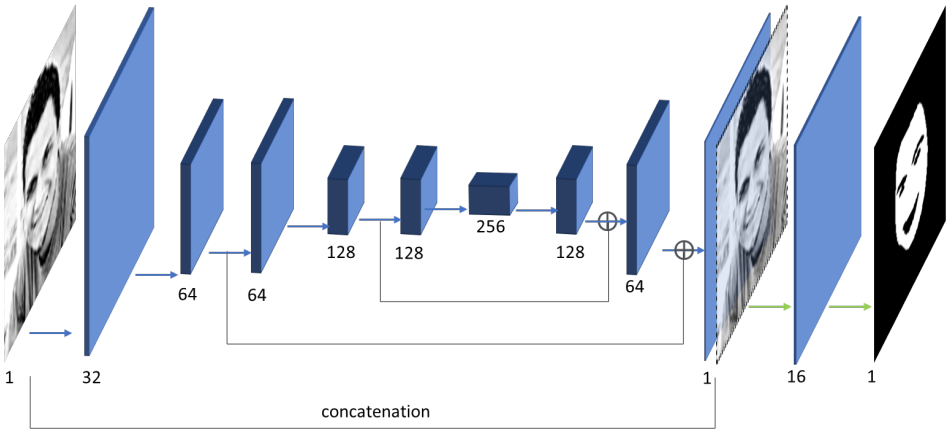

Figure 5.3: *Skin detection network architecture using depthwise separable convolution and skip connections. Blue arrows represent depthwise separable convolution while green arrows represent traditional convolution.*

5.2.2 Network architecture

As reported in Fig. 5.3, the architecture of depthwise separable convolution version of the skin detection network is quite similar to the one proposed in Chapter 4 but there are two main differences. The first one is that where blue arrows are shown in Fig. 5.3, the hidden layers are obtained by the adoption of depthwise separable convolution instead of traditional ones. This means, as described in Sec. 5.2.1, that each blue arrow represent the concatenated use of depthwise convolution (with ReLu nonlinear function and batch normalization as in [41]) and pointwise convolutions (also followed by ReLu nonlinear function and batch normalization as in [41]). On the other hand, green arrows represent convolution layers followed by ReLu activation functions. The activation function of the last layer, as in the architecture described in Chapter 4, is a sigmoid in order to produce output values between 0 and 1 representing a skin probability map.

The other difference respect to the skin detection network described in Chapter 4, is in the use of skip connection. Skip connections, firstly introduced in [35], could be implemented in different ways but the basic idea is to propagate information inside the network by adding connections between non consecutive layers. This operation has been introduced in order to help the training stage [35]. These kinds of connections are trivial ones as adding together the values of two hidden layers (with the same shape, i.e. spatial size and number of channels) or concatenating them (they must

have the same spatial size). As can be observed from Fig. 5.3, in the chosen architecture both these techniques have been used.

The overall shape of the network is quite similar to the one described in Chapter 4. In particular also in this network a decoder and encoder structure is adopted. The main difference is in the adoption of skip connection which makes easier the propagation of the information between these two consecutive steps. Another difference is the adoption at the end of a very simple result enhancer part of the network represented by the last two traditional convolution layers. The main idea behind the adoption of these layers is to use two small additional layers in order to enhance the accuracy of the mask obtained after the decoding step by using the more powerful traditional convolutions. These additional final layers could be viewed also as denoising layers which are able to adapt the estimated mask by comparing it directly to the original face input (thanks to the concatenation with the initial layer). The total number of parameters in this model is just 120 thousands while the architecture proposed in Chapter 4 has more than 6.2 millions, this means that the network depicted in Fig. 5.3 is roughly 50 times smaller.

5.2.3 Training procedure

The same skin detection dataset created as described in Chapter 4 was used in this case. The same training/testing/validation dataset split was adopted and data augmentation was performed by the means of horizontal flipping. Since the skin detection problem described in Chapter 4 is identical to the one faced here, the same custom loss function previously described was used.

$$E(\hat{y}, y) = \sum_{ij} (y_{ij} - \hat{y}_{ij})^2 (\alpha \cdot y_{ij} + (1 - \alpha)(1 - y_{ij})) \tag{5.8}$$

As for the method described in Chapter 4, also in this case, a transfer learning procedure was adopted. Starting from the trained network obtained as described in Chapter 4 (indicated from this point for simplicity as ConvNet) , exploiting the similar shapes between the two networks layers, for each one of the depthwise separable convolution, represented in Fig. 5.3 by blue arrows, an new incremental temporary network was created. In particular, the first new network was obtained by substituting the first layer of ConvNet with the first depthwise separable convolution represented in Fig. 5.3. This operation could be performed since the removed layers and the added one have the same input and output shape. This first obtained

hybrid network was then trained for 10 epoch by keeping frozen all the already trained layers and using a random initialization for the new one. This operation effectively force the newly introduce layer (with less parameters) to approximate the behavior of the removed one (with more parameters). This operation is performed for each depthwise separable convolution layer described in Fig. 5.3 each time starting from the previously obtained network and substituting the corresponding layer (or layers) with the new one. This process smoothly transform the original ConvNet into the one reported in Fig. 5.3, with the exception of the two last traditional convolution layers. Since these two do not have corresponding ones in ConvNet, they are simply added at the end using random initialization. The last step of the training procedure is the introduction of the additive skip connections and a final optimization of all the layers for an additional 50 epochs. The network was trained using Keras [22], with Tensorflow [1] as backend, with the Adam Optimization algorithm [56] and a learning rate of 0.0005. The transfer learning operations described above are summarized below.

Training procedure

1. Starting from the trained ConvNet, each depthwise separable convolution layer is introduce incrementally in order to substitute the corresponding traditional layer performing the following operations.

   For each depthwise separable convolution layer do:

   (a) Substitute the corresponding traditional convolution layer with the new depthwise separable convolution one
   (b) Froze the value of all the other layers
   (c) Train for 10 epochs

2. Add the last two convolution layers and the skip connections and train the whole model for 50 epochs.

5.3 Results

The proposed method was trained as described in Sec. 5.2.3 using the training set described in Sec. 4.2.4. In this section some results are reported highlighting the accuracy of the obtained method.

5.3.1 Skin detection accuracy

After the training process described in Sec. 5.2.3 was completed, the obtained network was run on the test set described in Sec. 4.2.4, obtaining re-

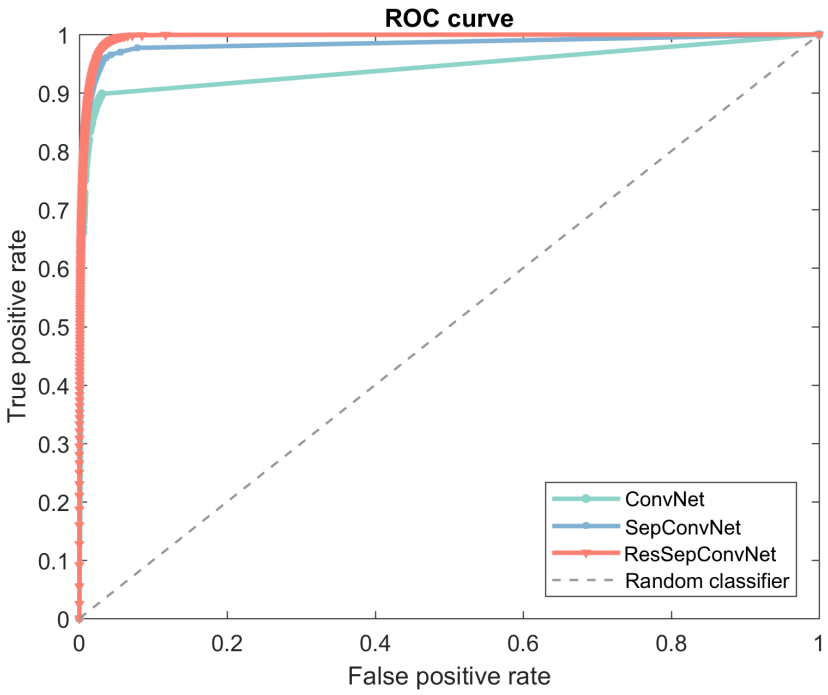

Figure 5.4: *Skin classification ROC curves obtained with the method described in Chapter 4 (ConvNet in green), the method that uses depthwise separable convolution but no skip connections (SepConvNet in blue) and the one that makes use of both (ResSepConvNet in red).*

sults able to evaluate the model both qualitatively and quantitatively. These results are shown in the following subsections.

Quantitative evaluation

ROC curves related to the per pixel skin classification task are reported in Fig. 5.4. In this image three different skin detection methods are compared. The first one is the one described in Chapter 4, called ConvNet and represented by the green line, the second one is very similar to the one described in Sec. 5.2.2 but without skip connection, indicated as SepConvNet and represented by the blue line, while the third one is the complete network as described in Sec. 5.2.2 that makes use of both depthwise separable convolution and skip connection and trained as described in Sec. 5.2.3 (ResSepConvNet in red). A random classifier is also reported as the gray dashed lines. As can be observed, both the red and blue lines are always above the green

Table 5.1: *Comparison between the method described in Chapter 4 (ConvNet), the method in [83] with or without using ground-truth information, and the two method that make use of depthwise separable convolution layers (with or without residual connections, ResSepConvNet and SepConvNet respectively).*

	IOU			F-score		
Method	MUCT	Helen	Complete	MUCT	Helen	Complete
[83]	70	56	63	82	71	76
[83] + GT	-	62	-	-	76	-
ConvNet	78	69	73	87	81	84
SepConvNet	77	71	74	86	83	85
ResSepConvNet	**83**	**79**	**81**	**90**	**88**	**89**

line, so in the per pixel skin classification task of both the smaller models (SepConvNet and ResSepConvNet) are able to reach higher true positive rates fixing the same false positive rate as the original model (ConvNet). This is an important results that shows that not always a larger model, that have a much higher approximation capability, is able to outperform models with less parameters. By reducing the dimension of parameter space, the training optimization algorithm was able to find a better local minimum in respect to the one fund in Chapter 4. Fig. 5.4 also show the importance of skip connections which are able to further increase the accuracy of the proposed method. By analyzing the ROC curve related to the best model (ResSepConvNet, red line in Fig. 5.4) it could be noticed that 90% of true positive rate (i.e. recall) is reached with just 1% of false positive rate (i.e. fallout), and 95% of true positive rate correspond to roughly 2% of false positive rate.

Moreover, the SepConvNet and ResSepConvNet have been quantitatively compared using the Intersection Over Union (IOU) and F-score metrics already described in Sec. 4.3.3. Results obtained are reported in Tab. 5.1, in order to produce this results the best work point (i.e. classification threshold) for each model was selected from Fig. 5.4. In this table results related to the method described in [83] are also reported. Moreover the second line is related to the same method enhanced with ground truth information for eyebrow, eye and mouth regions as explained in Sec. 4.3.3. The performance of the method described in Chapter 4 are reported in the third row. Both IOU and F-score results are reported for the complete test set and images of it originally belonging to the MUCT and Helen datasets, as explained in Sec. 4.3.3. As can be observed from Tab. 5.1, and as expected from the results shown in Fig. 5.4, ResSepConvNet outperforms SepCon-

Figure 5.5: *Collection of skin masks obtained using the model ResSepConvNet on test images. For each image the corresponding mask is superimposed in pink color.*

Table 5.2: *CPU Execution time comparison.*

	ConvNet	ResSepConvNet
FPS	12.4	**15.8**

vNet, ConvNet and [83], using or not ground truth information, reaching a IOU of 81% and an F-score of 89% on the complete test set.

Qualitative evaluation

Fig. 5.5 reports the results obtained on the test set described in Sec. 4.2.4. As can be observed, the model is able to achieve good quality results on all the input images shown. In particular, the model is always able to correctly locate the image avoiding also to annotate as skin the mouth, eyes and eyebrows regions. Moreover, glasses and sunglasses are correctly removed from the skin region as can be observed in the last result of the second row and second-last image in the third row. Beard and facial hair are also removed from the skin mask as can be observed from the second-last image in the third-last row. Comparing this results to the ones reported in Fig. 4.9 and Fig. 4.10 is another indication on the superiority in term of accuracy of the model described in Sec. 5.2.2 in respect to the one described in Chapter 4.

5.3.2 Real time performance

In order to evaluate the time performances of ResSepConvNet and ConvNet in limited computational power scenario the Odroid X-U4 ARM board described in Sec. 5.1.1 was used. In particular, a simple C++ program was created in order to evaluate the effective FPS required to run each model. This program make use of the library frugally-deep which is able to load a Keras model and run predictions using it. Results in term of frames per seconds are reported in Tab. 5.2. As can be observed ResSepConvNet is 30% faster than ConvNet while tested on Odroid ARM A15 CPU. Although this difference is significant it seems apparently smaller than expected given the considerable differences between the parameters number of the two models. The difference in the two network topology must be taken into account, considering that ConvNet is a simple straightforward convolutional network while in ResSepConvNet skip connection are present which increase the network complexity regardless the number of parameters. The

https://github.com/Dobiasd/frugally-deep

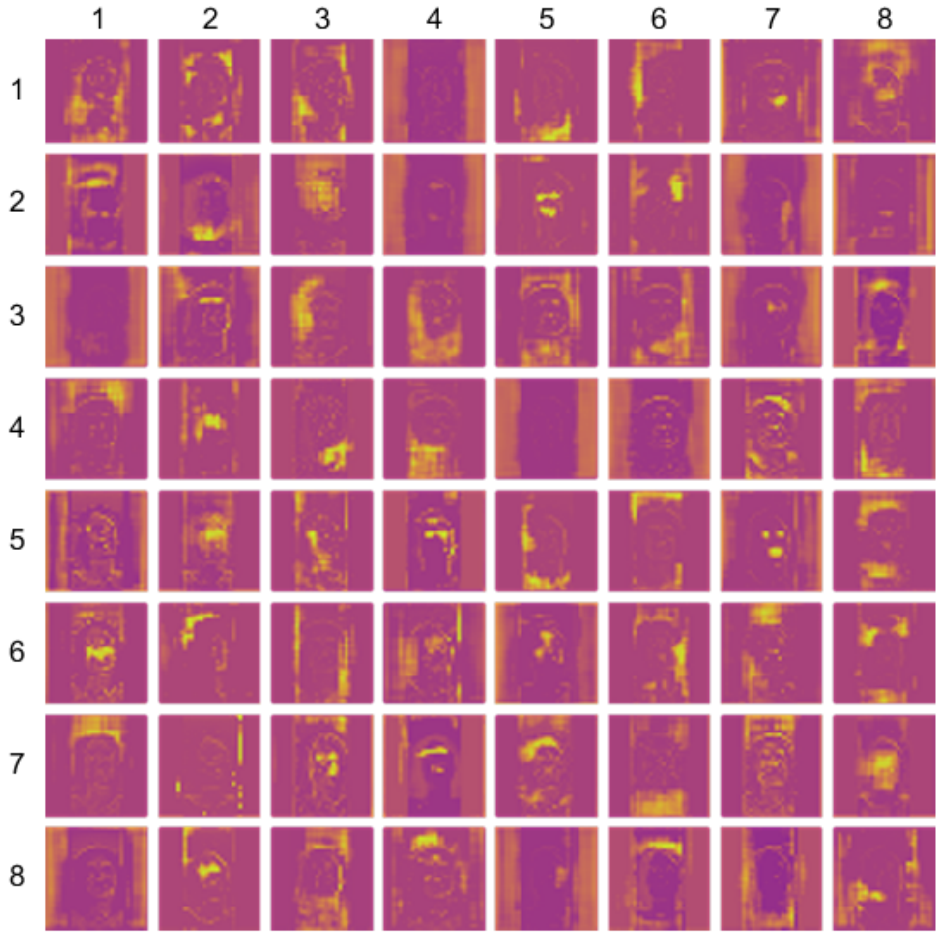

Figure 5.6: *Visual representation of the activations of the hidden layer that follows the second skip connection in the ResSepConvNet when tested on a face image acquired by the SPAD camera.*

difference is still significant especially in applications which must run with at least 10 FPS since the adoption of ResSepConvNet could save time that could be dedicated to other tasks.

5.3.3 Hidden layer output visualization

In Fig. 5.6 a visualization of the knowledge acquired by the ResSepConvNet network is reported. In particular, Fig. 5.6 visualizes the output of the activations of the hidden layer that follows the second skip connection

in the ResSepConvNet when the network is run on an image acquired by the SPAD camera, the same picture used to generate in Fig. 4.12. This particular hidden layer has been chosen due to its relationship with high level features and for being the equivalent of the one used in order to generate Fig. 4.12. As can be observed, after the training is completed, as for Fig. 4.12 and also in this case, some filters of this layer specialized in detecting same particular facial features relevant for the skin detection problem, e.g. eyes (fifth column on second row), the hair (seventh column on forth row), the background (first column on first row).

5.4 Discussion and conclusions

In this chapter, we presented a Deep Learning based method proposed in order to solve the facial skin detection problem on low-resolution grayscale images, motivated by the use of SPAD cameras in a rPPG application. As discussed in Sec. 5.1, a model that could run in real-time on hardware with limited computational power could be useful in many situations. In particular in the automotive domain even if more powerful GPU based computational power is available, having a compact dedicated solution to the rPPG problem solution could be the ideal scenario in order to not overload the main computational unit.

In order to achieve this goal a new Deep Learning based model for solving the skin detection problem have been proposed. This method is able to work in presence of low spatial resolution (64x64 pixels) coupled with the unavailability of color information (grayscale images) as the method described in Chapter 4. The main difference between the two methods is the adoption of depthwise separable convolution layer. As described in Sec. 5.2.1, these kinds of layers are obtained splitting the operation performed in a traditional convolution layer with two consecutive steps of spatial filtering and channel combination. In the same section we show also how and why this decoupling is able to drastically reduce the number of parameters. Moreover, the complete network architecture has been described in Sec. 5.2.2 highlighting the choice made in order to maximally increase the similarity with the one described in Chapter 4 and the usage of skip connections.

As discussed in Sec. 5.2.3, in order to exploit the maximum amount of knowledge possible gathered from the skin detection method described in Chapter 4, an incremental transfer learning strategy was adopted. Starting from the model described in Chapter 4 exploiting the similarity between the two networks, traditional convolution layers are one by one incrementally

substituted with depthwise separable ones. Ten epochs of training are performed between every substitution. This operation effectively force the new layers, with smaller amount of parameters, to approximate the much larger traditional ones. An additional 50 epochs of training were then applied on the complete network.

Furthermore in Sec. 5.3.1 the proposed model was tested in both precision and real time performance. In particular Fig. 5.4 shows that the proposed method is able to outperform the much larger one described in Chapter 4 also in the skin detection accuracy. This is probably due to the reduced number of parameters that increased the easiness for the optimization algorithm to converge to a better minimum of the loss function. Fig. 5.5 also shows some qualitative results highlighting the precision of the model in selecting the face region discarding parts not related to the skin. Lastly in Sec. 5.3.2 the time performances of the method proposed and one described in Chapter 4 are compared. In particular the proposed method is able to achieve realtime performance even when run on a limited compact single-board computer such as the Odroid XU-4.

CHAPTER 6

Dependable SPAD based rPPG application

SCOPE & AIMS: The main goal of this chapter is to introduce a rPPG system that, making use of a SPAD camera and an single-board ARM computer, is able to estimate biometric parameters, such as Heart Rate, in real time and in a dependable way.

METHODS: A rPPG pipeline is proposed which make use of a Deep Learning based method for facial skin segmentation and traditional signal processing techniques in order to estimate biometric parameters.

RESULTS: A set of experiments has been conducted highlighting the accuracy of this method and the beneficial impact of using a Deep Learning skin segmentation methodology coupled with traditional signal processing.

PUBLICATIONS: The main part of this chapter was published as a journal paper [133] and a conference paper [130].

6.1 Problem description

Being able to constantly check, in real time and without any contact, the health condition of a person could have a significant impact in many different situations. Possible applications include fitness assessments [103], medical diagnosis [103] and driver monitoring [130]. The aim of this chapter is to propose a method able to estimate the aforementioned biomedical measurements in real time and in a dependable fashion. Moreover, this work explores the possibility of adopting a SPAD (i.e. Single-Photon Avalanche Diode) array camera instead of traditional RGB camera, as done in most publications in this field, e.g. [103], [93]. In rPPG applications SPAD's high precision can accurately measure the intensity variations of the light reflected by the skin, caused by the blood flowing underneath it. Conversely, the main drawback of using a SPAD sensor is their low spatial resolution due to technical limitation. In order to overcome this problem and use as much spatial information as possible, an *ad-hoc* deep learning based method is proposed. Finally, since the rPPG estimation of biomedical measures is related to optical signals that could be affected by noise some dependability evaluation metrics are also proposed.

More recent publications, i.e. in 2008 [114], show that PPG could be performed remotely (i.e. rPPG) using ambient light as the optical source. Many other rPPG focused studies were published shortly after [24, 46, 82, 93, 94, 115]. Some surveys on the state of the art of this field could be found in [71, 103, 122] and [34]. While machine learning techniques are widely used in contact PPG applications [28], recent works [12, 57, 95] explored the opportunity of using deep learning methods also in remote PPG applications. All these works completely substitute the classical signal processing techniques with deep learning ones using an end-to-end network, as in [12] and [57], or by using two consecutive neural networks, as in [95]. On one hand, the use of an end-to-end deep learning model has proven to achieve state of the art results on many computer vision tasks such as image segmentation, object detection, and many others. On the other hand, this kind of methods required a massive amount of training data in order to learn how to extract heart related information directly from video frames and no prior domain knowledge is incorporated. This make the performance of this kind of methods tightly linked to the training dataset and potentially unable to generalize in different setting conditions. Moreover, the complete substitution of classical signal processing techniques developed using a solid theoretical background (signal filtering, Fourier transform, etc.) with data driven ones could lead to non-optimal

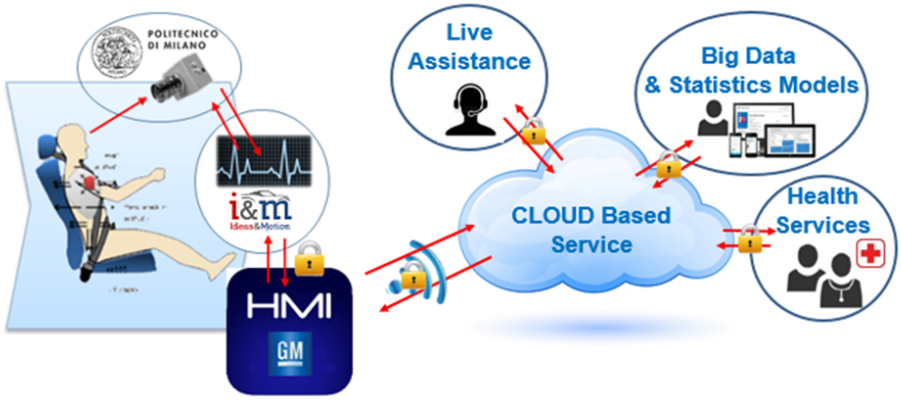

Figure 6.1: *A concept illustration of the rPPG based driver monitoring system developed inside the DEIS project.*

solutions. For the best of our knowledge no prior work has been done in trying to combine traditional and deep learning based signal processing in this field. Lastly, in all the considered studies the cameras used are traditional RGB cameras.

One of the main aim of this study is to validate the effectiveness of performing rPPG using SPAD camera coupled with a deep learning technique in order to compensate for low spatial resolution of Single-Photon cameras. On the other hand, the final goal is to develop a rPPG system for an automotive use case. In particular exploiting, the capability of SPAD camera, the propose system was developed in order to remotely monitor the health state of the driver. The idea is to develop an application that could run in real time on a computational unit equipped on the car that is able to extract the pulse signal and analyze it in real time in order to consistently monitor the driver's health condition. These data could then be used to enable particular features of the vehicle, such as autonomous driving, that could take control of the vehicle and avoid some accident in case of detected driver sickness or altered emotional state. All the acquired parameters could also be transmitted to a cloud based system in order to constantly monitor the health condition and the emotional state of the driver, for example for automatically activating health service or live remote assistance in case of necessity. An example of this application is depicted in Fig. 6.1. This task was a part of a H2020 project that ran from 2017 to 2020 called DEIS, which purpose was to develop methods in order to asses the dependability of Cyber-Physical

`http://www.deis-project.eu/`

Figure 6.2: *The proposed rPPG system tested in a driver simulator developed by General Motors.*

Systems. On the described automotive related tasks Politecnico di Milano jointly worked with General Motors and Ideas & Motion.

The rest of this work is organized as follow: in Sec. 6.2 the hardware and software components of the proposed method are described. Following that, in Sec. 6.3 a set of experiments are described in order to evaluate the proposed method and experimental results are reported. Lastly, in Sec. 6.4 the conclusions of the this work are discussed.

6.2 Methods

A complete system was developed in order to solve the rPPG problem described in Sec. 6.1. In particular in Sec. 6.2.1 a complete overview on the hardware involved in the system is described while in the following Sec. 6.2.2 and Sec. 6.2.3 all the software processing is described.

6.2.1 System overview

The complete rPPG system is shown in Fig. 6.2. In particular the SPAD camera in the center of Fig. 6.2 is recording the driver's face and it is connected to the on board computational unit shown in the bottom right of the

https://www.gm.com/
https://www.ideasandmotion.com/

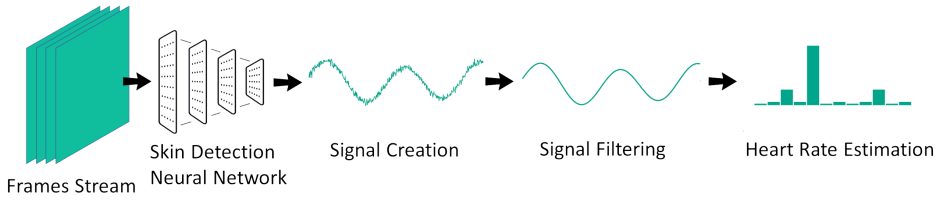

Figure 6.3: *The proposed rPPG method. The frame stream coming from the SPAD camera is firstly analysed with a Neural Network that generates a signal further processed with classical techniques.*

picture (which was developed by Ideas & Motion). Around the camera lens hood a ring light illuminator is mounted. This is composed by two circular stripes of 5 LEDs each. The emitted light spectrum is in the Infra Red (IR) range (in order not to distract the driver) and their intensity is controlled by the on-borad PC. In particular, there a feedback loop was implemented between the illuminator and the camera in which the LED receive current is adjusted in order to keep the received intensity in the SPAD working range. Moreover the maximum illumination power of this device was set to reach no more than $20W/m^2$ of that is the limit for eye safety for the considered wavelength, i.e Maximum Permissible Exposure (MPE). The computational unit was also connected to a small monitor that is able to show in real time the driver's current HR, the illumination power and the results of dependability checks described in Sec. 6.2.4.

6.2.2 Signal extraction

The signal extraction phase is composed by two components (facial skin detection and signal creation) which are depicted as the first two steps in Fig. 6.3. In particular, although the SPAD acquisition frame rate is set to 100 Hz, the deep learning skin detection method is executed at 10 Hz on key-frames obtained by averaging 10 consecutive frames. This is done mainly for computational reasons and in order to reduce acquisition noise (further detail on SPAD sensors noise could be found in [13]).

Skin detection

The majority of rPPG applications [93] make use of face detection methods in order to localize specific regions of the subject face where the pulse signal is extracted. In the proposed system a Convolutional Neural Network

is used instead. In particular the chosen architecture and training procedure are described in Chapter 5. The chosen network has a U-shape [91] and takes a low resolution grayscale image as an input (exactly the same kind of frames produced by the SPAD camera, 64×64) and produces as an output a single channel image, with values between zero and one. In particular, these represent for each pixel the estimated probability of depicting a skin region. As shown in Chapter 5, this method is robust to occlusions and, by considering all the visible facial skin surface, overcomes the problem of selecting *a-priori* a restricted skin region (that could be easy occluded).

As reported in Chapter 5, the first part of the network (i.e. Encoder) is composed by 6 consecutive depthwise separable convolution layers, using 3x3 kernels, coupled with ReLu non linear activation functions. In this phase, 3 of the 6 depthwise separable convolution use a stride of 2 in order to obtain in the last encoding layer a tensor with 1/8 of the original input spatial dimension. Conversely, the second part (i.e. Decoder) is constitute by 3 depthwise separable convolution layers using 3x3 kernels and ReLu activations. These are coupled with upconvolutional layers introduced in order to increase back the spatial dimension to the input one. As illustrated in Fig. 5.3, additive skip connections are used in order to better propagate information between the encoding and decoding parts. The decoding output is concatenated with the input image and two additional standard convolution layers used for denoising, using 3x3 kernels. For this two layers the first activation function is ReLu and the last one uses a sigmoid function in order to obtain output values in the desired range, i.e [0,1].

Signal pre-processing

For each frame acquired, once the relative skin detection output is available, a binary skin mask is obtained by comparing the skin detection output to a fixed threshold. This value was obtained as the best working point by looking at ROC in Fig. 5.4 in which 90% of true positive rate and just 1% of false positive rate are reached. The raw pulse signal is then obtained by averaging the intensity value of all the pixels inside the binary skin mask. The values respectively below and above the 5^{th} and 95^{th} percentiles are removed before computing the average in order to exclude possible outlier values that could be caused by errors in the skin detection step. Fig. 6.4 shows the effect on removing outlier values before computing the sampling mean. In this example, the original data (represented by the histogram in red) are obtained from a skin mask that could obtain some false positive error; in particular, values near the main central peak represent grayscale true positive values while the other smaller peak besides are related to bright-

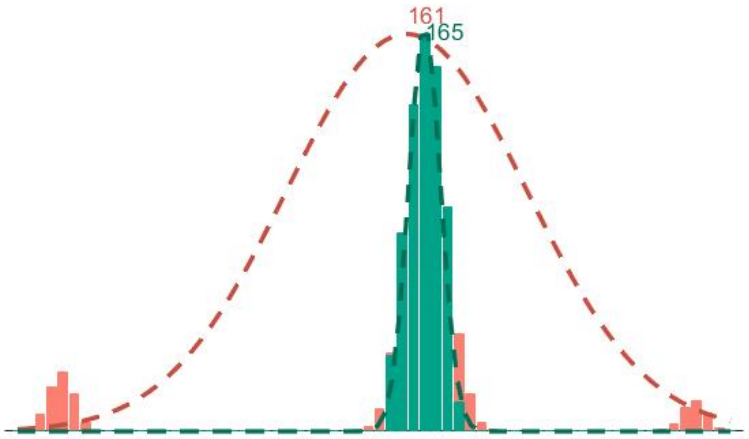

Figure 6.4: *Distribution tails removal effect on computed sample mean. Original data in red, obtained data after tails removal in green.*

ness value of other objects (hair, wearable objects, background) that are mistakenly attributed to the skin. As can be observed from the figure, if the distribution tail are not removed the sample mean is heavily influenced by the outlier and does not correspond to the mean of just the pixel related to the skin. As can be seen from Fig. 6.4, by using fixed percentile some values related to true positive are removed from the average computation. This does not cause any particular problem since the number of samples on which computing the average is not affected excessively. Some other possibilities such as clusterization algorithms or distribution fitting could lead to more accurate results but on the other hand would increase the excessively the computational load. The in time concatenation of the obtain average results in the creation of the pulse signal.

Moreover, in order to remove considerable jumps from the preprocessed pulse signal due to the skin mask variations, the operation illustrated in Fig. 6.5 is performed. In particular an offset value is removed before concatenating the new values to the pulse signal. Furthermore, the maximum signal buffer size has been set to 6000 which correspond to one minute of observations sampled at 100Hz. This has been done for the sake of increasing the estimations' stability by obtaining them on a sufficiently long period of time without increasing excessively their latency. Finally, in order to increase the application's stability if the skin mask could not be calculated for a small interval of time (1 second) the gap created in the sig-

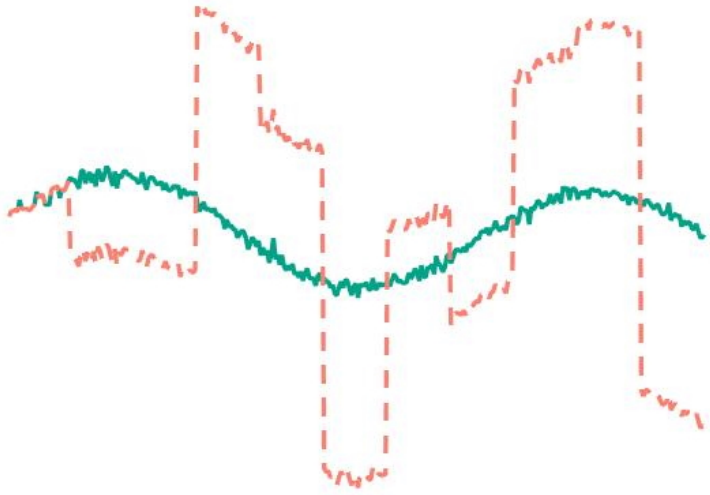

Figure 6.5: *Signal pre-processing opearation. The original signal (in red) could be affected affected by abrupt jumps due to the skin mask recomputation. The denoised signal, in green, is obtained removing them.*

nal is easily filled using linear interpolation between adjacent know signal values. This operation is performed in order to maintain time consistency in the signal and to increase the overall method stability even if small errors could occur in the one minute time frame.

6.2.3 Signal processing

After the signal has been extracted, the signal processing step is performed in order to extract relevant information from the obtained pulse signal.

Filtering

A bandpass Butterworth filter is applied to the signal obtained as described in Sec. 6.2.2. The filter bandwidth is between 0.4 Hz and 4 Hz which is equivalent to 24 bpm and 240 bpm. In particular the chosen filter has reals zeros in -1 and 1 and reals poles in 0.824 and 0.966. This is mainly done in order to cut out any other signal having a frequency very distant from a possible HR. A zero-phase filtering approach is used in this stage, in particular the filter is applied in the time domain to the signal then the returned filtered signal is reversed in time and the filter is applied again. The resulting filter is then flipped back to the original time direction.

Average heart rate estimation

In order to estimate the average heart rate from the camera signal, the following two operations are performed. Firstly, after applying the preprocessing steps described in Sec. 6.2.2, the first and last half seconds are removed from the original signal. This is done in order to remove transient effects that could be caused by the filter in the begin and end part of the signal. Since the following operations are performed in the frequency domain and since the length of the portion removed is much smaller than the total signal length, this operation do not compromise the final heart rate estimation. The power spectrum of the pulse signal is obtained applying a Fast Fourier Transform (FFT) on the filtered signal. Finally, in order to estimate the average Heart Rate, the frequency related to the peak of the power spectrum is chosen.

6.2.4 Dependability processing

Being rPPG an optical method it could be affected by optical alterations. In particular, some scenarios could occur in which the pulse signal could be masked by much stronger noise due to many different sources. The two main scenarios that we identified are the presence of subject head periodic movements and background pulsating light.

Periodic head movements

The estimation of the main pulse signal frequency could be affected by periodic head movements. In particular, such movements are in the HR frequency band and could mask the true HR frequency altering the rPPG HR estimation. For this reason, a visual based method able to detect periodic head movements have been developed. The first step of this method is to keep track of the head position for each analyzed frame. In order to do this, for each key-frame, the central skin mask point is tracked averaging the coordinates of the skin mask itself. An example of face position data gathered with an oblique periodic motion is reported in Fig. 6.6; in particular each red dot is the obtained position of the head central point in a particular frame inside the time frame considered. Although this simple method could introduce some errors, in particular in case of face rotation, it's suitable for a real time implementation due to its low computational cost. Once the two (vertical and horizontal coordinates) time varying variables, $\mathbf{x}$ and $\mathbf{y}$, related to the pixel position of the face has been estimated, a Principal Component Analysis (PCA) is used in order to combine this

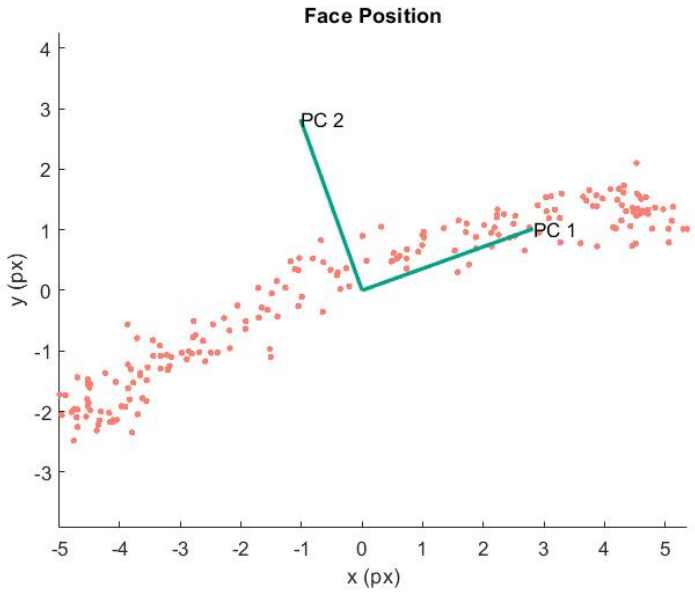

Figure 6.6: *Each red dot represents the face central pixel position in the considered time frame. The green axis are the computed principal components.*

information into a single signal. In particular, the two position column vectors are stored into a matrix $\mathbf{X} = [\mathbf{x}\ \mathbf{y}]$. The average of each column of matrix $\mathbf{X}$ is then shifted to zero just by subtracting $\bar{\mathrm{x}}$ and $\bar{\mathrm{y}}$, the empirical averages, to each element of the first and second column respectively. The PCA is then applied to obtained zero mean matrix $\bar{\mathbf{X}}$ by the use of Singular Value Decomposition (SVD):

$$[\mathbf{U}\ \mathbf{S}\ \mathbf{V}] = \mathrm{SVD}(\bar{\mathbf{X}}) \tag{6.1}$$

Where in particular the matrix $\mathbf{V}$ represents the PCA projection matrix. The data in the new PCA cohordinate system are subsequently obtained as:

$$\hat{\mathbf{X}} = \bar{\mathbf{X}}\mathbf{V} \tag{6.2}$$

In particular, the PCA is used to find the principal axes that compose the movement and the coordinates are projected to the principal component. After this operation only the first column of $\hat{\mathbf{X}}$ is kept which represents the values over time on the principal axes that compose the movement. As can be observed from Fig. 6.6, the first principal component is the one on which the variance is maximized, while the other one is just the perpendicular one since the data live in a bidimensional space. Fig. 6.7 shows the values

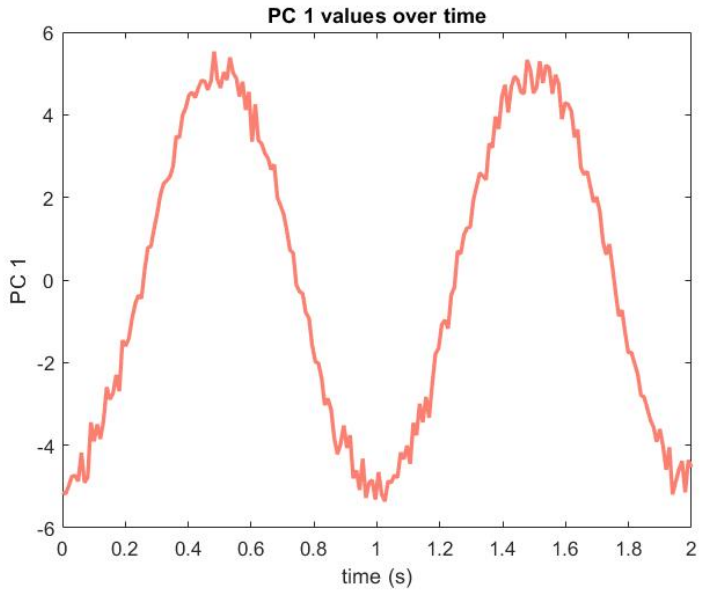

Figure 6.7: *First principal component values over time.*

assumed in the first principal component over time using the same data reported in Fig. 6.6. This process creates a 1D time varying signal on which FFT is applied in order to estimate its main frequency. Its signal power spectrum is then used in order to estimate a score defined as the percentage of the area below the peak in respect to the total area below the power spectrum graph (Fig. 6.8). In particular, the area under the peak is defined as the area below the graph between the interval defined by the two points respectively on the left and right of the peak in which the curve value reach 25% of the peak one. Ideally, in presence of periodic head movement, a single peak would be visible in the power spectrum so the score would be very close to 100% (its maximum value). On the other hand, if the peak would not be clearly visible in the power spectrum (due to noise) the score would be much lower. The periodic head movement is then detected using the aforementioned movement related score, in particular checking if the score value is greater a fixed threshold, optimized during a training procedure. In this way periodic head movements could be detected.

Pulsating light

Another possible situation in which the rPPG method could lead to incorrect results is in presence of strong pulsating ambient light in the same

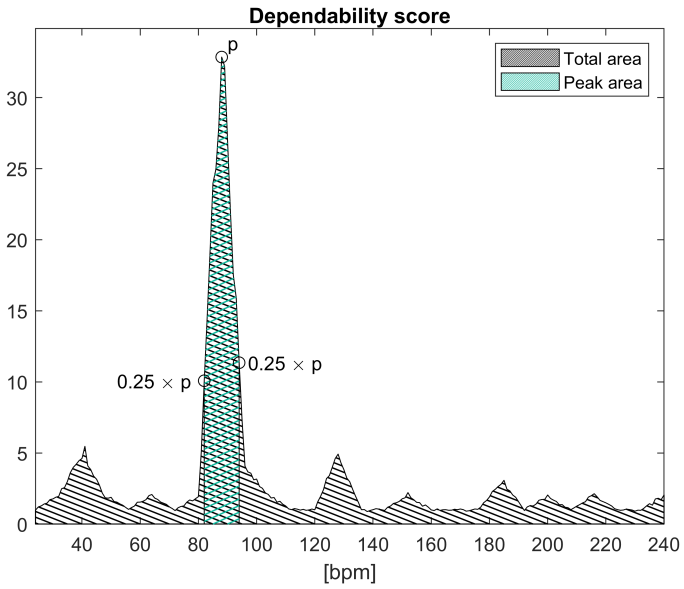

Figure 6.8: *Dependability score definition. Both periodic head movement and pulsating light scores are defined as the ratio between area under peak (blue) and total area (black)*

typical band of the HR. This situation could occur, for example, while driving in a tunnel; in this case the intensity of the light that illuminates the driver's face varies in time in respect to the distance of the closest lamp. In this situation the ambient light fluctuations would add up to the ones related to the heart activity in the observed pulse signal and, if the first ones are strong enough, would mask the HR related information. As for the periodic movement detection an auxiliary signal is needed in order to detect this situation. In particular, an additional environmental intensity signal is extracted averaging the value of background pixels. These are defined as the pixels of the image outside the detected skin mask. The background signal power spectrum is then extracted via FFT. Also in this case a score is defined as the area below the main peak divided by the total area below the power spectrum graph (i.e. the total power). The score obtained is then used in order to detected pulsating ambient light comparing its value to a fixed threshold, optimized during a training procedure.

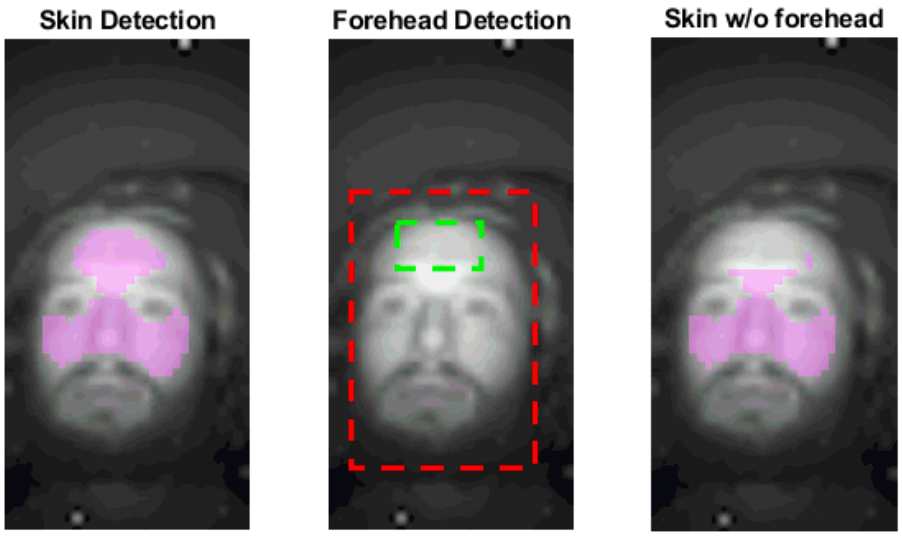

Figure 6.9: *Three different regions considered in the Deep Learning based signal extraction evaluation. On the left the region obtained from the DL skin detection algorithm. In the center the forehead region obtained with classical methods. The third region is the subtraction of the second from the first one.*

6.3 Results

In the following session two experiments will be described. The first one has been conducted in order to evaluate the impact of introducing a Deep Learning based method in the signal extraction stage while the second one is related to the impact of the dependability checks introduced.

6.3.1 Deep learning based signal extraction

In order to test the advantage of using a deep learning skin detection algorithm instead of a classical face detection method, a specific experiment has been performed. In particular the heart rate estimation obtained with the method described in Sec. 6.2.3 has been compared to the one obtained with a classical rPPG approach, as the one in [94]. In classic rPPG an optimal face region (usually the forehead) is detected by applying fixed proportion to a bounding box obtained with classical face detection methods (e.g. [117]). In order to test the differences between the two methods three signals have been extracted and analysed with the same processing described in Sec. 6.2.3. In particular one signal has been obtained using

the proposed skin detection algorithm (using the region depicted in the first panel of Fig. 6.9) while another one was extracted exploiting classical face detection method such as [117]. This is an accurate and efficient method for object detection widely used in many Computer Vision applications. It is a strong classifier build upon a cascade of weak classifiers based on Haar features. These rectangular features are able to compare different parts of a grayscale image, using different patterns, simply summing and subtracting the pixel value of the image in particular regions. In order to perform the feature evaluation step in an efficient fashion (constant in time), the original image is analysed only once in order to create an integral image. The integral image is defined as an image in which each pixel value is equal to sum of the pixel values in the area above and to the left of the considered pixel.

Since the SPAD camera output has a very small spatial resolution (32×64) the output frames are scaled by a factor of 10 before applying the face detection algorithm, using bicubic interpolation. A border padding of 50 pixel is also added in order to detect faces very near the image borders or partially outside of them. The ROI coordinates are then accordingly scaled back to the original resolution and the signal is extracted from the original resolution frames. As in [94], the face detector is coupled with a face tracking algorithm. In particular, if the face was already detected in the last iteration a tracking algorithm is used instead of the face detection one. Firstly same features are detected inside the face region returned by the detection algorithm on the previous frame, using the Shi's and Tomasi's "Good Features to Track" algorithm [99]. Consequently, these features are tracked forward to the current frame using the Kanade-Lucas-Tomasi (KLT) algorithm [120]. From the previous and current pixel positions of the tracked points a 2D rigid transformation (homography) is estimated and the face bounding box is transformed accordingly. From the bounding box containing the driver's face a Region Of Interest (ROI) centred around the subject forehead is calculated using fixed proportions. This ROI is depicted in the second panel of Fig. 6.9.

Lastly, an additional signal has been extracted by removing from the skin binary mask the forehead region obtained as described above, the obtained region is depicted in the last panel of Fig. 6.9. This was done in order to test the scenario in which the forehead region is unavailable, for example in case of occlusion due to hair presence of wearable objects. Two sequences with two different subjects (one male and one female) were recorded while driving in a car simulator. The SPAD camera, equipped with a 850 nm optical filter, was mounted approximately at 50 cm from the

Table 6.1: *Comparison of hearth rate estimation between signal extracted with deep learning based facial skin detection (**Skin**) versus classical face detection method (**Foreh.**).*

	Skin		Foreh.		Skin w/o Foreh.	
[bpm]	**RMSE**	**std**	**RMSE**	**std**	**RMSE**	**std**
Sbj 1	2	1.4	2	1.4	2	1.4
Sbj 2	1.4	0	1.4	1.4	1.4	0
Avg.	1.7	0.7	1.7	1.4	1.7	0.7

subject's face and the active infrared illumination described in Sec. 6.2.1 was used. The grand truth heart rate values was obtained with the Faros ECG device.

Experimental results

Results are reported in Tab. 6.1. As we can observe, the use of the proposed skin detection method performs as well as using classical face detection methods. In addition to that, the proposed method has the benefit of working also in situations in which the forehead skin intensity is not available, as can be observed from the last row of the table.

6.3.2 Dependability checks evaluation

The dependability checks described in Sec. 6.2.4 have been evaluated experimentally in two different set of acquisitions. For each one of the two checks two sequences, with two subjects (one male and one female), were recorded while using the same driving simulator described above. Also in this case, the SPAD camera, equipped with the 850 nm optical filter, was mounted approximately at 50 cm from the subject's face. In the two sequences recorded in order to test the ambient pulsating light, an incandescent lamp was used. This external light source was modulated at a frequency of 60 Hz and was turned on with a random delay from the record starting and the delay in the detection time (using the algorithm described in Sec. 6.2.4) was recorded. On the other hand, in order to test the periodic head movement detection the external light source was not used and instead the subject was asked to start moving periodically their head left to right at a fixed frequency of approximately 1 Hz. Also in this case, the detection time of the periodic head movement was recorded.

Experimental results

In all the 4 tested sequences the optical noise injected was correctly detected. In particular the delay detection for pulsating light has been of 13 seconds and 15.5 seconds for periodic head movements. This delays were expected due to the 1 minute signal window used and described in Sec. 6.2.2.

6.4 Discussion and conclusions

The work presented in this chapter describes a rPPG system based on SPAD camera. A detailed description of the system hardware is given in Sec. 6.2.1. In Sec. 6.2.2, the adoption of a Deep Learning based method for facial skin segmentation has been illustrated. In particular the main motivation for utilizing a segmentation method was to be able to use all the possible pixel surface related to the heart activity. As a matter of fact, using a traditional forehead region adopted in many rPPG systems [94], given the very low spatial resolution of SPAD cameras, would result in selecting very few pixel for the pulse signal estimation. Results reported in Sec. 6.3.1 shows a slight increment in heart rate estimation accuracy while using the deep learning skin segmentation method instead of forehead region obtain with traditional computer vision techniques. More importantly, this experiment highlight how the rest of the skin region detected by the Deep Learning method, excluding the forehead region, still carries pulse information and this method could achieve good quality results even in presence of occlusions (e.g caused by the presence wearable objects or hair) that could make the forehead region unavailable.

Moreover the proposed system is able to perform dependability checks in order to detect anomaly situations. In particular, since the rPPG biomedical parameters estimation is performed exclusively using optical information two scenarios were evaluated in which the pulse signal could be masked by other signals. The two scenarios that we identified are the presence of subject head periodic movements and background pulsating light. As a matter of fact, the periodic movement of the subject head could lead to the masking of the pulse signal with the one created by the light reflections introduced to the periodic movement. Moreover, the frequency of this kind of periodic movement is typically in the heart rate range. In this situation the pulse signal could be mistaken to the movement one and this could lead to false heart rate estimation. On the other hand, also the presence of pulsating ambient light with strong intensity could lead to the masking of the pulse signal. In Sec. 6.2.4 two different methods were described in order to

detect these situations and in Sec. 6.3.2 two experiments are conducted in order to test the precision of the introduced methods. Results shows that in all the acquired test sequences both the anomalies (pulsating ambient light and periodic head movements) were correctly detected.

CHAPTER 7

Conclusions

The main achievement of this work is the development of a rPPG system able to estimate numerous biomedical measurements in real time and in a dependable fashion adopting a SPAD camera as the imaging sensor and using deep learning jointly with traditional signal processing in order to achieve the biometric estimations. In particular, after introducing the problem in Chapter 1 and presenting the state of the art in this field in Chapter 2, the use of the SPAD camera for rPPG has been discussed in Chapter 3. In Chapters 4 and 5 the development of a deep learning skin segmentation for low resolution grayscale images has been described. Finally in Chapter 6 the overall rPPG system jointly adopting SPAD camera and Deep Learning has been introduced. In the following section an in-depth discussion of the major achievements of each chapter will be described.

7.1 General discussion and conclusions

In this session a discussion of all the major achievements reached in each chapter will be described.

Chapter 2

In this chapter an overview of the state of the art on the PPG and rPPG systems and methods has been discussed analysing also the different between them. Moreover a particular focus has been given to Deep Learning methods in general and their particular use in the rPPG field. As described, the adoption of deep learning methods in rPPG is very recent and the current work represent one of the first attempting in this direction. All the other works adopting deep learning in rPPG, completely substitute the classical signal processing techniques with data driven ones using an end-to-end approach. On one hand, the use of an end-to-end deep learning model has proven to achieve state of the art results on many computer vision tasks. On the other hand, this kind of methods required a massive amount of training data in order to learn how to extract heart related information directly from video frames and no prior domain knowledge is incorporated. This make the performance of this kind of methods tightly linked to the training dataset and potentially unable to generalize in different setting conditions. Moreover, the complete substitution of classical signal processing techniques developed using a solid theoretical background (signal filtering, Fourier transform, etc.) with data driven ones could lead to non-optimal solutions. Moreover even than some of this work claim to achieve realtime performances as [95] they require powerful GPU. For the best of our knowledge no prior work has been done in trying to combine traditional and deep learning based signal processing in this field. Lastly, in all the considered studies the cameras used are traditional RGB cameras and to the best of our knowledge no prior work explored the possibility of using SPAD cameras for rPPG applications.

Bullet point achievements:

- PPG is widely used in commercial and clinical devices.
- Many rPPG systems have been developed in recent years.
- SPAD cameras has never been used in this field.
- Deep Learning has been used in rPPG starting 2019 implementing end-to-end solutions.

Chapter 3

In this chapter the possibility of performing rPPG using a SPAD camera to compute HR, HRV and RR had been investigated. The working principle and reason behind the use of SPAD cameras had been discussed in Sec. 3.1. In this work two experiments have been set up, performing measurements

on a subject sat still in front of the camera with the artificial illumination directed on its face. The values of the pixels inside a manually obtained ROI were averaged resulting in a pulse wave. This was the starting signal that was processed in order to estimate HR, HRV and RR. In order to evaluate SPAD based rPPG five parameters were considered: single beat detection, heart rate estimation, tachogram estimation, LF/HF estimation and respiration rate estimation. In order to perform and validate biometric measurements with a SPAD camera and compare it to estimation that could be obtained from a traditional RGB camera, a portable ECG device was used for reference. One of the two experiments conducted (experimental setup described in Sec. 3.3.1) had the aim of comparing the SPAD rPPG performance using light with different wavelength. As can be observed from results reported in Sec. 3.5.1, 550 nm light (i.e. green light) is able to achieve the better results. Many parameters influence this result, in particular the most significant are light penetration depth in the tissues [8], absorption coefficient of the oxygenated hemoglobin [126], SPAD efficiency [13] and illumination power. Light below 500 nm is mostly reflected by stratum corneum, which is the most external skin layer, which being not reached by blood does not contain any information on pulse wave. Concerning light between 600 nm and 750 nm, the absorptivity of oxygenated hemoglobin is very low, thus reducing the modulation in rPPG signal. Therefore, only wavelengths between 500 nm an 600 nm and between 750 nm and 900 nm are able to carry useful signal. As a matter of fact, as shown from the results reported in Sec. 3.3.1, the best performance are achieved using 550 nm light but reasonable results are also achieved using near infrared light (750 nm to 850 nm). This is promising results since many scenarios could be imagined in which the use of non-visible light could be preferred (e.g. in the automotive field an rPPG system could be used in order to monitor the health state of the driver).

The second experiment (described in Sec. 3.3.2) was conducted in order to compare the rPPG SPAD based performance with the one obtainable using traditional RGB cameras. As can be observed in a normal light scenario, as reported in Sec. 3.5.2, SPAD cameras are able to achieve comparable results in respect to RGB cameras in heart rate estimation and slightly superior accuracy in estimation of the tachogram and respiration rate.

Bullet point achievements:

- It is possible to perform rPPG using SPAD cameras.
- Light with 550 nm wavelength performs best in implementing rPPG using a SPAD camera.

- It is also possible to achieve good quality results using light with 850 nm wavelength, which can be preferable in some scenarios being infrared light.
- SPAD cameras are able to achieve comparable to slightly superior results in respect to RGB cameras in rPPG estimations.

Chapter 4

In this chapter,a Deep Learning based method was proposed in order to solve the facial skin detection problem on low-resolution grayscale images. The low spatial resolution (64x64 pixels) coupled with the unavailability of color information (grayscale images) made this task particularly ambitious. Analyzing the state of the art of similar problems, in Sec. 4.1.1, we showed the peculiarity of the proposed task and how, to the best of our knowledge, the method described in this work is the first being proposed specifically to solve the specific task of facial skin detection.

Given the similarity between this problem and a semantic segmentation one, and the good accuracy achieved by neural network methods in this latter field, a Deep Learning based method was chosen. On the other hand, these kind of methods need massive amount of data to be trained on. Since the facial skin detection problem, tackled in his chapter, is very specific unfortunately only a limited amount of data are available for this specific problem. For this reason a transfer learning approach was adopted in the training phase. In particular, the proposed network architecture was chosen in order to have the majority of layers in common with a convolutional neural network proposed to solve the grayscale images colorization problem [9]. These apparently different problems are in reality tightly linked as a colorization method, in order to work on face images, must (implicitly) solve the skin detection problem, since it needs this information in order to color in a correct way pixels depicting skin regions. On the other hand, since the skin detection problem is only a small sub-task in respect to the colorization one, the proposed network was significantly simplified, as shown in Sec. 4.2.2. Further information about the similarities between the skin detection and colorization problems are described in Sec. 4.2.2.

As discussed in Sec. 4.2.3, in order to exploit the maximum amount of knowledge possible gathered from the colorization problem, a three step transfer learning strategy was adopted. Firstly the colorization method was trained on a large dataset of unlabeled face images. This was done in order to drive the preliminary method into the specific domain of face image analysis. The proposed skin detection network was subsequently trained starting from the colorization network weights and minimizing an asymmetric

loss function, described in Sec. 4.2.3, on a novel constructed dataset. This was done in two consecutive steps in order to train new and already trained layers at two different speed. In Sec. 4.2.4 the training dataset, containing more than 6000 labeled training images and 200 labeled test images, was described, detailing also all the operations performed in order to adapt the two existing and freely available datasets (MUCT [77] and Helen [124]) to the specific skin detection problem.

Lastly in Sec. 4.3.2 the proposed training procedure has been justified showing that, without using it, it would be not be possible to train the proposed network with the few data available. In particular Fig. 4.7 shows that the adapted training procedure is able to avoid overfitting since the validation error does not increase over the training epochs. Moreover, Fig. 4.11 demonstrate that evaluation of the model on new images produces qualitatively good results, reiterating the absence of overfitting in the training process. In addition, in Sec. 4.3.3 some quantitative results were reported providing accuracy evaluation for the proposed skin detection method and showing comparisons with a state of the art face segmentation method. In particular the proposed method is able to outperform [83] in the specific task of facial skin detection on low resolution grayscale images, even when GT information where integrated to [83]. Moreover, in Sec. 4.3, many skin detection outputs were shown for both images acquired in similar conditions with respect to the ones used to built the training set and for images completely independent from the training set, acquired with the SPAD camera. Both these results show how the proposed method is able to achieve quantitative and qualitative good results in the skin detection problem even in presence of different poses, ages, expressions, ethnicity, wearable objects and other occlusions.

Bullet point achievements:

- A labeled facial skin dataset has been built and made publicly available.
- A transfer learning approach was successfully implemented exploiting knowledge from an apparently unrelated task in order to overcome the scarcity of training data.
- A CNN for skin segmentation on low resolution grayscale facial images have been developed, achieving good accuracy.

Chapter 5

In this chapter, we presented a Deep Learning based method proposed in order to solve the facial skin detection problem on low-resolution grayscale images in real-time. As discussed in Sec. 5.1, a model that could run in real-time on hardware with limited computational power could be useful in many situations. In particular in the automotive domain even if more powerful GPU based computational power is available, having a compact dedicated solution to the rPPG problem solution could be the ideal scenario in order to not overload the main computational unit.

In order to achieve this goal a new Deep Learning based model for solving the skin detection problem have been proposed. This method is able to work in presence of low spatial resolution (64x64 pixels) coupled with the unavailability of color information (grayscale images) as the method described in Chapter 4. The main difference between the two methods is the adoption of depthwise separable convolution layer. As described in Sec. 5.2.1, these kinds of layers are obtained splitting the operation performed in a traditional convolution layer with two consecutive steps of spatial filtering and channel combination. In the same section we show also how and why this decoupling is able to drastically reduce the number of parameters. Moreover, the complete network architecture has been described in Sec. 5.2.2 highlighting the choice made in order to maximally increase the similarity with the one described in Chapter 4 and the usage of skip connections.

As discussed in Sec. 5.2.3, in order to exploit the maximum amount of knowledge possible gathered from the skin detection method described in Chapter 4, an incremental transfer learning strategy was adopted. Starting from the model described in Chapter 4 exploiting the similarity between the two networks, traditional convolution layers are one by one incrementally substituted with depthwise separable ones. Ten epochs of training are performed between every substitution. This operation effectively force the new layers, with smaller amount of parameters, to approximate the much larger traditional ones. An additional 50 epochs of training were then applied on the complete network.

Furthermore in Sec. 5.3.1 the proposed model was tested in both precision and real time performance. In particular Fig. 5.4 shows that the proposed method is able to outperform the much larger one described in Chapter 4 also in the skin detection accuracy. This is probably due to the reduced number of parameters that increased the easiness for the optimization algorithm to converge to a better minimum of the loss function. Fig. 5.5 also shows some qualitative results highlighting the precision of the model in selecting the face region discarding parts not related to the skin. Lastly

in Sec. 5.3.2 the time performances of the method proposed and one described in Chapter 4 are compared. In particular, a C++ implementation of the proposed method is able to achieve real-time performances even when run on a limited compact single-board computer such as the Odroid XU-4.

Bullet point achievements:

- A custom deep learning method was developed exploiting depth-wise separable convolution layers.
- Even if this method is significantly smaller than the one proposed in Chapter 4, is able to outperform it in term of skin detection accuracy, proving that not always greater amount of parameters correspond to superior models.
- The proposed CNN is able to run in real-time on a CPU of a single board computer such as the Odroid XU-4.

Chapter 6

The work presented in this chapter described a rPPG system based on SPAD camera. A detailed description of the system hardware is given in Sec. 6.2.1. In Sec. 6.2.2, the adoption of a Deep Learning based method for facial skin segmentation has been illustrated. In particular the main motivation for utilizing a segmentation method was to be able to use all the possible pixel surface related to the heart activity. As a matter of fact, using a traditional forehead region adopted in many rPPG systems [94], given the very low spatial resolution of SPAD cameras, would result in selecting very few pixel for the pulse signal estimation. Results reported in Sec. 6.3.1 shows a slight increment in heart rate estimation accuracy while using the deep learning skin segmentation method instead of forehead region obtain with traditional computer vision techniques. More importantly, this experiment highlight how the rest of the skin region detected by the Deep Learning method, excluding the forehead region, still carries pulse information and this method could achieve good quality results even in presence of occlusions (e.g caused by the presence wearable objects or hair) that could make the forehead region unavailable.

Moreover the proposed system is able to perform dependability checks in order to detect anomaly situations. In particular, since the rPPG biomedical parameters is performed exclusively using optical information two scenarios were evaluated in which the pulse signal could be masked by other signals. The two scenarios that we identified are the presence of subject

head periodic movements and background pulsating light. As a matter of fact, the periodic movement of the subject head could lead to the masking of the pulse signal with the one created by the light reflections introduced to the periodic movement. Moreover, the frequency of this kind of periodic movement is typically in the heart rate range. In this situation the pulse signal could be mistaken to the movement one and this could lead to false heart rate estimation. On the other hand, also the presence of pulsating ambient light with strong intensity could lead to the masking of the pulse signal. In Sec. 6.2.4 two different methods were described in order to detect this situation and in Sec. 6.3.2 two experiments are conducted in order to test the precision of the introduced methods. Results shows that in all the acquired test sequences both the anomalies (pulsating ambient light and periodic head movements) were correctly detected.

Bullet point achievements:

- A SPAD based rPPG system has been proposed.
- The adoption of the CNN skin segmentation method proved to be able to achieve good quality results and to be more robust to occlusions in respect to a traditional face detection algorithm.
- The adopted dependability checks proved to be able to detect situations in which an optical method such as rPPG could fail improving the reliability of results obtained with this method.

7.2 Directions for future work

The presented work, being one of the pioneering work in adopting deep learning in rPPG application and the first one in using SPAD camera in this field, have many different promising directions for future work.

Training dataset In evaluating the skin segmentation method on multiple test images, it could be notice that some labelling errors still occurs especially in presence of beard, glasses and other occlusions. This is due to the fact that the training dataset was automatically annotated using color similarity to label this kind of occlusions and this inevitably introduces some errors. The adoption of an accurate hand-labeled training dataset surly could increase this method precision.

Time-dependent skin segmentation method The implemented skin segmentation method could also benefit from some more advance deep learning architectures, developed for working with a stream of frame.

In this case a Recurrent Neural Network (RNN) [31], such as LSTM [40] and GRU [21], could integrate the current CNN architecture achieving the required result.

HR Deep Learning estimation Some part of the classical signal processing techniques could be substituted with deep learning based method. In particular, the pulse signal analysis could be performed by a CNN. The monodimensional signal could be transformed to 2D using spectrogram representation, or gammatonegram [68], and fed to a standard CNN architecture. This method is already used in sound signal analysis [68] and its adoption in this field could tested against classical signal processing.

Bibliography

[1] ABADI, M., AGARWAL, A., BARHAM, P., BREVDO, E., CHEN, Z., CITRO, C., CORRADO, G. S., DAVIS, A., DEAN, J., DEVIN, M., GHEMAWAT, S., GOODFELLOW, I., HARP, A., IRVING, G., ISARD, M., JIA, Y., JOZEFOWICZ, R., KAISER, L., KUDLUR, M., LEVENBERG, J., MANE, D., MONGA, R., MOORE, S., MURRAY, D., OLAH, C., SCHUSTER, M., SHLENS, J., STEINER, B., SUTSKEVER, I., TALWAR, K., TUCKER, P., VANHOUCKE, V., VASUDEVAN, V., VIEGAS, F., VINYALS, O., WARDEN, P., WATTENBERG, M., WICKE, M., YU, Y., AND ZHENG, X. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org.

[2] AGARWAL, R., DIAZ, O., LLADO, X., YAP, M. H., AND MARTI, R. Automatic mass detection in mammograms using deep convolutional neural networks. *Journal of Medical Imaging 6*, 3 (2019), 1 – 9.

[3] AHMED, A., HARNESS, J., AND MEARNS, A. Respiratory control of heart rate. *European Journal of Applied Physiology and Occupational Physiology 50* (02 1982), 95–104.

[4] AL-KHALIDI, F., SAATCHI, R., BURKE, D., AND ELPHICK, H. Facial tracking method for noncontact respiration rate monitoring. *Communication Systems Networks and Digital Signal Processing (CSNDSP), 2010 7th International Symposium on* (2010), 751–754.

[5] ALBOTA, M. A., HEINRICHS, R. M., KOCHER, D. G., FOUCHE, D. G., PLAYER, B. E., O'BRIEN, M. E., AULL, B. F., ZAYHOWSKI, J. J., MOONEY, J., WILLARD, B. C., AND CARLSON, R. R. Three-dimensional imaging laser radar with a photon-counting avalanche photodiode array and microchip laser. *Appl. Opt. 41*, 36 (Dec 2002), 7671–7678.

[6] ALLEN, J. Photoplethysmography and its application in clinical physiological measurement. *Physiological measurement 28*, 3 (March 2007), R1–39.

[7] ANTI, M., TOSI, A., ACERBI, F., AND ZAPPA, F. Modeling of afterpulsing in single-photon avalanche diodes. In - (2011), SPIE, pp. 79331R–1–79331R–8.

[8] ASH, C., DUBEC, M., DONNE, K., AND BASHFORD, T. Effect of wavelength and beam width on penetration in light-tissue interaction using computational methods. *Lasers in Medical Science 32*, 8 (2017), 1909–1918.

[9] BALDASSARRE, F., GONZALEZ-MORIN, D., AND RODES-GUIRAO, L. Deep-koalarization: Image colorization using cnns and inception-resnet-v2. *ArXiv:1712.03400* (Dec. 2017).

[10] BLACKFORD, E. B., AND ESTEPP, J. R. Effects of frame rate and image resolution on pulse rate measured using multiple camera imaging photoplethysmography. In *Medical Imaging 2015: Biomedical Applications in Molecular, Structural, and Functional Imaging* (2015), B. Gimi and R. C. Molthen, Eds., vol. 9417, International Society for Optics and Photonics, SPIE, pp. 639 – 652.

[11] BOBBIA, S., MACWAN, R., BENEZETH, Y., MANSOURI, A., AND DUBOIS, J. Unsupervised skin tissue segmentation for remote photoplethysmography. *Pattern Recognition Letters 124* (2019), 82 – 90. Award Winning Papers from the 23rd International Conference on Pattern Recognition (ICPR).

[12] BOUSEFSAF, F., PRUSKI, A., AND MAAOUI, C. 3d convolutional neural networks for remote pulse rate measurement and mapping from facial video. *Applied Sciences 9*, 20 (2019).

[13] BRONZI, D., VILLA, F., TISA, S., TOSI, A., AND ZAPPA, F. Spad figures of merit for photon-counting, photon-timing, and imaging applications: A review. *IEEE Sensors Journal 16* (2016), 3–12.

[14] BRONZI, D., VILLA, F., TISA, S., TOSI, A., ZAPPA, F., DURINI, D., WEYERS, S., AND BROCKHERDE, W. 100 000 frames/s 64 x 32 single-photon detector array for 2-D imaging and 3-D ranging. *IEEE Journal on Selected Topics in Quantum Electronics 20*, 6 (2014).

[15] BRONZI, D., ZOU, Y., VILLA, F., TISA, S., TOSI, A., AND ZAPPA, F. Automotive three-dimensional vision through a single-photon counting spad camera. *IEEE Transactions on Intelligent Transportation Systems 17*, 3 (March 2016), 782–795.

[16] BUTLER, M. J., CROWE, J. A., HAYES-GILL, B. R., AND RODMELL, P. I. Motion limitations of non-contact photoplethysmography due to the optical and topological properties of skin. *Physiological Measurement 37* (2016).

[17] CHALLONER, A. V. J., AND RAMSAY, C. A. A photoelectric plethysmograph for the measurement of cutaneous blood flow. *Physics in Medicine and Biology 19* (5 1974), 003.

[18] CHARLTON, P., BONNICI, T., TARASSENKO, L., CLIFTON, D. A., BEALE, R., AND WATKINSON, P. J. An assessment of algorithms to estimate respiratory rate from the electrocardiogram and photoplethysmogram. *Physiological Measurement 37*, 4 (2016), 610–626.

[19] CHEKMENEV, S. Y., FARAG, A. A., MILLER, W. M., ESSOCK, E. A., AND BHATNAGAR, A. Multiresolution approach for noncontact measurements of arterial pulse using thermal imaging bt - augmented vision perception in infrared: Algorithms and applied systems.

[20] CHEN, H., ENGKVIST, O., WANG, Y., OLIVECRONA, M., AND BLASCHKE, T. The rise of deep learning in drug discovery. *Drug Discovery Today 23*, 6 (2018), 1241 – 1250.

[21] CHO, K., VAN MERRIËNBOER, B., GULCEHRE, C., BAHDANAU, D., BOUGARES, F., SCHWENK, H., AND BENGIO, Y. Learning phrase representations using RNN encoder–decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (Doha, Qatar, Oct. 2014), Association for Computational Linguistics, pp. 1724–1734.

[22] CHOLLET, F., ET AL. Keras. `https://github.com/fchollet/keras`, 2015.

[23] COVA, S., GHIONI, M., LACAITA, A., SAMORI, C., AND ZAPPA, F. Avalanche photodiodes and quenching circuits for single-photon detection. *Applied Optics 35* (1996), 1956–1976.

[24] DE HAAN, G., AND JEANNE, V. Robust pulse rate from chrominance-based rPPG. *IEEE Transactions on Biomedical Engineering 60*, 10 (2013), 2878–2886.

[25] DENG, J., SOCHER, R., FEI-FEI, L., DONG, W., LI, K., AND LI, L.-J. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition(CVPR)* (06 2009), vol. 00, pp. 248–255.

[26] DING, W., HUANG, Z., HUANG, Z., TIAN, L., WANG, H., AND FENG, S. Designing efficient accelerator of depthwise separable convolutional neural network on fpga. *Journal of Systems Architecture 97* (2019), 278 – 286.

[27] DOCAMPO, N., AND CASAS, P. *Heart rate estimation using facial video information*. PhD thesis, 2011.

[28] EL-HAJJ, C., AND KYRIACOU, P. A review of machine learning techniques in photoplethysmography for the non-invasive cuff-less measurement of blood pressure. *Biomedical Signal Processing and Control 58* (2020), 101870.

[29] GHAMARI, M. A review on wearable photoplethysmography sensors and their potential future applications in health care. *International Journal of Biosensors & Bioelectronics 4* (2018).

[30] GIRAUD, G., SCHULZE, H., LI, D.-U., BACHMANN, T., CRAIN, J., TYNDALL, D., RICHARDSON, J., WALKER, R., STOPPA, D., CHARBON, E., HENDERSON, R., AND ARLT, J. Fluorescence lifetime biosensing with dna microarrays and a cmos-spad imager. *Biomedical Optics Express 1*, 5 (12 2010), 1302–1308.

[31] GOODFELLOW, I., BENGIO, Y., AND COURVILLE, A. *Deep Learning*. MIT Press, 2016. `http://www.deeplearningbook.org`.

[32] GUDI, A., BITTNER, M., LOCHMANS, R., AND GEMERT, J. V. Efficient real-time camera based estimation of heart rate and its variability. *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)* (2019), 1570–1579.

[33] GUO, Y., LIU, Y., GEORGIOU, T., AND LEW, M. S. A review of semantic segmentation using deep neural networks. *International Journal of Multimedia Information Retrieval 7*, 2 (Jun 2018), 87–93.

[34] HASSAN, M., MALIK, A., FOFI, D., SAAD, N., KARASFI, B., ALI, Y., AND MERIAUDEAU, F. Heart rate estimation using facial video: A review. *Biomedical Signal Processing and Control 38* (2017), 346 – 360.

[35] HE, K., ZHANG, X., REN, S., AND SUN, J. Deep residual learning for image recognition. *CoRR abs/1512.03385* (2015).

[36] HERTZMAN, A. B. Photoelectric plethysmography of the fingers and toes in man. *Proceedings of the Society for Experimental Biology and Medicine 37*, 3 (1937), 529–534.

[37] HERTZMAN, A. B. The blood supply of various skin areas as estimated by the photoelectric plethysmograph. *American Journal of Physiology-Legacy Content 124* (1938).

[38] HERTZMAN, A. B., AND DILLON, J. B. Applications of photoelectric plethysmography in peripheral vascular disease. *American Heart Journal 20* (1940).

[39] HEUSCH, G., ANJOS, A., AND MARCEL, S. A reproducible study on remote heart rate measurement, 2017.

[40] HOCHREITER, S., AND SCHMIDHUBER, J. Long short-term memory. *Neural Computation 9*, 8 (1997), 1735–1780.

[41] HOWARD, A. G., ZHU, M., CHEN, B., KALENICHENKO, D., WANG, W., WEYAND, T., ANDREETTO, M., AND ADAM, H. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *CoRR abs/1704.04861* (2017).

[42] HUANG, G. B., RAMESH, M., BERG, T., AND LEARNED-MILLER, E. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Tech. Rep. 07-49, University of Massachusetts, Amherst, October 2007.

[43] HYNDMAN, B., KITNEY, R., AND SAYERS, B. Spontaneous rhythms in physiological control systems. *Nature 233* (11 1971), 339–41.

[44] IIZUKA, S., SIMO-SERRA, E., AND ISHIKAWA, H. Let there be Color!: Joint End-to-end Learning of Global and Local Image Priors for Automatic Image Colorization with Simultaneous Classification. *ACM Transactions on Graphics (Proc. of SIGGRAPH 2016) 35*, 4 (2016).

[45] IIZUKA, S., SIMO-SERRA, E., AND ISHIKAWA, H. Let there be color! joint end-to-end learning of global and local image priors for automatic image colorization with simultaneous classification. *ACM Trans. Graph. 35*, 4 (July 2016).

[46] IOZZIA, L., CERINA, L., AND MAINARDI, L. Relationships between heart-rate variability and pulse-rate variability obtained from video-PPG signal using ZCA. *Physiological Measurement 37*, 11 (2016), 1934–1944.

[47] JAKUBOVITZ, D., GIRYES, R., AND RODRIGUES, M. R. D. Generalization error in deep learning. *ArXiv abs/1808.01174* (2018).

[48] JESORSKY, O., KIRCHBERG, K. J., AND FRISCHHOLZ, R. Robust face detection using the hausdorff distance. In *Proceedings of the Third International Conference on Audio- and Video-Based Biometric Person Authentication* (Berlin, Heidelberg, 2001), AVBPA '01, Springer-Verlag, pp. 90–95.

[49] JI, S., XU, W., YANG, M., AND YU, K. 3d convolutional neural networks for human action recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence 35*, 1 (2013), 221–231.

[50] KAKUMANU, P., MAKROGIANNIS, S., AND BOURBAKIS, N. A survey of skin-color modeling and detection methods. *Pattern Recogn. 40*, 3 (Mar. 2007), 1106–1122.

[51] KAMSHILIN, A. A., AND MARGARYANTS, N. B. Origin of photoplethysmographic waveform at green light. *Physics Procedia 86* (2017), 72 – 80. International Conference on Photonics of Nano- and Bio-Structures, PNBS-2015, 19-20 June 2015, Vladivostok, Russia and the International Conference on Photonics of Nano- and Micro-Structures, PNMS-2015, 7-11 September 2015, Tomsk, Russia.

[52] KAMSHILIN, A. A., NIPPOLAINEN, E., SIDOROV, I. S., VASILEV, P. V., EROFEEV, N. P., PODOLIAN, N. P., AND ROMASHKO, R. V. A new look at the essence of the imaging photoplethysmography. *Scientific Reports 5* (2015).

[53] KASINSKI, A., FLOREK, A., AND SCHMIDT, A. The put face database. *Image Processing and Communications 13* (01 2008), 59–64.

[54] KAWULOK, M., KAWULOK, J., AND NALEPA, J. Spatial-based skin detection using discriminative skin-presence features. *Pattern Recogn. Lett. 41*, C (May 2014), 3–13.

[55] KC, K., YIN, Z., WU, M., AND WU, Z. Depthwise separable convolution architectures for plant disease classification. *Computers and Electronics in Agriculture 165* (2019), 104948.

[56] KINGMA, D. P., AND BA, J. Adam: A method for stochastic optimization, 2014. cite arxiv:1412.6980Comment: Published as a conference paper at the 3rd International Conference for Learning Representations, San Diego, 2015.

[57] KOPELIOVICH, M., MIRONENKO, Y., AND PETRUSHAN, M. Architectural tricks for deep learning in remote photoplethysmography. In *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)* (Oct. 2019), IEEE.

[58] KRANJEC, J., BEGUŠ, S., GERŠAK, G., AND DRNOVŠEK, J. Non-contact heart rate and heart rate variability measurements: A review. *Biomedical Signal Processing and Control 13* (2014).

[59] KRIZHEVSKY, A. Learning multiple layers of features from tiny images. Tech. rep., 2009.

[60] KRIZHEVSKY, A., SUTSKEVER, I., AND HINTON, G. E. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems 25*, F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, Eds. Curran Associates, Inc., 2012, pp. 1097–1105.

[61] LE, V., BRANDT, J., LIN, Z., BOURDEV, L., AND HUANG, T. S. Interactive facial feature localization. In *Computer Vision – ECCV 2012* (Berlin, Heidelberg, 2012), A. Fitzgibbon, S. Lazebnik, P. Perona, Y. Sato, and C. Schmid, Eds., Springer Berlin Heidelberg, pp. 679–692.

[62] LECUN, Y., BOTTOU, L., BENGIO, Y., AND HAFFNER, P. Gradient-based learning applied to document recognition. In *Proceedings of the IEEE* (1998), pp. 2278–2324.

[63] LIN, Y.-P., AND JUNG, T.-P. Improving eeg-based emotion classification using conditional transfer learning. *Frontiers in Human Neuroscience 11* (2017).

[64] LIU, S., HUANG, D., AND WANG, Y. Receptive field block net for accurate and fast object detection. vol. 11215 LNCS.

[65] LIU, S., SHI, J., LIANG, J., AND YANG, M.-H. Face parsing via recurrent propagation. *CoRR abs/1708.01936* (2017).

[66] LUSSANA, R., VILLA, F., MORA, A. D., CONTINI, D., TOSI, A., AND ZAPPA, F. Enhanced single-photon time-of-flight 3d ranging. *Opt. Express 23*, 19 (Sep 2015), 24962–24973.

[67] MALLIANI, A., PAGANI, M., LOMBARDI, F., AND CERUTTI, S. Research Advances Series Cardiovascular Neural Regulation Explored in the Frequency Domain. *CV Neural Regulation in Frequency Domain* (1991), 482–492.

[68] MARCHEGIANI, L., AND NEWMAN, P. Listening for sirens: Locating and classifying acoustic alarms in city scenes. *ArXiv abs/1810.04989* (2018).

[69] MCCULLOCH, W., AND PITTS, W. A logical calculus of ideas immanent in nervous activity. *Bulletin of Mathematical Biophysics 5* (1943), 127–147.

[70] MCDUFF, D., AND BLACKFORD, E. iphys: An open non-contact imaging-based physiological measurement toolbox, 2019.

[71] MCDUFF, D., ESTEPP, J. R., PIASECKI, A. M., AND BLACKFORD, E. B. A survey of remote optical photoplethysmographic imaging methods. *2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)* (2015), 6398–6404.

[72] MCDUFF, D. J., BLACKFORD, E. B., AND ESTEPP, J. R. Fusing partial camera signals for noncontact pulse rate variability measurement. *IEEE Transactions on Biomedical Engineering 65*, 8 (2018), 1725–1739.

[73] MENABREA, L. F. Notions sur la machine analytique de M. Charles Babbage. *Bibliothèque Universelle de Genève 41* (1842–1843), 352–376.

[74] MESLEH, A., SKOPIN, D., BAGLIKOV, S., AND QUTEISHAT, A. Heart rate extraction from vowel speech signals. *Journal of Computer Science and Technology 27* (2012).

[75] MESSER, K., MATAS, J., KITTLER, J., AND JONSSON, K. Xm2vtsdb: The extended m2vts database. In *In Second International Conference on Audio and Video-based Biometric Person Authentication* (1999), pp. 72–77.

[76] MICHALET, X., INGARGIOLA, A., COLYER, R. A., SCALIA, G., WEISS, S., MACCAGNANI, P., GULINATTI, A., RECH, I., AND GHIONI, M. Silicon photon-counting avalanche diodes for single-molecule fluorescence spectroscopy. *IEEE Journal of Selected Topics in Quantum Electronics 20*, 6 (Nov 2014), 248–267.

[77] MILBORROW, S., MORKEL, J., AND NICOLLS, F. The MUCT Landmarked Face Database. *Pattern Recognition Association of South Africa* (2010).

[78] MOCO, A. V., STUIJK, S., AND DE HAAN, G. Skin inhomogeneity as a source of error in remote ppg-imaging. *Biomedical Optics Express 7* (2016).

[79] MOCO, A. V., STUIJK, S., AND HAAN, G. D. Ballistocardiographic artifacts in ppg imaging. *IEEE Transactions on Biomedical Engineering 63* (2016).

[80] MORAES, J. L., ROCHA, M. X., VASCONCELOS, G. G., VASCONCELOS FILHO, J. E., DE ALBUQUERQUE, V. H. C., AND ALEXANDRIA, A. R. Advances in photopletysmography signal analysis for biomedical applications. *Sensors 18*, 6 (2018).

[81] MORBIDUCCI, U., SCALISE, L., MELIS, M. D., AND GRIGIONI, M. Optical vibrocardiography: A novel tool for the optical monitoring of cardiac activity. *Annals of Biomedical Engineering 35* (2007).

[82] MORENO, J., RAMOS-CASTRO, J., MOVELLAN, J., PARRADO, E., RODAS, G., AND CAPDEVILA, L. Facial video-based photoplethysmography to detect HRV at rest. *International Journal of Sports Medicine 36*, 6 (2015), 474–480.

[83] NIRKIN, Y., MASI, I., TRAN, A. T., HASSNER, T., AND MEDIONI, G. On face segmentation, face swapping, and face perception, 2017.

[84] PAN, J., AND TOMPKINS, W. A Real-Time QRS Detection Algorithm. *IEEE Transactions on Biomedical Engineering BME-32*, 3 (1985), 230–236.

[85] PAN, S. J., AND YANG, Q. A survey on transfer learning. *IEEE Trans. on Knowl. and Data Eng. 22*, 10 (Oct. 2010), 1345–1359.

[86] POH, M., MCDUFF, D. J., AND PICARD, R. W. Non-contact, automated cardiac pulse measurements using video imaging and blind source separation. *Optics Express 18*, 10 (2010), 10762.

[87] PRANCE, R. J., BEARDSMORE-RUST, S. T., WATSON, P., HARLAND, C. J., AND PRANCE, H. Remote detection of human electrophysiological signals using electric potential sensors. *Applied Physics Letters 93* (2008).

[88] RAUBER, P. E., FADEL, S. G., FALCÃO, A. X., AND TELEA, A. C. Visualizing the hidden activity of artificial neural networks. *IEEE Transactions on Visualization and Computer Graphics 23*, 1 (2017), 101–110.

[89] RHEE, S., YANG, B. H., AND ASADA, H. H. Artifact-resistant power-efficient design of finger-ring plethysmographic sensors. *IEEE Transactions on Biomedical Engineering 48* (2001).

[90] ROBERTSON, A. R. The cie 1976 color-difference formulae. *Color Research & Application 2*, 1 (1976), 7–11.

[91] RONNEBERGER, O., FISCHER, P., AND BROX, T. *U-Net: Convolutional Networks for Biomedical Image Segmentation*. Springer International Publishing, Cham, 2015, pp. 234–241.

[92] ROSENBLATT, F. *Principles of Neurodynamics*. Spartan, New York, 1962.

[93] ROUAST, P., ADAM, M., CHIONG, R., CORNFORTH, D., AND LUX, E. Remote heart rate measurement using low-cost RGB face video: a technical literature review. *Frontiers of Computer Science*, August (2017), 1–15.

[94] ROUAST, P. V., ADAM, M. P., DORNER, V., AND LUX, E. Remote photoplethysmography: Evaluation of contactless heart rate measurement in an information systems setting. *Applied Informatics and Technology Innovation Conference* (2016), 1–17.

[95] SABOKROU, M., POURREZA, M., LI, X., FATHY, M., AND ZHAO, G. Deep-hr: Fast heart rate estimation from face video under realistic conditions, 2020.

[96] SARKAR, A., ABBOTT, A. L., AND DOERZAPH, Z. Universal skin detection without color information. In *2017 IEEE Winter Conference on Applications of Computer Vision (WACV)* (March 2017), pp. 20–28.

[97] SEITZ, P., AND THEUWISSEN, A. J. P. *Single-Photon Imaging (Springer Series in Optical Sciences)*. Springer, 2013.

[98] SHAFFER, F., AND GINSBERG, J. P. An overview of heart rate variability metrics and norms. *Frontiers in Public Health 5* (2017), 258.

[99] SHI, J., AND TOMASI, C. Good features to track. pp. 593–600.

[100] SHUSTERMAN, V., ANDERSON, K., AND BARNEA, O. Spontaneous skin temperature oscillations in normal human subjects. *The American journal of physiology 273* (10 1997), R1173–81.

[101] SIDOROV, I. S., ROMASHKO, R. V., KOVAL, V. T., GINIATULLIN, R., AND KAMSHILIN, A. A. Origin of infrared light modulation in reflectance-mode photoplethysmography. *PLoS ONE 11* (2016).

[102] SIMONYAN, K., AND ZISSERMAN, A. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations* (2015).

[103] SINHAL, R., SINGH, K., AND RAGHUWANSHI, M. M. An overview of remote photoplethysmography methods for vital sign monitoring. In *Computer Vision and Machine Intelligence in Medical Image Analysis* (Singapore, 2020), M. Gupta, D. Konar, S. Bhattacharyya, and S. Biswas, Eds., Springer Singapore, pp. 21–31.

[104] SMITH, B. M., ZHANG, L., BRANDT, J., LIN, Z., AND YANG, J. Exemplar-based face parsing. In *CVPR* (2013), IEEE Computer Society, pp. 3484–3491.

[105] SOLEYMANI, M., LICHTENAUER, J., PUN, T., AND PANTIC, M. A multimodal database for affect recognition and implicit tagging. *IEEE Transactions on Affective Computing 3*, 1 (2012), 42–55.

[106] STEINWART, I., AND CHRISTMANN, A. *Support Vector Machines*, 1st ed. Springer Publishing Company, Incorporated, 2008.

[107] SUN, Y., AZORIN-PERIS, V., KALAWSKY, R., HU, S., PAPIN, C., AND GREENWALD, S. E. Use of ambient light in remote photoplethysmographic systems: comparison between a high-performance camera and a low-cost webcam. *Journal of Biomedical Optics 17* (2012), 17 – 17 – 11.

[108] SUN, Y., AND THAKOR, N. Photoplethysmography revisited: From contact to noncontact, from point to imaging. *IEEE transactions on bio-medical engineering 63* (09 2015).

[109] TAMURA, T., MAEDA, Y., SEKINE, M., AND YOSHIDA, M. Wearable photoplethysmographic sensors—past and present. *Electronics 3*, 2 (2014), 282–302.

[110] TAN, W. R., CHAN, C. S., PRATHEEPAN, Y., AND CONDELL, J. A fusion approach for efficient human skin detection. *CoRR abs/1410.3751* (2014).

[111] TASLI, E., GUDI, A., AND UYL, M. Remote PPG based vital sign measurement usign adaptive facial regions Vicarious Perception Technologies Intelligent Systems Lab Amsterdam, University of Amsterdam, The Netherlands. *International Conference on Image Processing(ICIP)* (2014), 1410–1414.

[112] TULYAKOV, S., ALAMEDA-PINEDA, X., RICCI, E., YIN, L., COHN, J. F., AND SEBE, N. Self-Adaptive Matrix Completion for Heart Rate Estimation from Face Videos under Realistic Conditions. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2016), 2396–2404.

[113] VARANINI, M., BERARDI, P. C., CONFORTI, F., MICALIZZI, M., NEGLIA, D., AND MACERATA, A. Cardiac and respiratory monitoring through non-invasive and contactless radar technique. vol. 35.

[114] VERKRUYSSE, W., SVAASAND, L., AND NELSON, J. S. Remote plethysmographic imaging using ambient light. *Optics Express 16*, 26 (2008), 63–86.

[115] VIEIRA MOCO, A. *Towards photoplethysmographic imaging: modeling, experiments and applications*. PhD thesis, Department of Electrical Engineering, May 2019. Proefschrift.

[116] VILLA, F., LUSSANA, R., BRONZI, D., ZAPPA, F., AND GIUDICE, A. 3d spad camera for advanced driver assistance. In *2017 International Conference of Electrical and Electronic Technologies for Automotive* (2017), pp. 1–5.

[117] VIOLA, P., AND JONES, M. J. Robust real-time face detection. *Int. J. Comput. Vision 57*, 2 (May 2004), 137–154.

[118] WANG, Y., SKERRY-RYAN, R. J., STANTON, D., WU, Y., WEISS, R. J., JAITLY, N., YANG, Z., XIAO, Y., CHEN, Z., BENGIO, S., LE, Q. V., AGIOMYRGIANNAKIS, Y., CLARK, R., AND SAUROUS, R. A. Tacotron: A fully end-to-end text-to-speech synthesis model. *CoRR abs/1703.10135* (2017).

[119] YURTSEVER, E., LAMBERT, J., CARBALLO, A., AND TAKEDA, K. A survey of autonomous driving: Common practices and emerging technologies. *IEEE Access 8* (2020), 58443–58469.

[120] YVES BOUGUET, J. Pyramidal implementation of the lucas kanade feature tracker. *Intel Corporation, Microprocessor Research Labs* (2000).

[121] ZAPPA, F., TISA, S., TOSI, A., AND COVA, S. Principles and features of single-photon avalanche diode arrays. *Sensors and Actuators A: Physical 140*, 1 (2007), 103 – 112.

[122] ZAUNSEDER, S., TRUMPP, A., WEDEKIND, D., AND MALBERG, H. Cardiovascular assessment by imaging photoplethysmography – a review. *Biomedical Engineering / Biomedizinische Technik 63*, 5 (Oct. 2018), 617–634.

[123] ZHANG, G., LIU, C., JI, L., YANG, J., AND LIU, C. Effect of a percutaneous coronary intervention procedure on heart rate variability and pulse transit time variability: A comparison study based on fuzzy measure entropy. *Entropy 18* (2016), 246.

[124] ZHOU, F., BRANDT, J., AND LIN, Z. Exemplar-based graph matching for robust facial landmark localization. In *IEEE International Conference on Computer Vision (ICCV)* (2013).

[125] ZHOU, L., LIU, Z., AND HE, X. Face parsing via a fully-convolutional continuous CRF neural network. *CoRR abs/1708.03736* (2017).

[126] ZIJLSTRA, W. G., AND BUURSMA, A. Spectrophotometry of hemoglobin: Absorption spectra of bovine oxyhemoglobin, deoxyhemoglobin, carboxyhemoglobin, and methemoglobin. *Comparative Biochemistry and Physiology - B Biochemistry and Molecular Biology 118*, 4 (1997), 743–749.

List of publications

[127] BONETTINI, N., PARACCHINI, M., BESTAGINI, P., MARCON, M., AND TUBARO, S. Hyperspectral x-ray denoising: Model-based and data-driven solutions. *2019 27th European Signal Processing Conference (EUSIPCO)* (2019), 1–5.

[128] BROKALAKIS, A., TAMPOURATZIS, N., NIKITAKIS, A., PAPAEFSTATHIOU, I., ANDRIANAKIS, S., DOLLAS, A., PARACCHINI, M., MARCON, M., PAU, D. P., AND PLEBANI, E. An open-source extendable, highly-accurate and security aware simulator for cloud applications. In *2018 21st Conference on Innovation in Clouds, Internet and Networks and Workshops (ICIN)* (2018), pp. 1–3.

[129] BROKALAKIS, A., TAMPOURATZIS, N., NIKITAKIS, A., PAPAEFSTATHIOU, I., ANDRIANAKIS, S., PAU, D., PLEBANI, E., PARACCHINI, M., MARCON, M., SOURDIS, I., GEETHAKUMARI, P. R., PALACIOS, M. C., ANTON, M. A., AND SZASZ, A. Cossim: An open-source integrated solution to address the simulator gap for systems of systems. In *2018 21st Euromicro Conference on Digital System Design (DSD)* (2018), pp. 115–120.

[130] PARACCHINI, M., MARCHESI, L., PASQUINELLI, K., MARCON, M., FONTANA, G., GABRIELLI, A., AND VILLA, F. Remote photoplethysmography using spad camera for automotive health monitoring application. In *2019 AEIT International Conference of Electrical and Electronic Technologies for Automotive (AEIT AUTOMOTIVE)* (July 2019), pp. 1–6.

[131] PARACCHINI, M., MARCON, M., AND TUBARO, S. Fast and reliable facial landmarks localization in non frontal images. In *2019 8th European Workshop on Visual Information Processing (EUVIP)* (2019), pp. 88–92.

[132] PARACCHINI, M., MARCON, M., VILLA, F., AND TUBARO, S. Deep skin detection on low resolution grayscale images. *Pattern Recognition Letters 131* (2020), 322 – 328.

[133] PARACCHINI, M., MARCON, M., VILLA, F., ZAPPA, F., AND TUBARO, S. Biometric signals estimation using single photon camera and deep learning. *Sensors 20*, 21 (2020).

[134] PARACCHINI, M., PLEBANI, E., ICHE, M. B., PAU, D. P., AND MARCON, M. Embedded real-time visual search with visual distance estimation. In *Image Analysis and Processing - ICIAP 2017* (Cham, 2017), S. Battiato, G. Gallo, R. Schettini, and F. Stanco, Eds., Springer International Publishing, pp. 59–69.