Real-time Quantifying Heart Beat Rate from Facial Video Recording on a Smart Phone using Kalman Filters

Wen Jun Jiang[1], Shi Chao Gao[1], Peter Wittek[2] and Li Zhao[1]

***Abstract*— Photoplethysmography (PPG) can be carried out through facial video recording by a smart phone camera in ambient light. The main challenge is to eliminate motion artifacts and ambient noise. We describe a real-time algorithm to quantify the heart beat rate from facial video recording captured by the camera of a smart phone. We extract the green channel from the video. Then we normalize it and use a Kalman filter with a particular structure to eliminate ambient noise. This filter also enhances the heart pulse component in the signal distorted by Gaussian noise and white noise. After that we employ a band-pass FIR filter to remove the remaining motion artifacts. This is followed by peak detection or Lomb periodogram to estimate heart rate. The algorithm has low computational overhead, low delay and high robustness, making it suitable for real-time interaction on a smart phone. Finally we describe an Android application based on this study.**

***Index Terms*— Kalman filter, Photoplethysmography, Smart Phone, Real Time, Motion Artifact, Video Recording**

I. INTRODUCTION

Non-invasive measurement of cardiac activity is promising because it causes the least discomfort to patients. Photoplethysmography (PPG) is suitable for this kind of application. It is an optical technique that measures the change of blood volume in the microvascular vessels from light reflected from or transmitted through the skin. According to the Beer-Lambert law, the change of blood volume in the tissue is reversely proportional to the intensity of the light reflected or transmitted from skin [1]. This is hardly seen by the naked eye, but it can be captured by the commercial camera on board of smart phones.

Measuring the PPG signal in transmitted mode and also in reflected mode are under development in both academia and industry, promising future applications of non-invasive monitoring of cardiac functions. Nowadays, the transmitted mode is often used in measuring PPG signal from the fingertip [2]. We, on the other hand, focus on measuring the PPG signal from facial video recording, which is based on the reflected mode. Since there is more noise in the remote measurement, the main problem is how to extract a useful signal and how to remove noise.

Verkruysse et al., (2008) proposed a method of measuring plethysmographic signals remotely using ambient light with a consumer-grade digital camera [3]. They find that although all of the red, green, and blue channels contain plethysmographic information, the green channel contains the strongest signal.

Poh et al., (2011) suggested a procedure for computing heart rate from digital color video [4]. They split the red, green, and blue channels from the video, detrended each channel using a method based on smoothness priors approach [5], normalized them, and used a linear combination of these three channels as the blood pulse signal. They used Independent Component Analysis (ICA) based on the joint approximate diagonalization of eigenmatrices (JADE) algorithm [6] to calculate three candidate weights for each channel and used the fast Fourier transform (FFT) to select the best one.

The core idea in this approach is that the red, green, and blue signals come from three sources and these sources can be reconstructed from the obtained signals through ICA. They interpreted one source as the blood pulse signal, but they didn't explain what noises are composed of. In addition, their algorithm had a high computational overhead, so it could hardly run in real time on smart phones. To solve this problem, Kwon et al., (2012) left out ICA and used only the raw green channel to calculate heart rate [7]. In their experiments, the independent sources obtained from ICA were similar to, or not as good as the raw signal of the green channel. Roald, (2013) tested ICA on RGB, HSL and HSV color space and found the best ICA channels are not as good as the best non-ICA channels [8]. They also suggested that the Hue and Saturation channels were more robust against noise than the green channel. Furthermore, they divided the noise into several components:

$$\sigma = \sigma_w + \sigma_g + \sigma_m + \sigma_{other} \tag{1}$$

Here, σ_w is white noise, σ_g is Gaussian noise, σ_m is motion artifact and σ_{other} is other unknown sources of noise.

Among these types of noise, white noise and Gaussian noise exist in the whole frequency domain, while motion artifact is usually a low-frequency signal. Roald, (2013) used a band-pass Finite Impulse Response (FIR) filter, which is a frequency-domain filter, to remove these noises. But since white noise and Gaussian noise exist both in the band and out of band, it is hard to remove it by designing a filter in the frequency domain. Also, the attempt to smooth the signal using a time-domain moving average filter, such as a five-point average, would attenuate the heart pulse signal.

Forgoing the elimination of noise, Hao-Yu Wu et al., (2012) proposed a method to amplify heart pulse in the facial video and to make it seen by the naked eye [9]. They used both spacial and temporal processing of the video.

Here, we propose a structure used in Kalman filters, which, using only temporal processing, filter out Gaussian white noise in the PPG signal, and they also amplify the heart pulse component in the signal.

[1]W.J. Jiang, L. Zhao, and S.C. Gao are with the Tsinghua University.
[2]P. Wittek is with the University of Borås.

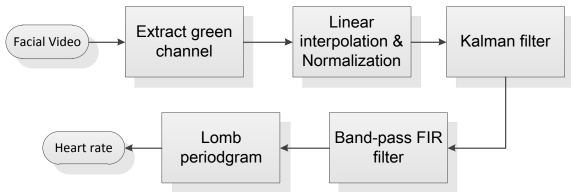


Fig. 1: System structure

The key technical contributions of this paper are summarized as follows:

- We propose an algorithm based on Kalman filter to smooth the PPG signal and amplify heart pulse.
- We implement our method as an Android application.

The rest of this paper is organized as follows. The experimental procedure and design details of our approach are described in Section II. Then, we conduct experiments and compare results in Section III. Finally, we conclude the paper in Section IV.

II. Proposed Method

A. Experimental Procedure

We describe our system structure by the flowchart in Figure (1). First, we extract the green channel from facial video and record the time stamp. Then, we interpolate the signal to 15 frames per second (fps). After that, we normalize the signal $y(t)$ as follows:

$$\mathrm{y}'(t) = \frac{y(t) - \mu}{\sigma} \tag{2}$$

where μ and σ are the mean and standard deviation of $y(t)$. Then we use Kalman filter to smooth the signal, and to amplify the heart pulse. After the noise in the signal has been attenuated, we filter it with a band-pass FIR filter (30-point, rectangular window, $0.66-2.66$ Hz) . Finally, we extract heart rate from the signal by a peak detection algorithm for real-time processing, or by the Lomb periodogram for accuracy and robustness. We developed an Android application Kiwi Face Cardiac Function Detection based on this procedure. In what follows, we describe our algorithm in more detail.

B. Smoothing and Amplifying Signal with Kalman Filter

The Kalman filter is a nonstationary recursive filter from the view of the minimum of variance which can estimate the useful signal in noisy time series [10]. A steady state Kalman approach can be described by two linear difference stochastic equations:

$$\mathbf{x_k} = A\mathbf{x_{k-1}} + \mathbf{w_k}, \tag{3}$$

$$z_k = H\mathbf{x_k} + v_k, \tag{4}$$

where,

$$\mathbf{x_k} = [x_k, x_{k-1}, x_{k-2}]^T, \tag{5}$$

$$\mathbf{w}_k = [w_k, 0, 0]^T. \tag{6}$$

In our experiment, the state variable vector $\mathbf{x_k}$ is a 3×1 column vector, representing the motionless signal vector. The measured value z_k is a scalar. The vector $\mathbf{w_k}$ is the state transaction noise and the value v_k is the measurement noise. We assume them to be independent of each other and that they satisfy Gaussian white noise distributions $\mathbf{w_k} \sim N(0, Q)$, $v_k \sim N(0, R)$. The 3×3 matrix A relates the state at the previous time step $k-1$ to the state at the current step k, in the absence of process noise. The 1×3 row vector H relates the state to the measurement [11]. Their values are set as follows:

$$A = \begin{bmatrix} 2 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \tag{7}$$

$$H = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \tag{8}$$

Kalman filtering equations contain two parts: time update equations and measurement update equations.

1. Time update equations

$$\mathbf{\hat{x}_k^-} = A\mathbf{\hat{x}_{k-1}} \tag{9}$$

$$P_k^- = AP_{k-1}A^T + Q \tag{10}$$

2. Measurement update equations

$$K_k = P_k^- H^T \left(HP_k^- H^T + R\right)^{-1} \tag{11}$$

$$\mathbf{\hat{x}_k} = \mathbf{\hat{x}_k^-} + K_k \left(z_k - H\mathbf{\hat{x}_k^-}\right) \tag{12}$$

$$P_k = (I - K_k H) P_k^- \tag{13}$$

where K_k is the Kalman gain, the 3×3 matrix P_k is the estimation of the error covariance, the 3×3 matrix P_k^- is prediction of the error covariance, the 3×1 column vector $\mathbf{\hat{x}_k}$ is estimation of the state variable, the 3×1 column vector $\mathbf{\hat{x}_k^-}$ is prediction of the state variable. Here we initialize P to be a unit matrix, R to be 1 and Q to be the following matrix:

$$Q = \begin{bmatrix} 0.02 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{14}$$

To derive the constants in A and H, we regard x_k as a function of a uniform time sampling t_k, $x_k = x(t_k), k = 1, 2, ...$ Assuming that the constant spacing of t is Δt, we get the equation $t_{k+1} = t_k + \Delta t$. When estimating x_{k+1}, we approximate it at its first-order Taylor series:

$$x(t_{k+1}) = x(t_k + \Delta t) \approx x(t_k) + \Delta t \left.\frac{\partial x}{\partial t}\right|_{t_k} \tag{15}$$

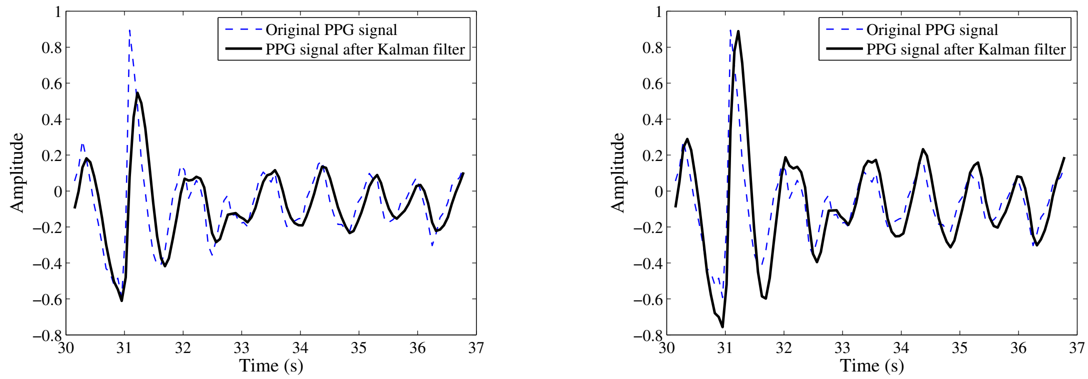
where we use *backward Euler* to approximate the derivative:

$$\left.\frac{\partial x}{\partial t}\right|_{t_k} \approx \frac{x(t_k) - x(t_{k-1})}{\Delta t} \tag{16}$$

Combining equation 15 with equation 16, we estimate x_{k+1} with $\hat{x}_{k+1}$ as follows:

$$\hat{x}_{k+1} = x_k + (x_k - x_{k-1}) = 2x_k - x_{k-1} \tag{17}$$

which is exactly the expression in the first row of A. If we want to amplify the heart pulse, we make $\hat{z}_{k+1}$, the estimation of z_{k+1}, lower than x_{k+1}, which implies the measured value



(a) (b)

Fig. 2: The comparison of the original PPG signal and that after Kalman filter. In Figure (2a), H = [1 0 0], the high-frequency noise in the signal is eliminated and the heart pulse is remained. In Figure (2b), H = [0 0 1], the heart pulse is enhanced after filtering.

is lower than the true value. We introduce a factor $\alpha \in [-1, 1]$ and construct the estimation expression as follows:

$$\hat{z}_{k+1} = x_k + \alpha(x_k - x_{k-1}) \tag{18}$$

The smaller α is, the more x_{k+1} exceeds z_{k+1}, which means the more the heart pulse is amplified. When $\alpha = 1$, $\hat{z}_{k+1} = 2x_k - x_{k-1} = \hat{x}_{k+1}$, which is equivalent to $H = [1\ 0\ 0]$. While when $\alpha = -1$, $\hat{z}_{k+1} = x_{k-1}$, which is equivalent to $H = [0\ 0\ 1]$. The comparison of these two cases are shown in Figure (2).

C. *Estimating Heart Rate Using Power Spectral Density*

After the Kalman filter, the filtered signal is smoothed by a band-pass filter (30 points, rectangle window, $0.66-2.66$ Hz). To estimate heart rate, we either use a peak detection algorithm for real-time performance. In this way, we get peaks in the PPG signal and the frequency of heart beat is the inverse of the average time interval between nearby peaks. Alternatively, we use the Lomb periodogram [12] for a more accurate and robust result. We calculate power spectral density (PSD) of the PPG signal with 256 points, oversampling factor $ofac = 4$, and maximum frequency to be the average Nyquist frequency [13]. In this way, the frequency of heart beat is the frequency of the peak in PSD and its resolution is about 1 beat per minute (bpm). Figure (3) shows the result of Lomb periodogram.

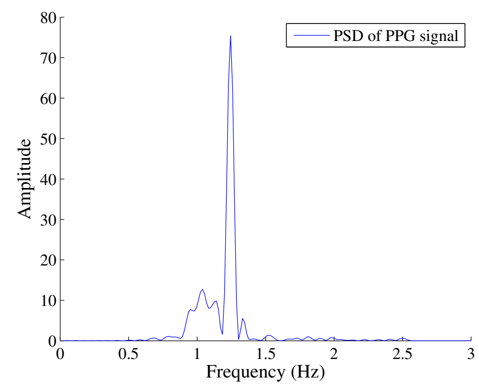


Fig. 3: Power spectral density (PSD) of the PPG signal calculated by Lomb periodogram

III. Experimental Results

A. *Data Collection*

We took samples from 15 participants, 12 males and 3 females. Their ages ranged between 25-35 years. A custom-developed application on an Android phone was used to collect data. During the experiment, the subjects were seated at a table, holding the smart phone by their right hand at a distance of 0.3m between the front camera and their face. A pulse oximeter was clipped on their left index finger to show the true heart beat value. The data-collection application ran at 15 frame per second for about 30 seconds. We asked the subjects to sit without movement and keep their face in the region of interest, which is a rectangular area at the center of the screen.

We used an LG G2 phone for all the subjects to acquire PPG trace. The phone ran Android 4.2.2. The pulse oximeter was an Etcomm HC-801. We set the region of interest as 400 by 400 pixel. The original picture format was YUV420sp. We converted it to RGB format and down-sampled it to 100 by 100 pixel. We recorded the time stamp and the mean value of the green channel as time and amplitude of PPG signal.

B. *Analysis*

We compared the efficiency of our algorithm against a state-of-art algorithm, ICA, on traces from the mobile phone. Table I shows the estimated heart rates and the error rate between the estimated one and reference one. We found that when the heart rate was between $60-80$ bpm, both algorithms had a low error rate. But when heart beat exceeded 80 bpm, the accuracy and robustness of both algorithms decreased. Our algorithm gave a closer value to the true heart rate than of ICA.

TABLE I: The estimated heart rates and the error rates

Subj.	Reference (bpm)	ICA (bpm)	Kalman (bpm)	ICA error rate(%)	Kalman error rate(%)
1	75	74.6	74.6	0.5	0.5
2	72	71.1	71.1	1.3	1.3
3	73	71.1	71.1	2.6	2.6
4	76	74.6	74.6	1.8	1.8
5	86	70.2	70.2	18.4	18.4
6	85	75.5	75.5	11.2	11.2
7	115	49.2	75.5	57.2	34.3
8	91	72.9	96.5	19.9	6.0
9	60	79	59.7	31.7	0.5
10	66	65.8	65	0.3	1.5
11	68	67.6	67.6	0.6	0.6
12	76	81.7	81.7	7.5	7.5
13	75	72	71.1	4.0	5.2
14	68	68.5	67.6	0.7	0.6
15	69	114.1	72	65.4	4.3
Mean				14.9	6.4

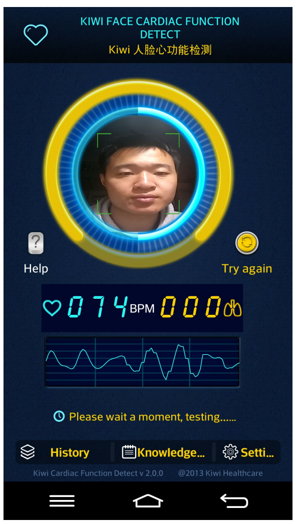


Fig. 4: Screenshot of our application, Kiwi Face Cardiac Function Detection

C. Android Application

We have developed Kiwi Face Cardiac Function Detection, an Android application to estimate heart rate from a facial video recording. We use the face detection function provided by Android API [14] to detect whether there is human face in the center 400×400 pixel region of the video frame. Since the speed of the face detection function is related to size of the picture and how many times the function is called, we extract a 400×400 pixel gray picture from the region, and down-sample it to 50×50 pixels. Also, we run the face detection function every second to keep a balance between the real-time performance and computational overhead. Once a human face is detected, this application shows the smoothed signal and the estimation of heart rate in real time. The screenshot of our application is shown in Figure (4).

IV. Conclusion and Future Work

In this paper, we proposed a real-time heart rate estimation algorithm on smart phone. Various types of noise, including motion artifacts and ambient noise were taken into consideration in our algorithm. Compared to the method described by Poh et al., (2011), we used only the green channel rather than all three RGB channels, because the green channel contained the strongest heart pulse signal. We also used a Kalman filter to replace the ICA algorithm for signal smoothing and enhancement in order to implement the real-time processing of the PPG signal, which means a more feasible embedding on smart phones. Moreover, we improved the accuracy of heart rate estimation in comparison with previous research. Since the measurement of low heart rate is relatively stable and accurate, in the future we will focus on improving the performance of the algorithm on people with a high heart rate. We believe that this approach will make the measuring of heart rate from facial video recording applicable to a wider range of areas, such as sports.

References

[1] John G Webster, *Design of pulse oximeters*, CRC Press, 2002.

[2] C 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*, vol. 59, no. 2, pp. 303–306, 2012.

[3] Wim Verkruysse, Lars O Svaasand, and J Stuart Nelson, "Remote plethysmographic imaging using ambient light", *Optics Express*, vol. 16, no. 26, pp. 21434–21445, 2008.

[4] Ming-Zher Poh, Daniel J McDuff, and Rosalind W Picard, "Advancements in noncontact, multiparameter physiological measurements using a webcam", *IEEE Transactions on Biomedical Engineering*, vol. 58, no. 1, pp. 7–11, 2011.

[5] 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*, vol. 49, no. 2, pp. 172–175, 2002.

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

[7] Sungjun Kwon, Hyunseok Kim, and Kwang Suk Park, "Validation of heart rate extraction using video imaging on a built-in camera system of a smartphone", in *Engineering in Medicine and Biology Society (EMBC), 2012 Annual International Conference of the IEEE*. IEEE, 2012, pp. 2174–2177.

[8] Nikolai Grov Roald, "Estimation of vital signs from ambient-light non-contact photoplethysmography", 2013.

[9] Hao-Yu Wu, Michael Rubinstein, Eugene Shih, John Guttag, Frédo Durand, and William Freeman, "Eulerian video magnification for revealing subtle changes in the world", *ACM Transactions on Graphics*, vol. 31, no. 4, pp. 65, 2012.

[10] Rudolph Emil Kalman, "A new approach to linear filtering and prediction problems", *Journal of Basic Engineering*, vol. 82, no. 1, pp. 35–45, 1960.

[11] Greg Welch and Gary Bishop, "An introduction to the Kalman filter", 1995.

[12] William H Press and George B Rybicki, "Fast algorithm for spectral analysis of unevenly sampled data", *The Astrophysical Journal*, vol. 338, pp. 277–280, 1989.

[13] William H Press, Saul A Teukolsky, William T Vetterling, and Brian P Flannery, "Numerical recipes in Fortran 77: The art of scientific computing", 1992.

[14] "Face detector", `http://developer.android.com/reference/android/media/FaceDetector.html`.