MOBILE CARDIAC PULSE MEASUREMENTS

Haiying Xia, Zhouxiao Bao, Haomiao Jiang

Electrical Engineering, Stanford
{hyxia,zhouxiao,hjiang36}@stanford.edu

ABSTRACT

Cardiac pulse measurements play an important role in diagnosing heart diseases. In this paper, we describe a methodology that simply depends on a series of facial images. By performing automatic face tracking and blind source separation of three color channels into independent components, we extract the blood volume pulse from the recorded facial frames. Heart rates can be subsequently quantified from previous results. The introduced method is featured as low-cost, automatic, absolutely contact-free, and motion-tolerant. We implement the measuring platform on a PC and web first, and then realize an Android phone version using server-client communication.

***Index Terms*—** facial images, face detection, detrend, ICA, peak detection, heart rate

1. INTRODUCTION

In current society, an increasing number of people are suffering from cardiovascular catastrophes. Resting heart rate (HR) is one of the simplest cardiovascular parameters. In [1], HR is identified as an independent risk factor for cardiovascular diseases. In this paper, we describe a remote, non-contact and non-invasive methodology to conveniently measure human heart rate.

Few non-contact measurement methods are proposed based on image signal processing algorithms. In order to measure the heart rate simultaneously, Chihiro and Yuji in [2] develop a non-contact device by applying autoregressive (AR) spectral analysis to a time-lapse image. In [3, 4], Ming-Zher et al. use face detection and Independent Component Analysis (ICA) to analyze a webcam captured face recording, and obtain the cardiac pulse measurements.

Most of the existing algorithms are achieved on PC or wired devices, which are non-portable and thus inconvenient. Our purpose is to realize the non-contact cardiac pulse measurements on a portable device. We first implement the algorithm on a PC by continuously taking facial images and extracting the corresponding HR using image processing techniques. Because of the increasingly developing market of mobile phones, we then make the application portable by using client-server communication for an Android phone.

Thanks to Kehuang Li for technical help; thanks to Huizhong Chen and David Chen for project advice; and thanks to Prof. Girod for the useful lecture.

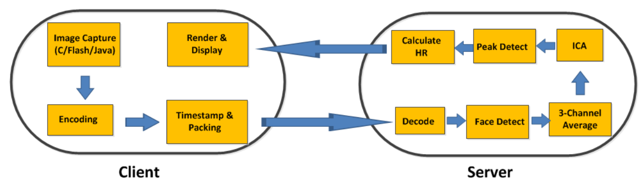

Fig. 1. The cardiac system structure.

This paper is organized as follows. The system structure is described in Section 2, where details in gathering the images for the three application versions (PC, flash/web and mobile) are provided in 2.1, and the post-processing algorithm for the HR detection and estimation is described in 2.2 and 2.3. To prove the feasibility of our system, we conduct experiments for each application, and demonstrate the corresponding relative error in Section 3. Finally, a conclusion is made in Section 4 for the paper.

2. THE CARDIAC SYSTEM

Fig.1 shows the system structure of our HR measurement application. To make the application more flexible and portable, we provide three different versions in this paper: PC (mainly implemented using Matlab and C++), Flash/Web and Mobile. In the client illustrated in Fig.1, a human face image is first recorded by webcam/camera. For mobile application, it is encoded and stamped by time (Note: timestampes are used to compute the sampling rate of the webcam/camera as $f_s = \frac{NumberOfFrames}{\Delta t}$ in Hz). Every 5 sequential frames are packaged and transmitted to the sever terminal via wireless network. The sever should first decode the signal if necessary (e.g., for mobile application). It implements all the post-processing algorithm (described in section 2.2 and section 2.3) and transmit the computed HR value to the client. Finally, the estimated HR is displayed in the client terminal.

2.1. Data gathering

2.1.1. PC Version

In the PC version, we programmed a C++ script to capture images from the laptop's webcam. The captured images were directly sent to the image processing program. After every 10 images were processed, the results were returned to the Matlab post-processing program as a 10×3 matrix. The capturing speed of this method was around 10 to 15 frames per second (fps) and the resolution of the captured image could be as high as 640×480

2.1.2. Flash/Web Version

In the flash/web version, we programmed a flash action script to capture images from the laptop's webcam. The captured images were first encoded as JPEG and then sent to the server as HTTP raw data. After sending, the flash traced the echo message from the server and display it with the captured images on the screen. The speed was around 10 fps and the image resolution was 320×240.

2.1.3. Mobile Version

In the mobile version, we implemented a Java program to capture images from the Android's camera. To enhance the transferring rate, images were encoded as JPEGs and every five images captured were concatenated into a data file to be sent in one connection. Also, we used 5 threads to send data in asynchronous mode. The total speed highly depended on the network. Generally, it could capture and send five 320×240 frames per second.

2.2. Frame Processing

To recover the blood volume pulse, we use an automated face tracker and localized face in each image captured. The Open Computer Vision (OpenCV) is utilized to obtain the coordinates of the face based on the algorithm in [5] and [6]. A cascade of boosted classifier uses 14 Haar-like digital image features trained with positive and negative examples. A small square image only containing the detected face is then obtained.

2.3. HR estimation

For each detected face frame, we define our region of interest (ROI) as middle 60% width and the full height of the box. As illustrated in Fig.2, the ROI contains three RGB channels. All the pixels in the ROI in every channel of a frame is summed to obtain three RGB raw sample points, which are then arranged sequentially to form three RGB raw traces $r_1(t), r_2(t)$, and $r_3(t)$. The raw traces are each 30 seconds. Note that these raw data are not necessarily stationary. During our signal processing procedure, only stationary segments could be

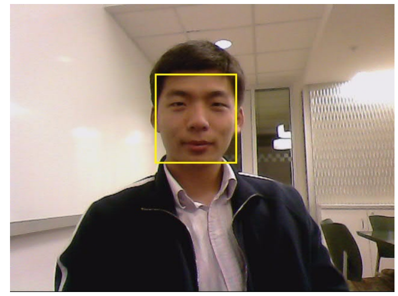

Fig. 2. The webcam captured face with the face-detect result in the yellow box.

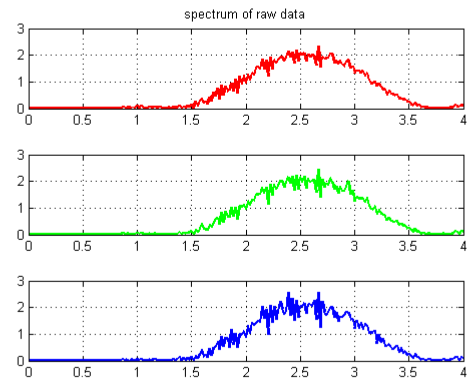

Fig. 3. The power spectra of the normalized RGB traces using Welch's method.

analyzed. To solve this problem, the three raw traces are detrended based on a smoothness priors approach proposed in [7], with a regularization parameter $\lambda = 10$ (corresponding cutoff frequency is $0.059 f_s$).

The raw traces in the RGB channels are then normalized as follows:

$$x_i(t) = \frac{r_i(t) - \mu_i}{\sigma_i}, \ i = 1, 2, 3 \tag{1}$$

where μ_i and σ_i are the mean and the standard deviation of $r_i(t)$. Fig.3 shows the estimated power spectra of $x_i(t)$ using Welch's method [8]. Since the detrending method operates like a time-varying FIR high pass filter, the power "spreads" on low frequency band while it is suppressed at DC. To further discover the relationship between the measured data components and blood volume pulse (BVP), an ICA algorithm based on the (joint) diagonalization of cumulant matrices - JADE in [9, 10, 11] is applied to the normalized RGB traces $x_i(t),\ i = 1, 2, 3$. The three separated source signals (Fig.4 are thus obtained, and their corresponding spectrum magnitudes are shown in Fig.5.

Meanwhile, the separated source signals are smoothed us-

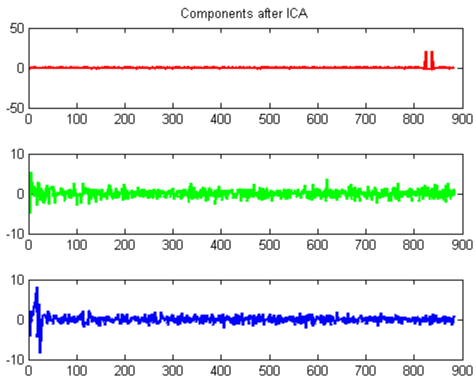

Fig. 4. Three independent signal components using the ICA algorithm described in [9, 10, 11].

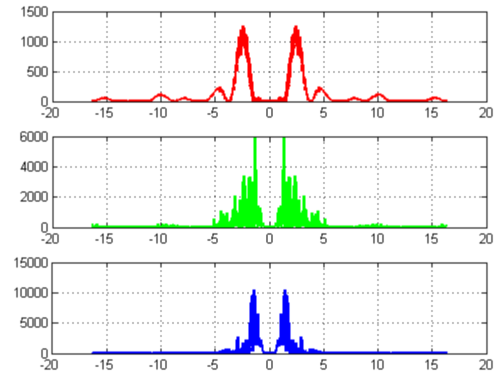

Fig. 5. The spectrum magnitudes of the Three independent signal components.

ing a lowpass filter and bandlimited to 1-4 Hz (corresponds to 60-240 bpm). In this example, we design the lowpass filter as a five-point moving average filter. As indicated in [3], although the order of the three resulting signals is random, the frequency point, at which the highest peak in the three spectra occurs, is selected as the BVP value (in Hz). However, such a BVP related peak should be refined to obtain a better estimation of BVP. In this paper, we apply the parabolic interpolation to the detected peak frequency. Finally, the heart rate (HR) is computed as $60\times$ (estimated BVP).

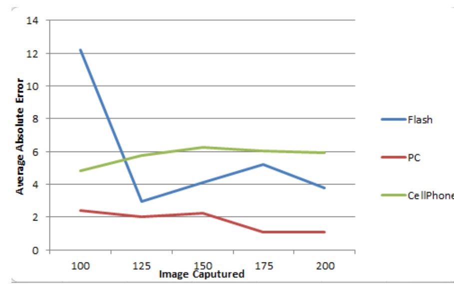

Fig. 6. Comparison of the HR errors in the three different versions: PC, web, and mobile.

3. EXPERIMENT RESULTS

3.1. Environment

For PC and flash versions, we use a basic webcam embedded in a laptop (built-in camera on a Lenovo Y570) to take images for analysis. Images are taken in color (24-bit RGB with 3 channels $\times$ 8 bits/channel) at 10 to 15 frames per second (fps) with resolution of 640 $\times$ 480. For mobile version, we use the embedded camera on the provided Android phone. Images are taken at 5 to 7 fps with resolution of 320 $\times$ 240. We encoded the time stamp for every image and pack 5 images to one file. Sending thread number is set to 5. We use C (Visual Studio 2010)/Flash (Adobe Flash CS5)/ Java (Eclipse SDK 3.7.2) for image capturing, OpenCV (2.4.0) for face detection, MATLAB (2012a) for data post-processing. The server code is written in PHP by Adobe Dreamweaver CS5.

3.2. Results

We analyze recordings more than one minute long of 4 participants at rest to achieve the experiment results. (To further test the robustness of our application, we plan to invite more participants.) Each participant sits in a room, with either natural light or bulb light. During the experiment, participants are asked to keep still, breathe spontaneously, and face the webcam. In the test for the web and mobile applications, in order to obtain accurate results, an experiment location with a strong wireless signal is highly recommended. Because a weak wireless signal causes unordered frames and missed packages, which impacts the final results. After the application starts, the participant faces the webcam/camera for 30-60 seconds before the HR appears on the screen. Since all the applications start post-processing just after the first 100 pictures are collected, the wait time for obtaining the first/updated HR value depends on the sampling rate of the camera and the transmission rate of the network. The applications cease the thread after 1024 frames are processed. Fig.6 shows the HR errors in the three different versions: PC, web, and mobile. The results from the PC implementation show a higher accuracy since the problem of out-of-order and package missing doesn't occur in this version. Besides, the pictures recorded from the webcam don't need to be transmitted, so the sampling rate is high and hence leads to better estimation. On the contrary, the results from the mobile implementation are the least accurate, mainly due to the transmission performance between server and client.

4. CONCLUSION

In this paper, a method for non-contact heart rate measurement is described and implemented. After the images are recorded via a webcam/camera, face detection algorithm is used to figure out the interest region and location. The sequential face data are then applied to construct the normalized RGB raw data. Since such raw data is not stationary, they should be set stationary by detrending; otherwise, the spectrum of these raw data would only contain large energy at DC component. In addition, Welch's method is used to estimate the power spectra of the detrended RGB data. The power spectra in this example could also be approximated using Periodogram method, but Welch's method achieves better estimation results. The RGB data are then decomposed into three independent components using ICA. As implied in the previous work, the magnitude peak location in the spectra of the three components is the corresponding BVP value in Hz. In order to achieve an accurate estimation, several signal processing algorithms are applied before the peak detection.

We design three application versions for this project, among which the PC version obtains the most accurate estimation during the experiments. In the ideal network environment (i.e., the wireless signal is good enough for client to transmit lossless package to the server, and the transmission speed is over 10 frames per second), the HR detection accuracy of the mobile version would approach that of PC version.

According to the experiment settings and results, our future work could include: Improving the accuracy and robustness of the mobile system; feature tracking and light adjustments for moving objects; and cardiac rate measurement for multiple faces.

5. REFERENCES

[1] S. Cook, M. Togni, M.C. Schaub, P. Wenaweser, and O.M. Hess, "High heart rate: a cardiovascular risk factor?," *European heart journal*, vol. 27, no. 20, pp. 2387–2393, 2006.

[2] C. Takano and Y. Ohta, "Heart rate measurement based on a time-lapse image," *Medical engineering & physics*, vol. 29, no. 8, pp. 853–857, 2007.

[3] M.Z. Poh, D.J. McDuff, and R.W. Picard, "Non-contact, automated cardiac pulse measurements using video imaging and blind source separation," 2010.

[4] M.Z. Poh, D. McDuff, and R. Picard, "A medical mirror for non-contact health monitoring," in *ACM SIGGRAPH 2011 Emerging Technologies*. ACM, 2011, p. 2.

[5] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*. IEEE, 2001, vol. 1, pp. I–511.

[6] R. Lienhart and J. Maydt, "An extended set of haar-like features for rapid object detection," in *Image Processing. 2002. Proceedings. 2002 International Conference on*. Ieee, 2002, vol. 1, pp. I–900.

[7] M.P. Tarvainen, P.O. Ranta-Aho, and P.A. Karjalainen, "An advanced detrending method with application to hrv analysis," *Biomedical Engineering, IEEE Transactions on*, vol. 49, no. 2, pp. 172–175, 2002.

[8] P. Welch, "The use of fast fourier transform for the estimation of power spectra: a method based on time averaging over short, modified periodograms," *Audio and Electroacoustics, IEEE Transactions on*, vol. 15, no. 2, pp. 70–73, 1967.

[9] J.F. Cardoso and A. Souloumiac, "Blind beamforming for non-gaussian signals," in *Radar and Signal Processing, IEE Proceedings F*. IET, 1993, vol. 140, pp. 362–370.

[10] J.F. Cardoso et al., "On the performance of orthogonal source separation algorithms," in *Proc. EUSIPCO*. Edinburgh, UK, 1994, vol. 94, pp. 776–779.

[11] J.F. Cardoso, "High-order contrasts for independent component analysis," *Neural computation*, vol. 11, no. 1, pp. 157–192, 1999.

6. APPENDIX: INDIVIDUAL WORK BREAK-DOWN

Haomiao Jiang: Mobile version server construction, Flash/web version implementation, PC version image capturing and face detection (in C) report writing.

Zhouxiao Bao: Mobile version client-side implementation, including image capturing, encoding, time-stamping, packing, multi-thread communication etc, face detection (in Java), report writing.

Haiying Xia: Implementation of the heart rate measurement algorithm (in Matlab, used for all versions), initial webcam image capturing using JavaCV, report writing.