ABSTRACT

Title of dissertation: ROBUST AND ANALYTICAL CARDIOVASCULAR SENSING

Qiang Zhu
Doctor of Philosophy, 2020

Dissertation directed by: Professor Min Wu
Department of Electrical and Computer Engineering

The photoplethysmogram (PPG) is a noninvasive cardiovascular signal related to the pulsatile volume of blood in tissue. The PPG is user-friendly and has the potential to be measured remotely in a contactless manner using a regular RGB camera. In this dissertation, we study the modeling and analytics of PPG signal to facilitate its applications in both robust and remote cardiovascular sensing.

In the first part of this dissertation, we study the remote photoplethysmography (rPPG) and present a robust and efficient rPPG system to extract pulse rate (PR) and pulse rate variability (PRV) from face videos. Compared with prior art, our proposed system can achieve accurate PR and PRV estimates even when the video contains significant subject motion and environmental illumination change.

In the second part of the dissertation, we present a novel frequency tracking algorithm called Adaptive Multi-Trace Carving (AMTC) to address the micro signal extraction problems. AMTC enables an accurate detection and estimation of one or more subtle frequency components in a very low signal-to-noise ratio condition.

In the third part of the dissertation, the relation between electrocardiogram (ECG) and PPG is studied and the waveform of ECG is inferred via the PPG signals. In order to address this cardiovascular inverse problem, a transform is proposed to map the discrete cosine transform coefficients of each PPG cycle to those of the corresponding ECG cycle. As the first work to address this biomedical inverse problem, this line of research enables a full utilization of the easy accessibility of PPG and the clinical authority of ECG for better preventive healthcare.

ROBUST AND ANALYTICAL CARDIOVASCULAR SENSING

by

Qiang Zhu

Dissertation submitted to the Faculty of the Graduate School of the
University of Maryland, College Park in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
2020

Advisory Committee:
Professor Min Wu, Chair/Advisor
Professor K. J. Ray Liu
Professor Behtash Babadi
Professor Chau-Wai Wong
Professor Yang Tao, Dean's Representative

To my parents and Chenqing

Acknowledgments

First and foremost I would like to express my sincere gratitude to my advisor, Professor Min Wu, for giving me an invaluable opportunity to work on challenging and interesting projects over the past four years. It is her critical thinking and patient guidance in every aspect and step of my research that lead to my achievements in my Ph.D. career. I learned from her that every detail matters. I learned from her the beauty of writing. I learned from her the importance of organizing the knowledge and research ideas. I also learned form her the importance and the skills to effectively deliver the research ideas.

I would like to thank all dissertation committee members, Prof. K. J. Ray Liu, Prof. Behtash Babadi, Prof. Chau-Wai Wong, and Prof. Yang Tao, for their time and invaluable feedback. I would also like to thank Prof. Liu and Prof. Babadi for their excellent courses which help lay the foundations for my subsequent research at UMD.

I would like to thank all the group members from the MAST family. I thank Dr. Wong, Dr. Adi Hajj-Ahmad and Dr. Abbas Kazemipour for their help in the initial phase of my research. I would like to thank Mingliang Chen, Xin Tian, Yiqi Li, Prof. Chang-Hong Fu, and Prof. Yuenan Li for their ideas shared through our discussions and group meetings. The collaboration with everyone is pleasant.

Last but not the least, I would like to thank my parents for their unconditional love and support. I thank my wife, Chenqing, who is always on my side, cheering for me and encouraging me to aim higher and take action. I am so lucky and grateful

to have you as my life-long partner and experience all the sweetness and bitterness of life. I dedicate this dissertation to them.

Table of Contents

List of Tables

List of Figures

Chapter 1: Introduction

1.1 Background

1.1.1 Photoplethysmography and Remote Photoplethysmography

Photoplethysmography (PPG) is a simple, low-cost, and non-invasive optical technology that detects blood volume changes in the microvascular bed of skin tissue [1,2]. PPG is nearly ubiquitous in clinics and hospitals in the form of finger/toe clips and oximeters and has increasing popularity in the form of consumer-grade wearable devices that offer continuous and long-term monitoring capability. The principle of the PPG is based on the fact that the blood has different optical behaviors compared with other skin tissues [3]. The circulation of the blood leads to the variations of the number of hemoglobin molecules, which causes the variations of the optical absorption across the light spectrum. As shown in Fig. 1.1, the PPG measurement requires a light source (e.g., Light Emitting Diode (LED)) to illuminate the skin and a phototransistor to receive the light propagated through the skin tissue. Based on the geometric location of the LED and the phototransistor, the working mechanism of PPG can be classified into two modes: transmissive mode or reflectance mode.

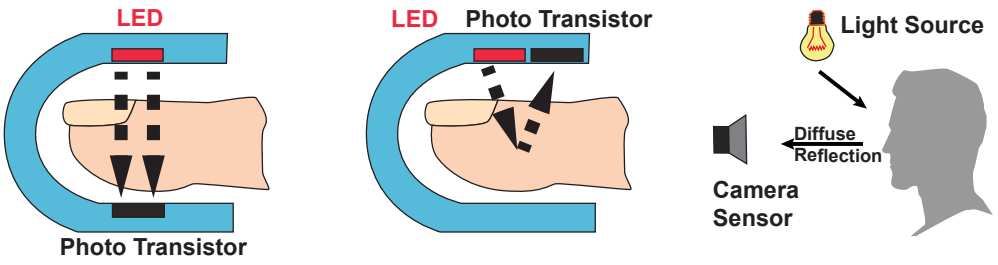


Figure 1.1: Three different operational modes of PPG. The difference between the setup of the transmissive mode (a) and the reflective mode (b) is the relative placement of the light sensor and the LED. The difference between the first two modes with the remote PPG is the distance of the sensor and the light source to the subject.

The contactless mode of PPG, also known as remote photoplethysmography (rPPG) is first introduced in [4], where multiple cardiovascular parameters are successfully extracted from face videos with ambient light. By allowing a certain distance between the sensor and skin surface, rPPG has become a more favorable pulse monitoring solution compared with the conventional contact-based method, especially for users with special needs or for certain application scenarios, such as liveness detection in rescue tasks. In the past decade, much progress has been made in the rPPG research community in terms of optical system modeling, robust signal processing, and system adaption for various applications scenarios. These progress has enabled the measurement of multiple physiological parameters from rPPG, such as pulse rate (PR) [4, 5], pulse-rate-variability (PRV) [6], respiration rate (RR) [7], SpO2 [8], and blood pressure [9], in highly challenging scenarios, such as long-distance between the sensor and subject. Applications of the rPPG include and are not limited to fitness monitoring [10], atrial fibrillation detection [11], and face anti-spoofing [12].

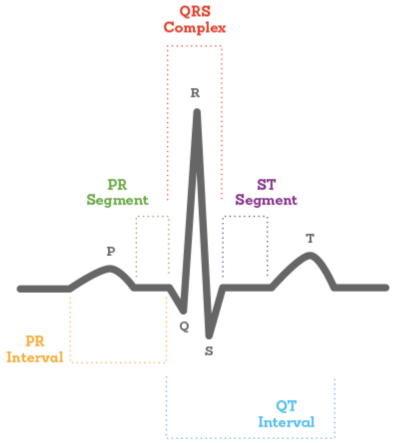


Figure 1.2: A typical ECG waveform of a heart in normal sinus rhythm. Retrieved from Wikipedia page of 'Electrocardiography' [14].

1.1.2 The ECG and PPG Waveforms

The ECG waveform: The ECG records the potential difference between prescribed sites on the body surface that varies during the cardiac cycle. It reflects differences in transmembrane voltages in myocardial cells that occur during depolarization and repolarization within each cardiac cycle [13].

Fig. 1.2 shows a typical ECG waveform of a heart in normal sinus rhythm. The cardiac electrical signal is initiated in the sinoatrial (SA) node located in the right atrium and travels to the left and right atria, causing the atria contraction and blood pumping into the ventricles. Such an atria depolarization process is represented as the P wave on the ECG cycle.

The electrical signal then passes from the atria to the ventricles through the atrioventricular (AV) node. The electrical signal slows down once it passes the AV node, which allows the blood to fill the ventricles. This process is recorded as the PR segment, which usually appears as a flat line on the ECG between the end of

the P wave and the starting point of the Q wave. The PR segment represents the electrical conduction through the atria and the delay of the electrical impulse in the atrioventricular node.

After the signal leaves the AV node, it travels along with the bundle of His and into the right and left bundle branches. The signal then travels across the heart's ventricles, causing them to contract, pumping blood to the lungs and the body. This signal is recorded as the QRS waves on the ECG.

The ventricles then recover to their normal electrical state, shown as the T wave. The muscles relax and stop contracting, allowing the atria to fill with blood, and the entire process repeats with each heartbeat. The ST segment connects the QRS complex and the T wave and represents the beginning of the electrical recovery of the ventricles. The QT interval represents the time during which the ventricles are stimulated and recover after the stimulation.

The PPG waveform: Fig. 1.3 shows a typical PPG waveform and its correspondence with the variations in light attenuation by pulsatile components in skin tissue. During the systole phase of the cardiac cycle, the oxygenated blood is pumped to the body from the left ventricle of the heart. This process causes the increase of the blood volume and oxyhemoglobin that reach to the capillaries in the skin surface. The variations of the amount of oxyhemoglobin and related protein in the blood result in the corresponding variation of light absorption and similarly oscillation of the received light by the PPG sensor in each cardiac cycle.

The PPG signal can be decomposed into 'AC' component and 'quasi-DC' component. The 'AC' component is related to the pulsatile component, and its funda-

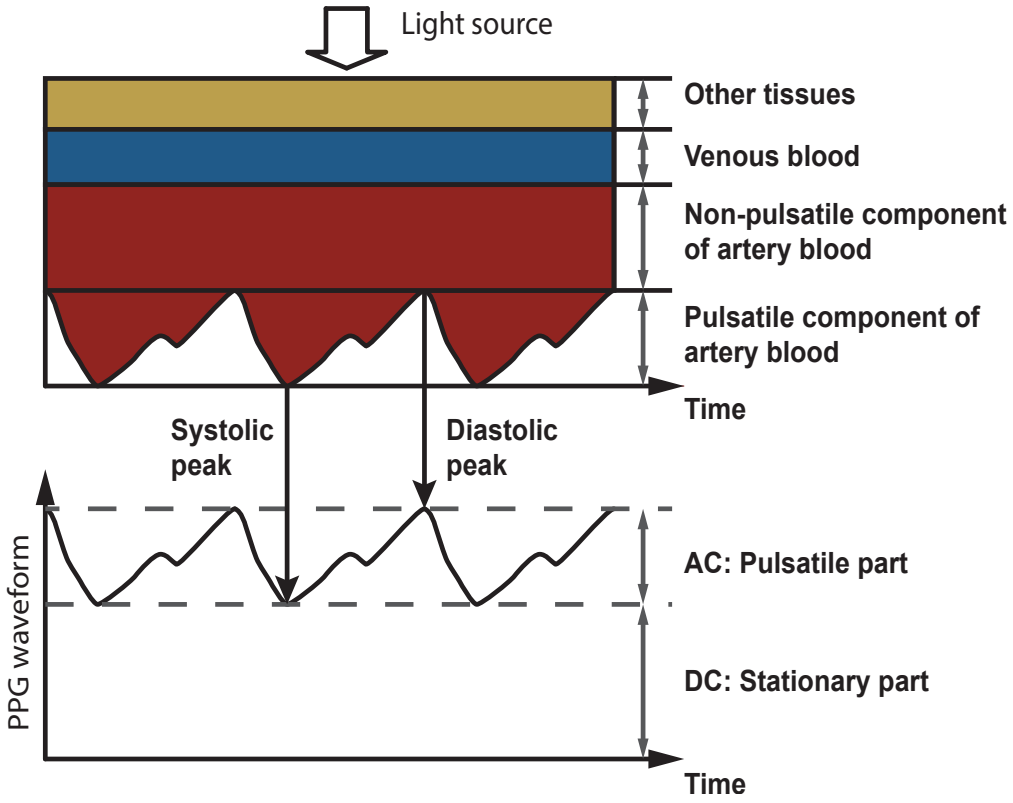


Figure 1.3: Variation in light attenuation by tissue and the corresponding PPG measurement (modified from [15]).

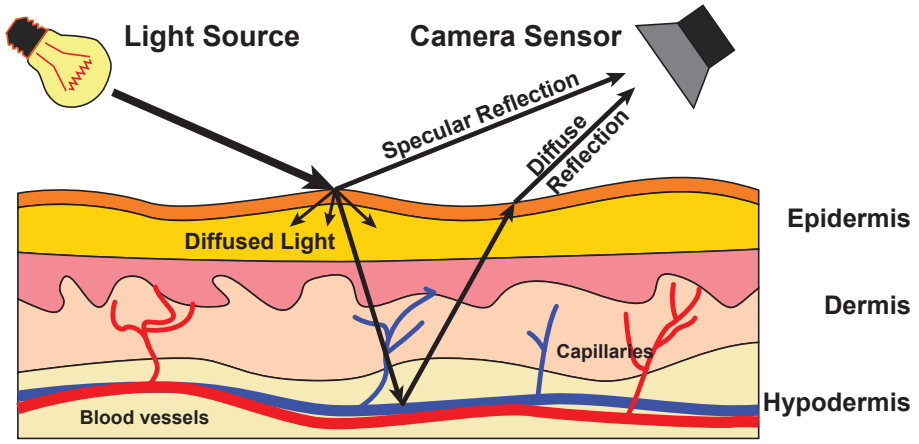


Figure 1.4: Anatomical cross-section structure of human skin tissues and the specular and diffuse reflections captured by a RGB sensor when the skin is illuminated by a light source (modified based on [16]).

mental frequency reflects the heart rate. The AC component is synchronized with each cardiac cycle, and it provides valuable information about the cardiovascular system. The quasi-DC component, which superimposes onto the AC component, is influenced by the respiration, vasomotor activity, and vasoconstrictor waves, Traube Hering Mayer (THM) waves, and also thermoregulation [2].

1.1.3 Skin Reflection Model

We discuss the face skin reflection model that is adopted in this thesis to facilitate the investigation of the rPPG system design. The discussion of the skin model allows us to analyze the problem in detail and offer insights on how the problems can be formulated and addressed.

Consider the situation when a piece of human skin containing pulsatile blood is illuminated by a light source as shown in Fig. 1.4. The reflected light from the skin surface can be characterized as the specular and diffuse reflections [1].

[1]Some literature [17] adopts the terms *interface* and *body* reflections rather than the *specular*

Approximately 4% to 7% of visible light is reflected from the stratum corneum in the epidermis layer [18, 19] as the specular reflection. The reflectance geometry among skin surface, light source, and the camera sensor determines the radiance of the specular reflection [17]. The spectral distribution of the measured specular reflection component in a camera is a function of the spectral distribution of the light source and the spectral response of the camera. Thus for a single light source with fixed spectral distribution, only the strength of the specular reflection will be modulated by the subject motion.

The diffuse reflection can be further decomposed into the epidermal reflection and dermal reflection [18]. The spectral distribution of the epidermal reflection is mostly determined by the concentration of the melanin in the epidermis layer. The dermal reflection, on the other hand, carries the blood pulse information. The variations of the blood volume, especially the amount of oxygenated and deoxygenated hemoglobin in the dermis layer, influence the color and intensity of the dermal reflection. Note that the dermal reflection also contains a part of the reflection, which exhibits similar spectral activity as the epidermal reflection because the light needs to pass through the epidermis layer so that it can reach the dermis layer.

Two assumptions about the skin reflection are made in this thesis:

1. Diffuse reflection from the skin surface is isotropic with respect to rotation about the surface normal;

2. No inter-reflection exists among surface, as we approximately treat head as a

and *diffuse* reflections. To avoid confusion and maintain the consistency of the terminology used in the rPPG community, we use the terms *specular* and *diffuse* reflections in this paper.

convex shape.

Based on the analysis and assumptions above, we arrive at the reflection formulation based on the skin characteristics and the Dichromatic Reflection Model (DRM) [16, 17, 20]:

$$\mathbf{C}_k(t) = I(t) \cdot (\mathbf{v_s}(t) + \mathbf{v_d}(t)) + \mathbf{v_n}(t), \tag{1.1}$$

where $\mathbf{C}_k(t)$ indicates the RGB channels of the kth skin pixel; $I(t)$ denotes the radiance of the light source arrived at the corresponding skin surface, which is a function of the distance between the light source and the surface; $\mathbf{v_s}(t)$ and $\mathbf{v_d}(t)$ denote the specular and diffuse reflection respectively; $\mathbf{v_n}(t)$ denotes camera's sensor noise and the video or image compression noise.[2]

Specifically, $\mathbf{v_s}(t)$ and $\mathbf{v_d}(t)$ can be decomposed as:

$$\begin{aligned} \mathbf{v_s}(t) &= \mathbf{u_s}(s_0 + s(t)), \\ \mathbf{v_d}(t) &= \mathbf{u_d} \cdot d_0 + \mathbf{u_p} \cdot p(t), \end{aligned} \tag{1.2}$$

where $\mathbf{u_s}$, $\mathbf{u_d}$, and $\mathbf{u_p}$ denote the unit color vector of the light spectrum, the skin tissue, and the pulse, respectively; s_0 and d_0 denote the strength of the DC part of the specular and diffuse refection respectively; $s(t)$ and $p(t)$ denote the strength of the AC part of the specular reflection and pulse signal respectively. Note that the variations of both $I(t)$ and $s(t)$ come from subject motion. The difference is that the variation of $I(t)$ comes from the distance of the light source to the skin surface, while

[2]For the completeness of this thesis, we briefly reiterate the modeling process, which has been discussed and detailed in [16, 20]. The terminology used in those two papers are propagated in this thesis paper for consistency considerations.

$s(t)$ comes from the variation of the surface normal direction. Let $I(t) \triangleq I_0(1+i(t))$, $\mathbf{u_c} \cdot c_0 \triangleq \mathbf{u_s} \cdot s_0 + \mathbf{u_d} \cdot d_0$, where $i(t)$ indicates the illumination change. We denote $\mathbf{C}(t)$ as the averaged RGB values over a sufficiently large number of the skin pixels. We assume that the pulse arrival time is identical for every face skin cite, and that $\mathbf{v_n}(t)$ exhibits zero mean white Gaussian distribution. Thus Eq. 1.1 can be rewritten as:

$$\begin{aligned} \mathbf{C}(t) \approx & I_0\left(1+i(t)\right) \cdot \left(\mathbf{u_c} \cdot c_0 + \mathbf{u_s} \cdot s(t) + \mathbf{u_p} \cdot p(t)\right) \\ \approx & \mathbf{u_c} \cdot I_0 \cdot c_0 + \mathbf{u_c} \cdot I_0 \cdot c_0 \cdot i(t) + \mathbf{u_s} \cdot I_0 \cdot s(t) + \mathbf{u_p} \cdot I_0 \cdot p(t), \end{aligned} \tag{1.3}$$

where the spatial averaging operation alleviates the white Gaussian noise $\mathbf{v_n}(t)$ in the first step, and cross products of AC-terms are neglected as the magnitude of AC-terms are considered as much smaller than the DC-terms.

As pointed in [20], the limitation of the model (1.3) lies at the assumption of a single light source and that the subject's motion creates a single specular variation direction in the RGB space. This is, unfortunately, unrealistic even when one single illumination source exists in the environment. This is because the skin surface might receive reflected light from other subjects with non-uniform light spectrum absorbance in the scene, and the spectrum of such a reflected light differs from that of the light source. If we include this variant in our model and assume that in total J light sources exist in the scene, including the reflected light from other subjects

in the scene, Eq. 1.3 becomes:

$$\mathbf{C}(t) \approx \overbrace{\sum_{j=1}^{J} \mathbf{u_{c,j}} \cdot I_{0,j} \cdot c_{0,j}}^{\textbf{DC}} + \overbrace{\sum_{j=1}^{J} \mathbf{u_{c,j}} \cdot I_{0,j} \cdot c_{0,j} \cdot i_j(t)}^{\textbf{Intensity}} + \overbrace{\sum_{j=1}^{J} \mathbf{u_{s,j}} \cdot I_{0,j} \cdot s_j(t)}^{\textbf{Specular}} + \overbrace{\left(\sum_{j=1}^{J} \mathbf{u_{p,j}} \cdot I_{0,j}\right) \cdot p(t)}^{\textbf{Pulse}}. \tag{1.4}$$

The DC component $\sum_{j=1}^{J} \mathbf{u_{c,j}} \cdot I_{0,j} \cdot c_{0,j}$ can be estimated and subtracted from (1.4) using the short term smoothing approach [16, 21] or the detrending method [22]. Since both $i_j(t)$ and $s_j(t)$ are functions of the subject motion, they can be approximately modeled as different linear combinations of the motion components, i.e., $i_j(t) = \sum_{k=1}^{K} a_{j,k} \cdot m_k(t)$, $_j(t) = \sum_{k=1}^{K} b_{j,k} \cdot m_k(t)$, where $m_k(t)$ denotes the k-th motion component. If we denote $\tilde{\mathbf{C}}(t)$ as the detrended signal after remove the DC component, we finally arrived at

$$\tilde{\mathbf{C}}(t) = \overbrace{\sum_{k=1}^{K} \mathbf{u_{m,k}} \cdot m_k(t)}^{\textbf{Motion}} + \overbrace{\mathbf{u_p}' \cdot p(t)}^{\textbf{Pulse}}, \tag{1.5}$$

where $\mathbf{u_{m,k}}$ and $\mathbf{u_p}'$ represent the strength and color direction of the k-th motion component and the pulse component, respectively. They are defined as $\mathbf{u_{m,k}} \triangleq \sum_{j=1}^{J} (a_{j,k} \cdot I_{0,j} \cdot \mathbf{u_{c,j}} \cdot c_{0,j} + b_{j,k} \cdot I_{0,j} \cdot \mathbf{u_{s,j}})$ and $\mathbf{u_p}' \triangleq \sum_{j=1}^{J} I_{0,j} \cdot \mathbf{u_{p,j}}$. According to (1.5), a linear projection of $\tilde{\mathbf{C}}(t)$ for eliminating the motion component would fail when $\mathbf{u_p}'$ is correlated with $\sum_{k=1}^{K} \mathbf{u_{m,k}} \cdot m_k(t)$. This is unfortunately almost always the case when a subject is performing physical exercises in an uncontrolled

environment, as multiple motion components might enter the pulse color direction.

1.2 Related Works on the Micro Signal Analytics of rPPG

In this thesis, we are generally interested in the micro pulse or cardiac signal extraction problems, where the signals-of-interest *often have smaller magnitudes-typically one order of magnitude or more-than the dominating signals*. In [23], the author has named such a problem as the *micro signal* extraction problem, and has discussed several related applications. We extend the discussion from [23] in the context of rPPG application that is covered in this thesis.

The objective of the rPPG technology is to extract a subject's physiological information, such as pulse rate, respiratory rate, or blood pressure, from a remote optical sensing device in a non-contact manner. This is not a trivial task as the challenges of this sensing mechanism come from each component of the system, namely, the camera, the illumination condition, and the subject. In a fitness setup, the motion-induced intensity and color change on the subject's skin may very well dominate over the reflected light from the facial skin, while the pulse-induced color variation is much subtler. The measurement is also associated with a group of nuisance signals, such as the sensor and quantization noise. To extract the subtle pulse signal that may have a much smaller magnitude than the dominating video components and simultaneously protect it from being corrupted by other nuisance signals, one usually has to tackle the problem with extra caution.

To exemplify the last-decade efforts in addressing this micro-signal problem,

we select 21 prior works and list them in Table 1.1 for an overview and comparison of the experiment setup (subject's motion type, lighting condition, video quality, etc.), claimed best performance, and the color handling method in each work. Note that the works listed and discussed in this paper can in no means cover all spectral of the rPPG technology. We extend our investigation below from the perspectives of *ROI selection* and *Motion robust pulse extraction*.

1.2.1 ROI Selection

ROI selection, aiming to locate the ROI consistently in each video frame in accordance with the subject's motion, is an indispensable first step to obtain reliable rPPG signals. The selection of face skin region as the ROI for pulse measurement is mainly due to the following two facts. First, compared with other parts of the human body, the face is less likely to be covered by other materials, such as clothes. Second, owing to the development of the recent computer vision technologies, a subject's face can be faithfully located and tracked from a video using off-the-shelf tools, even when the background is busy, and the video is compressed and noisy. We summarize the main approaches for ROI selection that are deployed in prior works below.

1. *Manual selection*: when the subject is completely motionless in the video, one may manually select a single ROI from the first frame of the video and extract face color signal using the same region in the subsequent video frames [4, 41]. This may not be a viable solution even when the subject is instructed

	year	Sensor Type Datasets	Subject's Motion	Lighting	Video	Performance	Color Handling	Ref. Sensor
Verkruysse et al. [4]	2008	RGB CCD cam	still	daylight, fluorescent, surgical lamp	15 fps	N/A	G	commercial pressure cuff with HR display
Poh et al. [24]	2011	RGB webcam	still	diffused sunlight	15 fps	1.24 bpm (RMSE)	ICA	PPG (256 Hz)
Sun et al. [25]	2011	monochrome CMOS cam	still, stationary bike	infrared LED	10 bits /channel	N/A	SCICA	PPG sensor
Wu et al. [26]	2012	DC	still	ambient light	N/A	N/A	EVM	ECG sensor
Scully et al. [27]	2012	mobile	still	N/A	25 fps	N/A	G	ECG sensor
Zhao et al. [28]	2013	near-IR cam	still, minor head motion	near-IR LED	8 bits /channel	3.10 bpm (RMSE)	SICA	OmiPlex
de Haan et al. [29]	2013	RGB CCD cam	still, stationary bike, elliptical machine	controlled studio light	8 bits /channel	48% success rate (elliptical machine)	CHROM	PPG sensor
Aarts et al. [30]	2013	RGB cam	still	ambient	8 bits /channel	high match with reference	G - R	ECG sensor
Li et al. [31]	2014	RGB webcam	still, minor head motion	ambient, screen lighting	8 bits /channel	1.53 % (relative error)	G	ECG sensor
Stricker et al. [32]	2014	RGB cam	still, controlled head motion	diffused sun light	30 fps	N/A	G/(R+G+B)	PPG sensor
Huang et al. [33]	2014	mobile phone	still	white LED		N/A		
Tarassenko et al. [34]	2014	High-end RGB cam	still	well-lit ward environment, fluorescent	8 bits /channel 12 fps	3 bpm (MAE)	G	PPG sensor
McDuff et al. [35]	2014	DSLR (RGBCO) cam	still	indoor light sun light	16 bits /channel 30 fps	1.00 (PCC)	ICA	ECG sensor
Feng et al. [36]	2015	webcam	still, head rotation	fluorescent light	8 bits /channel	0.96 (PCC)	G - R	PPG sensor
Wang et al. [16]	2016	RGB CCD cam	still, head rotation, stationary bike, elliptical machine	frontal fluorescent light	8 bits/channel 20 fps	5.16 dB (SNR)	POS	PPG sensor
Zhu et al. [10]	2017	mobile	elliptical machine treadmill	ceiling lighting sunlight	8 bits/channel 30 fps	1.10 bpm (RMSE)	ECG sensor	
Wang et al. [20]	2017	RGB CCD cam	treadmill	ceiling lighting	8 bits/channel 20 fps	4.78 dB (SNR)	sub-band POS (SB)	ECG sensor
Chen et al. [37]	2018	RGB cam	still, head rotation	indoor lighting	8 bits/channel 120 fps	1.50 bpm (MAE)	spatial-temporal Neural Network	ECG sensor
Niu et al. [38]	2019	webcam	still, talking, head rotation	filament bulb	uncompressed	7.99 (RMSE)	spatial-temporal Neural Network	PPG sensor
Song et al. [39]	2019	webcam	still, minor head motion	stable indoor lighting	8 bits/channel 30 fps	3.80 bpm (RMSE)	CHROM	ECG sensor
Gudi et al. [40]	2019	RGB cam	still, talking, head rotation	indoor lighting	25 fps	1.02 bpm (RMSE)	POS	ECG sensor

IR: infrared. RGBCO: red, green, blue, cyan, orange. RMSE: root mean square error. MAE: mean absolute error. PCC: Pearson's correlation coefficient. SNR: signal-to-noise ratio. ICA: independent component analysis. SCICA: single channel ICA. EVM: Eulerian video magnification.

Table 1.1: A research review of the past-decade technologies for PR estimation using rPPG.

to be motionless in the process, as Ballistocardiographic (BCG) motion is involuntary and may modulate the extracted rPPG signal as a result.

2. *Automatic Face detection*: clearly, a face detection process is necessary for automatically selecting the ROI when the video contains substantial subject's motion. This can be achieved by either conducting face detection on each video frame [24] or tracking an initial ROI by estimating the inter-frame transition matrix from some "good-features-for-tracking" [31,36]. Both these approaches might not be optimal as the former might lead to discontinuous face localization results due to possible false negatives, while the latter might lose track when the videos contain large motion or complex background.

3. *Skin detection*: The non-skin facial pixels have little-to-no contribution to the pulse extraction and might bring additional motion artifacts when the subject is talking or blinking. It is thus favorable to exclude those non-skin samples in each frame. The author in [42] proposed an online learning approach to train a skin pixel detector in the first several frames. This subject- and scene-specific learning approach is robust to the illumination source and the subject's skin tone. However, the system might generate false detection results when the illumination condition changes along the time.

1.2.2 Motion Robust Pulse Extraction

Green Channel generates the highest pulse signal quality compared with other color channels. As the oxyhemoglobin and deoxyhemoglobin have much

greater absorptivity in green light compared with the red or the blue channel, larger intensity variation in green channel during each cardiac cycle is observed. This observation laid the foundation for a series of works [4,27,31,34] which used the green channel for extracting the pulse information.

Blind source separation (BSS) method improves the system robustness by incorporating information from other color channels. As different components of the skin tissue have different optical responses in each spectral range (for example, red, green, and blue), it is possible to separate each component from available color channels based on proper assumptions on the sources and color channel measurement. The BSS schemes are applied to factorize the pulse signal from the RGB-signals by assuming the pairwise source uncorrelation (PCA-based method [43]) or independence (ICA-based method [44]). The pulse channel is selected as the most periodic one after the source separation is performed. Without side information such as the subject's motion or the change of the illumination intensity, each BSS algorithm produces the optimal factorization result when the noise and interference components exhibit the statistical behavior as assumed. However, in a fitness scenario, when strong periodic motion artifacts enter the RGB-signal measured from the face, the statistical assumptions about the source signals might be easily violated, and the channel selection scheme may mistakenly output motion source as the pulse, considering that the motion components may exhibit the highest strength in the frequency domain.

Skin model-based methods are proposed to address this source uncertainty problem in a line of research [16,20,21,29,42] by investigating the color characteris-

tics of pulse and other reflected components. With prior knowledge about the skin tone color vector obtained from a large scale experiments, CHROM algorithm [29] maps the temporally normalized RGB-signals to a color plane orthogonal to the specular component, and the pulse signal is obtained via the alpha-tuning operation. POS algorithm [16] adopts the same skin reflection model but instead maps the normalized RGB-signals to the color plane orthogonal to the intensity variation direction, to eliminate the motion artifacts. The pulse color direction is then searched for within a 90 degree sector, which outputs the highest pulse signal quality on the color plane. The hue change on the skin is tested by experiments to be another useful feature for pulse extraction [45]. 2SR [5] exploits such pulse-induced hue change in a subject-dependent manner by learning the principal direction of the hue channels. All these color mapping schemes use linear combinations of RGB color channels to factorize the pulse from other components. The difference concerning the assumptions of the relations of the source signals reflects on the demixing weights, which are applied on each color channel. For a more detailed discussion about the strength and weaknesses of the algorithms mentioned above, we referred the readers to [16].

Learning-based methods leverages the training data to perform PR estimation. Hsu et al. [46] treats the time-frequency representation of the extracted signal as an image and estimates the PR with a Convolutional Neural Network (CNN). The end-to-end rPPG learning systems [37, 38], which utilize the temporal and spatial attention module for automatic channel weighting and signal selection are appealing and easy-to-use. However, the training and testing of all the learning-based systems

are based on the same datasets with similar experimental setups in each video. The adaptation and robustness across different scenes, and the subject's motion type is still questionable.

The fundamental limitation of the color linear mapping schemes is a lack of measurement dimension. Specifically, a regular RGB camera only offers three degrees-of-freedom in color. A linear color mapping algorithm can thus maximally exclude two independent interferences from the pulse signal. **The extension of the signal degree-of-freedom** thus represents a promising research direction to address this "lack-of-dimension" problem. Note the possibility of treating each facial skin pixel as an rPPG sensor. The spatial sensor redundancy of rPPG sensor could be exploited to increase the measurement degree-of-freedom and thus the robustness of the algorithm. Such idea can be found in [10, 42, 47, 48], where the temporal correspondence of each rPPG sensor is estimated either via dense optical flow alogrithms [49] or estimated facial landmarks [50–52]. Noticeable improvement of the system performance was reported when multiple rPPG sensors became available. However, the large computational load for dense pixel alignment operation makes the system unfriendly to be deployed on a regular device with moderate computation capacity.

In [35], the author extended the signal's degree-of-freedom using a five-band RGBCO camera. Even though a performance gain in the estimation of PR/PRV was claimed, the cost and availability of a five-band camera restrict a wide adoption of the system. In [31,53], the benefit to include the background information in a rPPG system was presented. The illumination change on the face was compensated by that

of the background using an adaptive filter [31] or via a joint blind source separation scheme [53]. Such a system assumes a high correlation between variations of the background reflectance with the non-pulse reflectance on the face. This assumption might be true when the background is stationary and controlled, whereas it might be violated when the background contains additional illumination sources or moving objects.

The subjects' motion information was estimated from the video and exploited. When the camera sensor is fixed, and the subject exercises on the focal plane of the camera, the subject's face motion can be roughly estimated as the face motion trace appeared in the video [54]. Note the motion signal investigated in [54] is mainly pulse-induced ballistocardiography (BCG) motion signal. The face motion signal discussed in this paper contains little pulse component as the BCG component becomes negligible when the video contains voluntary subject motion. This property enables the author in [10, 55] to filter the pulse signal from the motion corrupted rPPG measurement.

A sub-band based approach is proposed in [20], where the essence of the algorithm is to perform a frequency-dependent POS pulse color mapping. Even though an increase of the measurement degree-of-freedom is claimed in the paper and the system performance improves in the fitness scenario, there is no gain in terms of the information level, and the motion residue can still dominate over pulse component in the processed signal.

1.3 Main Contributions

In this thesis, we explore the modeling, estimation, and inference problems with a focus on cardiovascular sensing applications. We first study the robust rPPG applications in Chs. 2 and 3. We next study the weak multiple frequency traces tracking problems in Ch. 4. We last study a biomedical inverse problem to reconstruct the ECG signal from PPG in Ch. 5.

Below we detailed the key contributions of this dissertation research.

1.3.1 Fitness Heart Rate Measurement using Face Videos

Recent studies showed that subtle changes in human face color due to the heartbeat could be captured by digital video recorders. Most existing work focused on still/rest cases or those with relatively small motions, while limited art addresses the large fitness motion scenarios either in a highly constrained setup or obtains unsatisfactory heart rate estimate. In this work, we propose an end-to-end heart-rate monitoring method for fitness exercise videos. We focus on designing a highly precise motion compensation scheme with the help of the localized facial optical flow and use motion information as a cue to adaptively remove ambiguous frequency components for improving the heart rate estimates. Experimental results show that our proposed method can achieve highly precise estimation with an average error of 3.3 beats per minute (BPM) or 1.74% in relative error.

1.3.2 Robust Pulse Rate and Pulse Rate Variability Measurement from Face Video

Following our first work focusing on accurate fitness pulse rate estimation from face video, we present another novel rPPG system that is robust for both pulse rate and pulse rate variability extraction from face video when the subject is exercising, and the video contains large subject motions. We focus on designing an online learning scheme for a precise subject- and scene-specific skin detection and use an adaptive filter bank system to clean the pulse signal so that the inter-beat-intervals and pulse rate variability are precisely estimated. The computation complexity is greatly reduced compared to the optical-flow method, and the system is capable of running in realtime in devices with just moderate computation power.

1.3.3 Adaptive Multi-Trace Carving based on Dynamic Programming

Many biomedical problems often boil down to the problem of frequency extraction. Previous works have studied to track the frequency components by incorporating the temporal correlation of the frequency component in their model. However, the limitations of prior frequency extraction approach, such as low performance in noisy conditions, inability to track multiple frequency components, or inefficient real-time implementation, restrict their deployment in many real-world tasks. To address this issue and facilitate the micro signal extraction process, we propose AMTC, an

unified approach for tracking one or more subtle frequency components in very low signal-to-noise ratio (SNR) conditions. AMTC treats the signal's time-frequency representation as an image and identifies all frequency traces with dominating energy through iterative dynamic programming and adaptive trace compensation. In addition, AMTC considers a long duration of high trace energy as an indicator of the presence of a frequency component. The trace detection problem is thus addressed with a robust test statistic characterizing the trace energy. By doing this, AMTC is capable of simultaneously detecting and tracking multiple frequency components accurately, even from highly-corrupted signals. Extensive experiments using both synthetic and real-world data reveal that the proposed method outperforms the state-of-the-art methods under low SNR conditions and can be implemented in near-realtime settings.

1.3.4 Learning Your Heart Actions From Pulse: ECG Waveform Reconstruction From PPG

We next study the relation between electrocardiogram (ECG) and photoplethysmogram (PPG) and infer the waveform of ECG via the PPG signals. In order to address this inverse problem, a transform is proposed to map the discrete cosine transform (DCT) coefficients of each PPG cycle to those of the corresponding ECG cycle. The resulting DCT coefficients of the ECG cycle are inversely transformed to obtain the reconstructed ECG waveform. The proposed method is evaluated with the different morphologies of the PPG and ECG signals on three benchmark

datasets with a variety of combinations of age, weight, and health conditions using different training modes. Experimental results show that the proposed method can achieve a high prediction accuracy greater than 0.92 in averaged correlation for each dataset when the model is trained subject-wise. With a signal processing and learning system that is designed synergistically, we are able to reconstruct ECG signal by exploiting the relation of these two types of cardiovascular measurement. The reconstruction capability of the proposed method may enable low-cost ECG screening for continuous and long-term monitoring. This work may open up a new research direction to transfer the understanding of clinical ECG knowledge base to build a knowledge base for PPG and data from wearable devices.

Chapter 2: Fitness Heart Rate Measurement using Face Videos

2.1 Introduction

Optimizing the adaptation and preparedness for enhanced performance is the goal for the athletic training and recovery [56]. The use of heart rate (HR) measures in sports represents a non-invasive, time-efficient method to monitor the training dose and quantify the athletes' response [56–60]. With the context information of the training and proper interpretations of the HR measures, such practice has direct implication for adjusting the training load in order to harness the individual or team training objectives in a safe and effective manner.

The conventional cardiac monitoring (e.g., electrocardiogram (ECG) [61] and photoplethysmography (PPG) [62, 63]) are obtrusive and may cause skin irritation problem or discomfort during prolonged use. Contact-free monitoring of the pulse rate using videos of human faces is a user-friendly approach compared to conventional contact-based ones such as electrodes, chest belts, and finger clips. Such monitoring system extracts from a face video a 1-D sinusoid-like face color signal that has the same frequency as the heartbeat. The ability to measure heart rate without touch-based sensors is attractive and gives it potentials in such applications as smart health and sports medicine for the following three reasons. First, it

brings more comfort to the end-users as no sensor needs to be worn, and free body movements are allowed during the monitoring process; Second, the data is collected unobtrusively. Thus invaluable recording will not be lost due to maloperation of the sensor or loose of contact; Third, the video content has a potential to offer valuable training context. A video-based solution could provide paired subject's motion information, which may enable a better interpretation for the subject's cardiac response with the quantified external load measure [57].

The last decade witnesses a rapidly increasing number of articles (seen Table 1.1) that published on rPPG address the pulse rate estimation on still/rest cases or with relative small motions [24, 26–28, 30–40]. Among a few art [16, 20, 21] addressing the fitness pulse rate extraction with strenuous subject motion, the pulse rate estimation result is either not reported [16, 20] or highly deviated from the reference [29]. It is still attractive to ask and answer the question in this paper as "Can we accurately estimate a subject's pulse rate remotely from his/her face video in a normal fitness setup with sufficient illumination?".

This is not a trivial question because the challenges of the fitness rPPG sensing come from each component of the rPPG sensing system, namely, the camera, the illumination conditions, and the subject. In a fitness setup, the motion-induced intensity and color change may very well dominate over the reflected light from the facial skin, while the pulse-induced color variation is much subtler. The measurement is also associated with a group of nuisance signals, such as the sensor and quantization noise. To extract the subtle pulse signal that may have a much smaller magnitude than the dominating video components and simultaneously pro-

tect it from being corrupted by other nuisance signals, one usually has to tackle the problem with extra caution.

The established arts provide solutions to improve the system performance in terms of the robustness against the sensor noise and the environmental noise, including the illumination changes and body motions. However, an existing rPPG system may still fail to estimate the subject's PR when the signal distortion problems are improperly addressed. It is thus attractive for the rPPG community if an end-to-end system provides the capability to accurately estimate the PR.

In this work, we aim to examine the best possible performance for fitness exercise videos when the registration error is minimized for the color-based heart-rate monitoring method. A block diagram of our proposed method is shown in Fig. 2.1. We minimize the registration error using pixel-level optical flow based motion compensation [64, 65] that is capable of generating almost "frozen" videos for best extracting the face color signals. We use the subject's motion information estimated from the video to reject the motion components in the extracted face color information by an adaptive least mean square filter. The final pulse rate estimate is updated in realtime via a frequency tracking algorithm which utilizes the temporal pulse rate correlation for generating the robust estimation results. We focus on the fitness scenarios that heart rate often wildly vary at different stages of fitness exercises, and present our results in widely adopted metrics [31, 66, 67] for comparison purpose.

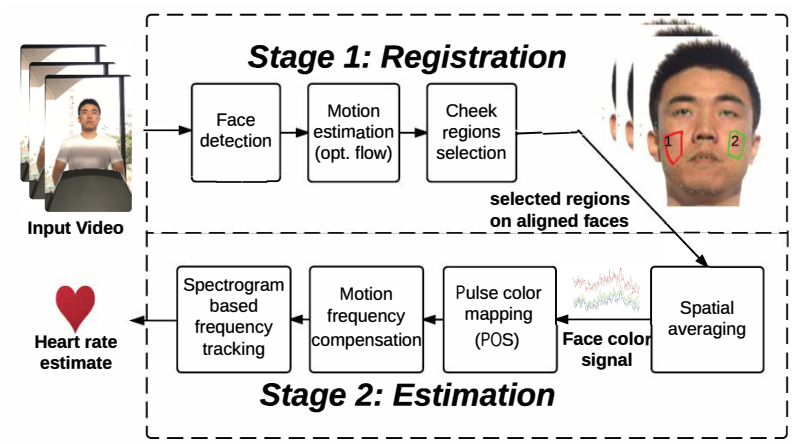


Figure 2.1: Flowchart for the proposed heart rate monitoring method for fitness exercise videos.

2.2 Proposed Method

2.2.1 Precise Face Registration via Localized Optical Flow

A highly precise pixel-level motion compensation is a crucial step toward generating a clean face color signal. Fitness exercise videos may contain large and periodic motions. Our proposed method focuses on a highly precise motion compensation scheme to allow generating a clean face color signal to facilitate the latter analysis steps, and uses the resulting motion cue as the guide to adaptively remove ambiguous frequency components that can be very close to the heart rate.

We use the Viola–Jones face detector [68] to obtain rough estimates of the location and size of the face. We clip and resize the face region of each frame to 180 pixels in height, effectively generating a prealigned video for the face region.

The prealignment significantly reduces the lengths of motion vectors, which in turn makes results of optical flow more reliable. In our problem, two face images are likely have a global color difference due to the heartbeat. In order to conduct

a precise face alignment, instead of using the illumination consistency assumption that is widely used, we assume more generally that the intensity I of a point in two frames are related by an affine model, namely,

$$I(x + \Delta x, y + \Delta y, t + 1) = (1 - \epsilon)\, I(x, y, t) + b \tag{2.1}$$

where ϵ and b control the scaling and bias of the intensities between two frames. Both of them are usually small.

Based on the model in (2.1), we have the following three remarks aiming to justify the validness of the optical flow method we adopted for pixel alignment in an rPPG application.

1. Traditional local-based optical flow techniques tackling the illumination consistency cases such as Taylor expansion and regularization can be similarly applied for (2.1). Our mathematical analysis showed that omitting the illumination change due to the heartbeat, and applying a standard optical flow method leads to *a bias term* that is at the same order magnitude compared to the intrinsic error (in terms of standard deviation) of the optical flow system.

2. The bias term mentioned above can be alleviated with a global flow estimation strategy with flow smoothness constraint in the optical flow estimation formulation [64]. In this way, the flow estimation in smooth skin region on face will be regularized globally by other flow estimates on salient facial features such as the noise and the eyes.

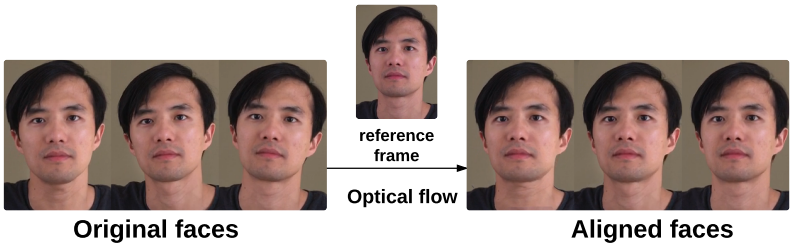


Figure 2.2: Face images from a same video segment before and after optical flow based motion compensation using the same reference face.

3. Even with prior alignment with face detection, the pixel displacement in two consecutive frames might still be large enough to fail a local method, such as [69]. A coarse-to-fine hierarchical searching strategy [64, 70] provides much better flow estimation results in presence of large subject motion as large displacement becomes small in a coarser image scale.

Based on the above discussion, we therefore use Liu's optical flow implementation [65] in our work.

We divide each video into small temporal segments with one frame overlaping for successive segments. We use the frame in the middle of the segment as the reference for optical flow based motion compensation. This would ensure two frames being aligned do not have significant occlusion due to long separation in time. Fig. 2.2 shows a couple of face images from a same segment before and after optical flow based motion compensation using the same reference.

With the precisely aligned face videos in short segments, we can estimate the face color for each frame by taking a spatial average over pixels of the cheek for R, G, and B channels, respectively. We call the three resulting 1-D time signals the *face color signals*.

When concatenating segments into color signals, the last point of the current

segment and the first point of the next segment may have different intensities because they correspond to the same frame whose motion compensation were conducted with respect to two different references. To address this problem, the difference of the intensity between the two points is calculated and the resulting value is used to bias the signal of the next segment in order to maintain the continuity.

To determine the cheek regions for conducting spatial averaging, we construct two conservative regions that do not contain facial structures and are most upfront in order to avoid strong motion-induced specular illumination changes. We use facial landmarks identified by the method proposed in [50] to facilitate the construction of the cheek regions. Each cheek region is constructed to be a polygon that has a safe margin to the facial structures protected by the landmarks. One example for such selected cheek regions and corresponding face landmarks is shown on the face in Fig. 2.1.

2.2.2 Motion Compensation via joint-channel NLMS

Once the skin pixels are detected in each frame, a temporal RGB sequence $\tilde{\mathbf{C}}(t)$ is generated by spatially averaging the RGB values of the detected skin pixels and temporally normalized in each color channel. $\tilde{\mathbf{C}}(t)$ is then linearly mapped to a specific color direction in the RGB space to generate a 1-D pulse signal. The pulse color mapping schemes have been extensively investigated in [16] and [20]. We note that the design of the pulse color mapping algorithms discussed in this paper is not within the contributions of this work, although different pulse color mapping

approaches [16, 20, 29, 44] are implemented and evaluated in Section 3.4.

Without loss of generosity, we assume that the face color signal $\tilde{\mathbf{C}}(t)$ is mapped to the POS direction [16], which is one of the most robust color features representing the highest relative pulse strength. We denote the projected 1-D processed signal as $c_{\text{pos}}(t)$. According to (1.5), we have

$$\begin{aligned} c_{\text{pos}}(t) &= \mathbf{p}^{\intercal} \cdot \tilde{\mathbf{C}}(t) \\ &= \overbrace{\mathbf{p}^{\intercal} \cdot \mathbf{u_p}' \cdot p(t)}^{\textbf{Pulse}} + \overbrace{\sum_{k=1}^{K} \mathbf{p}^{\intercal} \cdot \mathbf{u_{m,k}} \cdot m_k(t)}^{\textbf{Motion Residue}}, \end{aligned} \tag{2.2}$$

where $\mathbf{p} \in \mathrm{R}^{3\times 1}$ denotes the projection vector of POS algorithm. The motion residue term in Eq. (3.6) is negligible when the illumination source is single, as the POS direction is orthogonal to the color direction of the motion-induced intensity change, and the specular change is suppressed via "alpha tuning" [29]. However, if the video is captured in an uncontrolled environment, the motion residue term is often non-negligible, and sometimes can be more significant than the pulse term.

To address this problem, we rely on estimating the motion term in (3.6) using the estimate of the face motion in both horizontal and vertical directions. Note that the subject motion and the motion artifact in rPPG signal share the causal relation and are thus highly correlated. Meanwhile, we assume the pulse signal is uncorrelated with the subject motion. To capture this signal correlation, we propose to use the Normalized Least Mean Square (NLMS) filter [71], and the face motion signals in both horizontal and vertical directions are estimated and deployed

to approximate and mitigate the motion residue term in (3.6). We denote the estimated face motion sequence in horizontal and vertical directions as $m_x(t)$ and $m_y(t)$. The structure of the adaptive filter framework are shown in Fig. 2.3. We treat $c_{\text{pos}}(t)$ as the filter's desired response at time instant t. We treat the motion tap vector $\mathbf{m}^\intercal(t) \triangleq [m_x(t-M+1), m_x(t-M+2), ..., m_x(t), m_y(t-M+1), m_y(t-M+2), ..., m_y(t)]$ as the input and $\tilde{c}_{\text{pos}}$ as the output of the system and also the error signal. The estimated tap-weight vector of the transversal filter is denoted as $\hat{\mathbf{w}}(t)$, and weight control mechanism follows the Normalized Least Mean Square (NLMS) algorithm [71] as below

$$\begin{aligned} \tilde{c}_{\text{pos}} &= c_{\text{pos}} - \hat{\mathbf{w}}^\intercal(t) \cdot \mathbf{m}(t), \\ \hat{\mathbf{w}}(t+1) &= \hat{\mathbf{w}}(t) + \frac{\mu}{\|\mathbf{m}(t)\|^2} \mathbf{m}(t) \cdot \tilde{c}_{\text{pos}}, \end{aligned} \tag{2.3}$$

where μ denote the adaptation constant, which is normalized by the norm square of the input vector $\mathbf{m}(t)$.

2.2.3 Pulse Rate Tracking via dynamic programming

To this end, the signal quality is improved via precise facial pixel alignment, robust pulse color mapping, and adaptive motion filtering. As did most of the prior arts [21, 29, 44], one might now assume temporal stationarity of the processed pulse signal and consider to estimate the instantaneous heart rate by mapping the 1-D processed signal to the frequency domain and searching the spectral peak within the normal human heart rate range (50-240 bpm). Such highest-peak estimation

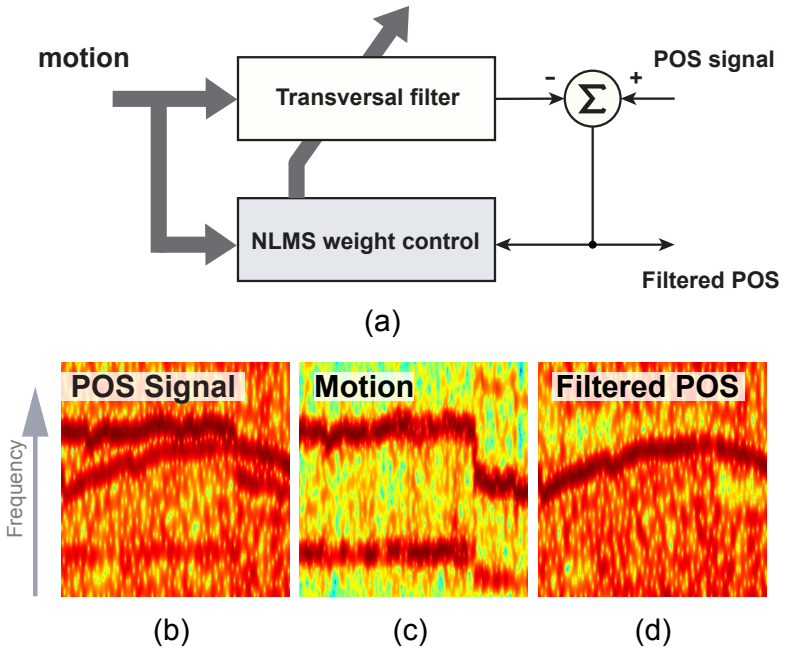


Figure 2.3: (a) Adaptive motion compensation filter framework. Spectrogram of the POS signal (b), the combined normalized subject motion in horizontal and vertical directions (c), and the filtered POS signal (d) as the output of (a).

method can give accurate result when the SNR is high, but may frequently generate outliers when SNR drops.

Note that for a healthy human being, two temporally consecutive heart/pulse rate measurements may not deviate too much from each other. We exploit this heart rate continuity property, and track people's heart rate by searching for the dominating frequency trace appearing in the signal's spectrogram image [72]. The process of the tracking algorithm is briefly described below.

Let $\mathbf{Z} \in \mathbb{R}_{+}^{M \times N}$ be the magnitude of a processed signal spectrogram image, which has N discretized bins along the time axis and M bins along the frequency axis. We model the change of the frequency value between two consecutive bins at $n-1$ and n as a one step discrete-time Markov chain, characterized by a transition probability matrix $\mathbf{P} \in \mathbb{R}^{M \times M}$, where $P_{m'm} = P(f(n) = m | f(n-1) = m'), \forall m, m' =$

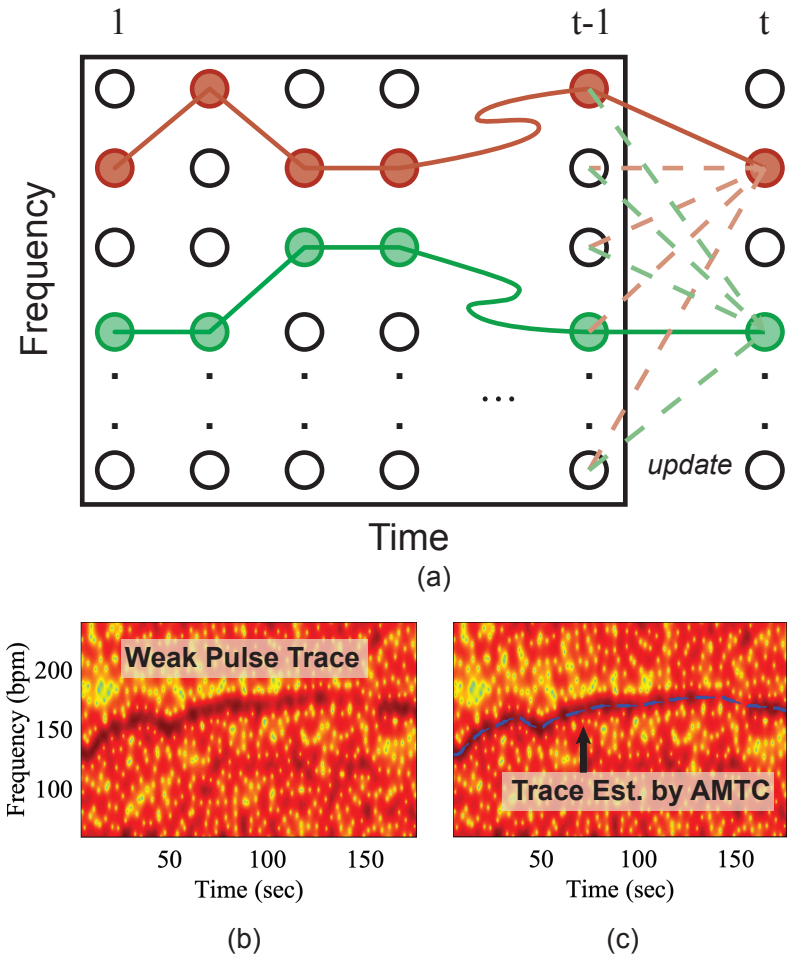


Figure 2.4: (a) Pulse rate tracking via dynamic programming. (b) Spectrogram of the rPPG signal with weak pulse frequency trace. (c) Spectrogram in (b) overlaid with trace estimation result in blue dashed line. The weak pulse trace is precisely estimated regardless of strong nearby noise and disconnectivity around 140 seconds.

$1, ..., M$, and $\forall n = 2, ..., N$. The regularized single trace frequency tracking problem is formulated as follows

$$\mathbf{f}^* = \underset{\mathbf{f}}{\operatorname{argmax}} \quad E(\mathbf{f}) + \lambda P(\mathbf{f}), \tag{2.4}$$

The regularized tracking problem in (4.3) can be solved efficiently via dynamic programming. First, we iteratively compute an *accumulated regularized maximum energy map* $\mathbf{G} \in \mathbb{R}_+^{N \times M}$ column by column for all entries (m, n) as follows

$$\mathbf{G}(m, n) = \mathbf{Z}(m, n) + \max_{m'=1,...,M} \{\mathbf{G}(m', n-1) + \lambda \log P_{m'm}\}. \tag{2.5}$$

After completing the calculation at column $n = N$, the maximum value of the Nth column is denoted as $f^*(N)$. Second, we find the optimal solution by backtracking from the maximum entry of the last column of the accumulated map $\mathbf{G}$. Specifically, we iterate n from $N-1$ to 1 to solve for $f^*(n)$ as follows

$$f^*(n) = \underset{f(n)}{\operatorname{argmax}} \quad (f(n), n) + \lambda \log P_{f(n)f^*(n+1)}. \tag{2.6}$$

Note that we can avoid transitions from state m' to state m by setting $P_{m'm} = 0$, as the regularized term would penalize the total energy to $-\infty$. If we assume uniform random walk transitions, i.e., $P_{mm'} = \frac{1}{2k+1}$, $|m' - m| \leq k$, then problem (4.3) is degenerated to the *seam carving* problem defined in [73], and in this case the value λ does not affect the solution.

This offline dynamic programming solution can be adapted naturally to a

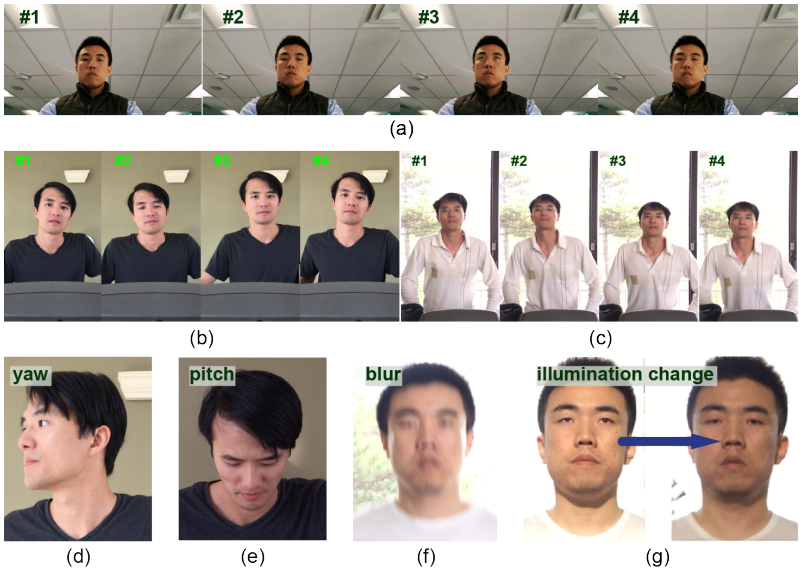


Figure 2.5: Sample frames in fitness video dataset with different types of motions: (a) stationary bike, (b) elliptical machine, (c) treadmill. The challenges in the dataset includes head rotation in (d) yaw and (e) pitch, (f) motion blurred frames, and (g) significant illumination variation from diffused sunlight.

realtime implementation. Suppose we stand at the time instance $n-1$, where $\mathbf{G}(n-1)$ has been updated based on previous input frames $\mathbf{Z}(1:n-1)$. At the arrival of the next innovation spectral frame $\mathbf{Z}(n)$, we update $\mathbf{G}(n)$ similarly according to (3.10). The PR estimation can be simultaneously updated by the same backtracking process describe in (3.11).

2.3 Experiment Setup

Our proposed method was evaluated on a self-collected fitness exercise dataset to demonstrate the efficacy of the PR estimation on dealing with motions. The dataset has 25 videos in which 10 contain human motions on an elliptical machine, 10 contain motions on a treadmill, 5 contains motions on a stationary bike. The experiment setups are detailed as below.

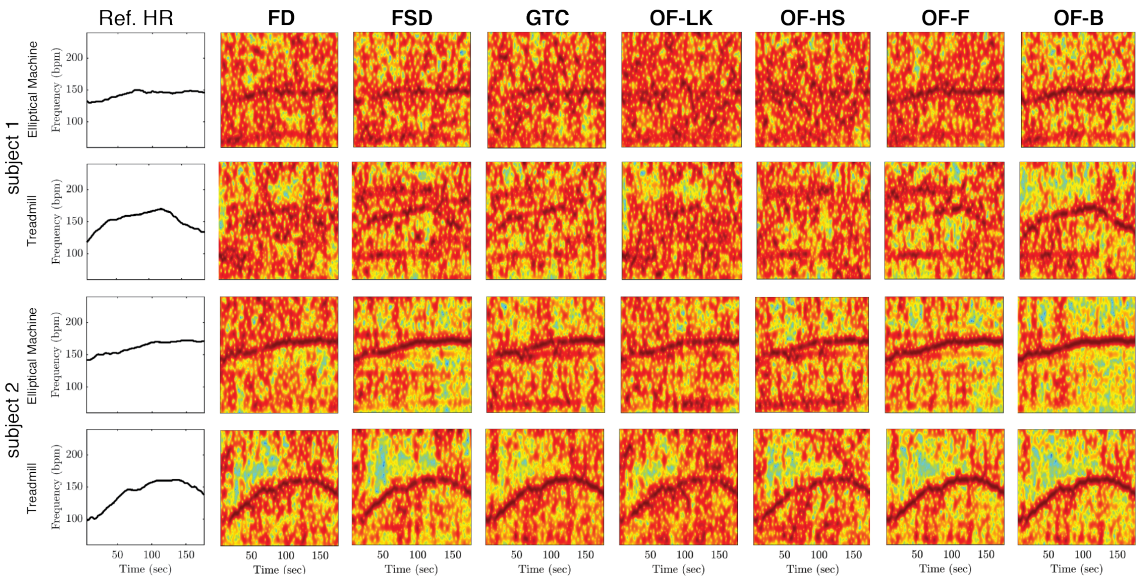


Figure 2.6: Four qualitative comparison results using different subject motion estimation schemes. (Column 1) The reference HR measured by ECG based chest belt. (Column 2-8) Spectrograms of the extract pulse signal using the proposed system with the motion estimation schemes as FD, FSD, GTC, OF-LK, OF-HS, OF-F, and OF-B, respectively.

Environment In order to test the robustness of the system, the experiment was conducted in two uncontrolled apartment fitting rooms. The active illumination sources only involve the existing lighting equipment in the scene. No additional illumination equipment or backdrop were placed during the recording. Both fitting rooms were well-lit with several over-the-top florescent lights and with diffuse daylight passing into the gym through glass walls. The presence of other subjects exercising or entering the scene is possible, and no regulation is placed to restrict people from entering the room.

Devices and Reference Signal The first 5 stationary bike videos were captured by the rear camera of a Huawei P9 mobile phone. The rest 20 videos involving the elliptical machine and treadmill motions were captured by the rear camera of a

iPhone 6s mobile phone. The shutter speed of both sensors was kept constant and the face in focus at all time. During the video recording, we obtained the subject's reference heart rate by simultaneously acquiring the subject's electrocardiogram (ECG) with a chest belt monitor (Model: Polar H7).

Placement of the Sensors The mobile camera was placed on the holder of the stationary bike, affixed on a tripod, or held by the hands of a person other than the test subject. The camera is placed in front the subject face at a distance of about 1 meter away at approximate same height with the subject's face during the recording. The ECG chest belt was correctly worn underneath the subject's cloth, and the sensor was in direct contact with the subject's chest skin to maximize the SNR of the reference ECG signal.

Participants Two male Asian subjects are involved in the experiment. The skin tone of both subjects is classified as Skin-type III according to the Fitzpatrick skin scale [74]. Among all the videos in the dataset, 5 treadmill videos and 5 elliptical machine videos belonged to one subject. The rest 15 belonged to the other. Based on the most recent medical examination results, none of the subject was diagnosed with any known CVDs or pulmonary diseases.

Compared Motion Estimation Methods In order to test the efficacy of the proposed motion estimation method, we compared it with other possible alternatives listed below for a thorough evaluation.

1. Face detection (FD): the face rectangle region is first estimated, and the two

check regions are localized according to the facial landmarks estimated by [50].

2. Face and skin detection (FSD): the face ROI is localized by a pixel color based skin detection algorithm [75] operating in the face detected rectangle region.

3. Geometric transform correction (GTC): we first detect the face ROI in the first frame in the same way as FD. We then estimate the ROI in the next frame by projecting each point in the ROI in the previous frame to the next frame according to the estimated 2D geometric transform. The geometric transform is estimated in the same way detailed in [31].

4. Proposed optical flow framework using the Lucas-Kanade method (OF-LK) [76], the Horn and Schunk method (OF-HS) [77], the Farneback method (OF-F) [70], and the Brox method (OF-B) [64].

Compared Pulse Color Mapping Methods As another comparison study, we evaluated the state-of-art pulse color mapping algorithms which include the Blind Source Separation (BSS) based approaches (ICA [24] and PCA and skin model based approaches (CHROM [29], POS [16], and SB [20]). Each method maps the RGB face color signal to a specific color direction aiming to provide the highest relative pulse strength based on different models or source-observation assumptions.

A detailed discussion of these approaches based on the human skin reflection model can be found in [16] and [20]. However, the evaluations and the conclusions in both papers are only based on the SNR metric, which may be insufficient in this fitness scenario. This is because a pulse signal with high interference, such

as a strong motion frequency component in the normal heart rate range, and low noise might confuse a frequency estimator or tracker much more significantly than a signal with only white noise at a same SNR level. In this paper, we re-evaluate these color mapping approaches using our proposed estimation framework and evaluate not only the SNR metric but accuracy measure of the pulse rate estimation results.

Compared Frequency Tracking/Estimation Methods In order to single out the contribution and demonstrate the effectiveness of our proposed frequency estimation method used in this paper, we compared it with three other trending frequency estimation methods listed below.

1. Maximum energy (ME): the pulse rate in each spectral frame is estimated as the frequency component with the highest spectral energy. This highest peak selection scheme yields the maximum likelihood frequency estimate when the noise component is independent with the source and is temporally independent. However, such method would frequent generate biased results when the measurement is highly corrupted by either noise or interference.

2. Particle filter (PF) [78]: PF first approximates the posterior distribution of the frequency state via the sequential Monte Carlo method. The pulse rate is then estimated as the one that maximize the posterior distribution in each time instance.

3. Yet Another Algorithm for Pitch Tracking (YAAPT) [79]: YAAPT estimate the frequency component from a set of local spectral peaks in the spectrogram

using the Viterbi algorithm.

Parameter Settings The following parameters are used in our investigation unless otherwise stated:

1. Each video lasts about 3 minutes.

2. The frame rate is 30 frame per second. The resolutions are 1280×720. The averaged bit rate is about 6 MB per second. The video codecs is AVC (H.264).

3. The tap number for joint-channels NLMS is 8, and the NLMS learning rate is 0.1.

4. Each video was divided into segments of 1.5 secs in order to guarantee small scene changes within each segment for optical flow's best performance.

5. The spectrum analysis window length was set to 10 secs with 98% overlap. A Hamming window is applied in each analysis window, and the number of frequency bins in the normal PR range (50 to 240 bpm) was set as 1024 via padding zeros at the end of the analysis signal sequence. The transition probability model used in the frequency tracking algorithm is a uniform random walk model with the width parameter $k = 1$ bpm. The number of particles used in PF is set as 1000.

2.3.1 Metrics of Performance Evaluation

Pulse Signal Quality The same SNR metric used in [20,29,42] is adopted in this paper. The value of this metric indicated the pulse signal quality using different

	SNR (dB)	PCC	p-value	E_{count} (%)	E_{rate} (%)	E_{RMSE} (bpm)
FD	-5.024 (4.019)	0.734 (0.375)	0.000 (0.001)	22.664 (25.426)	6.380 (8.935)	8.988 (16.834)
FSD	-1.626 (4.324)	0.855 (0.207)	0.000 (0.000)	14.381 (28.342)	5.273 (12.342)	7.280 (15.750)
GTC	-3.081 (2.866)	0.776 (0.325)	0.001 (0.002)	28.398 (34.234)	7.450 (2.985)	12.532 (15.754)
OF-LK	-7.638 (3.213)	0.670 (0.420)	0.003 (0.002)	35.962 (40.121)	11.858 (14.872)	12.613 (20.563)
OF-HS	-6.629 (3.631)	0.778 (0.341)	0.016 (0.082)	40.285 (46.568)	7.628 (12.957)	18.621 (20.845)
OF-F	-1.237 (5.012)	0.817 (0.281)	0.000 (0.001)	15.150 (25.923)	5.128 (12.534)	8.892 (12.355)
OF-B	**-0.771** (4.823)	**0.861** (0.208)	0.000 (0.000)	**8.892** (10.235)	**1.739** (2.234)	**3.273** (6.431)

Table 2.1: The system performance of the pulse rate estimation in terms of sample mean and standard deviation (in paranthesis) of SNR, PCC, p-value, $E_{(count)}$, E_{rate}, and E_{RMSE}. Different motion compensation schemes were evaluated. The top performed item is highlighted in bold in each metric.

	SNR (dB)	PCC	p-value	E_{count} (%)	E_{rate} (%)	E_{RMSE} (bpm)
ME	-0.771 (4.823)	0.168 (0.375)	0.067 (0.001)	34.157 (25.426)	13.424 (8.935)	33.601 (16.834)
PF	-0.771 (4.823)	0.598 (0.207)	0.003 (0.000)	38.564 (28.342)	14.355 (12.342)	22.605 (15.750)
YAAPT	-0.771 (4.823)	0.372 (0.325)	0.061 (0.002)	32.610 (34.234)	10.739 (2.985)	18.501 (15.754)
DP	-0.771 (4.823)	**0.861** (0.208)	0.000 (0.000)	**8.892** (10.235)	**1.739** (2.234)	**3.273** (6.431)

Table 2.2: The system performance of the pulse rate estimation in terms of sample mean and standard deviation (in paranthesis) of SNR, PCC, p-value, $E_{(count)}$, E_{rate}, and E_{RMSE}. Different frequency estimation algorithms were evaluated. The top performed item is highlighted in bold in each metric.

signal processing techniques. The SNR metric is computed on each power spectrum frame out of the spectrogram. The SNR metric is defined as the ratio between the energy around the first two harmonics of the reference PR and the remaining energy of the power spectrum:

$$\text{SNR} = 10 \log_{10} \left(\frac{\sum_{f \in \mathcal{F}} S_n(f) P(f)}{\sum_{f \in \mathcal{F}} (1 - S_t(f)) P(f)} \right), \tag{2.7}$$

where $S_n(f)$ is a defined binary window to select the frequency bins belong to the two harmonics region; $P(f)$ is the power spectrum of the pulse signal; set $\mathcal{F} \triangleq \{f | 50 \text{ bpm} \leq f \leq 240 \text{ bpm}\}$

PR Estimation Accuracy Three well-adopted metrics for pulse rate estimation accuracy were adopted in this study. They are specified as below:

1. Root mean squared error:

$$E_{\text{RMSE}} = \sqrt{\frac{1}{N}\sum_{n=1}^{N}(\hat{f}(n) - f(n))^2}, \tag{2.8}$$

2. Error rate:

$$E_{\text{rate}} = \frac{1}{N}\sum_{n=1}^{N}\left|\hat{f}(n) - f(n)\right| \Big/ f(n), \tag{2.9}$$

3. Error count ratio:

$$E_{\text{count}} = \frac{\left|\{n : \left|\hat{f}(n) - f(n)\right| \Big/ f(n) > \tau\}\right|}{N}, \tag{2.10}$$

4. Pearson's correlation coefficient:

$$\text{PCC} = \frac{\sum_{n=1}^{N}(\hat{f}(n) - \bar{\hat{f}})(f(n) - \bar{f})}{\sqrt{\sum_{n=1}^{N}(\hat{f}(n) - \bar{\hat{f}})^2}\sqrt{\sum_{n=1}^{N}(f(n) - \bar{f})^2}}, \tag{2.11}$$

where $|\{\cdot\}|$ denotes the cardinality of a countable set; N denote the total number of the PR estimate; $\hat{f}(n)$, $f(n)$, $\bar{f}$, and $\bar{\hat{f}}$ denote the PR estimate at time instant n, the ground-truth PR at time instant n, the average PR estimate, and the average reference PR. τ was chosen to be 0.03 empirically, determined from the spread of the frequency components.

2.4 Results and Discussion

As our proposed system consists of multiple modules with each focused on a different estimation and processing task, a holistic end-to-end system test would fall into insufficient evaluation on the contribution of each individual module. To address this commonly-seen issue in many prior publications, we discuss in this section the benchmark experiment result based on a fine-level comparison in terms of the motion estimation schemes, the pulse color mapping algorithm, the frequency estimation methods, and the motion adaptive filtering. Considering the presentation redundancy by expliciting all possible combinations with alternative modules, we show a marginal comparison on each module degree. For example, when different motion estimation schemes are evaluated, we choose and fix all other modules as the best-performed individual, such as POS algorithm for pulse color mapping and AMTC for pulse frequency tracking.

2.4.1 Comparison Study for Motion Estimation Schemes

In Fig. 2.6, we show four comparison examples with the spectrogram of each processed pulse signal using different motion estimation schemes. We listed the SNR estimates of the processed pulse signal and the PR estimation accuracy in terms of PCC, E_{count}, E_{rate}, and E_{RMSE} in Table 3.4.1. Note that OF-B outperforms all other methods. The SNR improves about 0.5 dB and the estimation accuracy improves about 2.5% in E_{rate} and about 5 bpm in E_{RMSE}. This is consistent with the qualitative results depicted in Fig. 2.6, where the pulse trace appeared most

significant in the spectrograms with OF-B. which shows that a precise alignment is a crucial step for the video-based heart-rate monitoring method for fitness scenarios. These results demonstrate that a precise alignment is a crucial step for the video-based heart-rate monitoring method for fitness scenarios.

Another interesting finding is that OF-LK and OF-HS generate worse performance even when compared with the straightforward FD. This may be due to the following two reasons. First, local optical flow estimation methods, such as OF-LK, are based on the assumptions of gray value and local flow constancy. Without global constrains such as the flow smoothness, these methods are sensitive to violations of the model assumptions and may generate highly biased estimate in smooth face regions. Second, without a coarse-to-fine motion estimation scheme which address the large motion displacement issue in the fitness scenario, the global flow estimation methods, such as OF-HS, would still underestimate the motion displacement.

2.4.2 Comparison Study for Pulse Color Mapping Algorithms

We show the system performance in averaged SNR and E_{rate} using different pulse color mapping schemes in Fig. 2.7(a-b). Notice that the blind source separation methods (ICA and PCA), in generate output less accurate PR estimate compared with the model-based methods, such as POS and SB. This is mainly due to the occasional failure of the pulse source selection out of the three de-mixed source components when face color measurement contains stronger motion components with the dominating frequency in the normal human PR range, for example 50-240

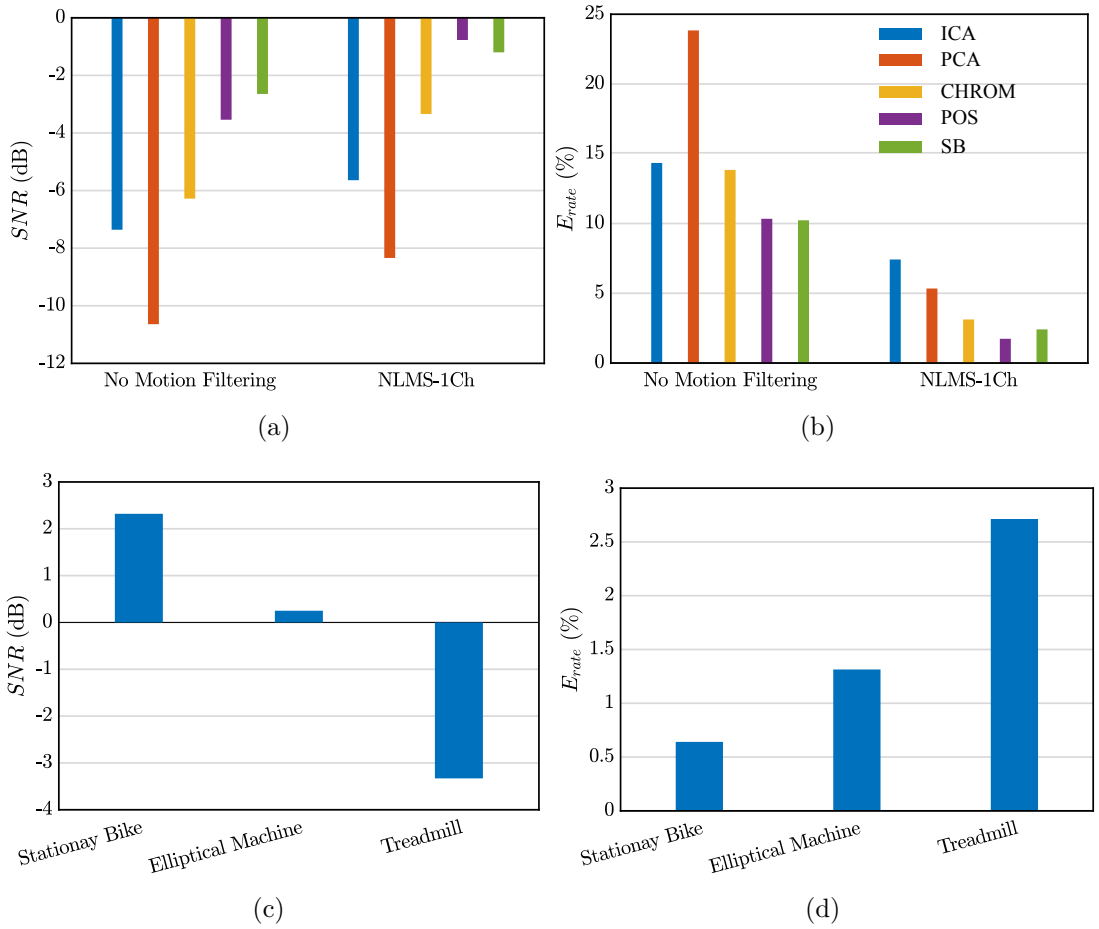


Figure 2.7: The barplots of the system performance using different combinations of the motion filtering and color mapping schemes (a, b), and with different subject motion types (c, d). The results are displayed in SNR (a, c) and E_{rate} (b, d).

bpm. Such violation of the assumption that pulse is the dominating components in the measurement is unfortunately commonly seen in the fitness scenario.

Notice that NLMS-1Ch has greatly improved the SNR by about 2 dB with almost every color mapping schemes. This is mainly due to the contribution of successful motion component removal. Out of the three model-based methods (CHROM, POS, and SB), SB generates the best performance when the NLMS-1Ch is turned off, whereas the POS performs slightly better than SB in the NLMS-1Ch mode.

2.4.3 Comparison Study for Frequency Estimation Methods

To study the contribution of the proposed frequency tracking algorithm for robust PR estimation, we list the performance result of four different frequency estimation or tracking methods in Table 2.3.1. The proposed AMTC tracking method clearly outperforms the other three methods, with a performance gain of more than 8% in relative error and five times better in E_{RMSE} than the second best YAAPT method.

2.4.4 Impact of the Fitness Motion Type

To study the effect of subject's exercise motion to the pulse signal and the PR estimation accuracy, we show the averaged SNR and E_{rate} using bar plots in Fig. 2.7(c) and (d) respectively. Notice that the highest pulse signal quality and the PR estimation accuracy is achieved in stationary bike scenario while the PR estimation in treadmill scenario is overall least accurate. As seen in the sample

video frames shown in Fig. 2.5(a-c), there is only minor face rigid motion when a subject is exercising on a stationary bike, especially in a sitting position. On the other hand, the subject motion is much more significant in the elliptical machine and the treadmill scenarios. Therefore the experimental results are consistent with the intuition that the more significant the subject exercising motion is, the more difficult it becomes to extract precise pulse rate from the face videos.

2.5 Conclusion

In this chapter, we proposed a heart rate monitoring method for fitness exercise videos. We focused on building a highly precise motion compensation scheme with the help of the localized facial optical flow, and used motion information as a cue to adaptively remove ambiguous frequency components for improving the heart rates estimates. Experimental results show that our proposed method can give precise estimates at an average error of 1.1 bpm in RMSE or 0.58% in relative error.

Chapter 3: Robust Fitness Pulse Rate and Pulse Rate Variability Measurement from Face Video

3.1 Introduction

In this chapter, we extend the discussion in the previous chapter and propose yet another rPPG system for robust fitness PR and PRV estimation.

Instead of densely estimating the displacement for each facial skin pixels, we proposed a novel unsupervised learning scheme to detect the skin pixels on face. An ellipsoid-shaped skin classifier is learned from the face pixel samples in the first frame of the video and is deployed in the subsequent frames to detect facial skins. This step not only alleviate the computational load for dense optical flow estimation, but also guarantees the quality of the extracted pulse signal as the skin surface contains the highest pulse signal-to-noise ratio (SNR). A filter-bank with an adaptive sub-band modification layer is designed for precise bandpassing operation to reconstruct the pulse signal and facilitate the estimation of the PRV.

The rest of the chapter is organized as follows. In Section 3.2, we discuss the challenges of the problem and related works. In Section 3.3, we introduce our proposed rPPG system. In Section 3.4, we discuss the experimental setup of our self-

collected benchmark dataset and evaluate the proposed system with the database. In Section 3.6, we discuss the impact on the system performance of various factors. In Section 5.6, we conclude the chapter.

3.2 Challenges and Related Work

The past decade has witnessed much progress in the rPPG community to improve the robustness of the system in challenging scenarios in terms of skin tones, subject motion, and environmental illumination change. The blind source separation (BSS) schemes are applied to factorize the pulse signal from the RGB-signals by assuming the pairwise source uncorrelation (PCA-based [43]) or independence (ICA-based [44]). The pulse channel is selected as the most periodic one after the source separation is performed. Each BSS algorithm produces the optimal factorization result when the noise and interference components exhibit the statistical behavior as assumed. However, in fitness scenario when strong periodic motion artifacts enter the RGB-signal measured from the face, the statistical assumptions about the source signals might be violated, and the channel selection scheme may mistakenly output motion source as the pulse.

The source uncertainty problem mentioned above is addressed in a line of research [16, 20, 21, 29, 42] by investigating the color characteristics of pulse and other reflected components. With prior knowledge about the skin tone color vector obtained from a large scale experiment, CHROM algorithm [29] maps the temporally normalized RGB-signals to a color plane orthogonal to the specular component,

and the pulse signal is obtained via the alpha-tuning operation. POS algorithm [16] adopts the same skin reflection model. Different from CHROM, POS first maps the normalized RGB-signals to the color plane orthogonal to the intensity variation direction, to eliminate the motion artifacts in that direction. The pulse color direction is then searched for within a 90 degree sector, which outputs the highest pulse signal quality on the plane orthogonal to the skin tone. The hue change on the skin is tested by experiments to be another useful feature for pulse extraction [45]. 2SR [5] exploits such pulse-induced hue change in a subject-dependent manner by learning the principal axes of the hue channels. All these color mapping schemes use linear combinations of RGB color channels to factorize the pulse from other components. The difference concerning the assumptions of the relations of the source signals reflects on the demixing weights applied on each color channel. For a more detailed discussion about the strength and weakness of the algorithms mentioned above, we referred the readers to [16].

The fundamental limitation of the color linear mapping schemes is a lack of measurement dimension. Specifically, a regular RGB camera only offers three degrees-of-freedom in color. A linear color mapping algorithm can thus maximally exclude two independent interferences from the pulse signal. Note the possibility to treat each facial skin pixel as an rPPG sensor. The spatial sensor redundancy of rPPG sensor could be exploited to increase the measurement degree-of-freedom and thus the robustness the algorithm. Such idea can be found in [10, 42, 47, 48], where the temporal correspondence of each rPPG sensor is estimated either via dense optical flow alogrithms [49] or estimated facial landmarks [50–52]. Noticeable system

improvement is seen when multiple rPPG sensors become available. However, the large computational load for dense pixel alignment operation makes the system unfriendly to be deployed on a regular device with moderate computation capacity.

In [35], the author extended the signal's degree-of-freedom using a five-band RGBCO camera. Even though a performance gain in the estimation of PR/PRV is claimed, the cost and availability of a five-band camera restrict a wide adoption of the system. In [31, 53], the benefit to include the background information in a rPPG system is presented. The illumination change on the face is compensated by that of the background using an adaptive filter [31] or via a joint blind source separation scheme [53]. Such a system assumes a high correlation between variations of the background reflectance with the non-pulse reflectance on the face. This assumption might be true when the background is stationary and controlled yet might be violated when the background contains additional illumination sources or moving objects.

The subjects motion information is estimated from the video and exploited. When the camera sensor is fixed, and the subject exercises on the focal plane, the subject's face motion can be roughly estimated as the face motion trace in the video [54]. Note the motion signal investigated in [54] are mainly pulse-induced ballistocardiography (BCG) motion signal. The face motion signal discussed in this paper contains little pulse component as the BCG component becomes negligible when the subject motion is voluntary. This property enables the author in [10, 55] to filter the pulse signal from the motion corrupted rPPG measurement.

A sub-band based approach is proposed in [20], where the essence of the algo-

rithm is to perform a frequency-dependent POS pulse color mapping. Even though an increase of the measurement degree-of-freedom is claimed in the paper and the system performance improves in fitness scenario, there is no gain in terms of the information level, and the motion residue can still dominate over pulse component in the processed signal.

3.3 Methodology

3.3.1 Face ROI localization

A highly precise motion compensation is a crucial step toward generating a clean face color signal. We first use a DNN face detector trained with the Single-Shot-Multibox detector (SSD) [80] and a ResNet [81] framework to obtain a rectangular face detection region. The SSD-ResNet is adopted in our system as it has better detection result compared with the traditional method, for example, the Viola-Jones detector [68], especially for face profile. We then use the CSR-DCF algorithm [82] to track the face region. The implementation of the SSD-ResNet and CSR-DCF is obtained from the OpenCV library [83]. A facial region of interest is located with the facial landmarks estimated with an ensemble of regression trees [51]. We follow the ROI selection principles discussed in [84] to include the cheek and forehead regions of a face.

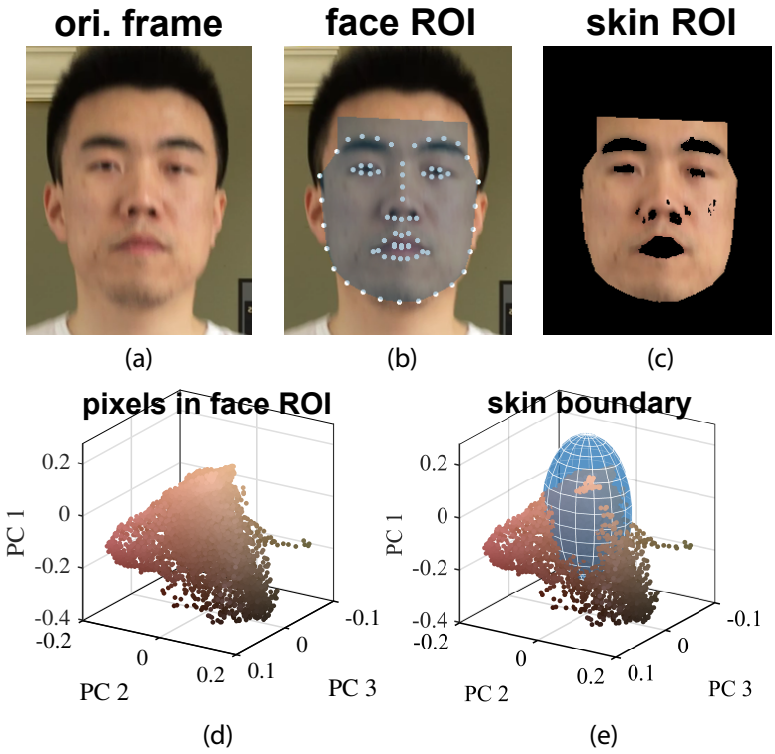


Figure 3.1: Face landmark localization and skin classification result example. (a) an example cropped video frame. (b) facial landmark localization result (blue dots) and the estimated face ROI (the transparent blue area). (d) Scatter plot of the skin pixels in face ROI in their top three dominating principle component directions. The dots' color is consistent with its original color in the frame. The scatter plot is overlaid with the boundary of the skin classifier in (e). (c) the corresponding detected skin pixels (in their original color) according to the classifier shown in (e).

3.3.2 Skin Tone Learning and Pruning

The idea of our proposed detection methodology is to learn the probability distribution of a skin pixel based on the samples collected from the facial ROI in the first few frames. Assuming the time-invariance of a person's intrinsic skin color and the surrounding illumination condition, we devise the skin detection method based on the learned parameters and the *a posteriori* (MAP) rule. Enlightened by the color space selection schemes [85] optimized for skin detection, we first map the pixel samples to the color space (R−G, R−B, Y−Cr, H), where H denotes the hue channel. Given a pixel random variable $\mathbf{s} \in \mathrm{R}^{4\times 1}$ from the face ROI, we make the following hypotheses:

Hypothesis H_0: $\mathbf{s}$ is a skin pixel.

Hypothesis H_1: $\mathbf{s}$ is a non-skin pixel.

Assume the *a priori* probability for the two hypotheses is $P(H_0) = p_0$ and $P(H_1) = 1 - p_0$. To capture the spatial variation of a subject's skin color on face, we model the conditional distribution of $\mathbf{s}$ under H_0 as a multivariate Gaussian distribution parameterized by the mean $\bar{\mathbf{s}}$ and covariance matrix Σ. We write the density function of $\mathbf{s}$ under H_0 as:

$$f_{\mathbf{s}|H}(\mathbf{s_i}|H_0) = \frac{1}{\sqrt{(2\pi)^4|\Sigma|}}\exp\left(-\frac{1}{2}(\mathbf{s_i} - \bar{\mathbf{s}})^{\intercal}\Sigma^{-1}(\mathbf{s_i} - \bar{\mathbf{s}})\right) \tag{3.1}$$

We model the conditional distribution of $\mathbf{s}$ under H_1 as a uniform distribution assuming that a non-skin pixel in the scene is equally likely to be any specific color.

The density function of $\mathbf{s}$ under H_1 is:

$$f_{\mathbf{s}|H}(\mathbf{s_i}|H_1) = \alpha, \tag{3.2}$$

where α is the model parameter satisfying the unitarity rule of probability.

According to the MAP rule [86], the decision is specified as:

$$P(H_1|\mathbf{s} = \mathbf{s}_i) \underset{H_0}{\overset{H_1}{\gtrless}} P(H_0|\mathbf{s} = \mathbf{s}_i), \tag{3.3}$$

which leads to the log likelihood ratio test as:

$$\log\left(\frac{f_{\mathbf{s}|H}(\mathbf{s_i}|H_1)}{f_{\mathbf{s}|H}(\mathbf{s_i}|H_0)}\right) \underset{H_0}{\overset{H_1}{\gtrless}} \log\left(\frac{p_0}{p_1}\right). \tag{3.4}$$

Substituting the conditional density functions (3.1) and (3.2) into (3.4), we have

$$(\mathbf{s_i} - \bar{\mathbf{s}})^\intercal \Sigma^{-1} (\mathbf{s_i} - \bar{\mathbf{s}}) \underset{H_0}{\overset{H_1}{\gtrless}} \phi, \tag{3.5}$$

where $\phi = 2\log p_0 - 2\log\left((2\pi)^2(1-p_0)\alpha\sqrt{|\Sigma|}\right)$. We observe from (3.5) that the skin detection boundary is defined by a hyper-ellipsoid shaped iso-density surface centered at $\mathbf{s_i}$. At this end, we introduced the decision rule for the skin detection based on the statistically modeling of both the skin pixels and non-skin pixels. We discuss next the estimation of the skin color model parameters $\bar{\mathbf{s}}$ and Σ.

3.3.2.1 Learning of the Skin Model Parameters

Letting $\mathbf{S} \in \mathrm{R}^{4\times N}$ denotes the sample pixels from the face ROI, our learning objective is to estimate the skin model parameter $\bar{\mathbf{s}}$ and Σ from $\mathbf{S}$. The direct use of the maximum likelihood estimator, i.e., the data mean and the data variance to estimate $\bar{\mathbf{s}}$ and Σ generates biases. This is because the non-skin pixels in the sample collection $\mathbf{S}$ does not satisfy the same distribution with the skin pixels.

To address this problem and exclude the negative effect from the non-skin pixels, we estimate the model parameters by iteratively excluding out a small amount of non-skin pixels. Specifically, in each iteration, we estimated $\bar{\mathbf{s}}$ and $\mathbf{V}$ using the data mean and data variance. We then compute the conditional density values according to (3.1) for each sample and discard 5% of the samples with the least probability values. After several iterations, the non-skin pixels will be all discarded and the estimates of the skin pixel distribution parameters become unbiased. The success of the exclusion of the non-skin pixels is based on the fact that most of the samples in the facial regions are skin pixels, and the initial estimate of $\bar{\mathbf{s}}$ will be closer to the cluster of the skin pixels rather than sparsely-distributed non-skin pixels.

In Fig. 3.1, we show one example of the parameter learning and the skin detection result. We represent Σ via its eigen structure as $\Sigma = \mathbf{EVE}^{\intercal}$, where $\mathbf{V} \in \mathrm{R}^{4\times 4}$ is a diagonal matrix with the descending diagonal entries as the eigen values of Σ, and $\mathbf{E}$ is a unitary matrix with each column as a eigen vector of Σ. We call the new random variables $\tilde{\mathbf{s}} \triangleq (\mathbf{s} - \bar{\mathbf{s}})^{\intercal}\,\mathbf{E}$ as the principle components of $\mathbf{s}$. In Fig. 3.1(d),

we display the pixel samples in terms of the first three components of $\tilde{\mathbf{s}}$ for better visualization. Note that the principle components are mutually independent based on the property of the multivariate Gaussian distribution, and the eigen vectors in $\mathbf{E}$ represent the principal axes of the ellipsoid $\{\mathbf{s_i}|(\mathbf{s_i}-\bar{\mathbf{s}})^\intercal\Sigma^{-1}(\mathbf{s_i}-\bar{\mathbf{s}})=\phi\}$, which is shown in Fig. 3.1(e) in transparent blue. The final skin detection result in face ROI is visualized in Fig. 3.1(c). Note that the detection process retains most of the skin pixels and successfully rejects most of the non-skin pixels and the pixels that are dominated by the specular reflection.

Algorithm 1: Skin Pixel Learning

procedure SKIN_CLUSTER($\mathbf{S}, I$) ▷ I: number of iteration.
  $\mathbf{S}^{(1)} \leftarrow \mathbf{S}$
  $N^{(1)} \leftarrow N$ ▷ $N^{(i)}$: number of inliers.
  for $i \leftarrow 1, I$ **do**
    $\bar{S}^{(i)} \leftarrow \frac{\sum_{j=1}^{N^{(i)}} S_j^{(i)}}{N^{(i)}}$, $\mathbf{V}^{(i)} = \frac{(\mathbf{S}^{(i)}-\bar{S}^{(i)})(\mathbf{S}^{(i)}-\bar{S}^{(i)})^\intercal}{N^{(i)}}$
    $D \leftarrow \mathrm{diag}((\mathbf{S}^{(i)}-\bar{S}^{(i)})^\intercal \left(\mathbf{V}^{(i)}\right)^{-1}(\mathbf{S}^{(i)}-\bar{S}^{(i)}))$
    $N^{(i+1)} \leftarrow 95\%N^{(i)}$
    $D \leftarrow \mathrm{sort}(D, \mathrm{ascent})$
    $th_D \leftarrow D_{N^{(i+1)}}$
    $\mathbf{S}^{(i+1)} \leftarrow 95\%$ samples from $\mathbf{S}^{(i)}$ which are smallest with respect to D.
  end for
  return th_D, $\bar{S} \leftarrow \bar{S}^{(I)}$, $\mathbf{V} \leftarrow \mathbf{V}^{(i)}$
end procedure

3.3.3 Motion Compensation via joint-channels NLMS

Once the skin pixels are detected in each frame, a temporal RGB sequence $\tilde{\mathbf{C}}(t)$ is generated by spatially averaging the RGB values of the detected skin pixels and temporally normalized in each color channel. $\tilde{\mathbf{C}}(t)$ is then linearly mapped to

a specific color direction in the RGB space to generate a 1-D pulse signal. The pulse color mapping schemes have been extensively investigated in [16] and [20]. We note that the design of the pulse color mapping algorithms discussed in this paper is not within the contributions of our work, whereas different pulse color mapping approaches [16, 20, 29, 44] are implemented and compared in the Section 3.4.

Without loss of generosity, we assume the face color signal $\tilde{\mathbf{C}}(t)$ is mapped to the POS direction [16]. We denote the projected 1-D processed signal as $c_{pos}(t)$. According to (1.5), we have

$$c_{\text{pos}}(t) = \mathbf{p}^{\intercal} \cdot \tilde{\mathbf{C}}(t) = \overbrace{\mathbf{p}^{\intercal} \cdot \mathbf{u_p}' \cdot p(t)}^{\textbf{Pulse}} + \overbrace{\sum_{k=1}^{K} \mathbf{p}^{\intercal} \cdot \mathbf{u_{m,k}} \cdot m_k(t)}^{\textbf{Motion Residue}}, \tag{3.6}$$

where $\mathbf{p} \in \mathrm{R}^{3\times 1}$ denotes the mapping vector of POS algorithm. The motion residue term in Eq. (3.6) is negligible when the illumination source is single, as the POS direction is orthogonal to the color direction of the motion-induced intensity change, and the specular change is suppressed via "alpha tuning" [29]. However, if the video is captured in an uncontrolled environment, the motion residue term is often nonnegligible, and sometimes can be more significant than the pulse term. To address this problem and further suppress the motion term in (3.6), two adaptive filter frameworks are deployed and compared, and the face motion signals in both horizontal and vertical directions are estimated and used to approximate and mitigate the motion interference component in (3.6). We denote the estimated face motion sequence in horizontal and vertical directions as $m_x(t)$

and $m_y(t)$. The pipelines of the two adaptive filter frameworks are shown in Fig. 3.2. In the first scheme, we treat $c_{\text{pos}}(t)$ as the filter's desired response at time instance t; $\mathbf{m}(t)$, a $2M$-by-1 tap vector at time t as the input, where $\mathbf{m}^\intercal(t) = [m_x(t-M+1), m_x(t-M+2), ..., m_x(t), m_y(t-M+1), m_y(t-M+2), ..., m_y(t)]$; and $\tilde{c}_{\text{pos}}$ as the output of the system and also the error signal. The estimated tap-weight vector of the transversal filter is denoted as $\hat{\mathbf{w}}(t)$, and weight control mechanism follows the Normalized Least Mean Square (NLMS) algorithm [71] as below

$$\tilde{c}_{\text{pos}} = c_{\text{pos}} - \hat{\mathbf{w}}^\intercal(t) \cdot \mathbf{m}(t), \tag{3.7}$$

$$\hat{\mathbf{w}}(t+1) = \hat{\mathbf{w}}(t) + \frac{\mu}{\|\mathbf{m}(t)\|^2}\mathbf{m}(t) \cdot \tilde{c}_{\text{pos}}, \tag{3.8}$$

where μ denote the adaptation constant, which is normalized by the norm square of the input vector $\mathbf{m}(t)$. The NLMS filter model assumes a high linear correlation between the motion residue term $\sum_{k=1}^{K} \mathbf{p}^\intercal \cdot \mathbf{u}_{\mathbf{m},\mathbf{k}} \cdot m_k(t)$ with the motion tap vector $\mathbf{m}(t)$. Considering such linear relation between the two might be time variant, the NLMS filter is thus designed to track the system change. The second adaptive filter scheme is inspired by the framework in [87]. This time, two NLMS filters ran in parallel with the same desired signal c_{pos} and different input signals as $\mathbf{m}_\mathbf{x}(t) = [m_x(t-M+1), m_x(t-M+2), ..., m_x(t)]$ and $\mathbf{m}_\mathbf{y}(t) = [m_y(t-M+1), m_y(t-M+2), ..., m_y(t)]$, respectively. The two output filtered signals $\tilde{c}_{\text{pos},x}$ and $\tilde{c}_{\text{pos},y}$ are then fused when transformed into the frequency domain. We evaluate the system performance with arbitrary pulse color mapping and adpative motion filtering schemes in Section 3.4.

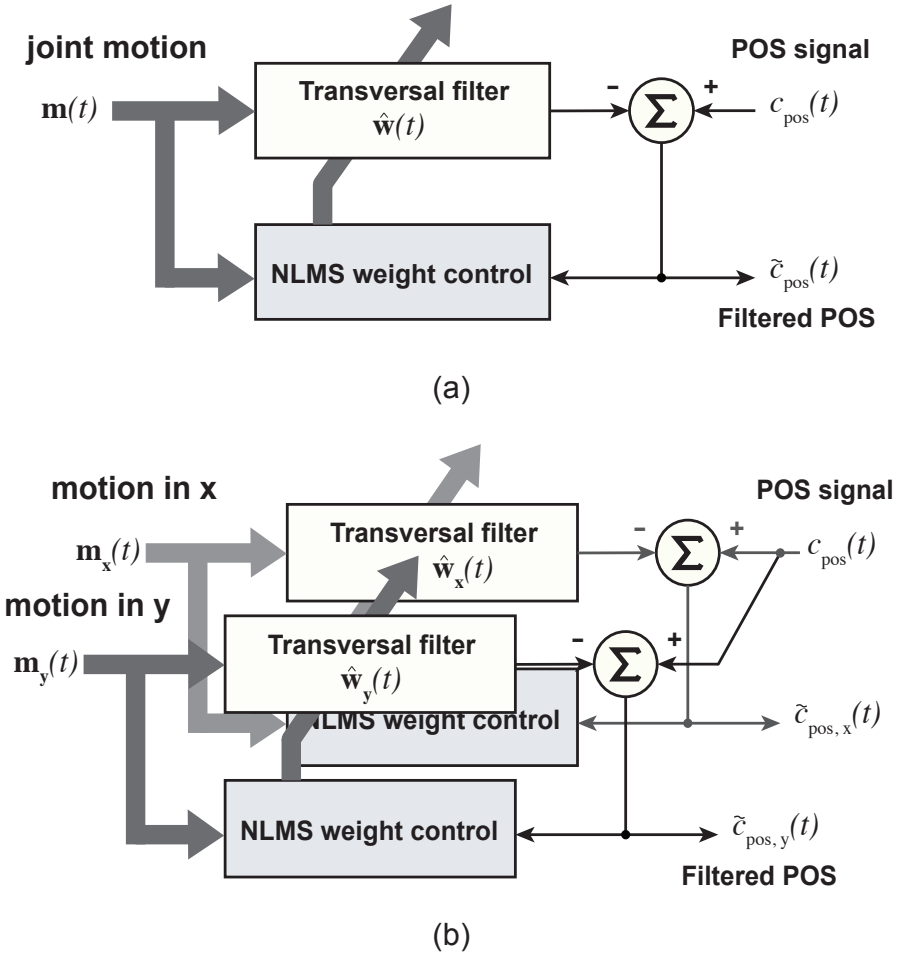


Figure 3.2: Two adaptive motion compensation filter frameworks: (a) NLMS-1Ch, (b) NLMS-2Ch.

3.3.4 Pulse Rate Tracking via Dynamic Programming

To this end, the signal quality is improved via precise face tracking, skin detection, pulse color mapping, and adaptive motion filtering. As did most of the prior arts [21, 29, 44], one might now assume temporal stationarity of the processed pulse signal and consider to estimate the instantaneous heart rate by mapping the 1-D processed signal to the frequency domain and searching the spectral peak within the normal human heart rate range (50-240 bpm). Such highest-peak estimation method can output accurate result when the SNR is high, but may frequently generate outliers when SNR drops.

Note that for a healthy human being, two consecutive heart/pulse rate measurements may not deviate too much from each other. We exploit this heart rate continuity property, and track people's heart rate by searching for the dominating frequency trace appearing in the signal's spectrogram image. The process of the tracking algorithm is briefly described below.

Let $\mathbf{Z} \in \mathbb{R}_{+}^{M \times N}$ be the magnitude of a processed signal spectrogram image, which has N discretized bins along the time axis and M bins along the frequency axis. We model the change of the frequency value between two consecutive bins at $n-1$ and n as a one step discrete-time Markov chain, characterized by a transition probability matrix $\mathbf{P} \in \mathbb{R}^{M \times M}$, where $P_{m'm} = P(f(n) = m | f(n-1) = m')$, $\forall m, m' = 1, ..., M$, and $\forall n = 2, ..., N$. The regularized single trace frequency tracking problem is formulated as follows

$$\mathbf{f}^* = \underset{\mathbf{f}}{\operatorname{argmax}} \quad E(\mathbf{f}) + \lambda P(\mathbf{f}), \tag{3.9}$$

The regularized tracking problem in (4.3) can be solved efficiently via dynamic programming. First, we iteratively compute an *accumulated regularized maximum energy map* $\mathbf{G} \in \mathbb{R}_+^{N \times M}$ column by column for all entries (m, n) as follows

$$\mathbf{G}(m,n) = \mathbf{Z}(m,n) + \max_{m'=1,\ldots,M} \{\mathbf{G}(m', n-1) + \lambda \log P_{m'm}\}. \tag{3.10}$$

After completing the calculation at column $n = N$, the maximum value of the Nth column is denoted as $f^*(N)$. Second, we find the optimal solution by backtracking from the maximum entry of the last column of the accumulated map $\mathbf{G}$. Specifically, we iterate n from $N-1$ to 1 to solve for $f^*(n)$ as follows

$$f^*(n) = \underset{f(n)}{\operatorname{argmax}} \quad (f(n), n) + \lambda \log P_{f(n)f^*(n+1)}. \tag{3.11}$$

Note that we can avoid transitions from state m' to state m by setting $P_{m'm} = 0$, as the regularized term would penalize the total energy to $-\infty$. If we assume uniform random walk transitions, i.e., $P_{mm'} = \frac{1}{2k+1}$, $|m' - m| \leq k$, then problem (4.3) is degenerated to the *seam carving* problem defined in [73], and in this case the value λ does not affect the solution.

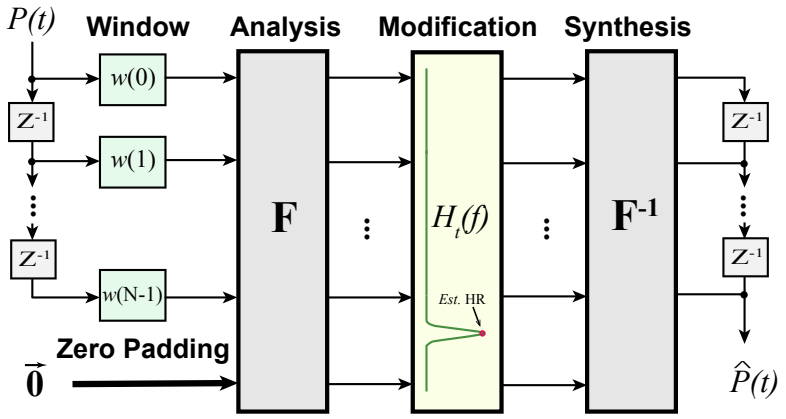


Figure 3.3: Adaptive filter bank design diagram for the pulse signal filtering. The modification layer rejects the frequency bands which are beyond a small frequency scope centered at the instaneous pulse rate tracking result at time instance t. Without the modification layer, the system perfectly reconstructs $P(t)$ when $\sum_{i=0}^{N-1} w(i) = 1$ [88].

3.3.5 Adaptive Filter Bank Modification and PRV Analysis

The analysis of the PRV requires clean pulse signal so that the temporal peak features are distinguishable along time to estimate the inter beat intervals. An LTI bandpass filter which rejects the frequency component outside the normal heart rate range may fail the task as noise power may be still significant within the pass band.

This problem is address in [42] with a design of the adaptive bandpass filter. A refined narrow passband is selected around the highest peak within the range of normal heart rate, and the final filtering result is obtained via the overlap add method [89]. This adaptive filter design improve the system performance, but it might suffers from the following three problems: First, the number of the frequency bins inside the heart rate range is relatively sparse, and the low frequency resolution makes the frequency selection ineffective; Second, in a low SNR scenario, the highest peak appears in the spectral might not represent the true pulse rate. A bandpass

filter based on a biased heart rate estimate generates completely adverse effect to amplify the noise and suppress the pulse component. Third, the rectangle analysis window is considered as less effective in terms of the band pass effect in the filter bank structure.

To address these problems, we exploit the heart rate estimates in Section 3.3.4 to precisely filter out all possible noise and interference existing in the processed pulse signal in a perfect reconstruction filter bank framework as discussed in [88]. The proposed filter bank system is shown in Fig. 3.3. Assume the processed 1-D pulse signal is $p(t)$. At the time instant t, the m-th ($m = 0, ..., M-1$) sub-band response is

$$P_t(m) = \sum_{\tau=t-(T-1)}^{t} w(t-\tau)p(\tau)e^{-j2\pi m(\tau-M+1)/M}, \tag{3.12}$$

where the $w(t)$ denotes a causal Hamming window, i.e.,

$$w(t) = \begin{cases} 0.54 - 0.46\cos(2\pi t/T), & 0 \le t \le T-1 \\ 0, & \text{otherwise.} \end{cases} \tag{3.13}$$

Note that in Eq. 3.12 and in Fig. 3.3, we pad a sequence of zeros at the end of the windowed time sequence $p_t(\tau) = p(\tau)w(t-\tau)$ to increase the number of the subbands. The number of appended zeros, and thus the DFT length M must be large enough to accommodate the spectrum modification which is to be made. Note that the zeros padding operation will not bring additional information to the system, and the perfect reconstruction property is maintained.

To achieve the bandpass filtering goal, we add a spectrum modification layer

Figure 3.4: Sample frames in fitness video dataset with different types of motions: (a) elliptical machine, (b) treadmill, (c) rowing machine, (d) head rotation and talking motion.

in between the analysis and the synthesis filter bank to suppress the noise outside the pulse rate frequency range. The modification is described as follows:

$$P'_t(e^m) = P_t(e^m) \cdot H_t(m), \quad m = 0, ..., M-1, \tag{3.14}$$

where $H_t(m)$ is a normalized Gaussian-shaped function with mean set as the current PR estimate. For a fixed value of t, $P_t(e^{j\omega_m})$ can be viewed as the normal Fourier transform of the modified sequence $p_t(\tau)$.

3.4 Experimental Results

3.4.1 Experiment Setups

Our proposed method was evaluated on a self-collected fitness exercise dataset to demonstrate the efficacy of the PR(V) estimation on dealing with motions. The

dataset has 25 videos in which 10 contain human motions on an elliptical machine, 10 contain motions on a treadmill, 1 contains motions on a rowing machine, and 4 videos contains rigid head motions and non-rigid face motions, for example, talking. The experiment setups are detailed as below.

Environment In order to test the robustness of the system, the experiment was conducted in three uncontrolled rooms. The active illumination sources only involve the existing lighting equipment in the scene. No additional illumination equipment or backdrop were placed during the recording. The presence of other subjects exercising or entering the scene is possible, as no regulation is placed to restrict people from entering the room.

The first 20 videos involving elliptical machine and treadmill motions were captured in a regular apartment gym room. The room was well-lit with several over-the-top florescent lights and with diffuse daylight passing into the gym through glass walls. The video containing rowing motions was captured in an athletic training room, with only over-the-top florescent lights. The videos with no body motions was captured in a regular lab with two over-the-top and one frontal florescent lights.

Devices and Reference Signal The first 20 videos, the rowing video, and the videos containing no body motions were captured by the rear camera of a iPhone 6s mobile camera, the rear camera of a iPhone X mobile camera, and a webcam (Model: Logitech C922x Pro Stream), respectively. The shutter speed was kept constant and the face was kept in focus at all time. During the recording of the first

20 videos and the rowing video, the heart rate of the subjects was simultaneously monitored by an electrocardiogram (ECG)-based chest belt (Model: Polar H7) for reference. The reference pulse signals of the last 4 videos were recorded with a pulse oximeter (Model: Contec CMS50E). Note that the raw PPG waveform is available with the pulse oximeter, but only the HR data is provided by the ECG chest belt.

Placement of the Sensors The mobile camera was affixed on a tripod or held by the hands of a person other than the test subject. The camera is placed in front the subject face at a distance of about 1 meter away at approximate same height with the subject's face during the recording of the first 20 video. The camera is placed in front of the subject face at a 45 degree angle when the rowing video was captured. The webcam was place in front of the subject in a distance of about half meter away. The ECG chest belt was correctly worn underneath the subject's cloth, and the sensor was in direct contact with the subject's chest skin. The pulse oximeter was worn at the subject's index finger of the right hand.

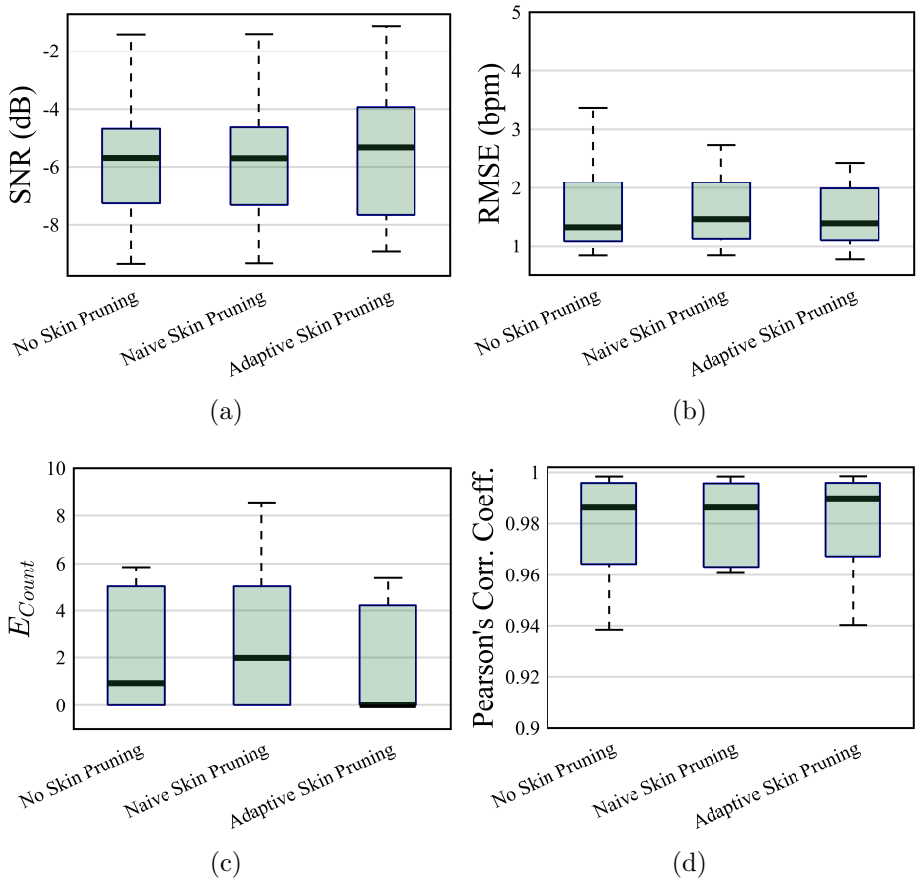


Figure 3.5: The boxplots of the system performance using different skin pruning schemes in terms of SNR (a), E_{RMSE} (b), E_{count} (c), and PCC (d). The pulse color mapping and motion compensation schemes are PCA and NLMS-1Ch, respectively.

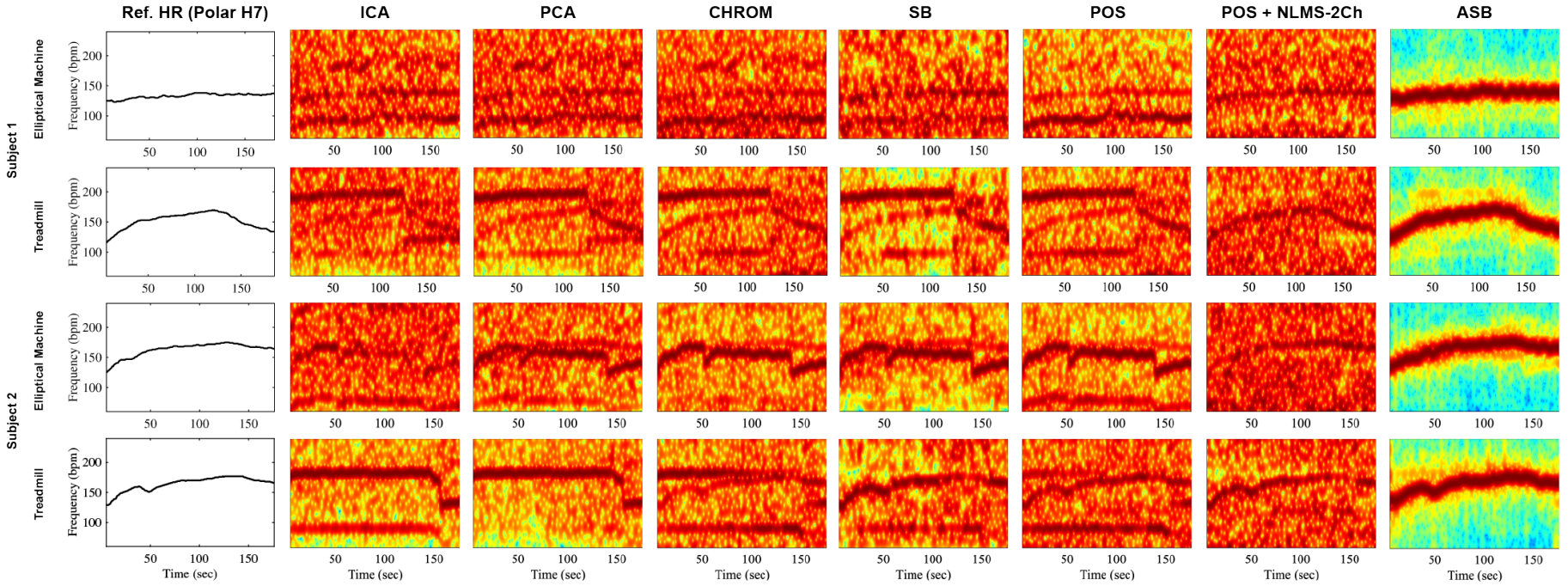


Figure 3.6: Four qualitative comparison results between different color mapping algorithms and the motion compensation result for NLMS-1Ch or NLMS-2Ch schemes. (Column 1) The reference HR measured by ECG based chest belt. (Column 2-8) Spectrograms of the extract pulse signal using different algorithms. (Column 9) Spectrograms of the filtered pulse signal using the adaptive filter bank with POS signal as input.

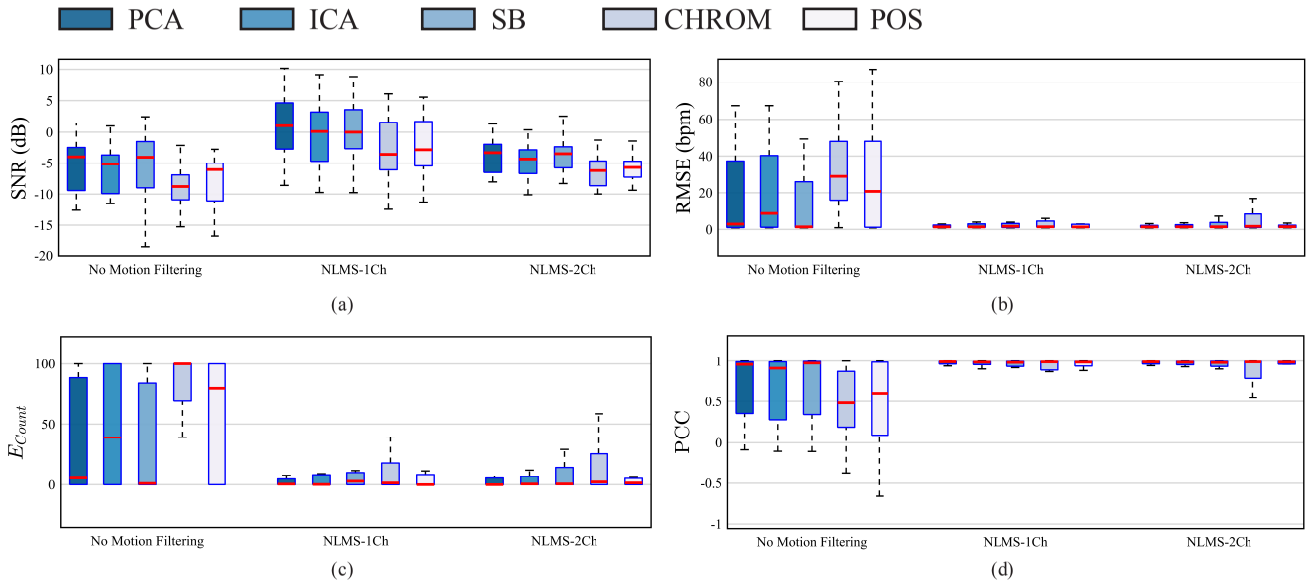


Figure 3.7: The system quantitative performance results with boxplots. The evaluation metrics are SNR (a), E_{RMSE} (b), E_{count} (c), and PCC (d). In each subplot (a-d), the result is group by three motion compensation schemes in x-axis: (from left to right) No motion filtering, NLMS-2Ch, NLMS-1Ch. In each group of motion compensation schemes, five pulse color mapping schemes are compared. They are (from left to right) PCA, ICA, SB, CHROM, and POS.

		SNR (dB)	PCC	p-value	E_{count} (%)	E_{rate} (%)	E_{RMSE} (bpm)
No Motion Filtering	PCA	-5.636 (4.019)	0.686 (0.431)	0.000 (0.002)	42.791 (46.426)	12.150 (15.321)	17.993 (20.875)
No Motion Filtering	ICA	-6.365 (3.785)	0.618 (0.422)	0.006 (0.026)	50.600 (44.680)	13.848 (15.216)	20.710 (20.567)
No Motion Filtering	SB	-4.555 (4.694)	0.770 (0.351)	0.001 (0.006)	33.260 (42.481)	9.157 (13.764)	14.076 (19.507)
No Motion Filtering	CHROM	-8.277 (3.185)	0.429 (0.401)	0.049 (0.192)	83.536 (30.210)	22.859 (15.754)	33.863 (20.949)
No Motion Filtering	POS	-7.859 (3.794)	0.521 (0.471)	0.043 (0.188)	56.724 (42.632)	14.766 (15.037)	22.783 (20.839)
NLMS-2Ch	PCA	**0.308** (4.851)	0.900 (0.244)	0.000 (0.000)	8.326 (22.621)	2.731 (8.060)	4.125 (10.449)
NLMS-2Ch	ICA	-0.753 (5.702)	0.886 (0.208)	0.000 (0.000)	13.614 (29.549)	4.499 (10.602)	6.649 (14.043)
NLMS-2Ch	SB	**0.927** (4.866)	0.925 (0.207)	0.008 (0.036)	10.080 (22.002)	3.395 (10.438)	5.232 (14.191)
NLMS-2Ch	CHROM	-1.230 (4.817)	0.884 (0.284)	0.000 (0.001)	12.279 (29.510)	5.025 (12.695)	7.244 (16.965)
NLMS-2Ch	POS	-1.329 (4.953)	**0.935** (0.155)	0.000 (0.000)	11.935 (24.819)	3.612 (10.496)	5.559 (14.308)
NLMS-1Ch	PCA	-4.347 (2.907)	0.863 (0.349)	0.000 (0.000)	12.022 (29.017)	3.873 (10.998)	5.732 (14.625)
NLMS-1Ch	ICA	-4.895 (3.311)	0.868 (0.329)	0.048 (0.210)	11.857 (26.211)	3.800 (10.862)	5.824 (14.489)
NLMS-1Ch	SB	-3.026 (2.871)	0.925 (0.179)	0.000 (0.000)	**7.152** (11.829)	**1.291** (1.433)	**2.711** (3.471)
NLMS-1Ch	CHROM	-4.555 (4.694)	0.770 (0.351)	0.001 (0.006)	33.260 (42.481)	9.157 (13.764)	14.076 (19.507)
NLMS-1Ch	POS	-5.133 (2.297)	**0.929** (0.235)	0.000 (0.001)	**4.058** (8.335)	**0.968** (0.751)	**1.804** (1.297)

Table 3.1: The system performance of the pulse rate estimation in terms of sample mean and standard deviation (in paranthesis) of SNR, PCC, p-value, $E_(count)$, E_{rate}, and E_{RMSE}. Difference combinations of motion compensation and pulse color mapping schemes were evaluated. The top two performed combinations are highlighted using bold red and bold blue respectively in terms of each metric.

Participants Two male Asian subjects are involved in the experiment. The skin tone of both subjects is classified as Skin-type III according to the Fitzpatrick skin scale [42,74]. Among all the videos in the dataset, 5 treadmill videos and 5 elliptical machine videos belong to one subject. The rest 15 belong to the other. Based on the most recent medical examination results, none of the subject was diagnosed with any known CVDs or pulmonary diseases.

Parameter Settings The following parameters are used unless otherwise stated:

1. Each video lasts about 3 minutes.

2. The frame rates for the videos captured by iPhone 6s, iPhone X, and Logitech webcam are about 30Hz, 30Hz, and 24Hz, respectively. The resolutions are 1280×720, 1920×1080, and 1920×1080. The averaged bit rates are 6MB, 8MB, and 13MB per second. The video codecs are AVC (H.264), HEVC (H.265), and AVC (H.264) respectively.

3. The threshold for the likelihood of the face DNN face detector is set as 0.5. The number of iterations during the skin tone learning is set as 4. The tap number for joint-channels NLMS is 8, and the NLMS learning rate is 0.1.

4. The spectrum analysis window length was set to 10 secs with 98% overlap. A Hamming window is applied in each analysis window, and the number of frequency bins in the normal PR range (50 to 240 bpm) was set as 1024 via padding zeros at the end of the analysis signal sequence. The transition probability model used in the frequency tracking algorithm is a uniform random

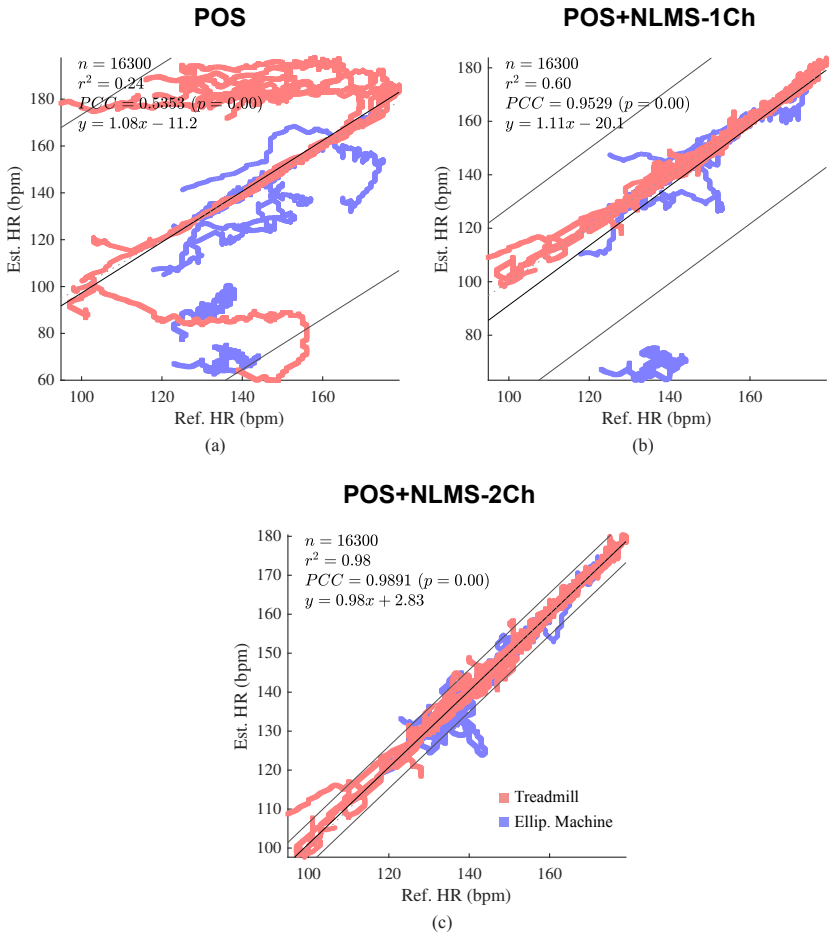


Figure 3.8: Correlation plots of the reference HR and the estimated PR using POS+No motion filtering (a), POS+NLMS-2Ch (b), and POS+NLMS-1Ch (c). In each subplot, a linear model $y\ x + 1$ is fitted according to the samples. Statistics about the correlation are listed in the left top corner of each subplot, where n denotes the number of the estimates; r^2 denotes the r-square statistics of the linear model; PCC denote the Pearson's correlation coefficient of the reference HR and the estimated PR.

walk model with $k = 1$ bpm.

5. In the Adaptive filter bank system, N is selected such that the effective signal length is about 7 secs. The length of the padded zeros sequence is selected so that the number of frequency channel is 1024 for a 30 Hz frame rate video. The standard deviation for the modification kernel is set as 1 bpm.

3.4.2 Metrics of Performance Evaluation

SNR of the processed rPPG pulse signal The same SNR metric used in [20, 29, 42] is adopted in this paper. The value of this metric indicated the pulse signal quality using difference signal processing techniques. The SNR metric is computed on each power spectrum frame out of the spectrogram. The SNR metric is defined as the ratio between the energy around the first two harmonics of the reference PR and the remaining energy of the power spectrum:

$$\mathrm{SNR} = 10 \log_{10} \left(\frac{\sum_{f \in \mathcal{F}} S_n(f) P(f)}{\sum_{f \in \mathcal{F}} (1 - S_t(f)) P(f)} \right), \tag{3.15}$$

where $S_n(f)$ is a defined binary window to select the frequency bins belong to the two harmonics region; $P(f)$ is the power spectrum of the pulse signal; set $\mathcal{F} \triangleq \{f | 50\text{bpm} \leq f \leq 240\text{bpm}\}$

PR Estimation Accuracy Three well-adopted metrics for pulse rate estimation accuracy were adopted in this study. They are specified as below:

1. Root mean squared error:

$$E_{\text{RMSE}} = \sqrt{\frac{1}{N}\sum_{n=1}^{N}(\hat{f}(n) - f(n))^2}, \tag{3.16}$$

2. Error rate:

$$E_{\text{rate}} = \frac{1}{N}\sum_{n=1}^{N}\left|\hat{f}(n) - f(n)\right| \Big/ f(n), \tag{3.17}$$

3. Error count ratio:

$$E_{\text{count}} = \frac{\left|\{n : \left|\hat{f}(n) - f(n)\right| \Big/ f(n) > \tau\}\right|}{N}, \tag{3.18}$$

4. Pearson's correlation coefficient:

$$\text{PCC} = \frac{\sum_{n=1}^{N}(\hat{f}(n) - \bar{\hat{f}})(f(n) - \bar{f})}{\sqrt{\sum_{n=1}^{N}(\hat{f}(n) - \bar{\hat{f}})^2}\sqrt{\sum_{n=1}^{N}(f(n) - \bar{f})^2}}, \tag{3.19}$$

where $|\{\cdot\}|$ denotes the cardinality of a countable set; N denote the total number of the PR estimate; $\hat{f}(n)$, $f(n)$, $\bar{f}$, and $\bar{\hat{f}}$ denote the PR estimate at time instance n, the ground-truth PR at time instance n, the average PR estimate, and the average reference PR. τ was chosen to be 0.03 empirically, determined from the spread of the frequency components.

PRV Estimation Accuracy Three PRV parameters estimated from the face videos were evaluated in this experiment using the reference finger-tip PPG signal[1].

[1]Minor difference between the PRV parameters measured from two different body cites is possible, as the the variability of the pulse transit time might be different. In this study, such difference

They are

1. normal to normal interval (NN): the time length between adjacent systolic pulse peaks of the (r)PPG signal. NN reflects the instaneous pulse rate. Almost all other PRV parameters are derived from NN.

2. Standard deviation of NN intervals (SDNN): a moving standard deviation calculation with the NN sequence. Each standard deviation window covers 60 consecutive NN intervals, and the window overlap is 10 NN intervals. SDNN is a time domain PRV parameter. It captures the variation of the NN around the mean statistic [90].

3. Low frequency to high frequency ratio (LF/HF): The NN intervals are first transformed to frequency domain via the fast Fourier transform. The ratio between the power of LF (0.04 0.15 Hz) to HF (0.15 0.4 Hz) was then estimated. LF/HF is a frequency domain PRV parameter to evaluate the sympatho-vagal balance controlling the HR [91].

All three PRV parameters are estimated from the rPPG signal and the reference PPG signal. The mean absolution difference is computed to measure the accuracy of the PRV estimation.

3.4.3 Performance of PR Estimation

Considering that the rPPG system consists of several layers of modules including the skin detection, color space mapping, NLMS-based motion compensation, we

is ignored.

perform a module-wise comparison to test the efficacy of each individual scheme.

Comparison of the Skin Pruning Schemes We first test the efficacy of the proposed skin pruning schemes by fixing the other modules of the system as discussed in 3.3 and evaluating the performance. We compared the proposed skin pruning scheme with the following two schemes:

1. No skin pruning. namely, no skin detection is performed and the face RGB value of certain frame is obtained via averaging on face ROI shown in Fig. 3.1(b).

2. Naive skin pruning [92]. A pixel is classified as a skin pixel if it lies inside a universal linear skin classifier.

The performance results in terms of the SNR, E_{RMSE}, E_{count}, and PCC is summarized in Fig. 3.5. We can see from the boxplots that the adaptive skin pruning scheme gives the best performance over the other two schemes.

Comparison of the Motion Compensation and Color Space Mapping Schemes We compare the system performance when different combinations of motion compensation and pulse color space mapping schemes are deployed. The compared adaptive filter based motion compensation schemes are no motion filtering, NLMS-1Ch, and NLMS-2Ch. The compared pulse color space mapping schemes are ICA [24], PCA [43], CHROM [29], SB [20], and POS [16]. Before performing the PCA or ICA pulse blind demixing, the face color signals are first detrended using the method introduced in [22], where the trend smooth regulator is set as $\lambda = 10$ for a 30 Hz frame rate video. Rather than selecting the demixed channel which outputs the

most "significant pulse peak" in frequency domain as the pulse channel, we select the one which generates the highest SNR when compared with the reference heart rate. This is because the assumption made in [43,44] that the dominating frequency components in the face color signal is pulse fails in a fitness scenario. Therefore, the result shown in this paper about the PCA and ICA algorithm is the best possible in terms of the channel selection.

A qualitative comparison result in the form of the spectrograms of the processed signals is visualized in Fig. 3.6. Compared with the reference HR traces depicted in the first column, we can observe that one or two motion frequency traces may present or even overshadow the pulse traces in the spectrograms if no adaptive motion compensation filtering operation is deployed (Column 2-6), whereas the NLMS-1Ch (Column 8) or NLMS-2Ch (Column 7) can effectively mitigate the motion interferences. In the last column, we show the spectrograms of the filtered POS signals using the adaptive filter bank system introduced in Section 3.3.5. Following the filtering instruction from the pulse rate estimates using the POS+NLMS-1Ch, the adaptive filter bank system is able to effectively filter out almost all the noise components, and the only trace stands out in the spectrogram belong to the pulse signal.

A quantitative comparison result is summarized in Table 3.4.1, where 15 combinations of the pulse color mapping and the motion compensation schemes are evaluated with five different metrics. The same statistics are also visualized in part using the boxplots in Fig. 3.7, and the correlation plots in Fig. 3.8. The key observations are summarized as follows:

Methods	NN	SDNN	LF/HF
LTI-BP [7]	0.028 (0.083)	0.031 (0.063)	0.511 (0.035)
ABP [42]	0.025 (0.038)	0.010 (0.009)	0.476 (0.038)
AFB (proposed)	**0.014** (0.026)	**0.005** (0.007)	**0.298** (0.117)

Table 3.2: The PRV estimation performance in terms of the sample mean and standard deviation (in parathesis) of the absolute error of NN, SDNN, and LF/HF using LTI-BP, ABP, and AFP (proposed), respectively.

- NLMS-2Ch scheme outputs the highest SNR, followed by NLMS-1Ch and no motion compensation. The NLMS-2Ch improves more than 5 dB in averaged SNR compared with no motion compensation using each best algorithm combination.

- The accuracy in term of the PR estimation is greatly improved when NLMS-2Ch or NLMS-1Ch is deployed. It is also interesting to note that the algorithm combination with the highest SNR does not lead to the highest accuracy of PR estimation. From Table 3.4.1, the NLMS-1Ch improves the POS algorithm by more than 50% in terms of E_{count} and the SB algorithm by more than 26%.

- Without motion compensation filtering, the SB algorithm outputs the best performance. This observation is consistent with the one in [20] that SB gives better result than other pulse color mapping algorithms in terms of the suppression of the motion components.

- The best PR estimation result comes from a combination of POS and NLMS-1Ch. In this system setup, the average RMSE can be as low as 1.8 bpm.

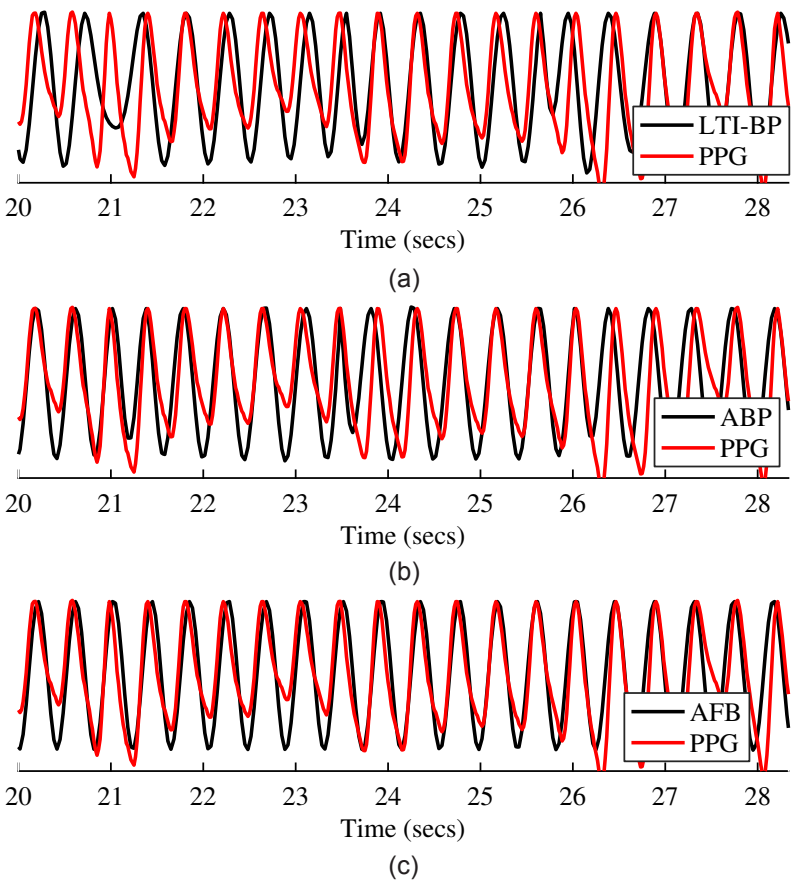


Figure 3.9: A qualitative comparison of the filtering results between LTI-BP [7] (a), ABP [42] (b), and AFB (proposed) (c). In each subplot, the black line denotes the filtered rPPG waveform, and the red line denotes the aligned reference finger-tip PPG signal waveform. Both signals are normalized according to their upper envelop for better visualization.

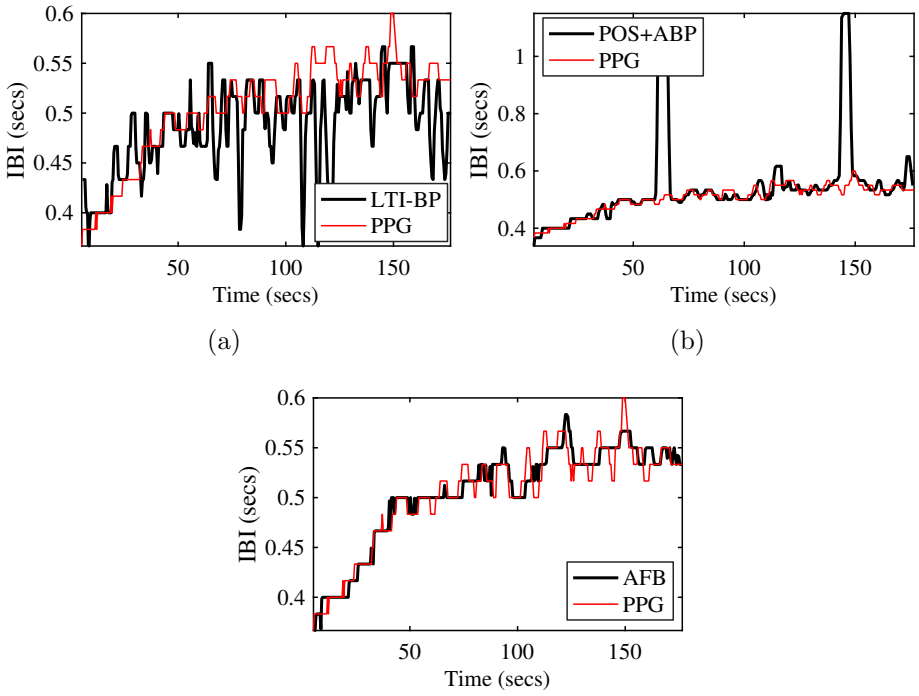


Figure 3.10: The instaneous inter beat intervals (IBI) estimated from the filtered rPPG signal (black line) versus the ones estimated from the finger-tip PPG signal. The three subplots differ in the rPPG filtering methods: LTI-BP (a), ABP (b), AFB (c).

3.4.4 Performance of PRV Estimation

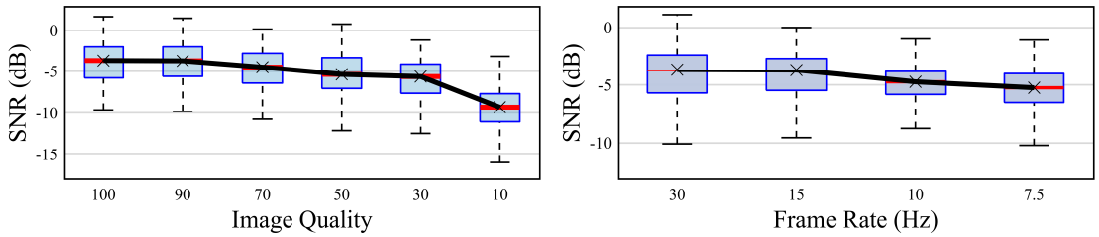
In this subsection, the effectiveness of the AFB algorithm in estimating the PRV is evaluated. Four videos where the subject is sitting in front of a webcam and a fluorescent light in a lab environment are used in this subsection. Before shooting each video, the subject performed 40 push ups to mimic the exercise process and to evaluate the post-exercise cardiovascular dynamics.

Two bandpass filtering operations are tested and served as the comparison groups. They are

- Linear time invariant bandpass filter (LTI-BP) [7]: A post-processed IIR bandpass filter with the passband set from the minimum pulse rate to the highest pulse rate.

- Adaptive bandpass filter (ABP) [42]: A FFT-based adaptive bandpass filter which passed the frequency components around the highest spectral peak within the normal human pulse rate range.

A qualitative comparison is visualized in terms of the waveform (Fig. 3.9) and the instaneous NN intervals (Fig. 3.10) using one of the face video. We summarize the quantative comparison results in Table 3.2. We observe that the proposed algorithm outperforms the other two state-of-the-art filtering schemes, and peaks are almost unbiased with the reference.



(a) (b)

Figure 3.11: Boxplots of the system response in terms of the pulse SNR when the image quality (a) and frame rate (b) is respectively changed.

3.5 Impact of Various Factors

3.5.1 Impact of Image Compression and Frame Rate

Image compression brings additional compression artifacts to the image and thus skin pixels, which distorts the pulse signal. The impact of the compression can be negligible when the compression rate is low, whereas the system performance may drop significantly when the image is highly compressed. To test the impact to the rPPG system when the source images of the face videos are compressed, an experiment is performed with one of a face video with elliptical machine motion. Without compression, the SNR is around -3.5 dB and the $E_{\text{count}} = 0\%$. We then applied different level of JPEG compression to the video by tuning the image quality parameter from 100 (no compression) to 10 (highly compressed).

The system performance in terms of the pulse SNR is shown in Fig. 3.11(a) using boxplots. A decreasing trend in the median SNR is observed when the image quality drops, which is consistent with our conjecture that a higher image compression leads to a lower pulse quality. Another observation is that the pulse SNR drops

only about 3 dB from 100 image quality to 30, while, in return, the compression operation saves more than 60% storage. A similar amount of pulse quality drop happens after the image quality is tuned from 30 to 10, which significantly distort the pulse signal.

3.5.2 Impact of Frame Rate

According to the Nyquist-Shannon sampling theorem, a minimum 8 Hz frame rate video is require to measure a person's PR via rPPG, if a person's maximum PR is 240 bpm (i.e., 4 Hz). From a statistical point of view, a higher frame rate reduces the estimation variance of a signal's power spectral density, thus improves the performance when PR is estimated.

An experiment is performed on the same video tested in Section 3.5.1 to evaluate the system response when we down-sampled the skin color sequence so that the frame rate equivalently drops to 1/2, 1/3, 1/4[2] of the original frame rate 30 Hz. The system performance in terms of the pulse SNR is shown in Fig. 3.11(b). Even though a system drop in terms of the median pulse SNR is observed when the frame rate decreases, we notice a SNR degradation of less than 2 dB from the original frame rate to only one fourth of the frame rate. The performance is comparable when half of the storage is saved by lower the frame rate by half, suggesting a possible temporal sampling redundancy for PR measurement.

[2]Note that the highest pulse rate in this experiment is lower than 200 bpm. Therefore the base pulse frequency component would not alias to a lower frequency even when the frame rate is 7.5 Hz (which is lower than the 8 Hz sampling requirement).

3.6 Discussion

3.6.1 The Detection of the Exercise

To this end, the exercise level of the subject is assumed to be known, and the motion compensation module (e.g., NLMS-1Ch) is turned on and off according to the algorithm needs. An accurate exercise detection method is thus needed to automatize the system and to avoid deploy the motion compensation module when the subject is in a rest condition as the head motion contains mostly BCG signal. This detection task can be accomplished via thresholding a statistics (e.g., standard deviation) of the subject's motion trace estimated from the video. When the distance between the subject and the camera sensor maintains, a rise in standard deviation of the motion suggests an increase of the intense level of the exercise.

3.6.2 The Evaluation of the Pulse Signal Quality

Even though we present extraordinary system performance in this paper, the pulse quality can still be poor due to various reasons. For a rPPG system aimed for commercialization, the evaluation of the pulse signal quality is necessary to demonstrate the estimation confidence to the users. The SNR statistic described in Section 3.4.2 offers a natural measure for this purpose, and the quality evaluation can be performed once the pulse rate is tracked.

3.7 Conclusion

In this chapter, we present another novel rPPG system that is robust for pulse rate and pulse rate variability extraction from fitness face video when the subject is exercising and the video contains large subject motions. We focus on designing an online learning scheme for precise subject- and scene-specific skin detection, and use motion information as a cue to adaptively remove the motion-induced artifacts in the corrupt rPPG signal. The computation complexity is greatly reduced compared with the optical-flow method, and the system is capable to run in realtime. An adaptive filter bank system is proposed to further clean the pulse signal so that the inter beat intervals and pulse rate variability are precisely estimated.

Chapter 4: Adaptive Multi-Trace Carving based on Dynamic Programming

4.1 Introduction

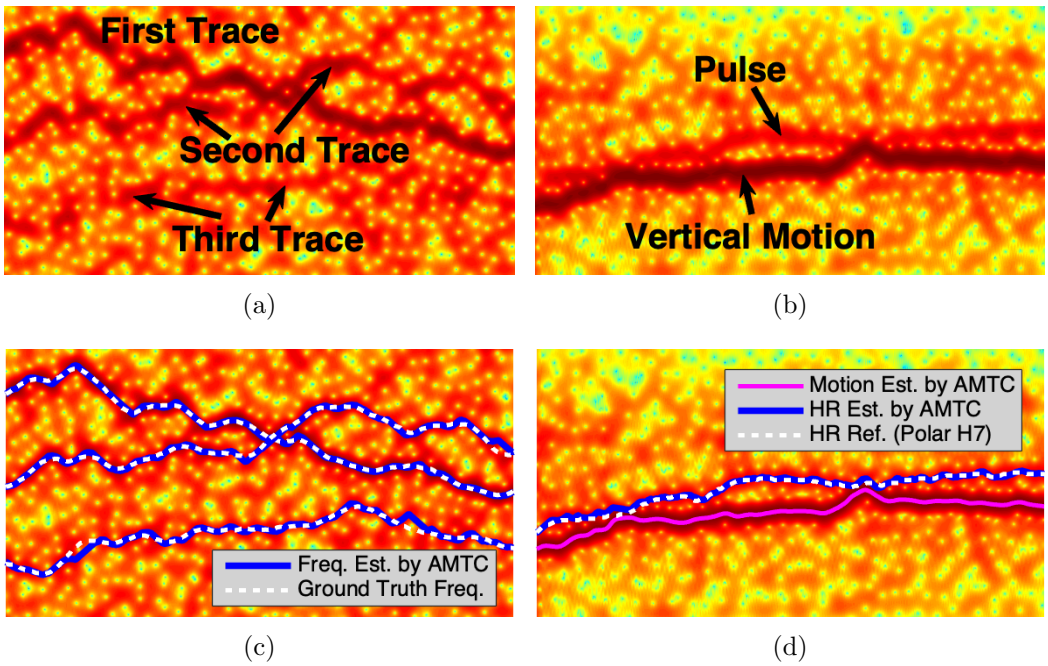
Many vital signs estimation problems often boil down to the problem of frequency extraction, such as PR estimation problem in the rPPG applications we discussed in Chs. 2 and 3. The problem becomes trivial when the SNR level is high, yet challenging in a low SNR condition, such as -10 dB.

As the extraction of frequency traces often plays a key role in the aforementioned applications, one needs to carefully answer the following questions before deploying a frequency estimator:

1. Can the frequency components be detected from the digital recording?

2. If a frequency component is detected, can the frequency be accurately estimated, especially in low signal-to-noise ratio (SNR) conditions?

Solving the above problems can be nontrivial due to the relatively low signal strength of the components-of-interest compared with those of other sources in the recordings. To successfully estimate the frequency of interest within the noisy signal, an algorithm must be robust under strong noise and has the capability to exclude strong



(a) (b)

(c) (d)

Figure 4.1: (a) Spectrogram image of a synthetic −10dB signal with three frequency components and (c) the same image overlaid with ground truth frequency components (white dashed line) and the frequency estimates using AMTC (blue line). (b) Spectrogram image of a remote-photoplethysmogram signal with weak heart pulse trace embedded in a strong trace induced by subject motion running on a elliptical machine [93] and (d) the same image overlaid with heart rate estimate (blue line) after compensating first trace estimate (magenta line) using AMTC. The estimation result is compared with the heart rate (white dashed line) simultaneously measured by a electrocardiogram based sensor.

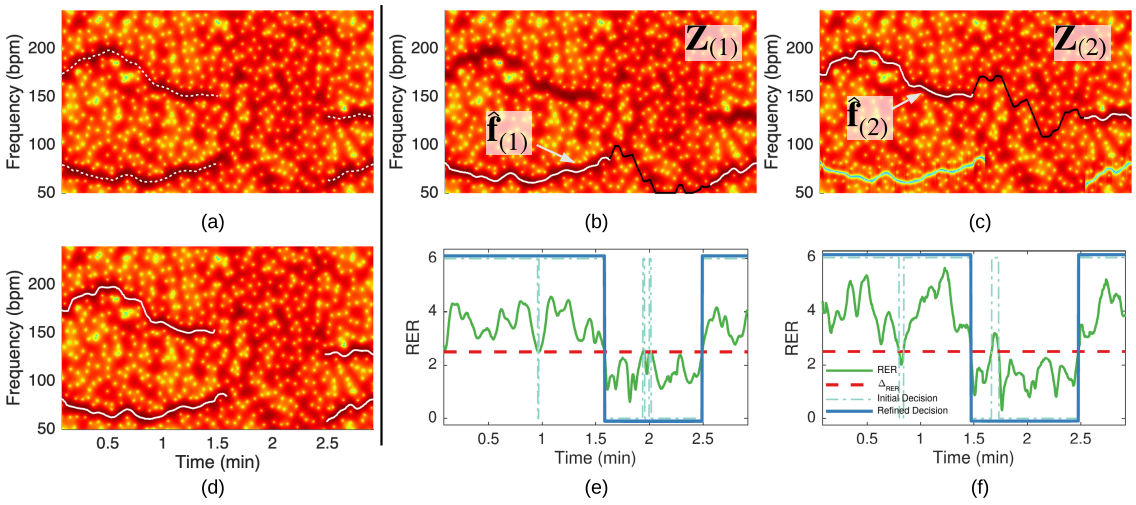


Figure 4.2: Example for offline AMTC estimation process: (a) spectrogram of a synthetic −8 dB signal with two frequency components. The unvoiced segment is from 1.5 to 2.5 min (white dots: ground truth); (b) first and (c) second trace estimates in voiced decision regions (white line) and unvoiced decision regions (black line) by AMTC; (e)–(f) test statistic RER and the corresponding voiced decision; (d) final trace estimates.

interference.

In this chapter, we exploit the signal's time-frequency feature map, such as spectrogram, to perform the frequency estimation. We propose a multiple frequency traces tracking and detection method based on iterative dynamic programming and adaptive trace compensation. Inspired by the seam carving algorithm for content-based aspect ratio adaptation of images [94], we treat finding a smooth frequency trace as finding the maximum energy trace in a spectrogram, with an additional regularization term that favors close frequency estimates in consecutive time bins. Such problem is efficiently solved using dynamic programming. In many applications, the presence of multiple traces within the frequency range of interest is possible. We propose an iterative frequency tracking method named *Adaptive Multi-Trace Carving* (AMTC) to track all candidate traces. We apply the proposed single frequency

tracking method to obtain the dominating frequency. We then compensate the previous trace energy at the end of each iteration to facilitate the estimation of the next trace. After several iterations, all traces within the frequency range of interest will be obtained. An efficient quasi-realtime algorithm is also proposed by utilizing the Markovian property of traces and introducing a bidirectional time window. We call it the online-AMTC. Note that we mainly consider spectrogram in this paper, while our proposed techniques can be applied to other visualizations for the signal for which the temporal tracking of signal traces is needed.

The contributions of this work are summarized as follows:

1. For the task of the frequency-based micro-signal parameter estimation, we proposed a robust frequency tracking and detection approach which could track multiple frequency traces in a very low (usually ≤ -10 dB) SNR condition accurately and efficiently. This method works in general for different levels of frequency variation and does not assume the availability of training data to learn prior knowledge of the signal characteristics.

2. We adapt the offline-AMTC algorithm into an efficient near-realtime implementation. We reduce the computational complexity with a queue data structure and maintain the performance compared with the offline version.

3. We conduct extensive experiments using challenging synthetic and real-world data. Several estimation methods initially proposed for other applications (*e.g.*, the pitch estimation) are implemented, re-trained (the factorial hidden Markov model based method [95]), and compared. The results demonstrate

that our approach outperforms other existing arts in terms of accuracy and efficiency.

4. We present a novel detection method based on the AMTC framework to accurately test the presence of trace and discuss other considerations using the approach, such as estimation of the number of frequency components and the benefit from human-in-the-loop involvement.

The rest of the chapter is organized as follows. In Section 4.2, the related work about the frequency tracking problems are discussed. In Section 4.3, we formulate the problem of single trace tracking and solve it using dynamic programming. In Section 4.4.1, we propose the offline multi-trace tracking method or the offline AMTC, based on a greedy search strategy. In Section 4.4.2, we present the online AMTC. In Section 4.5, we show that AMTC outperforms the state-of-the-art methods on both synthetic and real-world data. In Section 4.6, we evaluate the performance of AMTC in response to various factors. In Section 4.7, we discuss common problems which can be addressed with AMTC and the limitations of this algorithm. In Section 4.8, we conclude the paper.

4.2 Related Works on Frequency Tracking

Traditional frequency estimation algorithms are often applied individually to each temporal segment, assuming segment-wise signal stationarity. Subspace methods such as multiple signal classification (MUSIC) [96] and estimation of signal parameters via rotational invariance technique (ESPIRIT) [97] build pseudo power

spectra using parametric models of pure sinusoids. These frame-wise estimation algorithms cannot explicitly exploit the temporal correlation of neighboring segments and become less accurate as the SNR drops and frequently generate outliers.

The problem of tracking a single frequency component has been extensively studied. In [78], a sequential Monte Carlo method was proposed, and importance sampling was used to approximate the posterior distribution of each frequency state. However, without a backward smoothing procedure, the output tracking results tend to be inaccurate when substantial interference exists, and the resampling stage makes the algorithm time-consuming. In [98], a prior knowledge of trace dynamic was utilized, and the problem was formulated as a hidden Markov model (HMM) problem. The maximum a posteriori probability estimate was efficiently calculated by running a Viterbi solver. However, HMM requires both the modeling and calibration of a key building block, the emission probability. Such a pre-calibration requirement often makes this method hard to be deployed in real-world tasks, especially when the training data is unavailable. The recently developed Yet Another Algorithm for Pitch Tracking (YAAPT) [99] focused on single pitch estimation of speech signal based on both spectrogram and correlogram. The authors proposed using dynamic programming to estimate the fundamental frequency trace from a set of candidate peaks of proposed harmonic spectral features. A similar tracking method can be found in [100]. Such local-peak based methods guarantee excellent performance in high SNR cases, but often generate biased estimates under low SNR, as the probability that a local peak represents the actual signal frequency drops significantly.

The problem of tracking multiple frequency components from the spectrogram image has also been investigated. Image processing techniques such as morphological operators [101], active contour [102] methods have been applied to this area, but these methods may be difficult to be adapted to realtime tracking algorithms. Wohlmayr *et al.* [95] modeled the probability of pitch using Gaussian mixture models (GMMs), and then used the junction tree algorithm to decode a speaker-dependent factorial HMM (fHMM). A similar approach can be found in [103], where the emission probability was modeled by a deep neural network (DNN). Although both methods provide excellent performance in terms of accuracy, it is sometimes impossible to fit into real-world needs for the following two reasons. First, the training phase requires a large amount of real-world data, which is often unavailable for most tasks. Second, it is relatively time-consuming to compute the frame-wise joint emission probability and to decode the fHMM with the junction tree algorithm. The more recent studies [104, 105] proposed to use linear programming to find the best connection path of the frequency peaks on the spectrogram. These two methods first obtain all frequency peaks in the spectrogram as candidates and then find the best path from the candidates via linear programming. For low SNR scenarios, such approaches may find a large number of frequency peaks as the candidates, leading to huge memory and computational cost that is not scalable.

4.3 Track a Single Frequency Trace

4.3.1 Problem Formulation

We first formulate a frequency tracking problem for the scenarios that only a single trace exists in a frequency range of interest. Let $\mathbf{Z} \in \mathbb{R}_{+}^{M \times N}$ be the magnitude of a signal spectrogram image, which has N discretized bins along the time axis and M bins along the frequency axis. We define a *frequency trace* as

$$\mathbf{f} = \{(f(n), n)\}_{n=1}^{N}, \tag{4.1}$$

where $f\colon [1, N] \to [1, M]$ is a function. Given the spectrogram $\mathbf{Z}$ and a candidate trace $\mathbf{f}$, we define an energy function for the trace as $E(\mathbf{f}) = \sum_{n=1}^{N} \mathbf{Z}(f(n), n)$. A reasonable estimate of the frequency trace for the given signal is the trace $\hat{\mathbf{f}}$ that maximizes the energy function shown as follows

$$\hat{\mathbf{f}} = \underset{\mathbf{f}}{\operatorname{argmax}} \quad E(\mathbf{f}). \tag{4.2}$$

Problem (4.2) is equivalent to the peak finding method [106, 107] where $\hat{f}(n) = \underset{f(n)}{\operatorname{argmax}}\, \mathbf{Z}(f(n), n)$, $\forall n \in [1, N]$. It also shares similar spirit as the weighted average approach [107].

To take into consideration the smoothness assumption of the trace along the time, we add a regularization term that penalizes jumps in the frequency value. We model the change of the frequency value between two consecutive bins at $n - 1$

and n as a one step discrete-time Markov chain, characterized by the prior distribution function P_m and the transition probability matrix $\mathbf{P} \in \mathbb{R}^{M\times M}$, where $P_m = P(f(1) = m)$ and $P_{m'm} = P(f(n) = m|f(n-1) = m')$, $\forall m, m' = 1, ..., M$, and $\forall n = 2, ..., N$. Note that we assume P_m to be uniformly distributed throughout this paper to treat the initial presence of each frequency state equally. The regularized single trace frequency tracking problem is formulated as follows

$$\hat{\mathbf{f}} = \underset{\mathbf{f}}{\operatorname{argmax}} \quad E(\mathbf{f}) + \lambda P(\mathbf{f}), \tag{4.3}$$

where $P(\mathbf{f}) \triangleq \log P(f(1)) + \sum_{n=2}^{N} \log P(f(n)|f(n-1))$, and $\lambda > 0$ is a regularization parameter that controls the smoothness of the resulting trace.

4.3.2 Efficient Tracking via Dynamic Programming

The regularized tracking problem in (4.3) can be solved efficiently via dynamic programming. First, we iteratively compute an *accumulated regularized maximum energy map* $\mathbf{G} \in \mathbb{R}_{+}^{M\times N}$ column by column for all entries (m, n) as follows

$$\mathbf{G}(m,n) = \begin{cases} \mathbf{Z}(m,n) + \lambda \log P_m & n = 1; \\ \mathbf{Z}(m,n) + \max\limits_{m'}\{\mathbf{G}(m', n-1) + \lambda \log P_{m'm}\} & n > 1. \end{cases} \tag{4.4}$$

After completing the calculation at column $n = N$, the maximum value of the Nth column is denoted as $\hat{f}(N)$. Second, we find the optimal solution by backtracking from the maximum entry of the last column of the accumulated map $\mathbf{G}$. Specifically,

we iterate n from $N-1$ to 1 to solve for $\hat{f}(n)$ as follows

$$\hat{f}(n) = \underset{f(n)}{\operatorname{argmax}} \quad \mathbf{G}(f(n), n) + \lambda \log P_{f(n)\hat{f}(n+1)}. \tag{4.5}$$

Note that we can avoid transitions from state m' to state m by setting $P_{m'm} = 0$, as the regularized term would penalize the total energy to $-\infty$. If we assume uniform random walk transitions, *i.e.*, $P_{mm'} = \frac{1}{2k+1}$, $|m' - m| \leq k$, then problem (4.3) is degenerated to the *seam carving* problem defined in [94], and in this case the value λ does not affect the solution.

4.3.3 Trace Existence Detection for a Given Time Window

To determine the existence of a frequency component in a specific time interval, we first make independent decisions for every frame within the time interval on the existence of the frequency component and then refine the decisions by taking neighborhood correlations into consideration. We refer to those frames with a frequency component as *voiced* frames, or otherwise as *unvoiced* frames. We propose to test the existence of a frequency component by evaluating the relative energy of the detected trace. The test statistic called the *Relative Energy Ratio* (RER) is defined as follows:

$$\mathrm{RER}(n) = \frac{|\mathcal{F}(n)| \cdot \mathbf{Z}(\hat{f}(n), n)}{\sum_{m \in \mathcal{F}(n)} \mathbf{Z}(m, n)}, \tag{4.6}$$

where $\mathcal{F}(n) \triangleq [1, M] \backslash [\max(1, \hat{f}(n) - \delta_f), \min(M, \hat{f}(n) + \delta_f)]$ is a conservative set of frequency indices that does not contain the frequency indices around the estimated frequency; δ_f is a predetermined parameter, and $|\cdot|$ is the cardinality of a set. It is evident that the higher $\mathrm{RER}(n)$ is, the more probable that nth frame is voiced. The decision is made by comparing the test statistic $\mathrm{RER}(n)$ with an empirically determined threshold Δ_{RER}. A discussion about the optimal selection of Δ_{RER} will be presented later in Section 4.5.1.3.

In case when the length of the shortest possible unvoiced segment and voiced segment, *i.e.*, Δ_1 and Δ_2, are known, a post-process to smooth the initial detection result could further improve the detection accuracy. Specifically, we propose to group consecutive unvoiced frames into a segment when the length is greater than Δ_1, and then group consecutive unvoiced segments into one segment if the distance between the two is smaller than Δ_2. Figs. 4.2(e) and (f) illustrate two such decision making processes. Note that the final decisions can exclude all short segments, and the result is more robust compared to that of the initial decision.

4.4 Track Multiple Traces via Iterative Frequency Compensation

In the previous section, we have introduced a single frequency trace tracking and detection method using dynamic programming and trace existence testing, respectively. For some tasks such as extracting pulse rate from the face video containing subject's motion, as shown in Fig. 4.1(c), the existence of multiple traces

[1] DetectExistence$(\cdot)$ refers to the trace existence detection algorithm described in Section 4.3.3. $\hat{\mathbf{v}}_{(l)} \in \{0, 1\}^N$ is the trace existence decision with 0 as unvoiced and 1 as voiced.

Algorithm 2: Offline Adaptive Multi-Trace Carving (offline-AMTC)

procedure AMTC($\mathbf{Z}, L$) ▷ L: number of output traces
  $\mathbf{Z}_{(1)} \leftarrow \mathbf{Z}$
  $\hat{\mathbf{f}}_{(1)} \leftarrow \underset{\mathbf{f}}{\operatorname{argmax}}\, E_{\mathbf{Z}_{(1)}}(\mathbf{f}) + \lambda P(\mathbf{f})$
  $\hat{\mathbf{v}}_{(1)} \leftarrow \text{DetectExistence}(\mathbf{Z}_{(1)}, \hat{\mathbf{f}}_{(1)}, \Delta_{\text{RER}}, \Delta_1, \Delta_2)$[1]
  for $l \leftarrow 2$ to L **do**
    Update $\mathbf{Z}_{(l)}$ according to (4.7)
    $\hat{\mathbf{f}}_{(l)} \leftarrow \underset{\mathbf{f}}{\operatorname{argmax}}\, E_{\mathbf{Z}_{(l)}}(\mathbf{f}) + \lambda P(\mathbf{f})$
    $\hat{\mathbf{v}}_{(l)} \leftarrow \text{DetectExistence}(\mathbf{Z}_{(l)}, \hat{\mathbf{f}}_{(l)}, \Delta_{\text{RER}}, \Delta_1, \Delta_2)$
  end for
  return $\hat{\mathbf{f}}_{(1:L)}$, $\hat{\mathbf{v}}_{(1:L)}$
end procedure

in the frequency range of interest is possible, and the dominating trace in the spectrogram might not be the one of interest. A crude deployment of any single trace tracking method on such tasks would generate completely wrong answers. To address this problem, we strategically extend the single trace tracking method to be able to track multiple traces by extracting trace iteratively to find all candidates. We name this method the Adaptive Multi-Trace Carving (AMTC). In the rest part of this section, we first present the offline version of AMTC (offline-AMTC), when the trace estimate is optimized according to the entire available signal. We next adapt the offline-AMTC to an efficient online version (online-AMTC), which runs in quasi-realtime with low delay.

4.4.1 Offline-AMTC

Similar to the iterative nature of the seam carving algorithm [94], multiple traces can be greedily searched for by iteratively running the single trace tracker

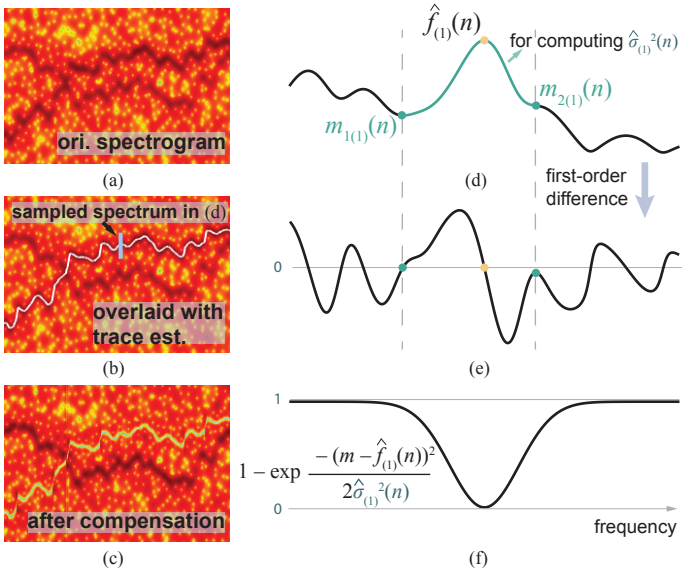


Figure 4.3: Illustrations for the trace compensation process: (a) the spectrogram of a synthetic -8 dB signal with two frequency components; (b) first trace estimate by AMTC; (c) the spectrogram after the first trace compensation; (d) the sampled spectral distribution centered at $\hat{f}_{(1)}(n)$ ($n = 400$, vertical line in (b)); (e) the first-order difference of the spectral function in (d); (f) the generated point-wise compensation weights. The value of $\hat{\sigma}^2_{(1)}$ is determined by the values within the green region in (d).

proposed in Section 4.3. However, as frequency energy is diffused around the center of each trace due to the windowing effect and violation of the signal stationary assumption, multiple consecutive trace estimates may belong to a single frequency component without compensating the diffused spectral energy.

To solve this problem, we attenuate the diffused energy around the estimated frequency trace at the end of each iteration once we obtain the estimated frequency trace. Specifically, suppose $\hat{\mathbf{f}}_{(l)}$ is the estimated frequency trace at the lth iteration. For each time frame of the spectrogram, *i.e.*, $\mathbf{Z}_{(l)}(1:M,n)$, we search for a left boundary point $m_{1(l)}(n)$ from $\hat{f}_{(l)}(n)$ to its left side. We set $m_{1(l)}(n) = m$, if m is the first point that is either a local minimum point in $\mathbf{Z}_{(l)}(1:M,n)$ or a local minimum point in the first-order difference of $\mathbf{Z}_{(l)}(1:M,n)$. The search of the right

boundary point $m_{2(l)}(n)$ works similarly except it considers the local maximum point in the first-order difference of $\mathbf{Z}_{(l)}(1:M,n)$. In this paper, we call $\mathbf{Z}_{(l)}(m_{1(l)}(n):m_{2(l)}(n),n)$ *the energy bump* of $\hat{f}_{(l)}(n)$.

One example of the trace compensation process is shown in Fig. 4.3. The plot in (d) shows the spectral energy distribution centered at $\hat{f}_{(1)}(n)$, which corresponds to the white vertical line in (b). In this case, $m_{1(1)}(n)$ is selected as the first local minimum point, and $m_{2(1)}(n)$ as the local maximum point in the first-order difference of $\mathbf{Z}_{(l)}(1:M,n)$. Based on $m_{1(l)}(n)$ and $m_{2(l)}(n)$, we propose to use a reverse Gaussian-shaped function to compensate the energy of the estimated frequency component. The updated equation for the compensated power spectrum at the $(l+1)$st iteration is as follows

$$\mathbf{Z}_{(l+1)}(m,n) \leftarrow \left[1-\exp\frac{-\left(m-\hat{f}_{(l)}(n)\right)^2}{2\hat{\sigma}^2_{(l)}(n)}\right]\cdot\mathbf{Z}_{(l)}(m,n), \tag{4.7}$$

where $\hat{\sigma}^2_{(l)}(n) = \frac{\sum_{m=m_{1(l)}(n)}^{m_{2(l)}(n)}\mathbf{Z}_{(l)}(m,n)(m-\hat{f}_{(l)}(n))^2}{\sum_{m=m_{1(l)}(n)}^{m_{2(l)}(n)}\mathbf{Z}_{(l)}(m,n)}$ is used to quantify the width of the energy bump at the lth iteration. The pseudo code of the offline-AMTC is shown in Algorithm 2. In Fig. 4.2, we give an example of two-trace estimation process on a synthetic heart beat signal. The final estimate is almost identical with the ground truth, and the unvoiced segments are successfully detected.

If we define L as the number of traces to track, the computational complexity for offline-AMTC is $O(NLM^2)$. To compare, the fHMM methods [95, 103] requires $O(NLM^{L+1})$ without considering operations for computing emission probability.

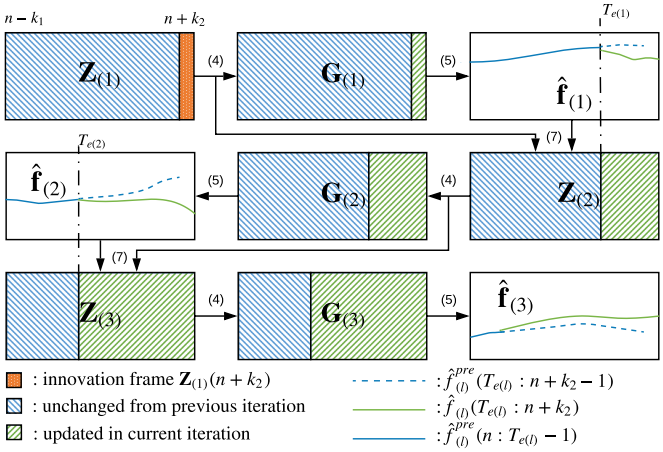


Figure 4.4: A flowchart for online-AMTC algorithm for three traces estimation process at tth iteration. $(\cdot)$ above arrows indicates the index of the equation being used.

The efficiency of offline-AMTC is mostly explained by the idea of the introduced iterative search. We will later show in Section 4.5 that the demonstrated efficiency is not achieved at the expense of performance drop.

4.4.2 Online-AMTC with Low Delay

The offline-AMTC algorithm minimizes the adverse effect of noise by making use of full-length signals. In a delay-sensitive scenario that a fixed-length delay k is allowed, the tracking objective at the time instant n is to estimate $\hat{f}_{(1:L)}(n)$ based on the available spectrogram information $\mathbf{Z}_{(1)}(1:n+k)$[2]. A simple approach runs offline-AMTC from the time instant 1 to $n+k$ at each time instant n, costing $O(nLM^2)$ in time. If the total length of the frame is N, this approach takes $O(NM)$ in space and $\sum_{n=1}^{N} O(nLM^2) = O(N^2LM^2)$ in time. Space and time complexities increase quadratically and linearly in N, respectively. This increasing

[2]For concise representation, we use $\mathbf{G}(n_1:n_2)$ and $\mathbf{Z}(n_1:n_2)$ as a shorthand for $\mathbf{G}(1:M, n_1:n_2)$ and $\mathbf{Z}(1:M, n_1:n_2)$, respectively.

Algorithm 3: Online-AMTC at time n

procedure AMTC($\mathbf{Z}_{(1:L)}(\tau_1 : \tau_2 - 1)$, $\mathbf{G}_{(1:L)}(\tau_1 : \tau_2 - 1)$, $\hat{f}^{\text{pre}}_{(1:L)}(\tau_1 : \tau_2 - 1)$, $\mathbf{Z}_{(1)}(\tau_2)$) ▷ $\tau_1 \triangleq n - k_1$, $\tau_2 \triangleq n + k_2$.
  $\mathbf{Z}_{(1)}(\tau_1 : \tau_2) \leftarrow$ concatenate $\mathbf{Z}_{(1)}(\tau_1 : \tau_2 - 1)$ and $\mathbf{Z}_{(1)}(\tau_2)$
  Update $\mathbf{G}_{(1)}(\tau_2)$ according to (4.4) using $\mathbf{G}_{(1)}(\tau_2 - 1)$ and $\mathbf{Z}_{(1)}(\tau_2)$
  $T_e \leftarrow \tau_2 - 1$
  for $l \leftarrow 1$ to L **do**
    Estimate $\hat{f}_{(l)}(T_e + 1 : \tau_2)$ according to (4.5) using $\mathbf{G}_{(l)}(T_e + 1 : \tau_2)$
    if $l < L$ **then**
      Update $\mathbf{Z}_{(l+1)}(T_e + 1 : \tau_2)$ according to (4.7) using $\mathbf{Z}_{(l)}(T_e + 1 : \tau_2)$ and $\hat{f}_{(l)}(T_e + 1 : \tau_2)$
    end if
    for $i \leftarrow T_e$ to τ_1 **do**
      Estimate $\hat{f}_{(l)}(i)$ according to (4.5) using $\hat{f}_{(l)}(i + 1)$ and $\mathbf{G}_{(l)}(i)$
      if $l < L$ **then**
        Update $\mathbf{Z}_{(l+1)}(i)$ according to (4.7) using $\mathbf{Z}_{(l)}(i)$ and $\hat{f}_{(l)}(i)$
      end if
      if $\hat{f}_{(l)}(i) == \hat{f}^{\text{pre}}_{(l)}(i)$ **then**
        Update $\mathbf{G}_{(l+1)}(i + 1 : \tau_2)$ according to (4.4) using $\mathbf{Z}_{(l+1)}(i + 1 : \tau_2)$ and $\mathbf{G}_{(l+1)}(i)$
        $\hat{f}_{(l)}(\tau_1 : i) \leftarrow \hat{f}^{\text{pre}}_{(l)}(\tau_1 : i)$
        $T_e \leftarrow i$
        break
      else if $i == \tau_1$ **then**
        Update $\mathbf{G}_{(l+1)}(\tau_1 : \tau_2)$ according to (4.4) using $\mathbf{Z}_{(l+1)}(i : \tau_2)$
        $T_e \leftarrow i$
      end if
    end for
    $\hat{v}_{(l)}(\tau_1 : \tau_2) = \text{DetectExistence}(\mathbf{Z}_{(l)}(\tau_1 : \tau_2), \hat{f}_{(l)}(\tau_1 : \tau_2), \Delta_{\text{RER}}, \Delta_1, \Delta_2)$
  end for
  return $\hat{f}_{(1:L)}(n)$, $\hat{v}_{(1:L)}(n)$, $\mathbf{Z}_{(1:L)}(\tau_1 + 1 : \tau_2)$, $\mathbf{G}_{(1:L)}(\tau_1 + 1 : \tau_2)$, $\hat{f}_{(1:L)}(\tau_1 + 1 : \tau_2)$
end procedure

trend of workload will eventually lead to either memory overflow or CPU overload, especially when we expect to run the system in days or even months.

An efficient, quasi-realtime algorithm called the online-AMTC is developed to address the storage and computational issues mentioned above. We propose to use a fixed-length queue buffer for storing and updating the intermediate result of $\mathbf{Z}_{(1:L)}$, $\mathbf{G}_{(1:L)}$, and $\hat{f}_{(1:L)}$. As a result, the running time and the memory requirement are greatly reduced and are independent of time n.

We introduce the algorithm by first discussing online iterations for the estimation process of the first trace. The processing flow of the online-AMTC algorithm at the instant n is illustrated in Fig. 4.4. Suppose the allowed delay length is k_2 and $\hat{f}_{(1)}(n-1)$ has been computed by backtracking from the accumulated regularized maximum energy map $\mathbf{G}_{(1)}(n-1:n+k_2-1)$. At the arrival of the next innovation frame $\mathbf{Z}_{(1)}(n+k_2)$ (the orange frame in Fig. 4.4), our goal is to estimate $\hat{f}_{(1)}(n)$. From the forward update rule of $\mathbf{G}$ in (4.4), it is clear that $\mathbf{G}_{(1)}(n:n+k_2-1)$ would remain unchanged compared to the output in the previous time instant $n-1$. We therefore only need to update the right most frame $\mathbf{G}_{(1)}(n+k_2)$ given $\mathbf{G}_{(1)}(n+k_2-1)$ and the innovation frame $\mathbf{Z}_{(1)}(n+k_2)$ as shown in the middle box of the first row of Fig 4.4. $\hat{f}_{(1)}(n)$ is then obtained via backtracking from $\mathbf{G}_1(n:n+k_2)$ according to (4.5). Now, we define the previous backtracking result at time $n-1$ as $\hat{f}^{\text{pre}}_{(1)}(n-1:n+k_2-1)$. During the backtracking process for $\hat{f}_{(1)}(n)$, if $\hat{f}_{(1)} = \hat{f}^{\text{pre}}_{(1)}$ at the time instant index $T_e \in [n, n+k_2)$, we have $\hat{f}_{(1)}(n:T_e) = \hat{f}^{\text{pre}}_{(1)}(n:T_e)$. This claim holds because $\mathbf{G}_{(1)}(n:T_e)$ remains the same during the process. In this regard, we consider storing and updating $\hat{f}^{\text{pre}}_{(1)}(n-1:n+k_2-1)$ in a buffer, whereby

the update process of $\hat{f}^{\text{pre}}_{(1)}$ stops at the instant T_e if $\hat{f}_{(1)}(T_e) = \hat{f}^{\text{pre}}_{(1)}(T_e)$, as shown in the right box of the first row of Fig. 4.4. In this way, the computation complexity is further reduced.

Different from the estimation process for the first trace, any change from previous trace estimation $\hat{f}_{(1:l-1)}$ would have influence on the formation of $\mathbf{Z}_{(l)}$, $\mathbf{G}_{(l)}$, and therefore $\hat{f}_{(l)}$. In order to obtain a robust estimate for $\hat{f}_{(l)}$, $l > 1$, we introduce a look-back length, $k_1 > 0$ in this process. As demonstrated from second and third rows in Fig. 4.4, for lth trace estimation at time instant n, we utilize the previous trace estimates $\hat{f}_{(l-1)}(n - k_1 : n + k_2)$ and $\mathbf{Z}_{(l-1)}(n - k_1 : n + k_2)$ to obtain new $\mathbf{Z}_{(l)}(n-k_1 : n+k_2)$ and $\mathbf{G}_{(l)}(n-k_1 : n+k_2)$, and thus $\hat{f}_{(l)}(n-k_1 : n+k_2)$. Efficient backtracking can also be achieved using the previous backtracking result, same as the case in estimating the first trace. The details of online-AMTC algorithm at nth iteration is shown in Algorithm 3.

The worst-case computational complexity for online-AMTC is $O(N(k_1+k_2)LM^2)$, which appears to be $(k_1 + k_2)$ times slower than offline version. In the statistical sense, we argue that the expected complexity of the online-AMTC is much less than the worst-case analysis result because the probability that an entire trace estimate changed from the previous one is low at each time instant. To demonstrate this, we will compare the average computation time running offline and online-AMTC in Section 4.5.1.2.

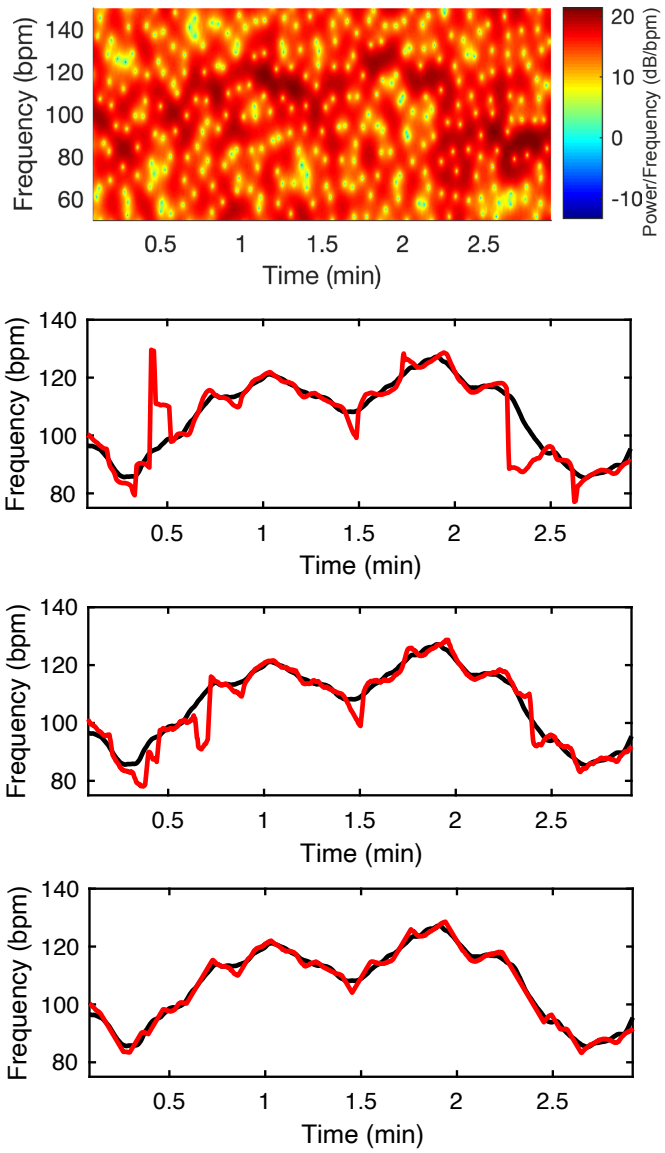


Figure 4.5: From top to bottom: Spectrogram of a synthetic -10 dB signal with one frequency component; Trace tracking results (red line) by YAAPT, particle filter, and offline-AMTC, respectively. The ground truth trace is shown in black line in each plot.

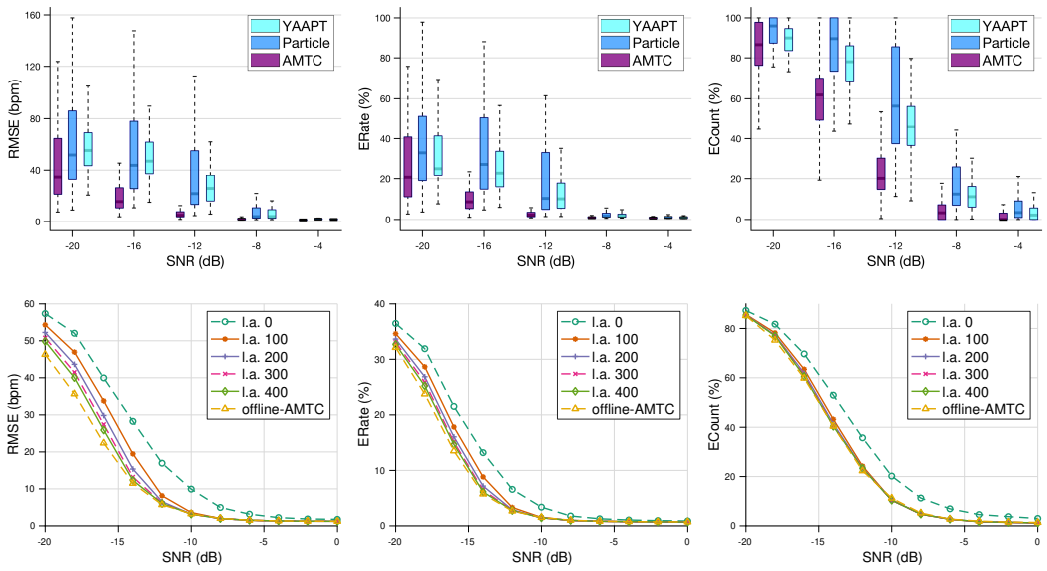


Figure 4.6: First row: Comparison of the performance of single trace tracking by the proposed offline-AMTC, particle filter, and YAAPT methods at different levels of SNR. Statistics of the RMSE, the ERATE, and ECOUNT of frequency estimates are summarized using box plots. Second row: RMSE, ERATE, and ECOUNT for trace estimation by online-AMTC with different levels of look-ahead window length and SNR. The result using offline-AMTC is shown in the plots for comparison purposes.

4.5 Performance Analysis of AMTC

4.5.1 Simulation Results and Comparison with Known Ground Truth

4.5.1.1 Single Trace

We first evaluated the performance of the AMTC algorithm using simulated data. For each test signal, we generated a time-varying pulse rate trace present from the beginning to the end of the timeline. More specifically, denote $s[n]$ as the temporal measurement of the corrupted frequency signal, $s[n] = \sin \Phi[n] + \epsilon[n]$, where $\Phi[n] = \Phi[n-1] + 2\pi f[n]/f_s$, $f[n]$ is the time-varying synthesis frequency, f_s is the sampling rate set to 30 Hz, and $\epsilon[n]$ is the noise quantified by a zero-mean white

Gaussian process. The variance of $\epsilon[n]$ is an adjustable parameter for achieving different SNR levels. To generate frequency signals $f[n]$ that behave similarly as real-world pulse rate signals, we trained a 9-tap autoregressive model using heart rate signals collected by a Polar H7 chest belt in both exercise mode and still mode. We use beat per minute (bpm) as the frequency unit. The duration of each test signal was 3 minutes. The spectrograms were generated by short-time Fourier transform (STFT) with window length 10 secs and 98% overlap between adjacent frames. We padded zeros to the end of each frame to make neighboring frequency bins 0.17 bpm apart.

We then compared our algorithm with the Particle filter method [78] and the local peak based YAAPT method [99] using a large scale synthetic dataset. The number of particles was set to 1024. We generated 500 trials under each of the five SNR conditions, or 250 for each mode (namely, the exercise and the still cases) using the estimated parameters of the autoregressive models. We used three metrics. Namely, the root mean squared error (RMSE), the error rate (ERATE), and the error count (ECOUNT) defined as follows to evaluate the performance:

- $\text{RMSE} = \sqrt{\frac{1}{T}\sum_{t=1}^{T}(\hat{f}_t - f_t)^2}$,
- $\text{ERATE} = \frac{1}{T}\sum_{t=1}^{T}\left|\hat{f}_t - f_t\right| \Big/ f_t$,
- $\text{ECOUNT} = \left|\{t \in [1,T] : \left|\hat{f}_t - f_t\right| \Big/ f_t > \tau\}\right| \Big/ T$,

where $|\{\cdot\}|$ denotes the cardinality of a countable set, $\hat{f}_t$ and f_t are the frequency estimate and the ground-truth frequency at tth time frame respectively, and τ was

chosen to be 0.03 empirically determined from the spread of the frequency components. Fig. 4.5 shows tracking results of a -10 dB synthetic signal with one frequency component using AMTC, YAAPT, particle filter, respectively. In this example, AMTC outputs the best trace estimate among the three without much deviation from the ground truth. The results of overall performance are shown in the first row of Fig. 4.6 in terms of box plots that each box compactly shows the median, upper, and lower quantiles, and the max and min values of a dataset. It is evident from the box plots that, under all SNR levels, AMTC generally outperforms the particle filter method and the YAAPT not only in terms of the average but also in the variance of the error statistics.

Next, we tested the online-AMTC algorithm using different look-ahead time lengths. The evaluation was conducted using the same setting mentioned above, and the averaged behavior of each look-ahead length is plotted in the second row of Fig. 4.6. The numbers in the legends indicate the lengths of look-ahead (l.a.) window lengths represented by the number of time bins in the spectrogram. We have two observations from these plots. First, a performance jump from no-look-ahead versus 100-bin look-ahead length is observed, but the performance saturates after further increasing the length. This observation coincides with the intuition that a small look-ahead length would cause the online trace estimator to find a locally optimum solution. Second, given the shape of the curve, the performance starts to converge from SNR $= -10$ dB and is almost identical among different levels of look-ahead length. This trend of convergence is also expected as the signal quality is high enough for AMTC to track the correct trace.

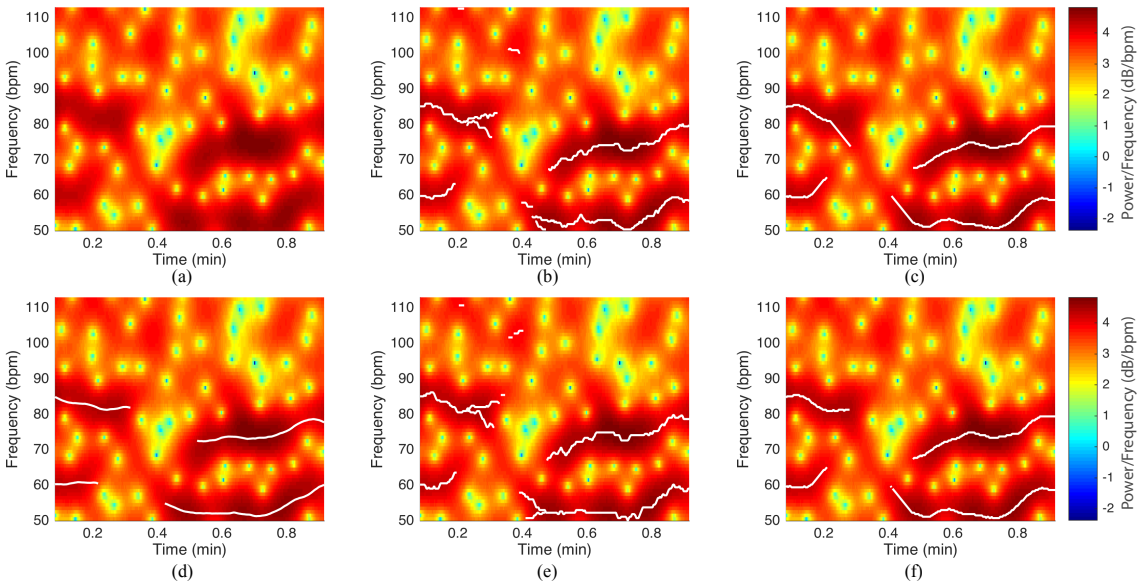


Figure 4.7: (a): spectrogram of one test instance with SNR = −8 dB; (d) same spectrogram overlaid by ground truth traces. SD-fHMM (b), SI-fHMM (e), offline-AMTC (c), and online-AMTC (f) tracking results.

	E_{01}	E_{02}	E_{10}	E_{12}	E_{20}	E_{21}	E_{Gross}	E_{Total}	E_{fine}
SD-fHMM	4.14%	1.62%	0.36%	15.39%	0.28%	1.78%	0.02%	23.59%	1.79%
SI-fHMM	3.48%	1.52%	0.61%	14.37%	0.29%	2.38%	0.03%	22.68%	1.82%
offline-AMTC	1.77%	0.28%	3.57%	2.16%	0.45%	9.99%	0.03%	18.27%	1.76%
online-AMTC	1.75%	0.38%	3.17%	2.65%	0.48%	8.41%	0.03%	16.87%	1.80%

Table 4.1: Averaged Performance of fHMM and AMTC on multi-trace tracking test

	***mixmax* likelihood (sec)**	**Tracking (sec)**
SD-fHMM	39.47	3.96
SI-fHMM	195.86	4.30
offline-AMTC	N/A	0.10
online-AMTC	N/A	0.44

Table 4.2: Average computation time in seconds per 100 frames

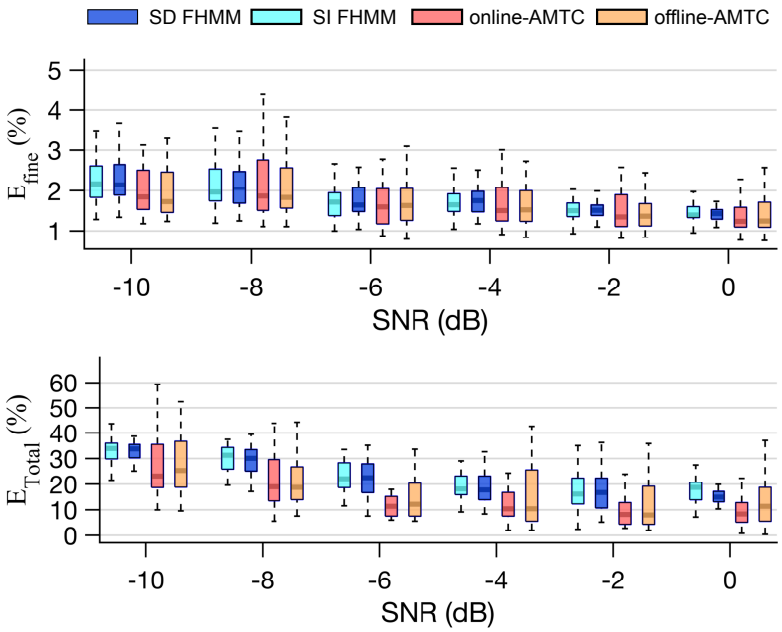
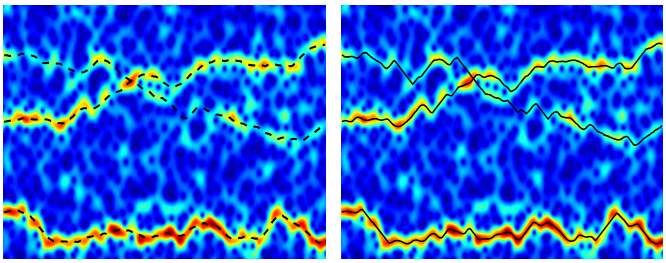
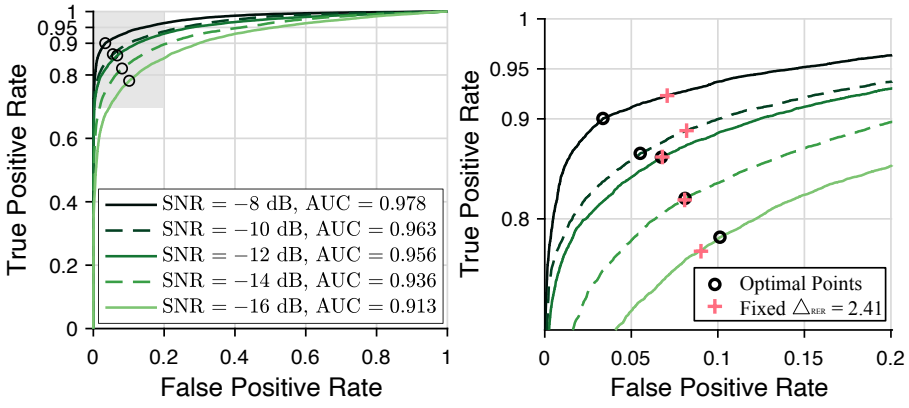


Figure 4.8: Box plots of E_{fine} (top) and E_{Total} (bottom) of two traces tracking using SD-fHMM, SI-fHMM, offline-AMTC, and online-AMTC on different levels of SNR.



(a) (b)

Figure 4.9: (a) Ground truth frequency traces at -10 dB in spectrogram of a synthetic signal. (b) Three trace estimates by AMTC.



(a) (b)

Figure 4.10: (a) ROC curve of the proposed trace detection method in different SNR conditions. (b) the zoomed plot of the shaded area in (a) with optimal operating points (black circle) and operating points using fixed threshold ($\Delta_{\text{RER}} = 2.41$, dark red plus sign).

4.5.1.2 Multiple Traces

In this section, we evaluate the performance of the offline- and online-AMTC using simulated data in the presence of multiple traces and compare them with the fHMM method. To allow a fair comparison of our methods with fHMM, we adopt the performance measure proposed in [108] with a slight change. We give details on our experiment setup as well as the error measure below.

To test both algorithms, we generated a corrupted frequency signal $s[n]$ with two frequency traces, *i.e.*, $s[n] = \sum_{l=1}^{2} \sin \Phi_(l)[n] + \epsilon[n]$. The variance of $\epsilon[n]$ is tuned to achieve six SNR levels from 0 to -10 dB. To cope with the high computational cost associated with running fHMM at a full scale, we cut signals to 1 minute, set the number of frequency bins to 64, and made neighboring frequency bins 1 bpm apart. The cardinality of frequency state was set to 169 so that it uniformly covers the whole frequency range of interest. For each trace, we also introduced a 20 seconds

unvoiced segment.

We estimate the GMM parameters of the fHMM framework using the EM algorithm [109]. For each SNR level, we generated 6000 spectrum frames with a single frequency component for each 169 frequency states (where the first state encodes unvoiced decision). We set the maximum number of components per GMM to 20 and used MDL [95] to determine the number of components automatically. The parameters were trained in an SNR-dependent (SD) and an SNR-independent (SI) fashion (*i.e.*, each SD model was trained only with samples of the corresponding SNR, and the SI model was trained with all samples). We adopted the mixture-maximization interaction model proposed in [95], and set the prior distribution for both fHMM and AMTC uniformly as $P(f_{(l)}(1) = m) = 1/169$, $\forall m$, and the transition probability follows a uniform distribution with width parameter $k = 2$. Moreover, the voiced to unvoiced transition probability for fHMM was empirically selected as $P(\text{voiced}|\text{unvoiced}) = 0.2$, and $P(\text{unvoiced}|\text{voiced}) = 0.1$.

To compare the tracking performance, we use the well-adopted error measure proposed in [108] as described below:

- E_{ij}: the percentage of time frames where i frequency components are misclassified as j.

- E_{Gross}: the percentage of frames where $\exists l$, s.t. $\Delta f_{(l)} > 20\%$. We define the relative frequency deviation $\Delta f_{(l)} \triangleq \min_i \frac{|\hat{f}_i - f_{(l)}|}{f_{(l)}}$, and $f_{(l)}$ is the reference frequency for lth component.

- E^l_{fine}: the average relative frequency deviation from the reference of the lth

frequency component for those frames where $\forall l$, $\Delta f_{(l)} \leq 20\%$.

Note that both E_{ij} and E_{Gross} represent a frame counting measure. We therefore group them together to form the total gross error: $E_{\text{Total}} = E_{01} + E_{02} + E_{10} + E_{12} + E_{20} + E_{21} + E_{\text{Gross}}$, and define $E_{\text{fine}} = E^1_{\text{fine}} + E^2_{\text{fine}}$.

To test the performance, we generated 30 tested signal for each SNR level using the same setting mentioned above. We compared the performance of SD-fHMM, SI-fHMM, offline-AMTC and online-AMTC using the aforementioned error measures and the results are listed in Table 4.1. We depict the distribution of E_{Total} and E_{fine} specifically in Fig. 4.8. All methods have a similar performance in terms of the fine detection error E_{fine}, while AMTC slightly outperforms fHMM in terms of E_{Total}, the main contributors of which are E_{12} and E_{21}. Table 4.2 shows the average computation time for the *mixmax* likelihood estimation procedure [95], together with the tracking time requirement tested on a 2014 MacBook pro with a 2.3 GHz Intel Core i5 processor. Note that the preprocessing stage of fHMM to compute the emission probability also consumes almost 0.4 sec/frame for the SD and 2.0 sec/frame for the SI model, which makes the real-time implementation almost impossible for a usual hardware setting. AMTC, on the other hand, is much more computationally efficient than fHMM even without considering the *mixmax* likelihood computing. For this task, online-AMTC reported a similar performance compared with the offline version at 4.4 msec/frame. It guarantees real-time adaptation with almost no performance drop. Fig. 4.7 shows the experimental results of the proposed algorithm and fHMM on a test signal with SNR $= -8$ dB. We can observe that in a low SNR

environment, the performance of online and offline-AMTC are better than fHMM algorithm in terms of accuracy and false-positive detections.

Fig. 4.9(b) shows an example of the tracking result of offline-AMTC when SNR is -10 dB and three traces are presented. We can see three traces have been accurately estimated as compared to the ground truth on the left when two weak traces with different levels of strength intersect.

4.5.1.3 Trace Detection

In this part, we evaluate the trace detection performance and the optimal selection of Δ_{RER} using the synthetic data in five levels of SNR conditions. We generated 100 trials for each level of SNR with the generative model described in Section VA-1. An unvoiced segment was selected in each test signal. The length of the selected segment ranged from 25% to 75% of the signal length, and the overall number of voiced spectral frames equaled the number of unvoiced frames. In this experiment, the voiced detection is treated as the positive case, and the detection result (without the post-processing operation using Δ_1 and Δ_2) is summarized using the Receiver-Operating Characteristic (ROC) plot in Fig. 4.10(a). From the plot, we observe highly accurate detection result in each SNR condition with the Area Under the Curve (AUC) higher than 0.9.

In Fig. 4.10(b), we show the zoomed plot of the shaded area in Fig. 4.10(a). The optimal operating points (black circles) were found by minimizing the sum of false positive rate and the false negative rate. The operating point corresponding

to a fixed Δ_{RER} selection is also shown in each SNR level. Note that the detection result using a fixed threshold value is highly accurate and is close to that with the optimal choice in every SNR level, demonstrating the insensitivity of the threshold parameter Δ_{RER}.

4.5.2 Experimental Results on rPPG Data

We evaluated the performance of the proposed method on a real-world dataset from the problem of the pulse rate estimation from facial videos. We show by experiment that AMTC can successfully extract the subtle pulse trace even when the trace is dominated by another frequency component. To test the robustness of the algorithm in a challenging situation, we use the dataset where the video contains significant subject motion [93]. In total, the dataset contains 20 videos in which 10 contain human motions on an elliptical machine, and the other 10 contain motions on a treadmill. Each video is about 3 minutes long in order to cover various stages of fitness exercise. Each video was captured in front of the subject's face by a commodity mobile camera (iPhone 6s) affixed on a tripod or held by the hands of a person other than the subject. The heart rate of the test subject was simultaneously monitored by an electrocardiogram (ECG)-based chest belt (Polar H7) for reference. The spectrogram of the preprocessed face color feature was estimated using the same set of parameters as in Section 4.5.1.

Fig. 4.11 gives an example of the tracking result using AMTC with a uniform Markov transition probability model with $k = 60$ for first motion-induced trace

	RMSE (bpm)		ERATE (%)		ECOUNT (%)	
	$\hat{\mu}$	$\hat{\sigma}$	$\hat{\mu}$	$\hat{\sigma}$	$\hat{\mu}$	$\hat{\sigma}$
MN+PF	5.29	5.51	9.41	14.13	2.20	2.24
offline-AMTC	2.21	1.11	3.16	6.04	1.02	2.24
online-AMTC	2.78	1.20	4.01	6.42	1.25	2.42

Table 4.3: Performance of proposed method and particle filter method on rPPG data

estimate and with $k = 2$ for second pulse-induced trace estimate. More freedom of trace dynamic ($k = 60$) was assigned to the first estimate as the variation of motion frequency can be much greater than the heart rate. We noticed for each spectrogram, the traces induced by subject motions dominate over the heart rate trace. Compared to the particle filter-based method that utilizes additional information to compensate for the motion trace [93], AMTC can faithfully track the dominating motion trace and recognize the PR trace as the second trace. Notice that the trace estimate from the particle filter would occasionally deviate to the vertical motion trace. We summarize the mean $\hat{\mu}$ and standard deviation $\hat{\sigma}$ of the error measures for all of our videos, and the results are listed in Table 4.3. The average error for AMTC is 2.21 bpm in offline mode and 2.78 bpm in online mode in RMSE and 3.16% in offline mode and 4.01% in online mode in relative error. The performance of AMTC is more than twice as good against the state-of-the-art motion notching + particle filter.

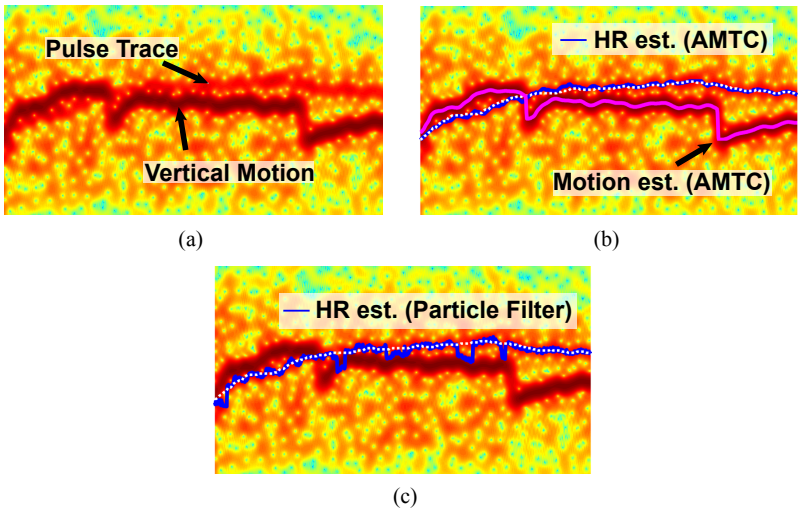


Figure 4.11: (a) Weak heart pulse embedded in a strong trace induced by vertical motion of the person running on an elliptical machine. The estimated pulse SNR equaled −4.5 dB. (b) Heart rate estimation after compensating the first trace estimate using offline-AMTC. (c) Heart rate estimation using motion spectrogram notching and particle filter method.

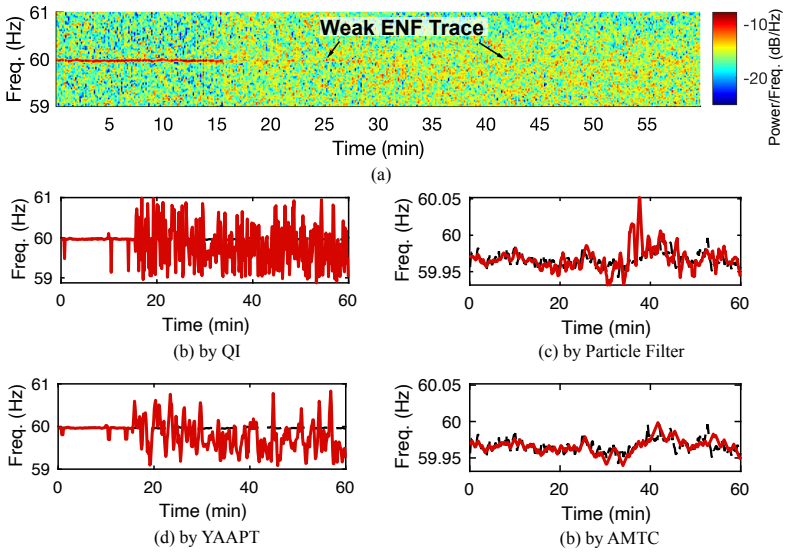


Figure 4.12: (a) Spectrogram for a sample ENF audio signal. The estimated ENF SNR equaled -8.2 dB. Trace estimates (red line) returned by Quadratic Interpolation (b), Particle Filter (c), YAAPT (d), and offline-AMTC (e). The reference ENF trace is shown in black line in plots (b)–(e).

	Rmse (Hz)		Pearson's ρ	
	$\hat{\mu}$	$\hat{\sigma}$	$\hat{\mu}$	$\hat{\sigma}$
QI	0.24	0.18	0.18	0.26
Particle Filter	0.04	0.07	0.55	0.37
YAAPT	0.16	0.12	0.23	0.28
offline-AMTC	0.01	0.01	0.85	0.18
online-AMTC	0.03	0.02	0.81	0.20

Table 4.4: Performance of various methods on ENF data

4.5.3 Experimental Results on ENF Data

In this subsection, we tested the performance of the proposed algorithm on a real-world ENF dataset. In total, 27 pairs of one-hour power grid signal and audio signal from a variety of locations in North America were collected and tested. Each pair of signals were simultaneously recorded using a battery-powered Olympus Voice Recorder WS-700M at a sampling rate of 44.1 kHz in MP3 format at 256 kbps. We recorded the reference ENF signal directly from the power mains of the electrical supply. To limit the voltage to the safe range of the input of a sound card or a digital recorder, we used a step-down transformer to convert the power supply voltage level to 5V and then used a voltage divider to obtain an input of 5 mV [107].

We downsampled the signals to 1 kHz to reduce the computational load, and applied *harmonic combining method* [110] to obtain robust frequency strips around the nominal frequency, *i.e.*, 60 Hz in North America. The harmonic combining method approach exploits different ENF components appearing in a signal, and adaptively combines them based on the local signal-to-noise ratio to achieve a more

robust and accurate estimate than that by using only one component. We obtained the ground truth from the corresponding power grid signals using *Quadratic Interpolation* (QI) [111], as the SNR is high and frame-wise highest peak method is proved to be the maximum likelihood estimator of signal frequency [106]. We use RMSE and Pearson correlation coefficient ρ of the estimated versus the ground-truth sequence of frequency variations as two performance indices. They are two well-adopted error measures for ENF estimation.

Fig. 4.12 gives a tracking example using a piece of the audio signal captured from 03:03 am to 04:03 am PT, Oct. 31st, 2012, San Diego, CA. Note that the ENF trace becomes weak after 15 mins, which we define as a checkpoint. AMTC can identify the trace from the noisy harmonic combined spectrum feature. Particle filter gives comparable results before the checkpoint but deviates from the true trace occasionally due to nearby interference. Local peak based tracking method YAAPT and frame-wise frequency estimator QI completely lost the target after the checkpoint as the peak information alone is not able to guarantee a good estimate.

The performance of various methods is summarized in Table 4.4. We calculated the mean and standard deviation of the error measures for 27 pieces of audio ENF signals. For this very noisy dataset, AMTC can achieve 0.01 Hz in offline mode and 0.03 Hz in online mode in average RMSE and 0.85 in offline mode and 0.81 in online mode in average correlation with ground truth, which outperforms all other tracking methods substantially both in average and variance of the error statistics.

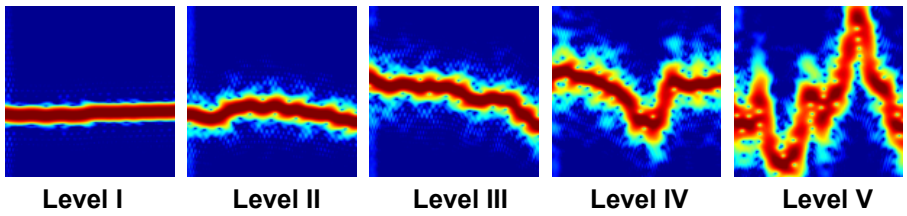


Figure 4.13: Spectrogram examples of clean signals with five trace variation levels.

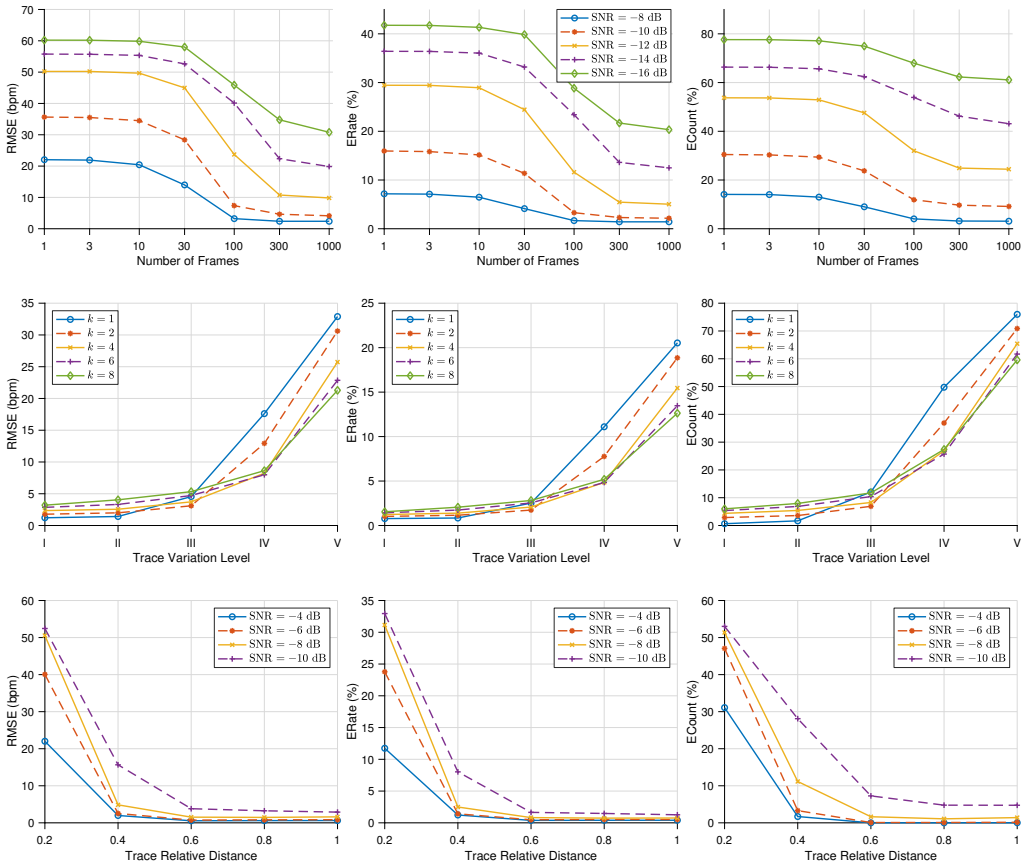


Figure 4.14: Impact evaluation of three factors: RMSE (first column), ERATE (second column), and ECOUNT (last column) of the trace estimates by offline-AMTC with different combinations of TRD and SNR levels (first row), with different trace variation levels and selections of k (second row), and with different TRD and SNR levels (last row).

4.6 Impact of Various Factors

In this section, we further investigate the performance of AMTC in respond to multiple factors. First, we study the performance when the signal length varies. Then, we discuss the effect of the trace variation level. Finally, we evaluate the impact of the distance between two traces to the estimation accuracy. The parameters are configured to be the same as introduced in Section 4.5.1 unless otherwise stated.

4.6.1 Impact of Signal Length

In this part, we show by experiment that more accurate tracking results are achieved when the signal length or the number of frames in the spectrogram increases. We quantitatively analyze the system performance concerning different levels of the signal length and argue that a minimum of 300 spectral frames is expected for the performance level demonstrated in the paper for SNR ranging from -8 dB to -14 dB.

The evaluation was conducted with the same signal synthetic setting described in Section 4.5.1.1. We generated 200 trials under each of the five SNR conditions ranging from -8 dB to -16 dB. The duration of the test signal was set as three and a half minutes, which is equivalent to 1000 spectral frames in the spectrogram. The 1000 spectral frames were then segmented uniformly based on the five levels of evaluated signal length, and the offline-AMTC was performed independently on each segment. The mean performance results with respect to different combinations of the signal length and SNR is shown in the first row of Fig. 4.14. Note that

when the spectrogram only contains one frame, the tracking result using AMTC is equivalent to that using the highest peak method. We can observe from the plots that the system performance gains significantly when the signal length exceeds 10 frames. More frames are needed in a lower SNR condition for converging to the best performance level, but overall the performance starts to converge when the number of frames equals 300 under all SNR levels except -16 dB.

4.6.2 Impact of Trace Variation

During the formulation process of the frequency trace tracking problem, we have assumed the change of the frequency value between two consecutive bins as a one-step discrete-time Markov chain, characterized by a transition probability matrix $\mathbf{P}$. With a training dataset of sufficient size available to the user, one may learn the model parameters of $\mathbf{P}$ to make a more precise tracking estimation. However, the training set is often unavailable in a real-world setting, and the user has to make their own choice of the $\mathbf{P}$ before deploying the algorithm. It is therefore important for a robust frequency tracker to successfully track the frequency components even when the variation of the frequency traces is at different levels.

We thus evaluated the system performance with respect to five different trace variation levels, and assumed the transition probability follows the uniform distribution parameterized by k. 200 trials were generated for each level of trace variation by tuning the variance of $f[n]$ in the generative signal model described in Section 4.5.1.1. Specifically, the five levels of the trace variation correspond to 0.001,

0.005, 0.01, 0.02, and 0.04 bpm as the standard deviation of $f[n]$. One example of the signal spectrogram for each level of trace variance is shown in Fig. 4.13 for comparison purposes. Note that a higher frequency energy diffusion is observed when the trace variation increases, as the signal within each analysis window becomes less stationary.

We show the averaged system performance in terms of RMSE, ERATE, and ECOUNT with respect to different combinations of the trace variation level and the selection of k in the second row of Fig. 4.14. The SNR was fixed as -10 dB. From the plots, we observe that the performance decreases when the trace variation level gets higher, especially above level III. Even though the optimal selection of k increases along with the trace variation level, ERATE are controlled below 5% when k is fixed as 4 or 6 with trace variation level lower than V, suggesting the robustness of AMTC again the trace variation levels with a proper selection of the transitional probability parameter.

4.6.3 Impact of Trace Distance

It is challenging for any frequency tracker to accurately distinguish and track two frequency traces that run close to each other. To quantify the distance between two frequency components in a meaningful manner, we first defined a metric called Trace Relative Distance (TRD) as the ratio of the distance of two frequency components in the frequency domain to the mean width of their energy bumps. In Fig. 4.15(a)-(e), we show examples of the spectral distribution when TRD = 0.2,

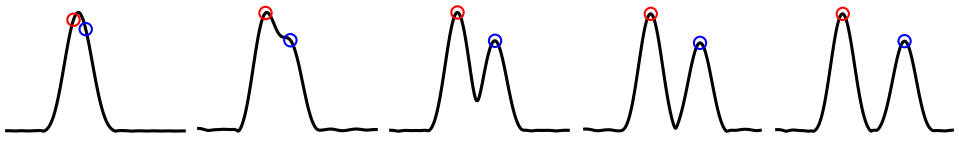


Figure 4.15: Examples of the spectral frame when the Trace Relative Distance (TRD) equals 0.2, 0.4, 0.6, 0.8, and 1, respectively from (a) to (e). Only part of the frame is displayed for better visualization.

0.4, 0.8, and 1, respectively.

We generated 200 trials for each level of TRD using the same generative signal model described in Section 4.5.1.2. No "unvoiced" segment was added to the tested signal and the TRD of two frequency traces was identical over time within each tested signal. We show the averaged system performance with respect to different levels of TRD and SNR in the last row of Fig. 4.14. From the plots, we know that AMTC is capable to track the frequency traces with ERate lower than 3% when SNR ≤ -8 dB, and TRD $>= 0.4$. The estimation result when TRD $= 0.2$ is highly deviated from the ground truth. In this level of TRF, more information or prior knowledge about the frequency components is expected to be incorporated for an improved estimation.

4.7 Discussions

4.7.1 Estimation of the Number of Traces

In previous sections, we presented both the offline and the online-AMTC algorithms with the assumption that the number of traces L is known. In some cases, L is unknown and needs to be estimated. Note that the process of estimating L in

the proposed AMTC system is equivalent to determining the number of iterations AMTC needs to take. The problem is then converted to deciding at which iteration should the AMTC stop. This problem can be solved by testing the hypothesis of the trace existence in the compensated spectrogram image $\mathbf{Z}_{(l)}$ at each iteration l.

In Section 4.3.3, we propose to use the RER measure to detect the existence of a frequency component in each frame. We are motivated by the fact that a low RER measure of a certain frame suggests low probability of the presence of a trace in that frame. Similarly, to test globally the trace existence at lth iteration of AMTC, we propose to evaluate the average of the statistics $\mathrm{RER}_{(l)}$, namely, $\overline{\mathrm{RER}}_{(l)} = \frac{1}{N}\sum_{n=1}^{N} \mathrm{RER}_{(l)}(n)$. As one example shown in Fig. 4.16, the ground truth number of traces in the spectrogram image is 3. We observe a significant drop in $\overline{\mathrm{RER}}_{(l)}$ from $l = 3$ to $l = 4$ in Fig. 4.16(c), when we run the offline-AMTC with four iterations. This observation coincides with the actual absence of the fourth trace. We therefore propose to estimate L as $l - 1$ if at lth iteration, $\overline{\mathrm{RER}}_{(l)}$ is less than a preset threshold. The selection of the threshold value may follow similarly with the optimal selection of Δ_{RER} that is discussed in Section 4.5.1.3.

4.7.2 Signals with Multiple Harmonics

In situations when multiple harmonic traces appear in the spectrogram (*e.g.*, audio signals, Electrocardiography (ECG) signals), AMTC might extract several harmonic traces that originated from one single source. Take the human speech signal as an example. The fundamental frequency range of interest, 85 Hz to 255

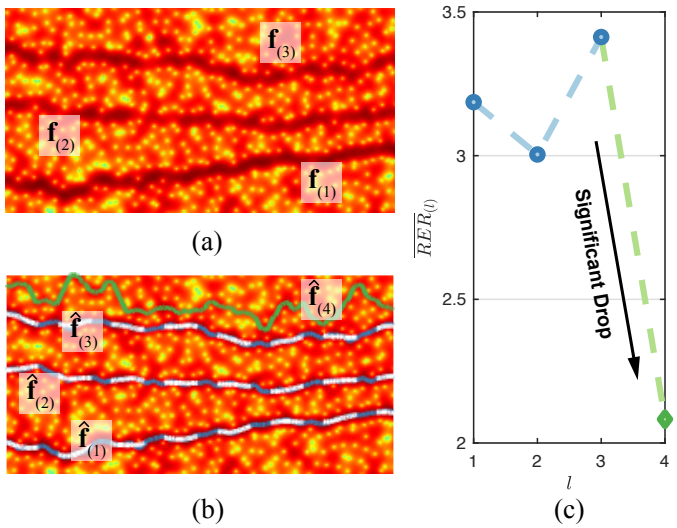


Figure 4.16: (a) Spectrogram image of a synthetic -8dB signal with three frequency components and (b) the same image overlaid with ground truth frequency components (white dashed line), the corresponding frequency estimates $\hat{\mathbf{f}}_{(1:3)}$(blue line) and one additional trace estimate $\hat{\mathbf{f}}_{(4)}$ (green line) using AMTC. (c) the corresponding $\overline{\text{RER}}$ of all four trace estimates in (b).

Hz [112,113], may cover both fundamental frequency components as well as second-order harmonics. For example, a peak in 200 Hz can be considered as the fundamental frequency component of a female speaker, or it can also represent the second-order harmonic of a male speaker. In this regard, the STFT spectrum feature might not be considered as a proper input of a robust fundamental frequency tracker. Instead, this problem can be addressed by introducing several alternative robust spectral features, *e.g.*, the subharmonic summation method [114], the discrete logarithmic Fourier transform [115], and the frequency autocorrelation function [99]. Similar to the idea of harmonic combining algorithm [110] used for ENF case, these methods are capable of combining harmonic spectral features and improving the SNR of the fundamental frequency. The tracking performance is therefore expected to be better by feeding in any of these three features rather than the STFT spec-

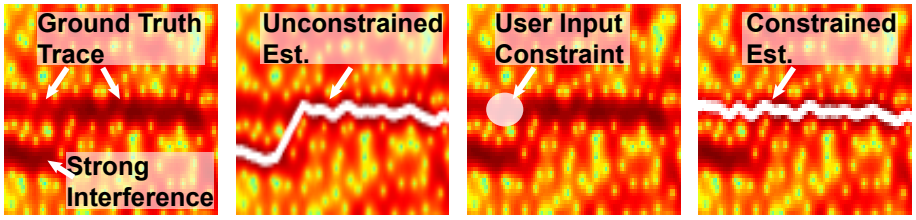


Figure 4.17: (a) Spectrogram of a synthesised signal with ground truth frequency around 95 bpm and strong nearby interference from 0–0.4 min. (b) Overlaid with user input constraint (in filled white circle). (c) Unconstrained trace estimate. (d) Constrained trace estimate.

trogram.

4.7.3 Benefits From Human-in-the-Loop Interactions

AMTC has its limitations in some specific cases. Due to the greedy nature of the searching strategy in each iteration, the algorithm may find incorrect traces when nearby strong interference is presented, or two traces with similar energies running closely in time. We show in Fig. 4.17(b) one such example that AMTC got confused when strong interference is presented near the ground truth frequency trace. Note that without extra information, even humans can make mistakes in this scenario. For some applications when the analysis is performed offline, and people have prior knowledge about the trace shape or part of the trace frequency range, it is beneficial to allow users to input high-level cues [94, 116] to guide our proposed estimator's priority to find the correct trace. As an example, Fig. 4.17(c) shows the user input constraint in the filled white circle for the estimated trace to pass through. Fig. 4.17(d) shows the constrained estimate, which was achieved by scaling up the spectrum entries in the constraint region until the estimated trace passed

through the region. The constrained tracking result reveals that AMTC correctly captured the true trace by shifting its attention from interference to the user-defined region.

4.8 Conclusion and Future Work

In this chapter, we addressed the problem of estimating and tracking of multiple weak frequency components from spectrogram image and proposed both offline and online versions of AMTC algorithm. By using iterative forward and backward dynamic trace estimation and adaptively trace carving, AMTC can provide accurate estimate even for weak frequency traces. Compared to the state-of-the-art spectrogram tracking methods, our method shows robustness and consistency on both synthetic and real-world data with different levels of noise in an efficient manner.

Chapter 5: Learning Your Heart Actions From Pulse: ECG Waveform Reconstruction From PPG

5.1 Introduction

Cardiovascular disease (CVD) has become the leading cause of human death – about 32% of all deaths worldwide in 2017 according to the Global Burden of Disease results [117]. Statistics also reveal that young people, especially athletes, are more prone to sudden cardiac arrests than before [118]. Those life-threatening cardiovascular diseases often happen outside hospitals, and the patients are recommended by cardiologists to be monitored in a long-term continuous manner [119].

The electrocardiogram (ECG) has become the most commonly used cardiovascular diagnostic procedure and is a fundamental tool of clinical practice [13,120]. Many modern wearable ECG systems have been developed in recent decades. They are simpler and more reliable than before, weighing only a fraction of a pound. However, the material used to provide good signal quality with the electrode may cause skin irratation and discomfort during prolonged use, which restricts the long-term use of the devices.

The photoplethysmogram (PPG) is a noninvasive circulatory signal related to

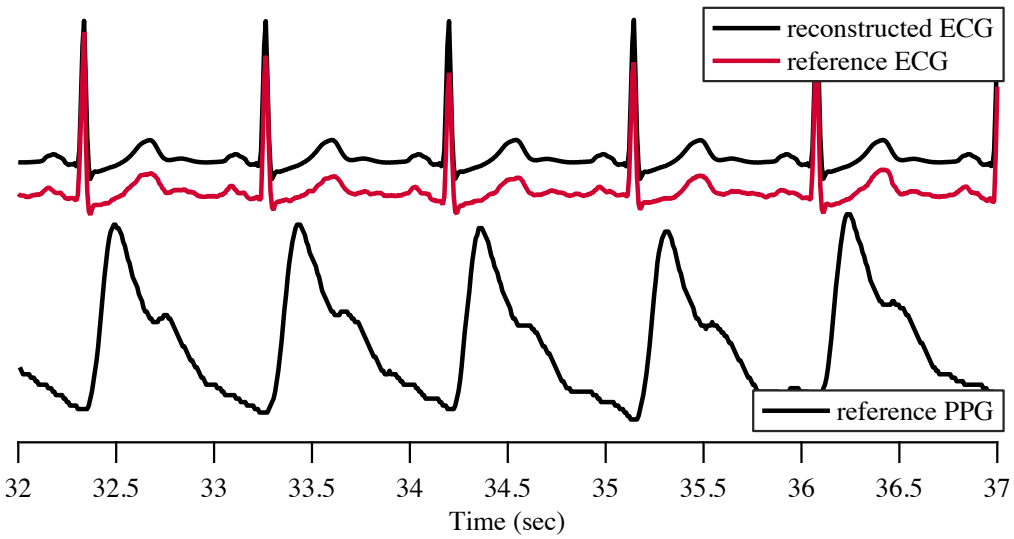


Figure 5.1: Upper: a five-second reconstructed ECG signal in test set (black line) vs. the reference ECG signal (red line) using the data from the MIMIC-III database [122]. Lower: the corresponding PPG signal used to reconstructed the ECG signal.

the pulsatile volume of blood in tissue [121]. In common PPG modalities, tissue is irradiated by a light-emitting diode, and the light intensity is measured by a photodetector on the same or other side of the tissue. A pulse of blood modulates the light intensity at the photodetector and by convention the PPG is inverted to correlate positively with blood volume [121]. Compared with ECG, PPG is easier to set up, more convenient, and more economical. PPG is nearly ubiquitous in clinics and hospitals in the form of finger/toe clips and oximeters and has increasing popularity in the form of consumer-grade wearable devices that offer continuous and long-term monitoring capability and do not cause skin irritations.

The PPG and ECG signals are intrinsically correlated, considering that the variation of the peripheral blood volume is influenced by the left ventricular myocardial activities, and these activities are controlled by the electrical signals originating from the sinoatrial (SA) node. The timing, amplitude, and shape characteristics

of the PPG waveform contain information about the interaction between the heart and connective vasculature. These features have been translated to measure heart rate, heart rate variability, respiration rate [123,124], blood oxygen saturation [125], blood pressure [126], and to assess vascular function [127–129]. As the prevailing use of wearable device capturing users' PPG signal on a daily basis, we are inspired to utilize this correlation to not only infer the ECG parameters but also reconstruct the ECG waveform from the PPG measurement. This exploration, if successful, can provide a low-cost ECG screening for continuous and long-term monitoring and take advantage of both the rich clinical knowledge base of ECG signal and the easy accessibility of the PPG signal.

There is a very limited amount of prior art addressing the ECG reconstruction/inference problem mentioned above. In [130], the authors trained several classifiers to infer the quantized level of RR, PR, QRS, and QT interval parameters, respectively, from selected time domain and frequency domain features of PPG. Even though the system yields 90% accuracy on a benchmark hospital dataset, the capability confined to only inferring ECG parameters may restrict the broad adoption of the prior art.

In this paper, we propose to estimate the waveform of the ECG signal using PPG measurement by learning a signal model that relates the two time series. We first preprocess the ECG and PPG signal pairs to obtain temporally aligned and normalized sets of signals. We then segment the signals into pairs of cycles and train a linear transform that maps the discrete cosine transform (DCT) coefficients of the PPG cycle to those of the corresponding ECG cycle. The ECG waveform is

then obtained via the inverse DCT. We evaluate our methodology on two publicly available datasets as well as a self-collected datasets which in total contains 147 subjects with a wide variety of age, weight, and health conditions. Experiment results show that the proposed method can achieve a high accuracy greater than 0.92 in averaged correlation in each datasets when the model is trained in a subject specified manner. Fig. 5.1 shows a five-second reconstructed ECG signal in test set with the proposed method. Note that the reconstructed ECG signal is almost identical with the reference one.

The significance of this work is threefold. First, the statistics of the system performance metrics evaluated on three databases show that our proposed system can reconstruct the ECG signal accurately. Second, to the best of our knowledge, this is the first work which addresses the problem of inferring ECG signal from the PPG signal. It may open up a new direction for cardiac medical practitioners, wearable technologists, and data scientists to leverage a rich body of clinical ECG knowledge and transfer the understanding to build a knowledge base for PPG and adta from wearable devices. Third, the technology may enable a more user-friendly, low-cost, continuous and long-term cardiac monitoring that support and promotes public health, especially for people with special needs.

The rest of this chapter is organized as follows. In Section 5.2, we first mathematically model the relationship between the ECG and PPG signals. In Section 5.3, we introduce the proposed system based on the proposed signal model in Section 5.2. We test the system and report the experimental results in Section 5.4, and discuss the possible extension and the limitations of the proposed method in Section 5.5.

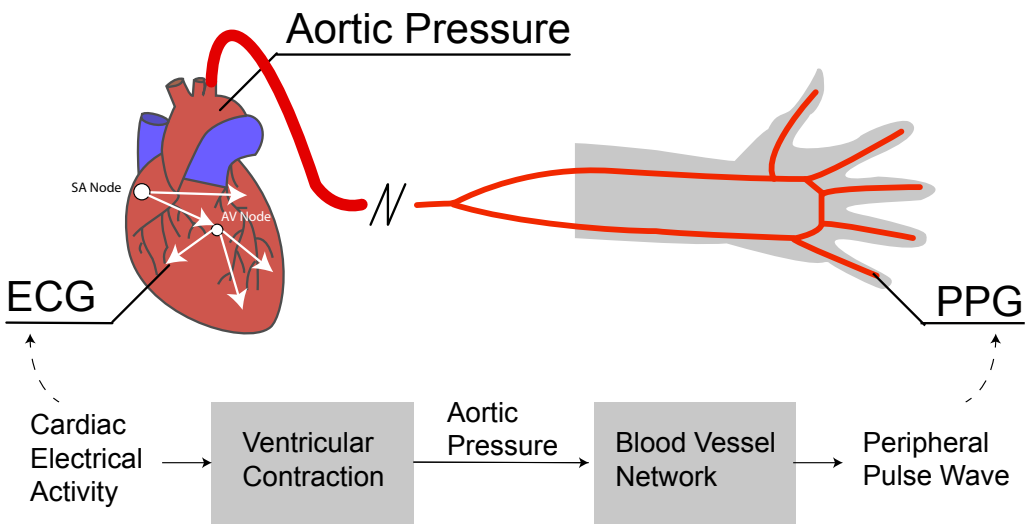


Figure 5.2: A visualization of the relationship between the ECG, the aortic pressure, and the PPG.

The conclusion is drawn in Section 5.6.

5.2 A Cycle-wise Signal Model of PPG and ECG

As shown in Fig. 5.2, during each cardiac cycle, the atrioventricular (AV) node receives the electrical signals originated from the SA node. The AV node then transmits this bio-electrical signals through the bundle of His, left bundle branches, and Purkinje fibers to the left ventricular myocardium, causing the depolarization and contraction of the left ventricle. As a result of this process, the pressure of the left ventricle rises and exceeds the aortic pressure, causing the opening of the aortic valves, bloodflow from the left ventricle into the aorta, and the corresponding rise of the aortic pressure. Upon closure of the aortic valves, the generated pulse wave transmits the blood to the peripheral parts of our body, such as finger tips or toes, through a network of blood vessels.

5.2.1 The ECG Signal and the Aortic Pressure

Consider one specific cardiac cycle. We denote the uniformly sampled cardiac electrical activity as $e(t)$, $t \in [1, L]$, where L is the total number of samples within the cycle. We denote the electrocardiogram measurement recording the potential difference between two electrodes placed on the surface of the skin as $c_y(t)$. Taking into account the human body electrical resistance and the sensor noise, we model the ECG signal as:

$$c_y(t) = \alpha e(t) + v_y(t), \tag{5.1}$$

where α is a subject-specific modulus reflecting the resistance of the electrical path between the heart and the skin surface. $v_y(t)$ denotes the ECG sensor noise, which is modeled as a zero-mean white Gaussian process.

The contraction and relaxation of the heart muscles follow the bio-electrical activities of the heart. These biomechanical activities further modulate the aortic pressure via the opening and closing of the aortic valves. The aortic pressure, denoted as $p_a(t)$, is thus highly correlated with the cardiac electrical activities $e(t)$. To model this correlation, we first map both $e(t)$ and $p_a(t)$ to their frequency domain via type II DCT, as DCT has the potential to provide a compact and effective representation of the signals [131]. We then model the relationship of the two signals with a linear transform from the DCT domain of $e(t)$ to that of $p_a(t)$ as:

$$\mathbf{P}_a = \mathbf{H}\mathbf{E}, \tag{5.2}$$

where $\mathbf{E}$, $\mathbf{P}_a \in \mathbb{R}^{L\times 1}$ are the DCT-II coefficients of $e(t)$ and the aortic pressure $p_a(t)$ respectively. $\mathbf{H} \in \mathbb{R}^{L\times L}$ is the transition matrix.

5.2.2 The Pulse Wave and the PPG Signal

When the pulse wave and blood flow travel through our body from the aorta to a peripheral site, it might experience different interactions with the blood vessels, for instance splitting and pushing. Assuming the structure of the blood vessel path of a specific person is time-invariant, we can model this channel from the aorta to the peripheral site as a linear-time-invariant system. We denote the peripheral pulse signal at a specific body site as $p_p(t)$. We write $p_p(t)$ according to the prior channel assumption as:

$$p_p(t) = b(t) \circledast p_a(t) + v_b(t), \tag{5.3}$$

where $b(t)$ denotes the impulse response of the channel of blood vessels, and $\circledast$ denotes a symmetric convolution process. $v_b(t)$ is the zero-mean white Gaussian noise, capturing the variance of this model.

Without loss of generosity, we assume the PPG sensor attached to the same peripheral site works in the transmission mode. It means that the photodetector of the PPG sensor is on the other side of the tissue with the light-emitting diode. We assume the light source has a constant intensity I on the spectral range of the receiver side. We further assume no relative motion between the attached skin and the photodetector is not influenced by the possible environmental illuminations. We

write the PPG measurement, denoted as $c_x(t)$, as:

$$c_x(t) = I\left[\tau_0 + \tau_1 p_p(t)\right] + v_x(t), \tag{5.4}$$

where τ_0 and τ_1 denote the stationary tissue transmission strength and relative pulsatile strength in the PPG light receiver side, respectively. $v_x(t)$ denotes the PPG sensor noise, which is modeled as a zero-mean white Gaussian process. We can rewrite (5.4) as:

$$c_x(t) = I_1 p_p(t) + I_0 + v_x(t), \tag{5.5}$$

where $I_1 = I\tau_1$ and $I_1 = I\tau_0$.

5.2.3 The Inverse Model from PPG to ECG

According to the property of the symmetric convolution [132], we can rewrite (5.3) as:

$$\mathbf{P}_p = \mathbf{B}\mathbf{P}_a + \mathbf{V}_b, \tag{5.6}$$

where $\mathbf{P}_p$, $\mathbf{P}_a$, and $\mathbf{V}_b$ are the DCT-II coefficients of $p_p(t)$, $p_a(t)$, and $v_b(t)$ respectively. $\mathbf{B} \triangleq \text{diag}(B_1, B_2, ..., B_L) \in \mathbb{R}^{L\times L}$, where B_k denotes the kth DCT-I coefficient of $b(t)$. We next apply a type II DCT on both sides of (5.1) and (5.5) and we arrive at:

$$\mathbf{C}_y = \alpha\mathbf{E} + \mathbf{V}_y \tag{5.7}$$

$$\mathbf{C}_x = I_1\mathbf{P}_p + \mathbf{I}_0 + \mathbf{V}_x \tag{5.8}$$

where $\mathbf{C}_y$, $\mathbf{V_y}$, $\mathbf{C}_x$, $\mathbf{I}_0$ and $\mathbf{V}_x$ denotes the DCT-II coefficients of $c_y(t)$, $v_y(t)$, $c_x(t)$, constant function I_0 and $v_x(t)$ respectively. Assuming the nonsingularity of the matrix B and H and according to (5.2), (5.6), (5.7), and (5.8), we have:

$$\mathbf{C}_y = \mathbf{F}\mathbf{C}_x + \mathbf{C}_0 + \mathbf{V} \tag{5.9}$$

where $\mathbf{F} \triangleq \frac{\alpha \mathbf{H}^{-1}\mathbf{B}^{-1}}{I_1}$, $\mathbf{C}_0 \triangleq -\frac{\alpha \mathbf{H}^{-1}\mathbf{B}^{-1}}{I_1}\mathbf{I}_0$, and $\mathbf{V} \triangleq \mathbf{V}_y - \alpha \mathbf{H}^{-1}\mathbf{B}^{-1}\left(\frac{\mathbf{V}_x}{I_1} + \mathbf{V}_b\right)$. When we look individually at each element of $\mathbf{C}_y$, we have:

$$C_y(k) = \mathbf{F}(k)\mathbf{C}_x + C_0(k) + V(k), \quad k \in [1, L], \tag{5.10}$$

where $\mathbf{F}(k)$ is the kth row of matrix $\mathbf{F}$; $C_0(k)$ and $V(k)$ denote the kth element of $\mathbf{C}_0$ and $\mathbf{V}$ respectively. We know $V(k)$ is a zero-mean Gaussian random variable, as it is a linear combination of zero-mean Gaussian random variables from v_y, v_b, and v_x. From the model listed in (5.10), we are motivated to explore the linear relationships between the DCT coefficients of PPG signal and those of the ECG signals.

5.3 Methodology

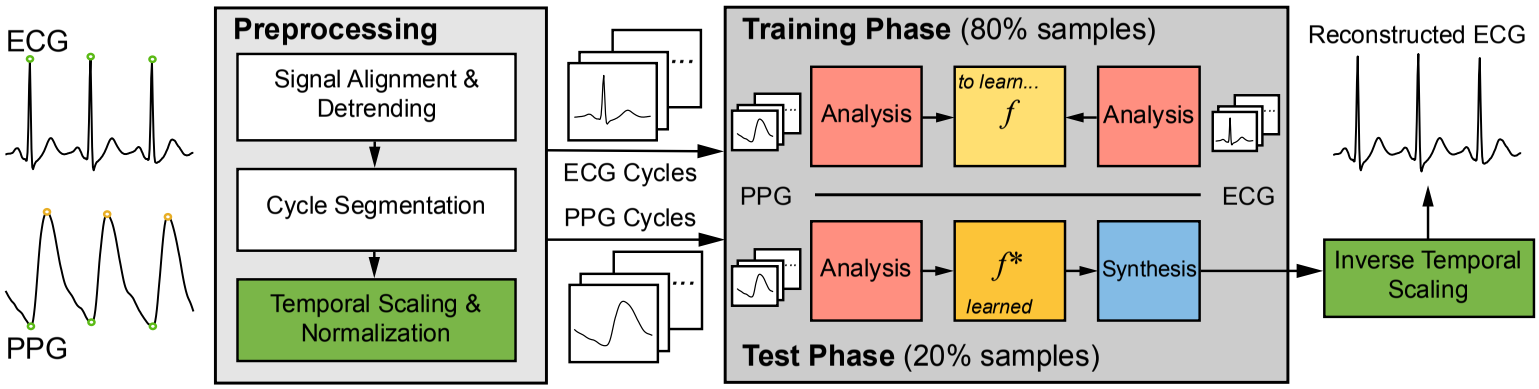


Figure 5.3: Flowchart of the proposed system. The ECG and PPG signals are first preprocessed to obtain physically aligned and normalized pairs of cycles. The selected DCT coefficients of 80% pairs of cycles are used for training a linear transform f which is used in the test phase to reconstruct the ECG signals.

According to the signal model we discussed in the previous section, we propose a system which learns the linear transform $\mathbf{F}$. The pipeline of the system is shown in Fig. 5.3. The pair of PPG and ECG signals are first preprocessed into pairs of synchronized cycles. The cycle pairs are then fed into the training system to learn the transform matrix. We discuss further the details of the system as follows.

5.3.1 Preprocessing: Cycle-Wise Segmentation

The goal of preprocessing ECG and PPG signals is to obtain temporally aligned and normalized pair of signals, so that the critical temporal features of both waveforms are synchronized to facilitate our investigation. As shown in Fig. 5.3, the preprocessing phase contains data alignment, signal detrending, cycle-wise segmentation, temporal scaling, and normalization stages that be explained as follows.

Data alignment Considering possible misalignment of the signal pair in each trial, we perform a two-level signal alignment to obtain physically aligned signal pairs. We first estimate the signal delay in the cycle level using the peak features as they are the most distinguishable features within the cycle. We then align the signals to the sample level based on their physical correspondence.

Suppose we have a pair of almost simultaneously recorded PPG and ECG signals, denoted as $\mathbf{x} \in \mathbb{R}^T$ and $\mathbf{y} \in \mathbb{R}^T$ respectively. We name the coordinate of the systolic peak in the ith cycle of PPG as $t_{\text{sp}}(i)$ and the R peak of ECG as $t_{\text{rp}}(i)$. The cycle delay n_{delay} is searched for in a set $\mathbb{D} \triangleq [-k, k]$, where the search radius $k = 5$ as we expect the cycle delay to be small. For each evaluated

$n \in \mathbb{D}$, we first preliminarily align the signal with respect to $t_{\text{sp}}(1-n\cdot\mathbb{1}(n<0))$, and $t_{\text{rp}}(1-n\cdot\mathbb{1}(n>0))$. The aligned coordinates of PPG and ECG peaks are $\{t'_{\text{sp}}(n)\}$ and $\{t'_{\text{rp}}(n)\}$. We then estimate the cycle delay $\hat{n}_{\text{delay}}$ by solving the following problem:

$$\hat{n}_{\text{delay}} = \underset{n\in\mathbb{D}}{\operatorname{argmin}} \sum_{i=1}^{i=N-k} \left| t'_{\text{sp}}(i - n\cdot\mathbb{1}(n<0)) - t'_{\text{rp}}(i + n\cdot\mathbb{1}(n>0)) \right|, \tag{5.11}$$

where N is total number of cycles, $\mathbb{1}$ is the indicator function. We align the signals by shifting PPG signal so that the systolic peaks of PPG and the R peaks of ECG are temporally matched.

Next, we align the signal to the sample level according to the R peak of the ECG and the onset point of PPG in the same cycle (the local minimum point before the systolic peak), considering that the R peak corresponds approximately to the opening of the aortic valve, and the onset point of PPG indicates the arrival of the pulse wave [121]. In this way, we eliminate the pulse transit time and align the signals.

Detrending The non-stationary trend in both signals can be problematic for temporal pattern analysis. Such slowing-varying trend can be estimated and then subtracted from the original signals. The trend is assumed to be a smooth, unknown version of $\mathbf{x}$ and $\mathbf{y}$ with a property that its accumulated convexity measured for every point on the signal is as small as possible, namely,

$$\hat{\mathbf{x}}_{\text{trend}} = \underset{\hat{\mathbf{x}}}{\operatorname{argmin}} \left\|\mathbf{x} - \hat{\mathbf{x}}\right\|_2^2 + \lambda \left\|\mathbf{D}_2\hat{\mathbf{x}}\right\|_2^2, \tag{5.12}$$

where $\mathbf{x}$ is the original signal, $\hat{\mathbf{x}}_{\text{trend}}$ is the estimated trend in $\mathbf{x}$, λ is a regularization parameter controlling the smoothness of the estimated trend, and $\mathbf{D}_2 \in \mathbb{R}^{T\times T}$ is a Toeplitz matrix that acts as a second-order difference operator. The closed-form solution of (5.12) is $\hat{\mathbf{x}}_{\text{trend}} = (\mathbf{I}+\lambda\mathbf{D}_2^{\intercal}\mathbf{D}_2)^{-1}\mathbf{x}$, where $\mathbf{I}$ is the identity matrix, Hence, the detrended signal is $\tilde{\mathbf{x}} = \mathbf{x} - \hat{\mathbf{x}}_{\text{trend}}$, and similarly, $\tilde{\mathbf{y}} = \mathbf{y} - \hat{\mathbf{y}}_{\text{trend}}$.

Segmentation & Normalization After the signal alignment and detrending, we segment each cycle of the signal $\tilde{\mathbf{x}}$ and $\tilde{\mathbf{y}}$ to prepare for the learning phase. In our experiment, we introduce the following two cycle segmentation schemes: SR and R2R.

- *SR*: we segment the signal according to the points which are 1/3 of the cycle length to the left of the R peaks of the ECG signal. We call this scheme SR as it approximately captures the standard shape of sinus rhythm.
- *R2R*: we segment the signal according to the location of the R peak of the ECG signal to mitigate the reconstruction error in the QRS complex.

After the segmentation, we temporally scale each cycle sample via linear interpolation to make it of length L in order to mitigate the influence of the heart rate variation. We then normalize each cycle by subtracting the sample mean and dividing by the sample standard deviation. We denote the normalized PPG and ECG cycle samples as $\mathbf{C}_x$, $\mathbf{C}_y \in \mathbb{R}^{N\times L}$.

5.3.2 Learning a Linear Transform for DCT Coeffients

The right part of Fig. 5.3 shows our proposed learning framework. In the training phase, we build and train a linear transform to model the relation between the DCT coefficients of PPG and ECG cycles. We then use the trained matrix to reconstruct the ECG waveform in the test phase.

Specifically, we first perform cycle-wise DCT on $\mathbf{C}_x$ and $\mathbf{C}_y$, which yields $\mathbf{X}$, $\mathbf{Y} \in \mathbb{R}^{N\times L}$. Then the first L_x, L_y DCT coefficients of $\mathbf{X}, \mathbf{Y}$ are selected to represent the corresponding waveform as the signal energy is concentrated mostly on the lower frequency components per our observation. We denote them as $\tilde{\mathbf{X}} \in \mathbb{R}^{N\times L_x}$ and $\tilde{\mathbf{Y}} \in \mathbb{R}^{N\times L_y}$. We next separate $\tilde{\mathbf{X}}$ and $\tilde{\mathbf{Y}}$ into training and test sets as $\mathbf{X}_{\text{train}} \in \mathbb{R}^{N_{\text{train}}\times L_x}$, $\mathbf{Y}_{\text{train}} \in \mathbb{R}^{N_{\text{train}}\times L_y}$ and $\mathbf{X}_{\text{test}} \in \mathbb{R}^{N_{\text{test}}\times L_x}$, $\mathbf{Y}_{\text{test}} \in \mathbb{R}^{N_{\text{test}}\times L_y}$, where $N_{\text{train}} + N_{\text{test}} = N$.

In the training process, a linear transform matrix $f^* \in \mathbb{R}^{L_x\times L_y}$ that maps from PPG to ECG DCT coefficients is learned through ridge regression as described below:

$$f^* = \underset{f}{\operatorname{argmin}} \left\|\mathbf{X}_{\text{train}}\ f - \mathbf{Y}_{\text{train}}\right\|_{\text{F}}^2 + \gamma \left\|f\right\|_{\text{F}}^2, \tag{5.13}$$

where $\|\cdot\|_{\text{F}}$ denotes the Frobenius norm of a matrix, and $\gamma > 0$ is a complexity parameter that controls the shrinkage of f toward zero. The idea of penalizing the sum-of-squares of f is to reduce the variance of the predictions and to avoid overfitting [133]. The analytic solution to (5.13) is $f^* = (\mathbf{X}_{\text{train}}^{\intercal}\mathbf{X}_{\text{train}} + \gamma\mathbf{I})^{-1}\mathbf{X}_{\text{train}}^{\intercal}\mathbf{Y}_{\text{train}}$, where $\mathbf{I}$ is the identity matrix.

In the test phase, we apply the optimal linear transform f^* learned in training

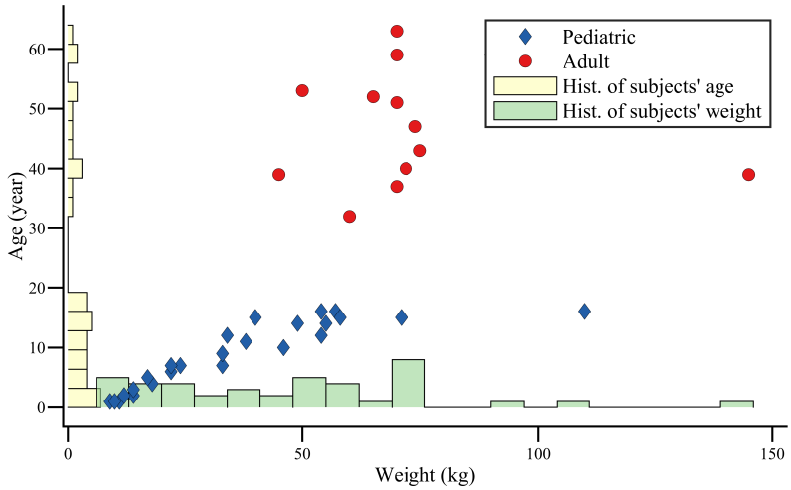


Figure 5.4: Scatter plot of the age vs. weight of all the subjects in the database [124]. The bar plot on each x and y axis shows the histogram of subject's weight and age, respectively.

stage on $\mathbf{X}_{\text{test}}$ and estimate the corresponding DCT coefficients of ECG cycles. We denote the estimate as $\hat{\tilde{\mathbf{Y}}}_{\text{test}} \triangleq \mathbf{X}_{\text{test}}\ f^*$. To reconstruct ECG, we first augment each row of $\hat{\tilde{\mathbf{Y}}}_{\text{test}}$ to be in the same dimension as L (by padding zeros). We denote the zero-padded matrix as $\hat{\mathbf{Y}}_{\text{test}} \in \mathbb{R}^{N_{\text{test}} \times L}$. We then apply inverse DCT to each row of $\hat{\mathbf{Y}}_{\text{test}}$, interpolate the resulted matrix row by row to its original temporal scale, and concatenate the inversely scaled pieces of cycles to obtain the reconstructed ECG signal $\hat{\mathbf{y}}_{\text{test}}$.

5.4 Experiments

5.4.1 Experiment I: TBME-RR database

We first use the Capnobase TBME-RR [124] to evaluate the performance of the proposed system. The dataset contains 42 eight-min sessions of simultaneously

recorded PPG and ECG measurements from 29 pediatric and 13 adults [1], sampled at 300 Hz. The 42 cases were randomly selected from a larger collection of physiological signals collected during elective surgery and routine anesthesia for the purpose of development of monitoring algorithms in adults and children. Each recorded session corresponds to a unique subject. The PPG signal was acquired on subjects' fingertips via a pulse oximeter. As shown in Fig. 5.4, the dataset has a wide variety of patient's age and weight and is thus a favorable dataset for testing the performance of our proposed system.

We first pruned the signals according to the human-labeled artifact segments and processed the pairs of ECG and PPG signal using the method introduced in Section 5.3.1 to obtain aligned and normalized pairs of the signal cycles. We set $L = 300$ and $L_y = 100$, as most of the diagnostic information of ECG is contained below 100 Hz [13]. We set $\lambda = 500$ and $\gamma = 10$ empirically as they offer the best regularization results in the tasks. In order to test the consistency of the system, we selected the first 80% of each session as the training set and the rest for testing. We use the following two metrics to evaluate the system performance in the test set:

- Relative root mean squared error:

$$\text{rRMSE} = \frac{\|\mathbf{y}_{\text{test}} - \hat{\mathbf{y}}_{\text{test}}\|_2}{\|\mathbf{y}_{\text{test}}\|_2}, \tag{5.14}$$

[1] Note that the recording in this database is of high signal quality. In cases when the signal is corrupted by noise or subject's motion artifacts, a denoising process is needed to clean the signal before the preprocessing stage.

- Pearson's correlation coefficient:

$$\rho = \frac{(\mathbf{y}_{\text{test}} - \bar{y}_{\text{test}})^{\intercal}(\hat{\mathbf{y}}_{\text{test}} - \bar{\hat{y}}_{\text{test}})}{\|\mathbf{y}_{\text{test}} - \bar{y}_{\text{test}}\|_2 \left\|\hat{\mathbf{y}}_{\text{test}} - \bar{\hat{y}}_{\text{test}}\right\|_2}, \tag{5.15}$$

where $\mathbf{y}_{\text{test}}$, $\bar{\hat{y}}_{\text{test}}$, and $\bar{y}_{\text{test}}$ denote the ECG signal in test set, the average of all coordinates of the vectors $\hat{\mathbf{y}}_{\text{test}}$ and $\mathbf{y}_{\text{test}}$ respectively.

In this study, we evaluate the system in the following two training modes:

- *Subject Independent (SI)* mode: we trained a single linear transform f^* using all the training data, i.e., the trained model is independent with each subject in the dataset.

- *Subject Dependent (SD)* mode: a linear transform f^* is trained and tested in each session. In this way, we obtain subject dependent model for each individual.

We first cross-validated the number of DCT coefficients of the PPG signal L_x used in the learning system. It is clear that the more variables as predictors, i.e., more PPG DCT coefficients are used in the linear system, the better the performance can be achieved in training. However, we can observe from Fig. 5.5 that the performance of our system in the test set using either SR and R2R becomes saturated as L_x gets larger from approximate 18 and 12 in SI and SD mode respectively. The trends of convergence in both mode suggest potential model overfitting. Another observation is the convergence rate is slower in SI mode compared with SD mode. This is expected because the data diversity is much higher in SI mode than

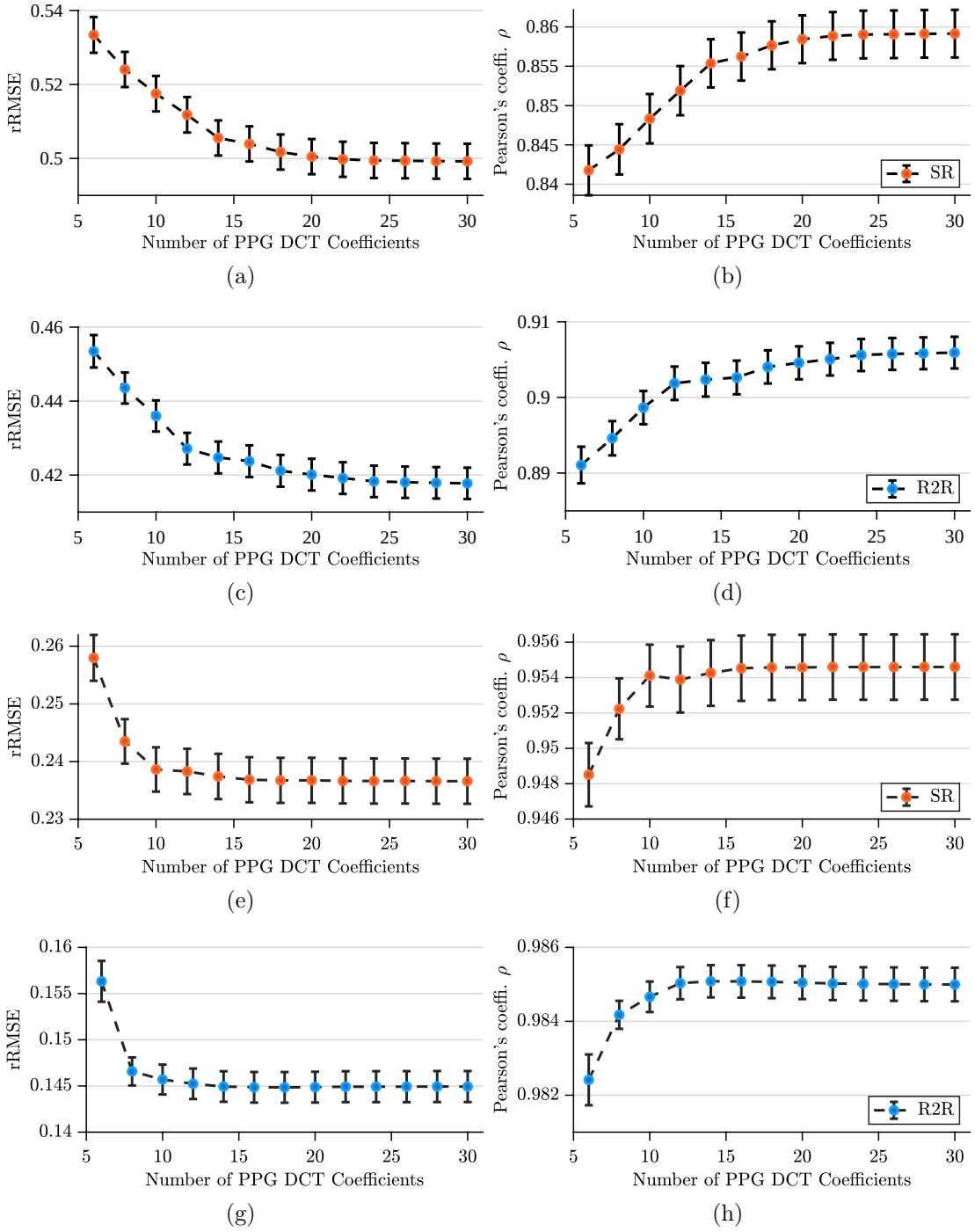


Figure 5.5: The line plots give the average of rRMSE in (a), (c), (e), and (g) and ρ in (b), (d), (f), and (h) of all sessions in the test set for different choices of number of PPG DCT coefficient m_1 using SR (a), (b), (e), and (h) and R2R (c), (d), (g), (h) segmentation scheme and SI (a)–(d) and SD (e)-(h) model respectively. The vertical bars at each data point shows 3% standard deviation above and below the sample mean.

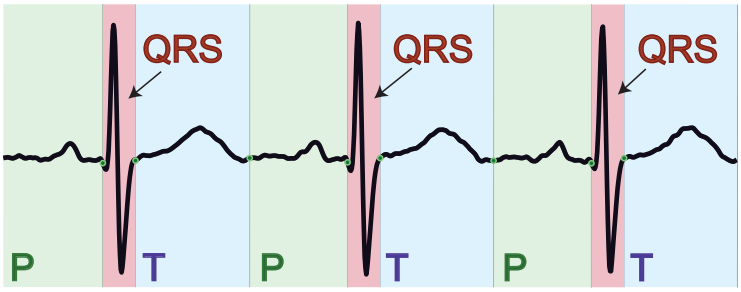


Figure 5.6: An example of the ECG segmentation result on three cycles of the signal in the 1st session of TBME-RR database. The green, red, and blue areas in the plot denote the estimated P waves, the QRS waves, and the T waves, respectively. For each cycle, the ratio between the duration of the QRST wave is 3/2 of the duration of the T wave.

that in SD mode, and more variables are needed to capture the additional variance in SI mode. $L_x = 18$ in SI mode and $L_x = 12$ in SD mode are thus favorable to us as the system has comparable performance and the model is simpler than those with larger L_x.

The norm of one cycle of ECG signal is dominated by that of the QRS complex. This condition might lead to insufficient evaluation on P wave and T wave of the ECG signal. To address this problem, we further separate the ECG cycle into segments of P wave, QRS wave, and T wave. The evaluation is performed in terms of rRMSE and ρ on each segment as well as the whole length of signal. Specifically, we adopted the QRS detection algorithm introduced in [134] to locate the onset and end point of the QRS complex. We empirically select the 60% point between the onset points of two adjacent QRS complexes as the separating point for the P and T wave. Fig. 5.6 shows one example of the ECG segmentation result on three cycles sampled from the first subject in the database. Note that the onset and end point of all waves in each cycle are accurately estimated.

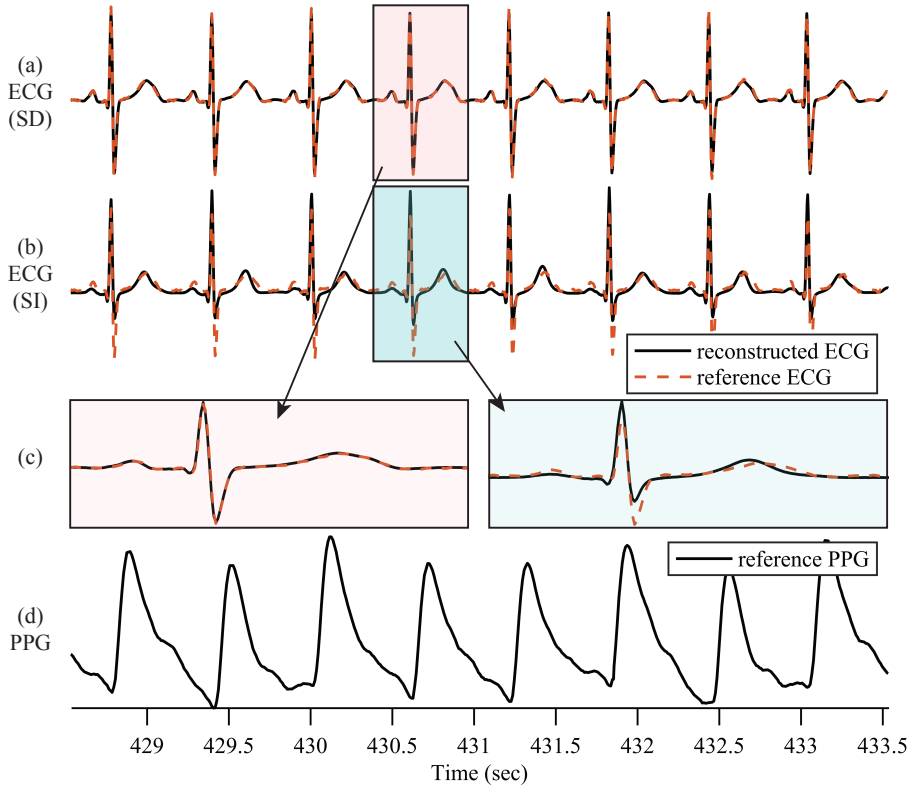
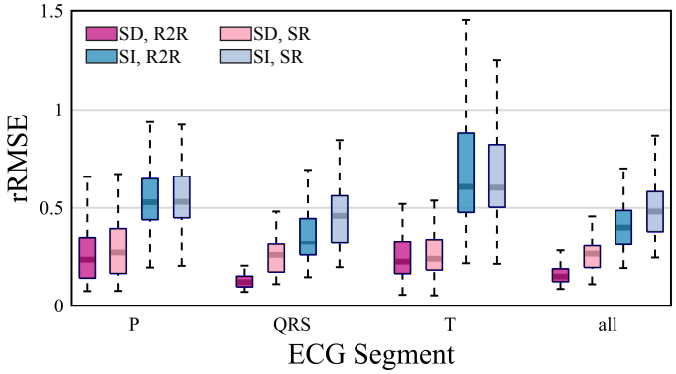
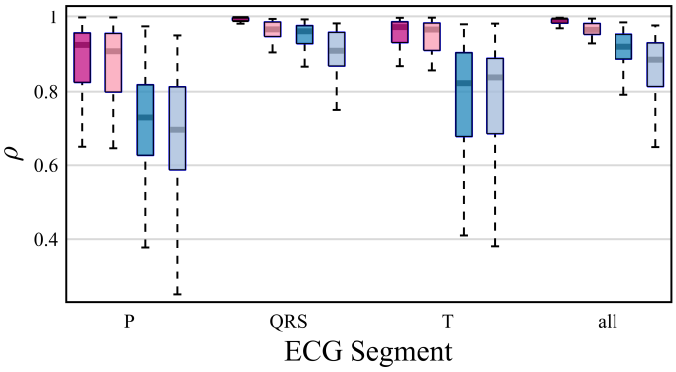


Figure 5.7: The reconstructed ECG (black line) in (a) SD and (b) SI and the reference ECG (orange dashed line) waveform of the last 5 seconds of the first session (age: 4 years old, weight: 18 kg) in TBME-RR database. Zoomed-in version of the shaded cycle in each mode is shown in (c). The corresponding PPG waveform is shown in (d).



(a)



(b)

Figure 5.8: Comparison of the performance of the proposed method in test set of the TBME-RR database in different combinations of the SR or R2R segmentation schemes and SD or SI test modes evaluated at P, QRS, T, and all waves. Statistics of the (a) rRmse and (b) ρ are summarized using the box plots.

We listed the average performance using R2R and SR cycle segmentation schemes in different training modes in Table 5.1 and plotted the results using the box plots in Fig. 5.8. In Table 5.1, the performance is characterized by the sample mean and standard deviation of rRMSE and ρ on P, QRS, T, and all waves, where all wave denotes the whole length of the signal including every wave. From the statistics, we learn that overall R2R gives better performance than SR, and model trained in SD mode gives better performance compared with that trained in SI mode in this dataset as possible subject differences in terms of $\mathbf{H}$ in (5.2) and $b(t)$ in (5.3) are expected. In general, R2R outputs comparable results on P and T waves compared with SR, whereas R2R outperforms SR on QRS and all waves. In SD mode, the average performance in ρ on T wave is about 0.92 using R2R and 0.90 using SR, much higher values than those on P wave. There are two possible reasons that explain this result. First, compared with the QRS and T waves, the amplitude of P wave is much smaller. As a result, the P wave becomes more sensitive to the noise compared with the T wave. Second, the shape of T wave signifies the repolarization of the ventricles, and the ventricular repolarization is correlated with the shape of dichrotic notch in PPG signal. This is because during the ventricular repolarization process, the closure of the aortic valve is associated with a small backflow of blood into the ventricle and a characteristic notch in the aortic pressure tracings. This connection between the P wave of ECG and the dichrotic notch of PPG may facilitate the system performance on P wave.

		rRmse				ρ			
		P	QRS	T	all	P	QRS	T	all
TBME-RR (*SI*)									
	lr	0.563 (0.197)	0.465 (0.173)	0.736 (0.422)	0.499 (0.142)	0.660 (0.208)	0.879 (0.102)	0.717 (0.262)	0.859 (0.091)
SR	ridge	0.561 (0.199)	0.465 (0.173)	0.734 (0.423)	0.499 (0.141)	0.659 (0.210)	0.880 (0.101)	0.718 (0.267)	0.859 (0.090)
	lasso	0.565 (0.200)	0.468 (0.173)	0.734 (0.421)	0.502 (0.140)	0.652 (0.210)	0.879 (0.102)	0.718 (0.266)	0.858 (0.090)
	lr	0.564 (0.202)	0.359 (0.139)	0.726 (0.434)	0.418 (0.124)	0.686 (0.203)	0.937 (0.059)	0.709 (0.261)	0.906 (0.061)
R2R	ridge	0.562 (0.204)	0.360 (0.142)	0.722 (0.424)	0.418 (0.126)	0.687 (0.204)	0.937 (0.060)	0.713 (0.264)	0.906 (0.063)
	lasso	0.564 (0.204)	0.363 (0.142)	0.721 (0.435)	0.420 (0.125)	0.684 (0.203)	0.937 (0.060)	0.711 (0.267)	0.905 (0.062)
TBME-RR (*SD*)									
	lr	0.329 (0.288)	0.275 (0.153)	0.285 (0.213)	0.284 (0.155)	0.825 (0.206)	0.946 (0.071)	0.898 (0.180)	0.947 (0.068)
SR	ridge	0.289 (0.165)	0.273 (0.134)	0.285 (0.173)	0.277 (0.121)	0.836 (0.179)	0.949 (0.061)	0.906 (0.146)	0.951 (0.052)
	lasso	0.314 (0.155)	0.291 (0.139)	0.306 (0.170)	0.294 (0.122)	0.810 (0.188)	0.943 (0.065)	0.899 (0.145)	0.947 (0.054)
	lr	0.283 (0.180)	0.131 (0.049)	0.285 (0.233)	0.170 (0.080)	0.859 (0.159)	0.990 (0.011)	0.906 (0.173)	0.982 (0.025)
R2R	ridge	0.273 (0.165)	0.128 (0.039)	0.275 (0.178)	0.165 (0.061)	0.869 (0.141)	0.991 (0.006)	0.917 (0.131)	0.984 (0.014)
	lasso	0.287 (0.158)	0.135 (0.038)	0.295 (0.173)	0.175 (0.058)	0.856 (0.137)	0.990 (0.006)	0.911 (0.131)	0.983 (0.012)

Table 5.1: The system performance in test set of the TBME-RR database in terms of sample mean and standard deviation (in paranthesis) of rRMSE and ρ. R2R segmentation using difference combinations of traning mode (SI/SD), segmentation schemes (SR/R2R) and linear regression methods (lr/ridge/lasso).

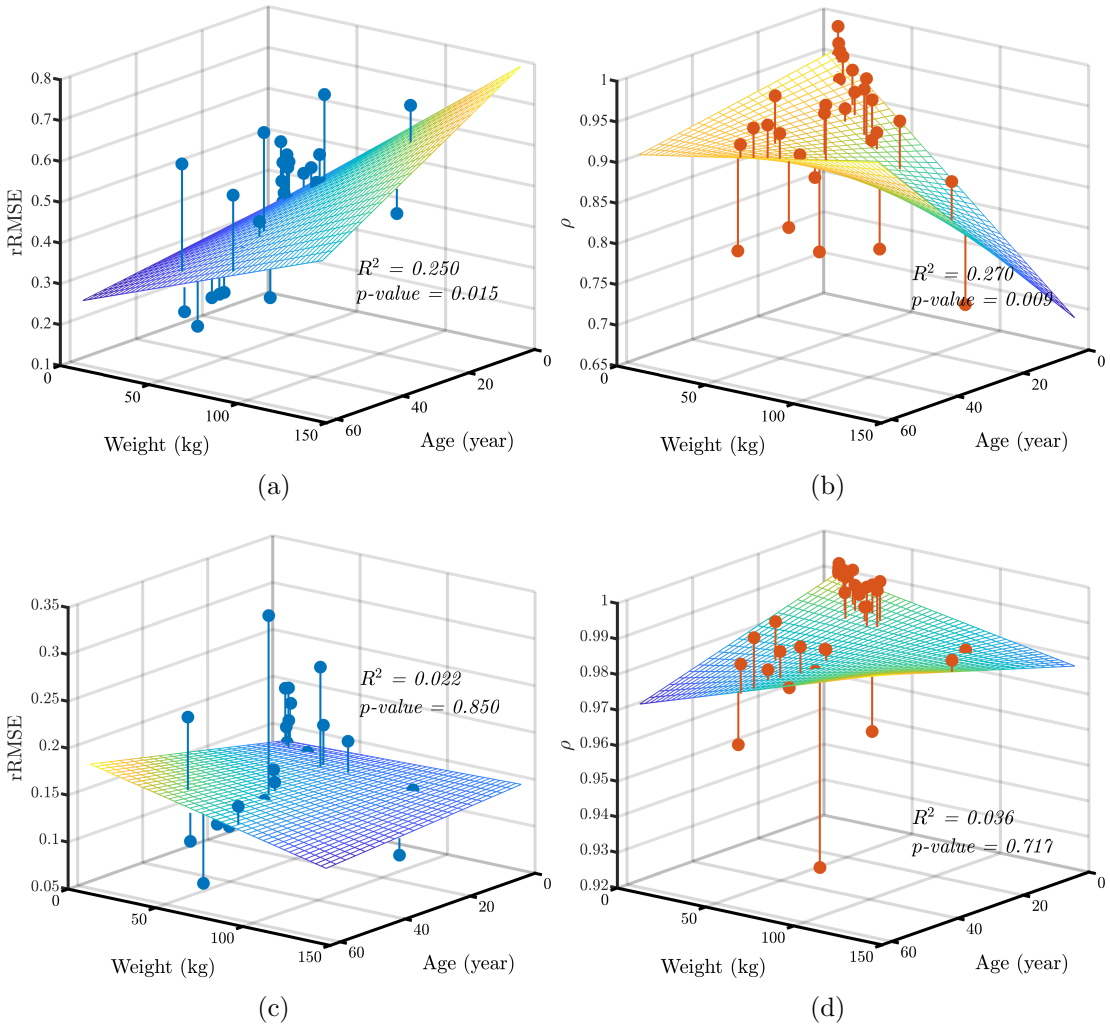


Figure 5.9: Scatter plots of (a) rRMSE and (b) ρ vs. subjects' weight and age using R2R scheme. Each sample corresponds to one of 42 sessions. The surface mesh on each plot shows the regressed linear model: rRMSE or $\rho \sim$ intercept + age + weight + age $\times$ weight. The R^2 and the p-value of F-test is shown on each plot.

As an example, we show a five-second segment of the reconstructed ECG waveform in the test set of the first subject in Fig. 5.7 using the R2R cycle segmentation scheme with $L_x = 18$ in SI mode and $L_x = 12$ in SD mode. We choose the first subject to be the example as the system performance evaluated on this subject approximates the average performance over the database. We see from the plot that the system retains most of the shape of the original ECG waveform except for the S

peaks in SI mode and almost perfectly reconstructs the shape of the ECG waveform and maintains the location of each PQRST peaks in SD mode.

In Fig. 5.9, we plot the rRMSE and ρ of each session with respect to subjects' age and weight respectively in two 3-D plots in SI and SD mode. We then fitted a linear model with an interaction term for each combination of training mode and evaluation metric according to the least squares criterion. An F-test is performed to test whether subjects' profile, i.e., age and weight, can significantly affect the performance of the algorithm in each metric and training mode combination. F-tests results of high p-values shown in Fig. 5.9(c) and 5.9(d) reveal the independent relationship between the performance of the algorithm and the subject's age and weight in SD mode, whereas the test results of low p-values shown in Fig. 5.9(a) and 5.9(b) indicate the dependent relationship in SI mode. Moreover, we notice that the performance tends to be lower as the subject's weight gets larger. This trend of performance degradation might be due to the bias of the training sample that the number of new-borns is much larger than the number of other groups of subjects in the database.

5.4.2 Experiment 2: MIMIC-III database

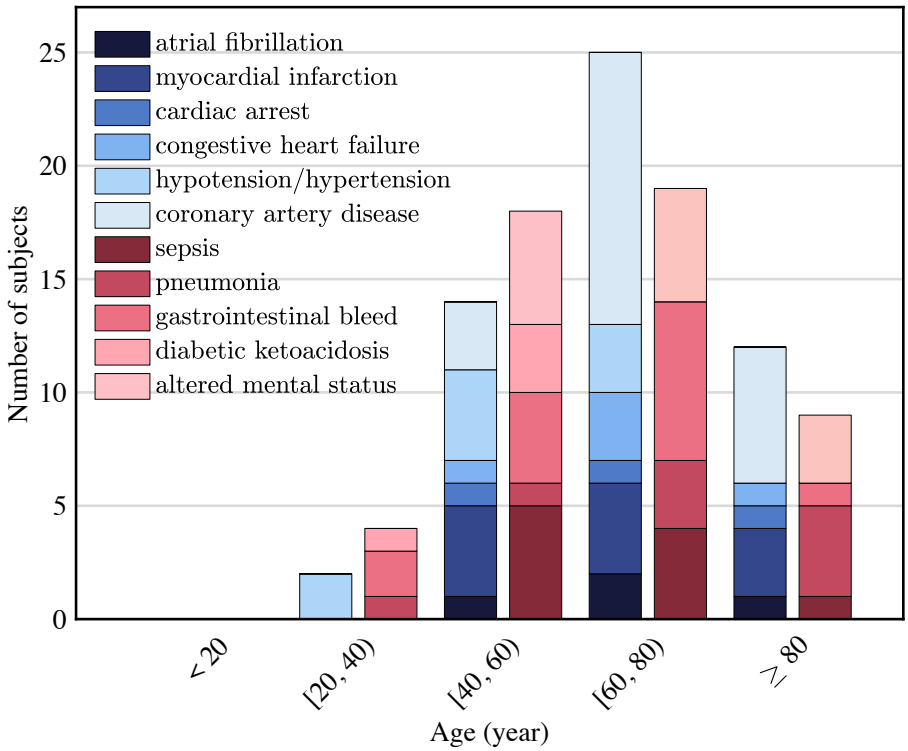


Figure 5.10: Distribution of subjects collected from the MIMIC-III database in five age groups and eleven disease type. Within each age group, the cardiac-related diseases are colored as different shades of blue on the left, and the noncardiac-related diseases are colored as different shades of red on the right.

		rRmse				ρ			
		P	QRS	T	all	P	QRS	T	all
MIMIC-III (*SD*)									
	lr	0.451 (0.183)	0.320 (0.115)	0.367 (0.175)	0.333 (0.119)	0.807 (0.150)	0.936 (0.045)	0.896 (0.103)	0.935 (0.055)
R2R	ridge	0.436 (0.175)	0.311 (0.113)	0.356 (0.169)	0.324 (0.114)	0.819 (0.141)	0.939 (0.044)	0.903 (0.097)	0.939 (0.053)
	lasso	0.439 (0.171)	0.310 (0.110)	0.358 (0.162)	0.324 (0.109)	0.817 (0.139)	0.940 (0.042)	0.903 (0.094)	0.940 (0.049)
MIMIC-III (*SI*)									
	lr	0.844 (0.240)	0.503 (0.166)	0.773 (0.211)	0.599 (0.148)	0.533 (0.252)	0.880 (0.082)	0.627 (0.318)	0.790 (0.118)
R2R	ridge	0.844 (0.240)	0.503 (0.166)	0.773 (0.211)	0.599 (0.148)	0.533 (0.253)	0.881 (0.082)	0.627 (0.318)	0.790 (0.118)
	lasso	0.844 (0.240)	0.503 (0.166)	0.773 (0.211)	0.598 (0.148)	0.533 (0.253)	0.881 (0.082)	0.627 (0.318)	0.790 (0.118)

Table 5.2: The system performance in test set of the MIMIC-III database in terms of sample mean and standard deviation (in paranthesis) of rRMSE and ρ. R2R segmentation using difference combinations of traning mode (SD/SI) and linear regression methods (lr/ridge/lasso).

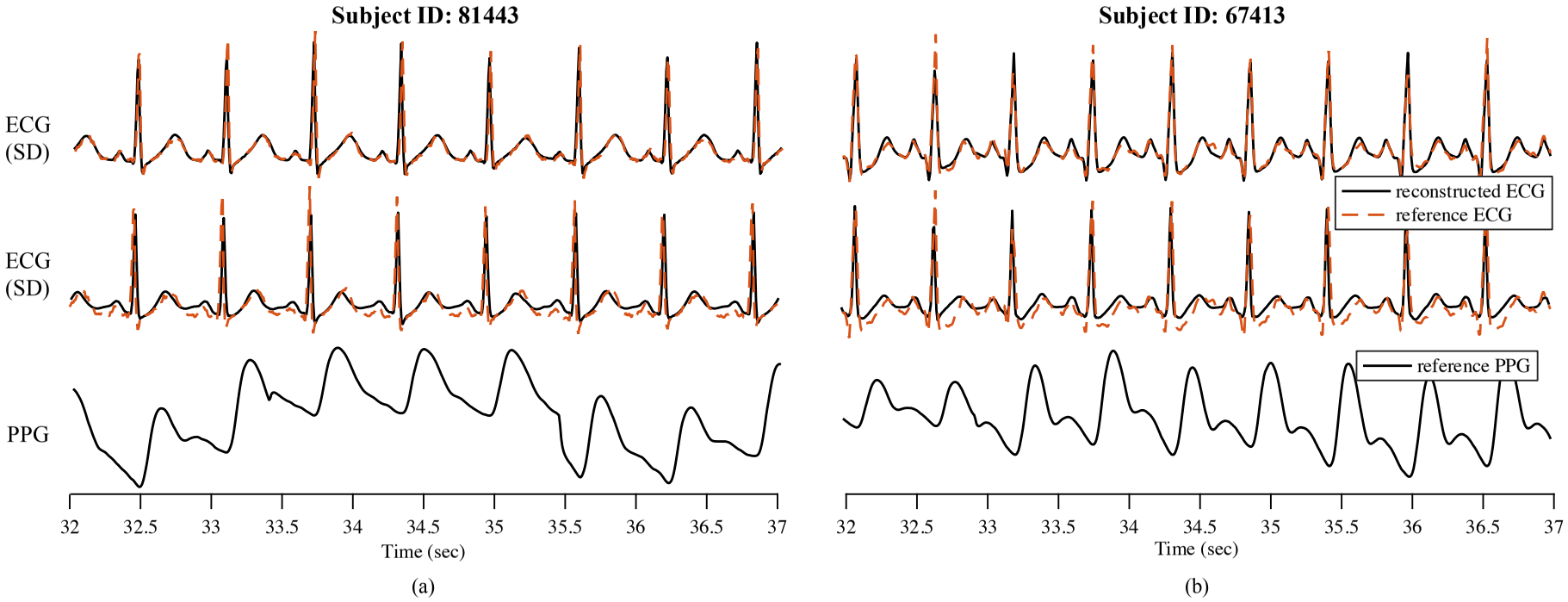
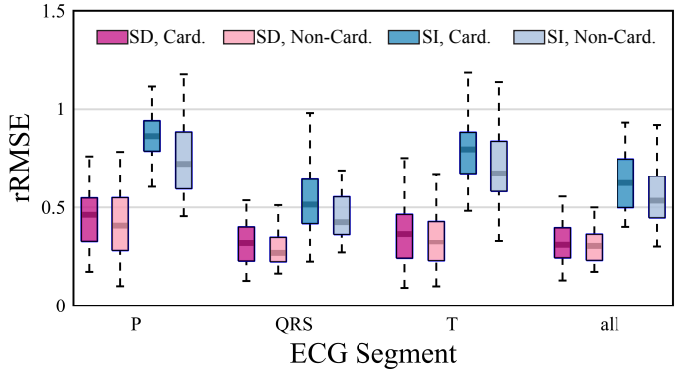


Figure 5.11: Two qualitative comparisons between the reconstructed ECG signals tested in SD (1st tow) and SI (2nd row) mode from the MIMIC-III database. (a) The subject is male, 54 years old, and with upper gastrointestinal bleeding. The Pearson's correlation coefficients are 0.969 in SD mode, and 0.923 in SI mode. (b) The subject is male, 52 years old, and with congestive heart failure. The Pearson's correlation coefficients are 0.959 in SD mode, and 0.881 in SI mode.

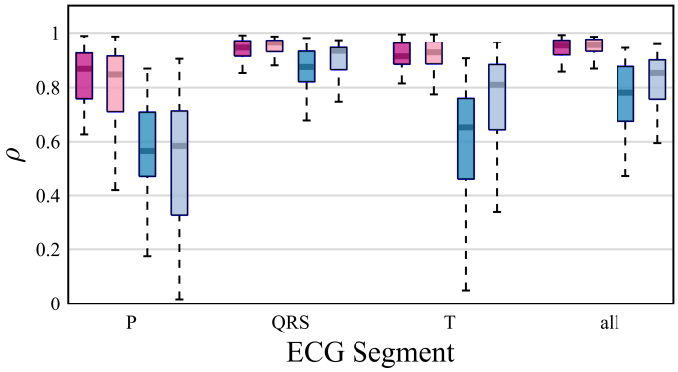
Medical Information Mart for Intensive Care III (MIMIC-III) [122] is a large, freely available database comprising vital sign measurements at the bedside documented in MIMIC-III waveform database and part of the patients' profile in the MIMIC-III clinical database. The database is publicly available and encompasses a large population of ICU patients. In this experiment, the MIMIC-III database is used to evaluate the system's performance when the subjects are with various cardiac or non-cardiac malfunctions. In order to collect the required signals from the raw database to facilitate the evaluation, we developed a data collector procedure that be detailed as follows.

First, we selected waveforms that contain both lead II ECG and PPG signals from folder 35 in MIMIC-III waveform database. Then we linked the selected waveforms with the MIMIC-III clinical database by subject ID to match with the corresponding patient profile. Among the patients, we selected those with specific cardiac/non-cardiac diseases and removed low signal quality PPG/ECG pairs. [selecting patients with specific cardiac/non-cardiac diseases and removing low signal quality PPG/ECG pairs]

The resulting collected database consists of 53 patients with six typical cardiac diseases and 50 patients with five types of non-cardiac diseases. Each patient has three sessions of 5-min ECG and PPG recordings collected within several hours. Cardiac diseases in the resulting database include atrial fibrillation, myocardial infraction, cardiac arrest, congestive heart failure, hypotension, hypertension and coronary artery disease, while non-cardiac diseases are composed of sepsis, pneumonia, gastrointestinal bleed, diabetic ketoacidosis and altered mental status. Fig.5.10



(a)



(b)

Figure 5.12: Comparison of the performance of the proposed method in test set of the MIMIC-III database in different combinations of the disease types and Sub.D test modes. Statistics of the (a) rRMSE and (b) ρ are summarized using the box plots.

shows the distribution of the collected dataset with respect to age and disease.

In this part of experiment, we evaluate our proposed system in the following two training modes (both under R2R segmentation scheme):

- *Subject Independent (SI)* mode: we trained one linear transform f^* using training data from patients with cardiac diseases and another linear transform f^* from non-cardiac disease patients, i.e., the trained model is independent with each subject in terms of disease type.

- *Subject Dependent (SD)* mode: a linear transform f^* is trained and tested in each session. In this way, we obtain subject dependent model for each individual.

In addition to quantitative analysis of the reconstruction performance by Pearson correlation and rRMSE, we also execute a disease classification experiment on the reconstructed ECG signals to show the potential of our proposed method in applications within biomedical health informatics.

First, from the collected MIMIC-III database, we select 28 patients with five types of cardiac diseases, including congestive heart failure, ST-segment elevated myocardial infraction, non-ST segment elevated myocardial infraction, hypotension, and coronary artery disease. For each patient, we perform the SD mode ECG reconstruction experiment to obtain the reconstructed ECG signals. To simulate the diagnosis process of cardiologists, we connect the cycle-wise ECG signals into pieces of 30-cycle length for training and classification. The training data is composed of 70 % from the original ECG signals, and the testing data constitutes of the rest 30

Disease	Number of patients	Number of training data	Number of test data (original ECG)	Number of test data (reconstructed ECG)
CHF	7	163 (23.6%)	65 (25.8%)	67 (23.9%)
STMI	7	171 (24.7%)	59 (23.4%)	68 (24.3%)
NSTMI	5	114 (16.5%)	40 (15.9%)	46 (16.4%)
HYPO	5	158 (22.8%)	57 (22.6%)	64 (22.9%)
CAD	4	86 (12.4%)	31 (12.3%)	35 (12.5%)
Total	28	692 (100%)	252 (100%)	280 (100%)

CHF: congestive heart failure
STMI: ST-segment elevated Myocardial infraction
NSTMI: non-ST segment elevated Myocardial infraction
HYPO: hypotension
CAD: coronary artery disease

Table 5.3: Distribution of training and testing data for disease classification in the MIMIC-III dataset

% from original ECG signals and all of the reconstructed ECG signals. The detailed distribution of training and testing data with respect to disease types are shown in table 5.4.2.

We applied PCA for dimension reduction and SVM classifier with polynomial kernel from SVM library [135]. The confusion matrices for classification are illustrated in figure 5.13 with the reduced dimension equals to 100. Comparing figure 5.17(a) and 5.17(b), we conclude that our reconstructed ECG has a comparable classification performance as the original ECG signals. We also include the confusion matrix for original PPG classification in figure 5.13(c) for reference. The superior performance of classification from the reconstructed ECG signals compared to that of the original PPG signal indicates the fidelity of the reconstructed ECG recordings in the presence of cardiac pathologies.

5.4.3 Experiment 3: Self-collected data

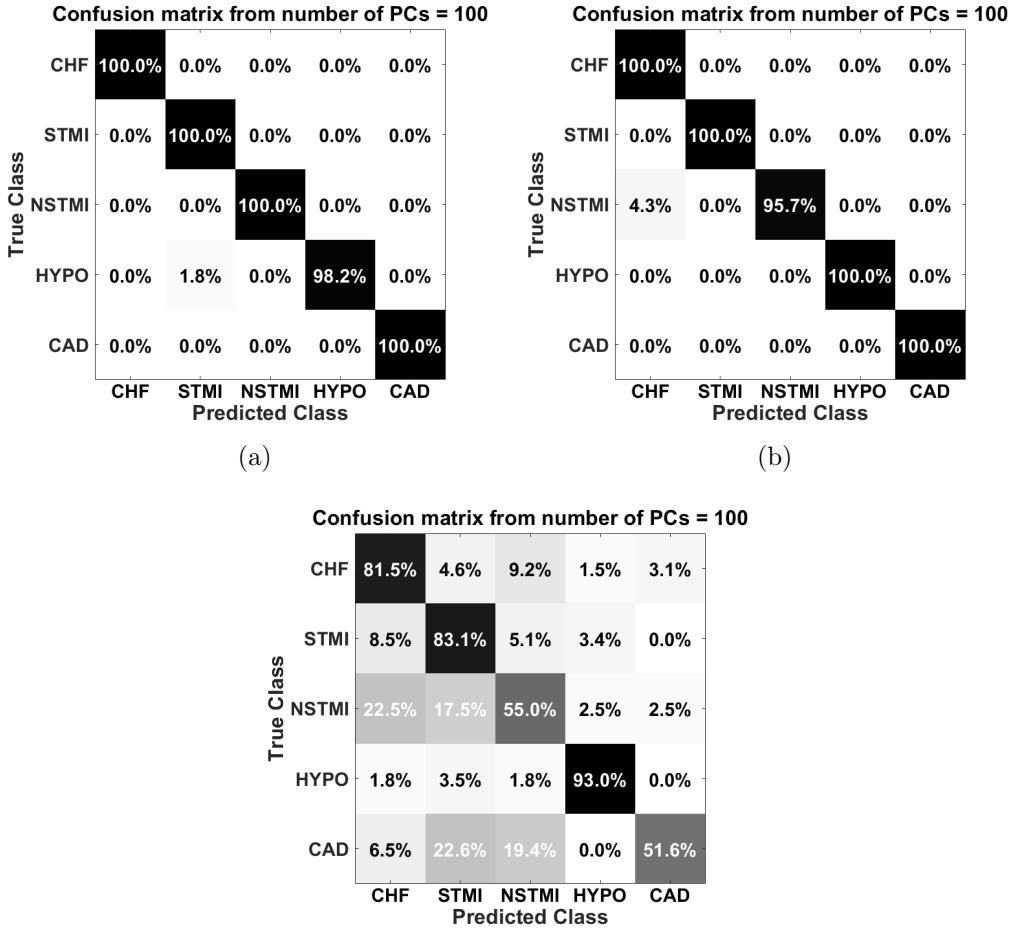


Figure 5.13: Confusion matrices of SVM with polynomial kernel on three types of data: original ECG, inferred ECG and original PPG

		rRMSE				ρ			
		P	QRS	T	all	P	QRS	T	all
Self-collected (*Sess. D*)									
	lr	0.591 (0.102)	0.230 (0.051)	0.491 (0.103)	0.372 (0.068)	0.620 (0.155)	0.970 (0.013)	0.863 (0.057)	0.926 (0.028)
R2R	ridge	0.589 (0.101)	0.229 (0.051)	0.490 (0.105)	0.370 (0.068)	0.623 (0.147)	0.970 (0.013)	0.864 (0.057)	0.926 (0.027)
	lasso	0.593 (0.102)	0.235 (0.051)	0.494 (0.107)	0.376 (0.071)	0.618 (0.149)	0.968 (0.013)	0.861 (0.058)	0.924 (0.028)
Self-collected (*Sess. I*)									
	lr	0.660 (0.070)	0.278 (0.021)	0.569 (0.052)	0.427 (0.047)	0.575 (0.125)	0.966 (0.009)	0.835 (0.039)	0.903 (0.024)
R2R	ridge	0.660 (0.071)	0.278 (0.021)	0.567 (0.053)	0.426 (0.049)	0.575 (0.125)	0.966 (0.009)	0.836 (0.039)	0.904 (0.024)
	lasso	0.664 (0.073)	0.280 (0.022)	0.568 (0.056)	0.428 (0.051)	0.569 (0.125)	0.965 (0.010)	0.834 (0.041)	0.903 (0.026)
Self-collected (*Sub. I*)									
	lr	0.724 (0.058)	0.302 (0.024)	0.591 (0.111)	0.447 (0.046)	0.503 (0.146)	0.956 (0.013)	0.830 (0.044)	0.895 (0.025)
R2R	ridge	0.724 (0.059)	0.302 (0.024)	0.591 (0.111)	0.447 (0.046)	0.503 (0.147)	0.956 (0.013)	0.830 (0.044)	0.895 (0.025)
	lasso	0.725 (0.059)	0.303 (0.025)	0.592 (0.110)	0.448 (0.047)	0.500 (0.146)	0.956 (0.014)	0.829 (0.045)	0.895 (0.025)

Table 5.4: The system performance in test set of the self-collected database in terms of sample mean and standard deviation (in paranthesis) of rRMSE and ρ. R2R segmentation

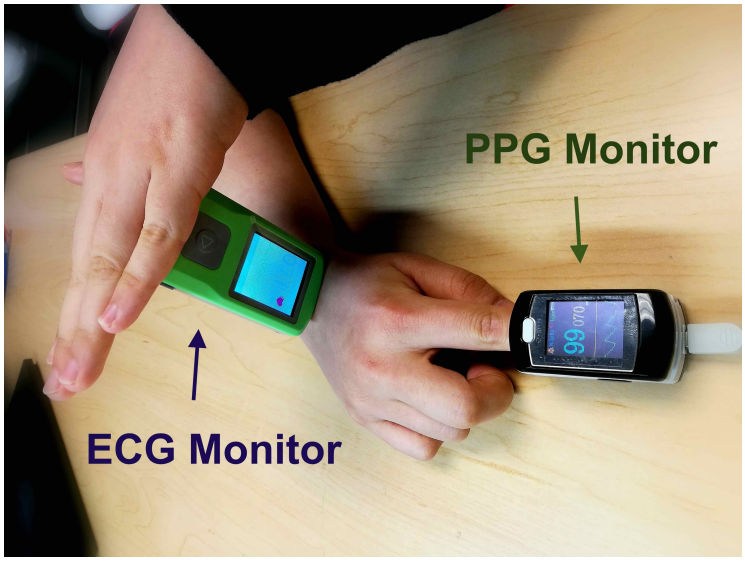


Figure 5.14: This figure shows how the signals were collected in the self-collected database. The subject was asked to wear a PPG sensor and an ECG sensor to capture her fingertip PPG and Lead I ECG signal simultaneously in a sitting position.

We next test our system with the self-collected data using commercially available sensors to test the temporal consistency of the system. Two subjects participated in this two-weeks long experiment. One subject is male, 31 years old. The other is female, 23 years old. Both of them are Asian. According to the most-recent medical examinations received by both subjects, none of them has been diagnosed with any known CVDs or mental illness. As shown in Table 5.4.3, we recorded six 5-min trials for the first subject and seven trials for the second subject in different times of two consecutive weeks. In each trial, the subject was asked to wear 1. EMAY FDA-clear handheld single-lead ECG monitor (Model: EMG-10) and 2. CONTEC pulse oximeter (Model: CMS50E) to record their lead I bipolar ECG signals[2] and finger-tip PPG signals simultaneously. As shown in Fig. 5.14, we asked the subject to wear the PPG sensor on his/her index finger of the right hand, and

[2]We measure the lead I ECG signal in this experiment considering the easiest accessibility among all leads using the handheld ECG sensor.

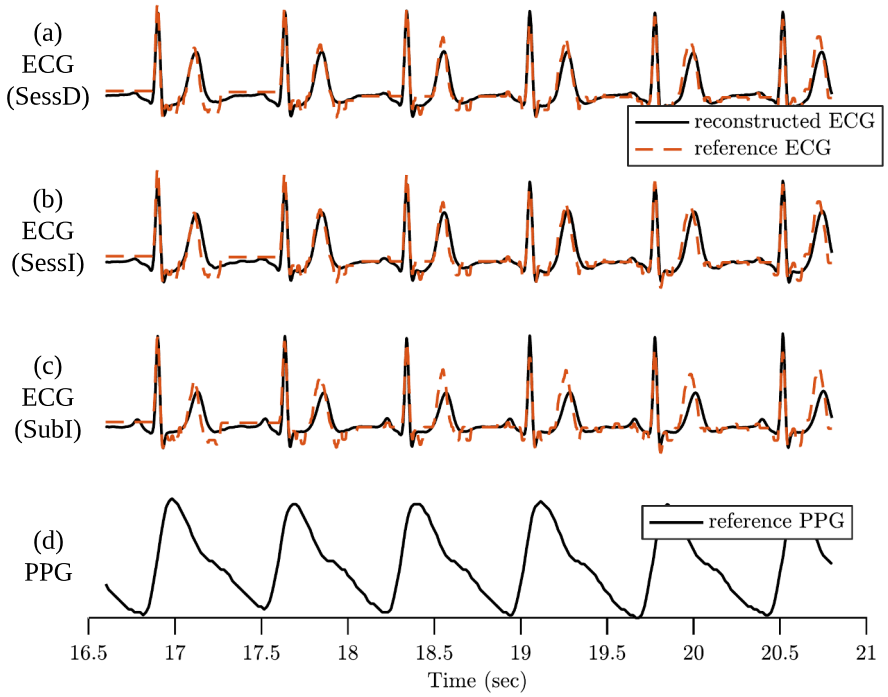
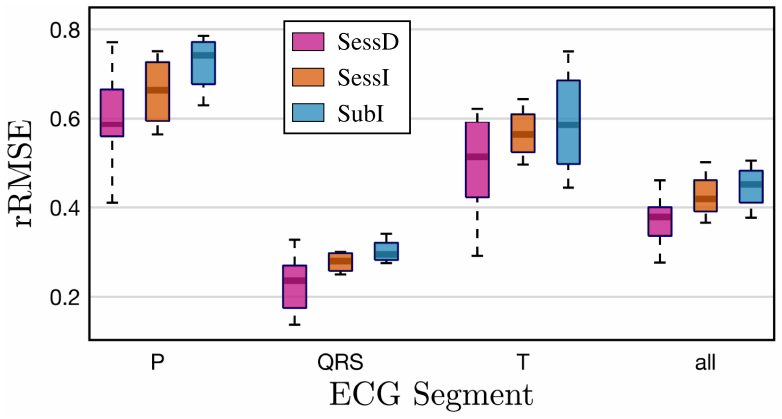
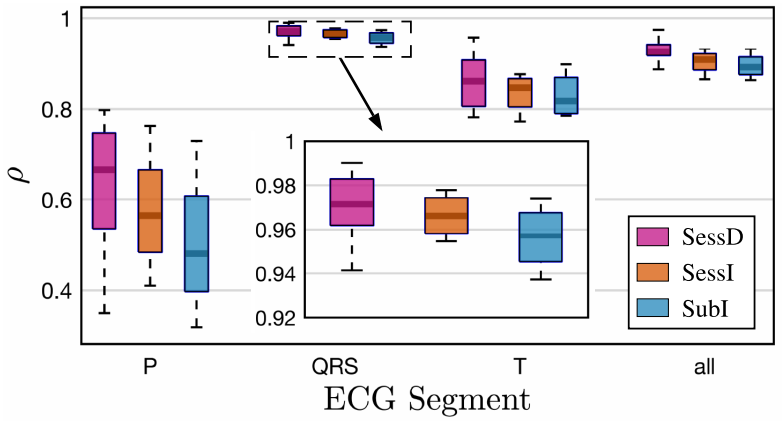


Figure 5.15: A qualitative comparison among the reconstructed ECG signals tested in (a) SessD, (b) SessI, and (c) SubI modes respectively, from the 6th session of the first subject in self-collected database. In (a-c), the black line indicates the reconstructed ECG and the orange dashed line refers to the reference ECG. The Pearson's correlation coefficients for these three cases are 0.937 in SessD, 0.917 in SessI, and 0.869 in SubI. (d): the corresponding PPG waveform.



(a)



(b)

Figure 5.16: Comparison of the performance of the proposed method in test set of the self-collected database in SessD, SessI, and SubI mode. Statistics of the (a) rRMSE and (b) ρ are summarized using the box plots.

attach the electrodes of the ECG sensor to the palm of left hand and the back of the right hand. The subjects were asked to sit in front of a table and put their arms on the table as motionless and peacefully as possible to reduce the motion-induced artifacts during the recording time. The sampling rates of the ECG and PPG sensors are 150 and 60 Hz respectively. We up-sampled both signals to 300 Hz via the bilinear interpolation for consistency consideration, and aligned the pair of signals by the USB protocol.

We evaluate the system performance in the following three training modes:

- *Session Dependent (SessD)* mode: Same to SD mode we investigated in Section 5.4.1. f^* is trained and tested separately in each session.

- *Session Independent (SessI)* mode: The sessions of each subject are first listed chronologically. f^* is trained on the first 80% of the sessions and is tested on the rest of the sessions in order to maximize the temporal difference of the training and test set.

- *Subject Independent (SubI)* mode: We combined the subject dependent training sets used in SessI mode and trained a subject independent model to test on the same test set in SessI mode.

In this experiment, we use the R2R segmentation scheme and set $L_x = 12$ in SessD and SessI mode and $L_x = 18$ in SubI mode. The cycle segmentation process is guided by the peak detection algorithms introduced in [134] and [136]. The two algorithms are deployed to detect the R peak of ECG and the onset point of PPG signal respectively. Fig. 5.15 shows one example of the reconstructed waveforms

Session	Time	Session	Time
Subject 1		**Subject 2**	
1	2019-01-07, 10:04	1	2019-01-07, 10:12
2	2019-01-07, 11:11	2	2019-01-07, 11:18
3	2019-01-07, 14:08	3	2019-01-07, 13:29
4	2019-01-09, 19:42	4	2019-01-09, 19:54
5	2019-01-10, 19:14	5	2019-01-09, 22:12
6	2019-01-18, 09:49	6	2019-01-09, 23:21
		7	2019-01-10, 10:00

Table 5.5: The date and time of each session in self-collected database.

from the 6th session of the first subject. Note that this session is recorded more than one-week after the other sessions. From the qualitative result in 2nd and 3rd rows of Fig. 5.15, we notice that the reconstructed signals match well with the reference ECG in all waves in the condition of long temporal separation from the training set.

Similar to the previous two experiments, we summarize the average performance in different combinations of training modes and regression methods and evaluate each combination in terms of RRMSE and ρ in P, QRS, T waves respectively. In Fig. 5.16, we use the box-plots to visualize the more detailed statistics of the evaluation metrics. Notice that in general, the system perform best in SessD mode, followed by SessI and SubI. Again, this difference may suggest possible subject-wise difference of the model parameter $b(t)$, $\mathbf{H}$, or α. Consistent observations in this dataset also include better performance in T wave than P wave, and our conjecture remains with the one claimed in Section 5.4.1.

Segmentation	rRMSE (SD)	ρ (SD)	rRMSE (SI)	ρ (SI)
O2O	0.553	0.823	0.689	0.717
R2R	0.324	0.940	0.599	0.790

Table 5.6: Performance comparison using O2O and R2R cycle segmentation schemes on the MIMIC-III test dataset.

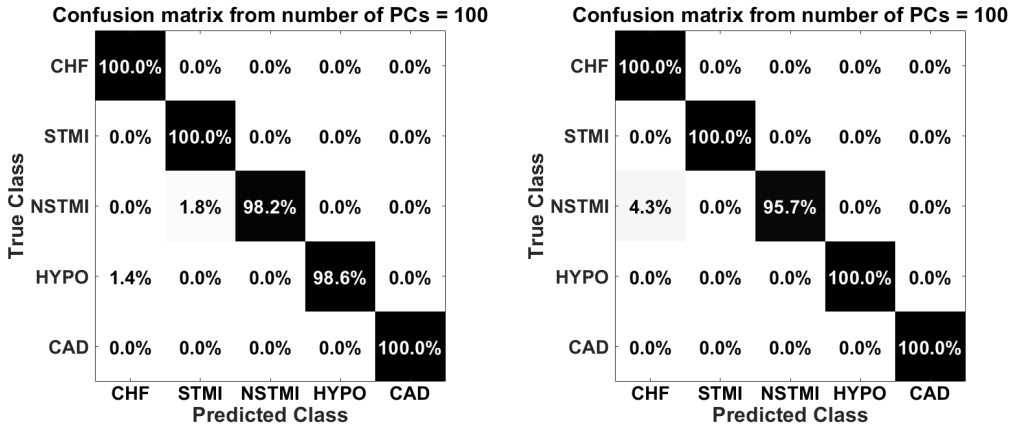
5.5 Discussion and Extensions

5.5.1 Cycle Segmentation via PPG

We have evaluated the system in Section 5.4 assuming the availability of the ground truth cardiac cycle information obtained from the ECG signal. We now examine a more practical setting when the cycles are estimated solely from the PPG signal, thereby accounting for the real-world constraint that the reference cycle information is unavailable.

The MIMIC-III database introduced in Section 5.4.2 was adopted in this experiment. We segmented the signal according to the onset points of the PPG signal, considering the onset point represents one of the most distinct features within the PPG cycle. We name this segmentation scheme *O2O*.

To single out the contribution to the reconstruction error due to the discrepancy in the waveform shape rather than the misalignment of the ECG peaks, we evaluate O2O after each reconstructed cycle was post-processed to align with the original ECG signal. This was done by shifting each reconstructed ECG cycle in time so that the original and reconstructed ECG signals were matched according to their R peaks. We list the performance metrics in the SD and SI modes and



(a) (b)

Figure 5.17: Confusion matrices for classification results using kernel SVM on three types of data: (a) inferred ECG (O2O) (b) inferred ECG (R2R).

compare the results with the R2R segmentation in Table 5.6. Note that $\rho = 0.510$ when using O2O segmentation without the peak alignment in the SD mode, and ρ increases to 0.823 once the peak is aligned. The performance statistics reveal that the shape of the waveform is inferred well, and increased error in reconstruction by O2O compared with R2R is mainly due to the misalignment of the signal that has a sample mean and standard deviation of 0.38% and 3.98% in relative cycle length, respectively. This observation is consistent across the SI and SD training modes.

The disease classification experiment was conducted using the O2O segmentation without the peak alignment. We observed a comparable classification accuracy of the reconstructed ECG signal compared with the result when the model was trained with the R2R segmentation. This observation indicates that the ECG reconstruction deviation does not affect the diagnostic power of the reconstructed ECG signal.

The confusion matrices for the disease classification using the O2O scheme

without the shifting operation are illustrated in Fig. 5.17. Note that the classification accuracy is comparable with the result we had before where the model is trained using R2R. This observation indicated that the ECG reconstruction deviation does not affect the diagnostic power of the signal.

5.5.2 Extensions of the Proposed Methodology Using Joint Dictionary Learning

In this chapter, a linear transform in the DCT domain has been designed to reconstruct the ECG signal from the PPG signal. The use of this universal orthogonal DCT basis offers an efficient and compact representation for both signals and enables a robust system with less concerns on overfitting the model even when the data size is relatively small.

As the growing size of our available dataset, we would like to explore the possibilities of a more complicated model as an extension of the current system. For the sake of this problem, we have asked therefore the following questions before we started to design the extended learning mechanism:

- *Universal v.s. Signal-specific Basis*: instead of using an universal basis which is optimized for a broader range of signals, can we learn a PPG- and a ECG-specific set of dictionary atoms which is optimized for representing the signal in the sensing of signal transformation?

- *Complete v.s. Overcomplete Basis*: in order to effectively capture the variations of the signal that is possible in a large population and in different health

conditions, can we instead learn an overcomplete basis for sparse and precise signal representation without restricting ourselves in the design constraint of the completeness of the functions?

- *Individual Learning v.s. Joint Learning Strategy*: if we could address the first two questions we asked, can we design a joint learning objective to globally optimize the representation of PPG, the representation of ECG, and the transform in between the two sparse representation domain instead of marginally tuning each part individually?

The three questions we asked naturally lead us to a joint dictionary learning method, whose block diagram is shown in Fig. 5.18. In the training phase, a dictionary pair for ECG and PPG signal representation is learned, meanwhile, a linear transform is acquired which maps from the sparse space of PPG to that of ECG. Different from [137, 138], this learning system is designed to optimize the capability of the obtained dictionaries for both signal representation as well as the signal transform via a joint problem formulation. We briefly present below the formulation of the objective of the learning task.

Let $\mathbf{P} = [\mathbf{X}_p, \mathbf{T}_p] \in \mathbb{R}^{d\times(n+m)}$ and $\mathbf{E} = [\mathbf{X}_e, \mathbf{T}_e] \in \mathbb{R}^{d\times(n+m)}$ be PPG and ECG data sets respectively. Each column of $\mathbf{P}$ and $\mathbf{E}$ is denoted as $\mathbf{p}_i \in \mathbb{R}^{d\times 1}$ and $\mathbf{e}_i \in \mathbb{R}^{d\times 1}$, representing one PPG/ECG cycle during the same cardiac cycle. The objective of ECG waveform reconstruction from PPG is to utilize the training PPG/ECG cycles from $\mathbf{X}_p$ and $\mathbf{X}_e$ to learn some patterns (dictionaries, mappings, etc.) that can be applied to the testing PPG dataset $\mathbf{T}_p \in \mathbb{R}^{d\times m}$ for accurate

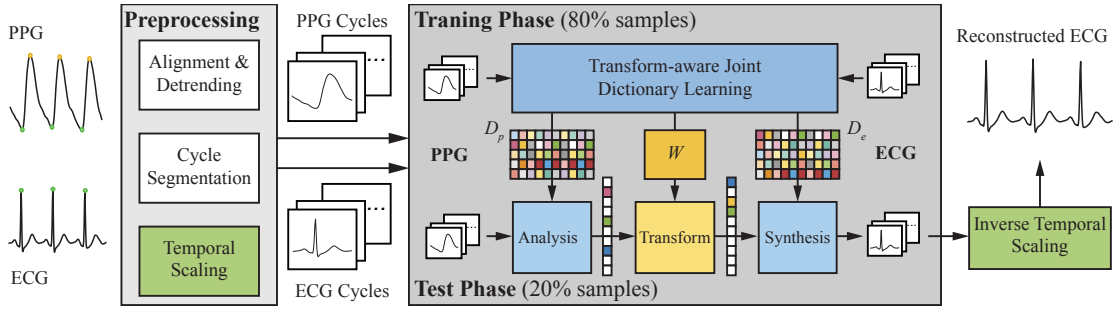


Figure 5.18: Block diagram of the proposed joint dictionary learning framework. The ECG and PPG signals are first preprocessed to obtain temporally aligned and normalized pairs of cycles. 80% pairs of ECG and PPG signal cycles are used for training paired dictionaries $\mathbf{D}_p$, $\mathbf{D}_e$ and a linear transform $\mathbf{W}$ which will be applied in the test phase to reconstruct the ECG signals.

approximation and inference of testing ECG dataset $\mathbf{T}_e \in \mathbb{R}^{d\times m}$.

We formulate our learning objective as:

$$\begin{aligned} \min_{\substack{\mathbf{D}_e,\mathbf{A}_e,\mathbf{D}_p,\\ \mathbf{A}_p,\mathbf{W}}} & \|\mathbf{X}_e - \mathbf{D}_e\mathbf{A}_e\|_F^2 + \alpha\|\mathbf{X}_p - \mathbf{D}_p\mathbf{A}_p\|_F^2 + \quad \beta\|\mathbf{A}_e - \mathbf{W}\mathbf{A}_p\|_F^2 \\ \text{s.t. } & \|\mathbf{a}_{p,j}\|_0 \le t_p, j = 1, ..., n. \\ & \|\mathbf{a}_{e,j}\|_0 \le t_e, j = 1, ..., n. \end{aligned} \tag{5.16}$$

where $\mathbf{D}_p \in \mathbb{R}^{d\times k_p}, \mathbf{D}_e \in \mathbb{R}^{d\times k_e}$ are dictionaries learned for $\mathbf{X}_p, \mathbf{X}_e$ respectively. $\mathbf{A}_p \in \mathbb{R}^{k_p\times n}, \mathbf{A}_e \in \mathbb{R}^{k_e\times n}$ are the corresponding sparse coding matrices related with the data matrices $\mathbf{X}_p, \mathbf{X}_e$ when $\mathbf{D}_p, \mathbf{D}_e$ are the current dictionaries. Each column of $\mathbf{A}_p, \mathbf{A}_e$ is denoted as $\mathbf{a}_{p,j}$, $\mathbf{a}_{e,j}$ with the sparsity upper bounded by t_p, t_e, respectively. $\|\mathbf{X}_e - \mathbf{D}_e\mathbf{A}_e\|_F^2$ and $\|\mathbf{X}_p - \mathbf{D}_p\mathbf{A}_p\|_F^2$ are the data fidelity terms for ECG and PPG cycle sets, respectively.

The problem in (5.16) can be solved using a variant of the well-known K-SVD algorithm, where the dictionary atoms and the linear transforms are updated

iteratively. We skip the details of the optimization process for the conciseness of this thesis. In Fig. 5.19, we show one preliminary result reconstructing the ECG signal. The result looks promising in this case compared with the DCT-based system in the SI mode.s

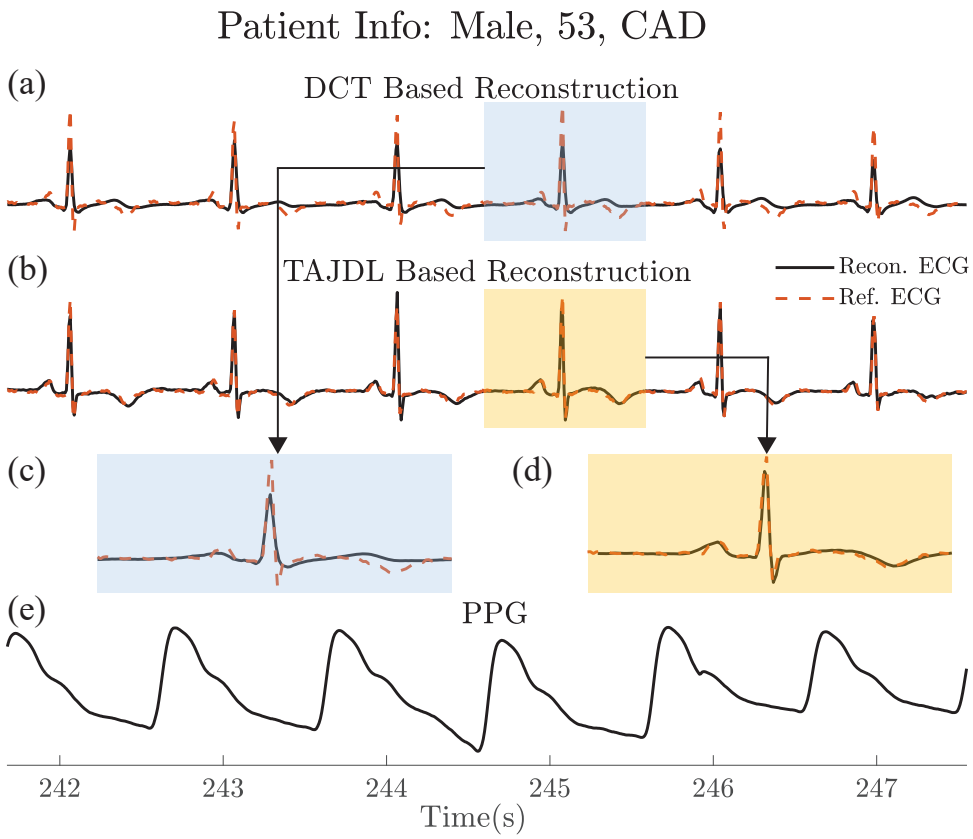


Figure 5.19: Qualitative comparison between the reconstructed ECG signals from (a) DCT based method and (b) Joint dictionary learning method. (c)(d) are the zoomed-in version of the 4^{th} cycle from (a)(b). (e) shows the PPG signal along the time.

5.6 Conclusion

This chapter presents a learning-based approach to reconstruct ECG signal from PPG. The algorithm is successfully evaluated in both subject-dependent and

subject-independent fashions on two widely-adopted databases as well as a self-collected database. We cross-validate the system hyper-parameters, test the CVD diagnosis performance using the reconstructed ECG signal, and justify the algorithm's accuracy and consistency in a fine ECG wave level. As a pilot study, this work demonstrates that with a signal processing and learning system that is justified in each design step, we are able to precisely reconstruct ECG signal by exploiting the relation of the two measurements.

Chapter 6: Conclusion

In this dissertation, we study the modeling and analytics of PPG signal to facilitate its applications in both robust and remote cardiovascular sensing.

In the first part of this dissertation (Ch. 2 and Ch. 3), we studied the remote photoplethysmography (rPPG) sensing and presented a robust and efficient rPPG system to extract pulse rate (PR) and pulse rate variability (PRV) from face videos. Compared with prior art, our proposed system achieves accurate PR and PRV estimates even when the video contains significant subject motion and illumination change. We have implemented a robust rPPG system using Python to achieve the state-of-the-art performance with just a regular web-cam running in realtime. We optimistically look forward to the coming era of wide deployment of the rPPG technology into everyone's life for better preventable health care.

In the second part of the dissertation (Ch. 4), we presented a novel frequency tracking algorithm called Adaptive Multi-Trace Carving (AMTC) as an unified tool to address the micro signal extraction problems that can be applied to multiple application scenarios including physiological measurement and media forensics. AMTC enables an accurate detection and estimation of one or more subtle frequency components in a very low signal-to-noise ratio condition.

In the third part of the dissertation (Ch. 5), the relation between electrocardiogram (ECG) and PPG have been studied and the waveform of ECG was inferred via the PPG signals. In order to address this cardiovascular inverse problem, a linear transform was proposed to map the discrete cosine transform coefficients of each PPG cycle to those of the corresponding ECG cycle. As the first work to address this biomedical inverse problem, this line of research enables a full utilization of the easy accessibility of PPG and the clinical authority of ECG for better preventive healthcare.

As we finalize this dissertation, the COVID-19 pandemic is hitting many regions in the world and changing many practices in unprecedented ways. We are seeing rapidly rising needs in remote healthcare. We hope the research in this thesis can help advance the technology development toward continuous health monitoring, remote triage, and remote rehabilitation to address the needs of the current and future management of public health.

Bibliography

[1] Mohamed Elgendi. On the analysis of fingertip photoplethysmogram signals. *Current Cardiology Reviews*, 8(1):14–25, February 2012.

[2] John Allen. Photoplethysmography and its application in clinical physiological measurement. *Physiological Measurement*, 28(3):R1, February 2007.

[3] Tom Lister, Philip A Wright, and Paul H Chappell. Optical properties of human skin. *Journal of biomedical optics*, 17(9):090901, September 2012.

[4] Wim Verkruysse, Lars O Svaasand, and J Stuart Nelson. Remote plethysmographic imaging using ambient light. *Optices Express*, 16(26):21434–45, December 2008.

[5] Wenjin Wang, Sander Stuijk, and Gerard De Haan. A novel algorithm for remote photoplethysmography: Spatial subspace rotation. *IEEE transactions on biomedical engineering*, 63(9):1974–1984, December 2015.

[6] Daniel McDuff, Sarah Gontarek, and Rosalind Picard. Remote measurement of cognitive stress via heart rate variability. In *2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 2957–2960, August 2014.

[7] Mingliang Chen, Qiang Zhu, Harris Zhang, and Min Wu. Respiratory rate estimation from face videos. In *IEEE-EMBS International Conference on Biomedical and Health Informatics (BHI)*, Chicago,IL, May 2019.

[8] Lingqin Kong, Yuejin Zhao, Liquan Dong, Yiyun Jian, Xiaoli Jin, Bing Li, Yun Feng, Ming Liu, Xiaohua Liu, and Hong Wu. Non-contact detection of oxygen saturation based on visible light imaging device using ambient light. *Optics express*, 21(15):17464–17471, July 2013.

[9] In Cheol Jeong and Joseph Finkelstein. Introducing contactless blood pressure assessment using a high speed video camera. *Journal of medical systems*, 40(4):77, April 2016.

[10] Qiang Zhu, Chau-Wai Wong, Chang-Hong Fu, and Min Wu. Fitness heart rate measurement using face videos. In *2017 IEEE International Conference on Image Processing (ICIP)*, pages 2000–2004, Beijing, China, September 2017.

[11] Jean-Philippe Couderc, Survi Kyal, Lalit K Mestha, Beilei Xu, Derick R Peterson, Xiaojuan Xia, and Burr Hall. Detection of atrial fibrillation using contactless facial video monitoring. *Heart Rhythm*, 12(1):195–201, 2015.

[12] Wenjin Wang, Sander Stuijk, and Gerard de Haan. Living-skin classification via remote-ppg. *IEEE Transactions on Biomedical Engineering*, 64(12):2781–2792, 2017.

[13] Paul Kligfield, Leonard S Gettes, James J Bailey, Rory Childers, Barbara J Deal, E William Hancock, Gerard Van Herpen, Jan A Kors, Peter Macfarlane, David M Mirvis, et al. Recommendations for the standardization and interpretation of the electrocardiogram: part I: the electrocardiogram and its technology. *Journal of the American College of Cardiology*, 49(10):1109–1127, January 2007.

[14] Wikipedia. Electrocardiography, 2019.

[15] Yu Sun and Nitish Thakor. Photoplethysmography revisited: from contact to noncontact, from point to imaging. *IEEE Transactions on Biomedical Engineering*, 63(3):463–477, September 2015.

[16] Wenjin Wang, Albertus C den Brinker, Sander Stuijk, and Gerard de Haan. Algorithmic principles of remote ppg. *IEEE Transactions on Biomedical Engineering*, 64(7):1479–1491, 2016.

[17] Steven A Shafer. Using color to separate reflection components. *Color Research & Application*, 10(4):210–218, December 1985.

[18] R Rox Anderson and John A Parrish. The optics of human skin. *Journal of investigative dermatology*, 77(1):13–19, July 1981.

[19] Hirotsugu Takiwaki et al. Measurement of skin color: practical application and theoretical considerations. *Journal of Medical Investigation*, 44:121–126, February 1998.

[20] Wenjin Wang, Albertus C den Brinker, Sander Stuijk, and Gerard de Haan. Robust heart rate from fitness videos. *Physiological measurement*, 38(6):1023, May 2017.

[21] G de Haan and A van Leest. Improved motion robustness of remote-PPG by using the blood volume pulse signature. *Physiological Measurement*, 35(9):1913, August 2014.

[22] Mika P Tarvainen, Perttu O Ranta-Aho, and Pasi A Karjalainen. An advanced detrending method with application to hrv analysis. *IEEE Transactions on Biomedical Engineering*, 49(2):172–175, August 2002.

[23] Chau-Wai Wong. *Micro Signal Extraction and Analytics*. PhD thesis, University of Maryland, College Park, 2017.

[24] M.-Z. Poh, D. J. McDuff, and R. W. Picard. Advancements in noncontact, multiparameter physiological measurements using a webcam. *IEEE Transactions on Biomedical Engineering*, 58(1):7–11, January 2011.

[25] Yu Sun, Sijung Hu, Vicente Azorin-Peris, Stephen Greenwald, Jonathon Chambers, and Yisheng Zhu. Motion-compensated noncontact imaging photoplethysmography to monitor cardiorespiratory status during exercise. *Journal of Biomedical Optics*, 16(7):077010:1–9, July 2011.

[26] Hao-Yu Wu, Michael Rubinstein, Eugene Shih, John Guttag, Frédo Durand, and William T. Freeman. Eulerian video magnification for revealing subtle changes in the world. *ACM Transactions on Graphics (Proc. SIGGRAPH 2012)*, 31(4), 2012.

[27] Christopher G Scully, Jinseok Lee, Joseph Meyer, Alexander M Gorbach, Domhnull Granquist-Fraser, Yitzhak Mendelson, and Ki H Chon. Physiological parameter monitoring from optical recordings with a mobile phone. *IEEE Transactions on Biomedical Engineering*, 59(2):303–306, July 2011.

[28] Fang Zhao, Meng Li, Yi Qian, and Joe Z Tsien. Remote measurements of heart and respiration rates for telemedicine. *PloS one*, 8(10):e71384, October 2013.

[29] G. de Haan and V. Jeanne. Robust pulse rate from chrominance-based rPPG. *IEEE Transactions on Biomedical Engineering*, 60(10):2878–2886, October 2013.

[30] Lonneke AM Aarts, Vincent Jeanne, John P Cleary, C Lieber, J Stuart Nelson, Sidarto Bambang Oetomo, and Wim Verkruysse. Non-contact heart rate monitoring utilizing camera photoplethysmography in the neonatal intensive care unit—a pilot study. *Early human development*, 89(12):943–948, December 2013.

[31] Xiaobai Li, Jie Chen, Guoying Zhao, and Matti Pietikainen. Remote heart rate measurement from face videos under realistic situations. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4264–4271, Columbus, OH, June 2014.

[32] Ronny Stricker, Steffen Müller, and Horst-Michael Gross. Non-contact video-based pulse rate measurement on a mobile service robot. In *The 23rd IEEE International Symposium on Robot and Human Interactive Communication*, pages 1056–1062, Edinburgh, UK, August 2014. IEEE.

[33] Sheng-Chieh Huang, Pei-Hsuan Hung, Chung-Hung Hong, and Hui-Min Wang. A new image blood pressure sensor based on ppg, rrt, bptt, and harmonic balancing. *IEEE sensors Journal*, 14(10):3685–3692, June 2014.

[34] L Tarassenko, M Villarroel, A Guazzi, J Jorge, DA Clifton, and C Pugh. Non-contact video-based vital sign monitoring using ambient light and autoregressive models. *Physiological measurement*, 35(5):807, March 2014.

[35] Daniel McDuff, Sarah Gontarek, and Rosalind W Picard. Improvements in remote cardiopulmonary measurement using a five band digital camera. *IEEE Transactions on Biomedical Engineering*, 61(10):2593–2601, May 2014.

[36] L. Feng, L. M. Po, X. Xu, Y. Li, and R. Ma. Motion-resistant remote imaging photoplethysmography based on the optical properties of skin. *IEEE Transactions on Circuits and Systems for Video Technology*, 25(5):879–891, May 2015.

[37] Weixuan Chen and Daniel McDuff. Deepphys: Video-based physiological measurement using convolutional attention networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 349–365, 2018.

[38] Xuesong Niu, Xingyuan Zhao, Hu Han, Abhijit Das, Antitza Dantcheva, Shiguang Shan, and Xilin Chen. Robust remote heart rate estimation from face utilizing spatial-temporal attention. In *2019 14th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2019)*, pages 1–8. IEEE, 2019.

[39] Rencheng Song, Senle Zhang, Juan Cheng, Chang Li, and Xun Chen. New insights on super-high resolution for video-based heart rate estimation with a semi-blind source separation method. *Computers in Biology and Medicine*, November 2019.

[40] Amogh Gudi, Marian Bittner, Roelof Lochmans, and Jan van Gemert. Efficient real-time camera based estimation of heart rate and its variability. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 0–0, 2019.

[41] Lingqin Kong, Yuejin Zhao, Liquan Dong, Yiyun Jian, Xiaoli Jin, Bing Li, Yun Feng, Ming Liu, Xiaohua Liu, and Hong Wu. Non-contact detection of oxygen saturation based on visible light imaging device using ambient light. *Optics express*, 21(15):17464–17471, July 2013.

[42] W. Wang, S. Stuijk, and G. de Haan. Exploiting spatial redundancy of image sensor for motion robust rPPG. *IEEE Transactions on Biomedical Engineering*, 62(2):415–425, February 2015.

[43] M. Lewandowska, J. Rumiński, T. Kocejko, and J. Nowak. Measuring pulse rate with a webcam – a non-contact method for evaluating cardiac activity. In

Federated Conference on Computer Science and Information Systems (FedCSIS), pages 405–410, Szczecin, Poland, September 2011.

[44] Ming-Zher Poh, Daniel J McDuff, and Rosalind W Picard. Non-contact, automated cardiac pulse measurements using video imaging and blind source separation. *Optics express*, 18(10):10762–74, May 2010.

[45] Gill R Tsouri and Zheng Li. On the benefits of alternative color spaces for noncontact heart rate measurements using standard red-green-blue cameras. *Journal of biomedical optics*, 20(4):048002, April 2015.

[46] Gee-Sern Hsu, ArulMurugan Ambikapathi, and Ming-Shiang Chen. Deep learning with time-frequency representation for pulse estimation from facial videos. In *2017 IEEE International Joint Conference on Biometrics (IJCB)*, pages 383–389. IEEE, 2017.

[47] Zhenyu Guo, Z Jane Wang, and Zhiqi Shen. Physiological parameter monitoring of drivers based on video data and independent vector analysis. In *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4374–4378, Florence, Italy, 2014. IEEE.

[48] Sergey Tulyakov, Xavier Alameda-Pineda, Elisa Ricci, Lijun Yin, Jeffrey F Cohn, and Nicu Sebe. Self-adaptive matrix completion for heart rate estimation from face videos under realistic conditions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2396–2404, Las Vegas, NV, 2016.

[49] Ce Liu et al. *Beyond pixels: exploring new representations and applications for motion analysis*. PhD thesis, Massachusetts Institute of Technology, 2009.

[50] Xiang Yu, Junzhou Huang, Shaoting Zhang, Wang Yan, and Dimitris N Metaxas. Pose-free facial landmark fitting via optimized part mixtures and cascaded deformable shape model. In *IEEE International Conference on Computer Vision (ICCV)*, pages 1944–1951, Sydney, Australia, December 2013.

[51] Vahid Kazemi and Josephine Sullivan. One millisecond face alignment with an ensemble of regression trees. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1867–1874, 2014.

[52] Brais Martinez, Michel F Valstar, Xavier Binefa, and Maja Pantic. Local evidence aggregation for regression-based facial point detection. *IEEE transactions on pattern analysis and machine intelligence*, 35(5):1149–1163, September 2012.

[53] Juan Cheng, Xun Chen, Lingxi Xu, and Z Jane Wang. Illumination variation-resistant video-based heart rate measurement using joint blind source separation and ensemble empirical mode decomposition. *IEEE journal of biomedical and health informatics*, 21(5):1422–1433, October 2016.

[54] Guha Balakrishnan, Fredo Durand, and John Guttag. Detecting pulse from head motions in video. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3430–3437, Portland,Oregon, 2013.

[55] W. Wang, B. Balmaekers, and G. de Haan. Quality metric for camera-based pulse rate monitoring in fitness exercise. In *IEEE International Conference on Image Processing (ICIP)*, pages 2430–2434, Phoenix, AZ, September 2016.

[56] Christoph Schneider, Florian Hanakam, Thimo Wiewelhove, Alexander Döweling, Michael Kellmann, Tim Meyer, Mark Pfeiffer, and Alexander Ferrauti. Heart rate monitoring in team sports—a conceptual framework for contextualizing heart rate measures for training and recovery prescription. *Frontiers in physiology*, 9, 2018.

[57] Juha Karvonen and Timo Vuorimaa. Heart rate and exercise intensity during sports activities. *Sports Medicine*, 5(5):303–311, May 1988.

[58] Mikko P Tulppo, Timo H Makikallio, Tapio Seppänen, Raija T Laukkanen, and Heikki V Huikuri. Vagal modulation of heart rate during exercise: effects of age and physical fitness. *American Journal of Physiology-Heart and Circulatory Physiology*, 274(2):H424–H429, February 1998.

[59] Martin Buchheit. Monitoring training status with hr measures: do all roads lead to rome? *Frontiers in physiology*, 5:73, February 2014.

[60] Hein AM Daanen, Robert P Lamberts, Victor L Kallen, Anmin Jin, and Nico LU Van Meeteren. A systematic review on heart-rate recovery to monitor changes in training status in athletes. *International journal of sports physiology and performance*, 7(3):251–260, September 2012.

[61] Willem Einthoven. Galvanometrische registratie van het menschelijk electrocardiogram. *In: Herinneringsbundel Professor S. S. Rosenstein. Leiden, Netherlands: Eduard Ijdo*, pages 101–106, 1902.

[62] Alrick B Hertzman. The blood supply of various skin areas as estimated by the photoelectric plethysmograph. *American Journal of Physiology-Legacy Content*, 124(2):328–340, October 1938.

[63] John Allen. Photoplethysmography and its application in clinical physiological measurement. *Physiological measurement*, 28(3):R1, February 2007.

[64] Thomas Brox, Andrés Bruhn, Nils Papenberg, and Joachim Weickert. High accuracy optical flow estimation based on a theory for warping. In *European conference on computer vision*, pages 25–36, May 2004.

[65] Ce Liu. *Beyond pixels: exploring new representations and applications for motion analysis*. PhD thesis, Massachusetts Institute of Technology, 2009.

[66] M. A. Haque, R. Irani, K. Nasrollahi, and T. B. Moeslund. Heartbeat rate measurement from facial video. *IEEE Intelligent Systems*, 31(3):40–48, May 2016.

[67] S. Tulyakov, X. Alameda-Pineda, E. Ricci, L. Yin, J. F. Cohn, and N. Sebe. Self-adaptive matrix completion for heart rate estimation from face videos under realistic conditions. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2396–2404, Las Vegas, NV, June 2016.

[68] Paul Viola and Michael J. Jones. Robust real-time face detection. *International Journal of Computer Vision*, 57(2):137–154, May 2004.

[69] Bruce D. Lucas and Takeo Kanade. An iterative image registration technique with an application to stereo vision. In *International Joint Conference on Artificial Intelligence*, volume 2, pages 674–679, Vancouver, Canada, August 1981.

[70] Gunnar Farnebäck. Two-frame motion estimation based on polynomial expansion. In *Scandinavian conference on Image analysis*, pages 363–370, June 2003.

[71] Simon Haykin. *Adaptive filter theory*. Prentice Hall, Upper Saddle River, NJ, 4th edition, 2002.

[72] Qiang Zhu, Mingliang Chen, Chau-Wai Wong, and Min Wu. Adaptive multi-trace carving based on dynamic programming. In *2018 52nd Asilomar Conference on Signals, Systems, and Computers*, pages 1716–1720, Pacific Grove, CA, October 2018.

[73] Shai Avidan and Ariel Shamir. Seam carving for content-aware image resizing. In *ACM Transactions on graphics (TOG)*, volume 26, page 10. ACM, August 2007.

[74] Thomas B Fitzpatrick. The validity and practicality of sun-reactive skin types i through vi. *Archives of dermatology*, 124(6):869–871, June 1988.

[75] Michael J Jones and James M Rehg. Statistical color models with application to skin detection. *International Journal of Computer Vision*, 46(1):81–96, January 2002.

[76] Bruce D Lucas, Takeo Kanade, et al. An iterative image registration technique with an application to stereo vision. 1981.

[77] Berthold KP Horn and Brian G Schunck. Determining optical flow. *Artificial intelligence*, 17(1-3):185–203, August 1981.

[78] Yu Shi and Eric Chang. Spectrogram-based formant tracking via particle filters. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, Hong Kong, April 2003.

[79] Kavita Kasi and Stephen A Zahorian. Yet another algorithm for pitch tracking. In *IEEE International Conference On Acoustics, Speech, and Signal Processing*, volume 1, pages I–361, May 2002.

[80] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. SSD: Single shot multibox detector. In *European conference on computer vision*, pages 21–37, October 2016.

[81] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, June 2016.

[82] Alan Lukezic, Tomas Vojir, Luka ˇCehovin Zajc, Jiri Matas, and Matej Kristan. Discriminative correlation filter with channel and spatial reliability. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6309–6318, 2017.

[83] G. Bradski. The OpenCV Library. *Dr. Dobb's Journal of Software Tools*, 2000.

[84] Georg Lempe, Sebastian Zaunseder, Tom Wirthgen, Stephan Zipser, and Hagen Malberg. Roi selection for remote photoplethysmography. In *Bildverarbeitung für die Medizin 2013*, pages 99–103. 2013.

[85] Elgammal Ahmed, M Crystal, and H Dunxu. Skin detection-a short tutorial. *Encyclopedia of Biometrics by Springer-Verlag Berlin Heidelberg*, 2009.

[86] Harry L Van Trees. *Detection, estimation, and modulation theory, part I: detection, estimation, and linear modulation theory*. John Wiley & Sons, 2004.

[87] Tim Schäck, Christian Sledz, Michael Muma, and Abdelhak M Zoubir. A new method for heart rate monitoring during physical exercise using photoplethysmographic signals. In *2015 23rd European Signal Processing Conference (EUSIPCO)*, pages 2666–2670. IEEE, 2015.

[88] Jonathan Allen. Short term spectral analysis, synthesis, and modification by discrete fourier transform. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 25(3):235–238, June 1977.

[89] Thomas G Stockham Jr. High-speed convolution and correlation. In *Proceedings of the April 26-28, 1966, Spring joint computer conference*, pages 229–233, April 1966.

[90] U Rajendra Acharya, K Paul Joseph, Natarajan Kannathal, Choo Min Lim, and Jasjit S Suri. Heart rate variability: a review. *Medical and biological engineering and computing*, 44(12):1031–1051, December 2006.

[91] Eduardo Gil, Michele Orini, Raquel Bailon, José María Vergara, Luca Mainardi, and Pablo Laguna. Photoplethysmography pulse rate variability as a surrogate measurement of heart rate variability during non-stationary conditions. *Physiological measurement*, 31(9):1271, August 2010.

[92] S Kolkur, D Kalbande, P Shimpi, C Bapat, and J Jatakia. Human skin detection using rgb, hsv and ycbcr color models. *arXiv preprint arXiv:1708.02694*, 2017.

[93] Qiang Zhu, Chau-Wai Wong, Chang-Hong Fu, and Min Wu. Fitness heart rate measurement using face videos. In *IEEE International Conference on Image Processing*, pages 2000–2004, Beijing, China, September 2017.

[94] Shai Avidan and Ariel Shamir. Seam carving for content-aware image resizing. 26(3):10, July 2007.

[95] Michael Wohlmayr, Michael Stark, and Franz Pernkopf. A probabilistic interaction model for multipitch tracking with factorial hidden markov models. *IEEE Transactions on Audio, Speech, and Language Processing*, 19(4):799–810, May 2011.

[96] Ralph Schmidt. Multiple emitter location and signal parameter estimation. *IEEE Transactions on Antennas and Propagation*, 34(3):276–280, March 1986.

[97] Richard Roy and Thomas Kailath. ESPRIT-estimation of signal parameters via rotational invariance techniques. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 37(7):984–995, July 1989.

[98] Roy L Streit and Ross F Barrett. Frequency line tracking using hidden Markov models. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 38(4):586–598, April 1990.

[99] Stephen A Zahorian and Hongbing Hu. A spectral/temporal method for robust fundamental frequency tracking. *The Journal of the Acoustical Society of America*, 123(6):4559–4571, June 2008.

[100] Ode Ojowu, Johan Karlsson, Jian Li, and Yilu Liu. ENF extraction from digital recordings using adaptive techniques and frequency tracking. *IEEE Transactions on Information Forensics and Security*, 7(4):1330–1338, May 2012.

[101] Jonathan S Abel, Ho John Lee, and Augustus P Lowell. An image processing approach to frequency tracking (application to sonar data). In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 2, pages 561–564, San Francisco, CA, March 1992.

[102] Thomas A Lampert and Simon EM O'Keefe. An active contour algorithm for spectrogram track detection. *Pattern Recognition Letters*, 31(10):1201–1206, July 2010.

[103] Yuzhou Liu and DeLiang Wang. Speaker-dependent multipitch tracking using deep neural networks. *The Journal of the Acoustical Society of America*, 141(2):710–721, February 2017.

[104] Nicholas Esterer and Philippe Depalle. A linear programming approach to the tracking of partials. *arXiv preprint arXiv:1901.05044*, 2019.

[105] Julian Neri and Philippe Depalle. Fast partial tracking of audio with real-time capability through linear programming. In *21st International Conference on Digital Audio Effects*, 2018.

[106] DCBP Rife and Robert Boorstyn. Single tone parameter estimation from discrete-time observations. *IEEE Transactions on Information Theory*, 20(5):591–598, September 1974.

[107] Ravi Garg, Avinash L Varna, Adi Hajj-Ahmad, and Min Wu. 'Seeing' ENF: power-signature-based timestamp for digital multimedia via optical sensing and signal processing. *IEEE Transactions on Information Forensics and Security*, 8(9):1417–1432, September 2013.

[108] Mingyang Wu, DeLiang Wang, and Guy J Brown. A multipitch tracking algorithm for noisy speech. *IEEE Transactions on Speech and Audio Processing*, 11(3):229–241, May 2003.

[109] Arthur P Dempster, Nan M Laird, and Donald B Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 1–38, 1977.

[110] Adi Hajj-Ahmad, Ravi Garg, and Min Wu. Spectrum combining for ENF signal estimation. *IEEE Signal Processing Letters*, 20(9):885–888, September 2013.

[111] Julius O Smith and Xavier Serra. *PARSHL: An analysis/synthesis program for non-harmonic sounds based on a sinusoidal representation*. CCRMA, Department of Music, Stanford University, 1987.

[112] Ingo R Titze. *Principles of Voice Production*. Englewood Cliffs, N.J., Prentice-Hall, 1994.

[113] Ronald J Baken and Robert F Orlikoff. *Clinical Measurement of Speech and Voice*. Cengage Learning, 2000.

[114] Dik J Hermes. Measurement of pitch by subharmonic summation. *The Journal of the Acoustical Society of America*, 83(1):257–264, 1988.

[115] Chao Wang and Stephanie Seneff. Robust pitch tracking for prosodic modeling in telephone speech. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 3, pages 1343–1346, Istanbul, Turkey, June 2000.

[116] Hung-Kuo Chu, Chia-Sheng Chang, Ruen-Rone Lee, and Niloy J Mitra. Halftone QR codes. *ACM Transactions on Graphics*, 32(6):217: 1–8, November 2013.

[117] The Global Burden of Disease: 2017 update.

[118] Sudden death in young people: Heart problems often blamed.

[119] Xiaomao Fan, Qihang Yao, Yunpeng Cai, Fen Miao, Fangmin Sun, and Ye Li. Multiscaled fusion of deep convolutional neural networks for screening atrial fibrillation from single lead short ecg recordings. *IEEE Journal of Biomedical and Health Informatics*, 22(6):1744–1753, nov 2018.

[120] W Bruce Fye. A history of the origin, evolution, and impact of electrocardiography. *American Journal of Cardiology*, 73(13):937–949, May 1994.

[121] Andrew Reisner, Phillip A Shaltis, Devin McCombie, and H Harry Asada. Utility of the photoplethysmogram in circulatory monitoring. *Anesthesiology: The Journal of the American Society of Anesthesiologists*, 108(5):950–958, May 2008.

[122] Alistair EW Johnson, Tom J Pollard, Lu Shen, H Lehman Li-wei, Mengling Feng, Mohammad Ghassemi, Benjamin Moody, Peter Szolovits, Leo Anthony Celi, and Roger G Mark. Mimic-iii, a freely accessible critical care database. *Scientific data*, 3:160035, 2016.

[123] Lena Nilsson, Anders Johansson, and Sigga Kalman. Respiration can be monitored by photoplethysmography with high sensitivity and specificity regardless of anaesthesia and ventilatory mode. *Acta anaesthesiologica scandinavica*, 49(8):1157–1162, September 2005.

[124] Walter Karlen, Srinivas Raman, J Mark Ansermino, and Guy A Dumont. Multiparameter respiratory rate estimation from the photoplethysmogram. *IEEE Transactions on Biomedical Engineering*, 60(7):1946–1953, July 2013.

[125] Takuo Aoyagi and Katsuyuki Miyasaka. Pulse oximetry: its invention, contribution to medicine, and future tasks. *Anesthesia and Analgesia*, 94(1 Suppl):S1, 2002.

[126] RA Payne, CN Symeonides, DJ Webb, and SRJ Maxwell. Pulse transit time measured from the ecg: an unreliable marker of beat-to-beat blood pressure. *Journal of Applied Physiology*, 100(1):136–141, January 2006.

[127] William A Marston. PPG, APG, duplex: which noninvasive tests are most appropriate for the management of patients with chronic venous insufficiency? In *Seminars in Vascular Surgery*, volume 15, pages 13–20. Elsevier, March 2002.

[128] John Allen and Alan Murray. Development of a neural network screening aid for diagnosing lower limb peripheral vascular disease from photoelectric plethysmography pulse waveforms. *Physiological Measurement*, 14(1):13, February 1993.

[129] John Allen and Alan Murray. Similarity in bilateral photoplethysmographic peripheral pulse wave characteristics at the ears, thumbs and toes. *Physiological Measurement*, 21(3):369, August 2000.

[130] Rohan Banerjee, Aniruddha Sinha, Anirban Dutta Choudhury, and Aishwarya Visvanathan. PhotoECG: Photoplethysmography to estimate ECG parameters. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 4404–4408, Florence, Italy, May 2014. IEEE.

[131] H GholamHosseini, H Nazeran, and B Moran. ECG compression: evaluation of FFT, DCT, and WT performance. *Australas Phys Eng Sci Med*, 21(4):186–192, December 1998.

[132] Stephen A Martucci. Symmetric convolution and the discrete sine and cosine transforms. *IEEE Transactions on Signal Processing*, 42(5):1038–1051, May 1994.

[133] Hastie Trevor, Tibshirani Robert, and Friedman JH. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer, New York, 2009.

[134] Jiapu Pan and Willis J Tompkins. A real-time qrs detection algorithm. *IEEE Transaction on Biomedical Engineering*, 32(3):230–236, March 1985.

[135] Chih-Chung Chang and Chih-Jen Lin. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2(3):27, 2011.

[136] Ikaro Silva and George B Moody. An open-source toolbox for analysing and processing PhysioNet databases in MATLAB and octave. *Journal of Open Research Software*, 2(1), September 2014.

[137] Kai Li, Zhengming Ding, Sheng Li, and Yun Fu. Discriminative semi-coupled projective dictionary learning for low-resolution person re-identification. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[138] Shenlong Wang, Lei Zhang, Yan Liang, and Quan Pan. Semi-coupled dictionary learning with applications to image super-resolution and photo-sketch synthesis. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2216–2223. IEEE, 2012.