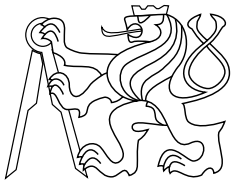
CENTER FOR
MACHINE PERCEPTION



CZECH TECHNICAL
UNIVERSITY IN PRAGUE

BACHELOR THESIS

ISSN 1213-2365

Heart Beat Rate Estimation from Video

Pavel Černý

cerny.pav@gmail.com

CTU–CMP–2014–07

May 22, 2014

Available at
ftp://cmp.felk.cvut.cz/pub/cmp/articles/franc/Cerny-TR-2014-07.pdf

Thesis Advisor: Vojtěch Franc

The author was supported by the Grant Agency of the Czech Republic under Project P202/12/2071

Research Reports of CMP, Czech Technical University in Prague, No. 7, 2014

Published by

Center for Machine Perception, Department of Cybernetics
Faculty of Electrical Engineering, Czech Technical University
Technická 2, 166 27 Prague 6, Czech Republic
fax +420 2 2435 7385, phone +420 2 2435 7637, www: http://cmp.felk.cvut.cz

Czech Technical University in Prague
Faculty of Electrical Engineering

Department of Cybernetics

BACHELOR PROJECT ASSIGNMENT

Student: Pavel Č e r n ý

Study programme: Open Informatics

Specialisation: Computer and Information Science

Title of Bachelor Project: Heart Beat Rate Estimation from Facial Video

Guidelines:

Implement an algorithm for heart beat rate estimation from facial videos captured by a common camera working in a visible spectrum. The implemented method should run in real time on a common PC, it should be invariant against face movements and it should work indoor under standard lighting conditions. Create a statistically representative database of videos along with ground truth annotation of contained people measure by a contact device (e.g. a sport tester). Use the created database to estimate precision of the implemented algorithm.

Bibliography/Sources:

[1] M.Poh et al.: Non-contact, automated cardiac pulse measurements using video imaging and blind source separation. Optics Express. 18(10):10762010774, 2010.
[2] Guha Balakrishnan, Fredo Durand, John Guttag: Detecting Pulse from Head Motions in Video. The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2013, pp. 3430-3437.

Bachelor Project Supervisor: Ing. Vojtěch Franc, Ph.D.

Valid until: the end of the summer semester of academic year 2014/2015

L.S.

doc. Dr. Ing. Jan Kybic
Head of Department

prof. Ing. Pavel Ripka, CSc.
Dean

Prague, January 10, 2014

České vysoké učení technické v Praze
Fakulta elektrotechnická

Katedra kybernetiky

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

Student: Pavel Č e r n ý

Studijní program: Otevřená informatika (bakalářský)

Obor: Informatika a počítačové vědy

Název tématu: Odhad tepové frekvence srdce z videa tváře

Pokyny pro vypracování:

Implementujte algoritmus pro odhad tepové frekvence lidí na základě analýzy videa jejich tváře snímaného běžnou kamerou ve viditelném spektru. Navrhněte algoritmus tak, aby odhadoval tepovou frekvenci na běžném PC v reálném čase. Algoritmus by měl být invariantní vůči pohybům a otočením tváře, a měl by fungovat v běžně osvětlené místnosti. Vytvořte statisticky reprezentativní databázi videí, u kterých bude tepová frekvence lidí změřena pomocí dotykového snímače. Použijte vytvořenou databázi k odhadu přesnosti implemetovaného algoritmu.

Seznam odborné literatury:

[1] M.Poh et al.: Non-contact, automated cardiac pulse measurements using video imaging and blind source separation. Optics Express. 18(10):10762010774, 2010.
[2] Guha Balakrishnan, Fredo Durand, John Guttag: Detecting Pulse from Head Motions in Video. The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2013, pp. 3430-3437.

Vedoucí bakalářské práce: Ing. Vojtěch Franc, Ph.D.

Platnost zadání: do konce letního semestru 2014/2015

L.S.

doc. Dr. Ing. Jan Kybic
vedoucí katedry

prof. Ing. Pavel Ripka, CSc.
děkan

V Praze dne 10. 1. 2014

Author's declaration

I declare that I have developed the presented work independently and that I have listed all information sources used in accordance with the Methodical guidelines on maintaining ethical principles during the preparation of higher education theses.

Prohlášení autora práce

Prohlašuji, že jsem předloženou práci vypracoval samostatně a že jsem uvedl veškeré použité informační zdroje v souladu s Metodickým pokynem o dodržování etických principů při přípravě vysokoškolských závěrečných prací.

V Praze dne ....

Podpis autora práce

Abstract

The thesis describes an implementation of a method estimating the heart beat rate based on an analysis of the color changes measured in a video of a human face. We use a 3D facial landmark tracker that allows to extract a stable region on a face needed to acquire a robust measurement of the color changes. A simple model is proposed to approximate a linear projection of the color signal by sinusoid superimposed on a line. The unknown parameters of the model are estimated by the least squares method which has a closed form solution for fixed heart beat rate. The proposed method is implemented partially in C++ and Matlab and it runs real time on a common PC. Another contribution of the thesis is a database of 43 videos containing 8 human subjects annotated with their heart beat rate measured by a precise contact method. The proposed method evaluated on the benchmark provides estimate of the heart rate with average deviation 5.25 BPM from the ground truth annotation.

Abstrakt

Práce popisuje implementaci metody, odhadující tepovou frekvenci srdce ve videu s lidským obličejem, na principu analyzování barevných změn. Používáme 3D tracker významných bodů tváře k vyjmutí stabilní oblasti z obličeje, což je nezbytné pro získání robusního měření barevných změn. Je navržen jednoduchý model pro aproximaci lineární projekce barevného signálu pomocí sinusoidy posazené na přímku. Neznámé parametry modelu jsou určeny pomocí metody nejmenších čtverců, ta má pro fixní tepovou frekvenci analytické řešení. Navrhovaná metoda je implemetovaná částečně v C++ a v Matlabu a běží na běžném PC v reálném čase. Dalším přínosem práce je databáze 43 videí, obsahujících 8 osob, anotovaných jejich tepovou frekvencí, změřenou pomocí přesné kontaktní metody. Navrhovaná metoda, spuštěna na benchmark testu, odhaduje tep s průměrnou odchylkou od anotace 5.25 tepů za minutu.

Heart Beat Rate Estimation from Video

Pavel Černý

May 22, 2014

Acknowledgement

I would like to thank my thesis supervisor Ing. Vojtěch Franc Ph.D. for guidance, support and patience.

Contents

1 Introduction **3**

2 Proposed Method **7**
2.1 Description of the Process 7
2.1.1 Video Recording 7
2.1.2 Tracker of Facial Landmarks 9
2.1.3 Determining the ROI 9
2.1.4 Converting The Pixel Values to the Low Dimensional Signal 10
2.1.5 Sampling Rate Increasing 11
2.1.6 Data Filtering 12
2.1.7 Model Fitting to the Signal and the HR Estimating 12
2.1.8 Smoothing of the Estimates 13
2.2 Model of the Low Dimensional Signal 13
2.3 Fitness Function 17

3 Experiments **18**
3.1 Database with Annotated Grand Truth 18
3.2 Benchmarks of Our Implementation 20
3.3 Influence of the Video Frame Rate on the Estimating 21
3.4 Real Time Implementation 23

4 Conclusions **25**

Bibliography **26**

1 Introduction

The non-contact measuring of the heart rate is a comfortable method in comparison with conventional contact methods. An electrocardiograph (ECG) is an example of the traditional device for heart rate measuring. The ECG is an expensive device which requires to connect several electrodes to the body. There are some other cheaper and more proliferated devices such a sport-tester with a connected chest-band. However they require more preparations before the measuring compared to a contactless solution. These constraints prevent contact methods from their comfortable everyday use. The contactless methods are unobstructive and cheap, but they estimate the heart rate (HR) typically with a lower accuracy.

Non-contact measuring can be applied in many situations. It can be used for a long time monitoring of hospital-based patients, a monitoring of the activity during a sleep, a personal home healthcare and many other non-medical scenarios. Examples of these non-medical usages can be sportsmen in a gym or some applications computing calories burned in activities such as video-gaming.

There are two main video based methods for the non-contact measuring at the present time. The first one is based on monitoring small movements of the body caused by reaction to the waves of blood pushed from the heart according to the third Newton's law "For every action, there is an equal and opposite reaction". The second one is based on small changes in skin color caused by the blood flow in tissues. The reflection capability of a skin with capillaries supplied with blood differs from the reflection capability of a skin containing less blood. The blood volume is changing in correspondence with the HR. The observing and analysing of such signal is called a photoplethysmographfy (PPG) [3]. The PPG waveforms could be obtained in a regular red-green-blue (RGB) color spectrum using the ambient daylight as an illumination.

The best way to extract the PPG signal is monitoring of a large area of the skin without shades on it. However finding such an area is a challenging problem. There are such areas of the skin on the body but they are mostly covered by clothing. We also need to track the selected part of the skin. These factors are the reason why we use the facial area of the head. This part of the skin is commonly visible, uncovered and also stays still, because it is natural for human to keep the head in one position and therefore this area keeps the same illumination over time.

In this thesis we propose a new method for the HR estimation based on measuring the PPG signal in the video of face. We use a tracker capable of tracking distinctive points in the face such as the eyes, the mouth and the nose to track the same selected region in

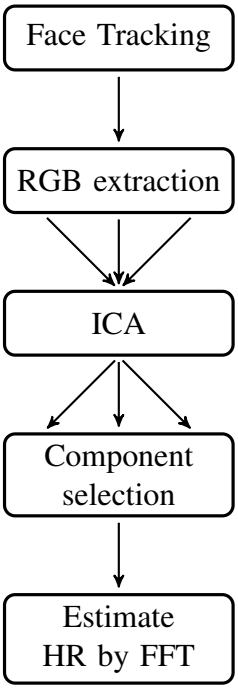


Figure 1.1 Procedure used by Poh et al

a video with frontally oriented head. It allows us to determine always the same region of interest (ROI) from which the color information is extracted. Our algorithm describes the color signal acquired from the ROI as a simple sinusoid model. The model parameters are estimated by the least square (LS) method. The LS problem has a closed form solution for a fixed HR. We compute the LS problem for each admissible HR and select the one with the lowest LS error.

We implement the proposed method in MATLAB [8] with some parts written in C++ language. The implementation runs in real time on a standard PC. Provided that the person is still and looking forward to the camera our solution has nearly the same accuracy as a contact sport-tester device with a chest-band.

Relation to the State of the Art

There have been many papers on this topic since the idea to use the PPG signal from face was first published in the article [12] by Poh at al. The method described in [12] demonstrates that a common web-camera working in RGB can be used for obtaining the PPG signal sufficient to estimate the HR.

Following works use a process similar to [12] (see figure 1.1). They use the same OpenCV [2] face detector in each frame to locate the facial area. Some experiments have

Figure 1.2 1.2a ROI suggested by the work of Lewandowska, 1.2b counterexample from our database, why not to use such a specific ROI

been done with intention to improve the detection of the ROI in order to prevent suffering from the noise caused by drifting of the detected ROI. However using such a specific ROI can be incompatible with the appearance of some people and causing total invalidity of the extracted data. The figure 1.2a shows the ROI proposed by Lewandowska [7] and the example from our video database where such ROI is covered by hair and thus not suitable for PPG analysis.

A lot of works use independent component analysis (ICA) [6] or principal component analysis (PCA). These methods try to decorrelate the 3D color signal assuming that the PPG will pop up. However both methods are working blindly without knowing anything about the properties of the PPG signal. These methods just decorrelate the three input signals. They suppose that the output is the PPG signal, but it is difficult to decide which component contains the information we are looking for. The Poh et al use the second component obtained by the ICA every time. The work [9] proves that there could be some useful information in all of the three ICA components. The last step in estimating the HR frequency is selecting the strongest harmonic signal using the Fast Fourier Transform (FFT).

In our work we use a principle similar to the one demonstrated in the thesis of Plesek [11]. A sinusoid curve

$$model(t) = \sin(\omega \cdot t + \phi) + a_1 \cdot t + a_0$$

is compared to a linear projection of the measured RGB signal.

Plesek implemented the HR estimator on an iPad tablet. In his thesis he uses exhausted search for the best HR ω and the phase shift ϕ, the rest of the parameters are computed by the LS method. In this thesis we enhance the method used by Plesek with two things.

First: we propose a method using the LS to estimate the parameters Plesek does and also the phase shift ϕ and coefficients of a linear projection of the signal. It reduces the search only to the variable ω and allows better extraction of the PPG signal by different illumination conditions.

Second: we use a 3D Facial Landmarks Tracker [5] to select the ROI. This method suffers significantly less from the drifting of the ROI than running a common face detector on every frame.

Our method brings an important speed up essential for the real-time implementation. Plesek was forced to implement some reductions in the computational procedure to achieve smooth running on a low-performance mobile device as the iPad is.

The next possibility for estimating the HR is monitoring the movements of the head or the whole body. Balakrishnan et al in their work [4] use the Kanade-Lucas-Tomasi Feature Tracker (KLT) [13] to track the position of feature points on the face and estimate the movement of the whole head from their position. The advantage of this motion-based technique is that it does not need the clean and direct view on a face. It can even be used for a person wearing a mask.

Some works focus on computing the heart rate variability (HRV) which is important for diagnosing some diseases e.g. cardiac arrhythmia. [4] [10]

2 Proposed Method

2.1 Description of the Process

The process of estimating the HR from facial video can be split into two main phases. In the first phase, video processing, a video of a face is transformed into a low dimensional signal which contains the PPG information and suppresses the other irrelevant information such as signal changes because of movement of the subject, changing lightning conditions etc. In the second phase the low dimensional signal is used to find the best estimation of the HR of the tested person. These two phases are shown in the figures 2.1 and 2.2.

1. Video Processing
   a) video recording
   b) tracker of the facial landmarks
   c) determining the ROI
   d) converting the pixel values to the low dimensional signal
2. Low dimensional signal processing
   a) sampling rate increasing
   b) data filtering
   c) model fitting to the signal
   d) HR estimating
   e) smoothing of the estimates

2.1.1 Video Recording

The program processes the video with a human face. It can be streamed from a camera or read from a file.

The tested person on the video should be looking in the direction of the camera. The movement of the observed subject should be reduced to minimum. Stronger moves can cause failures of the tracker of the face and also remarkable changes in the illumination conditions thus making the heart rate estimation less precise. In our sessions we used an ordinary web-camera with resolution 640×480 px.

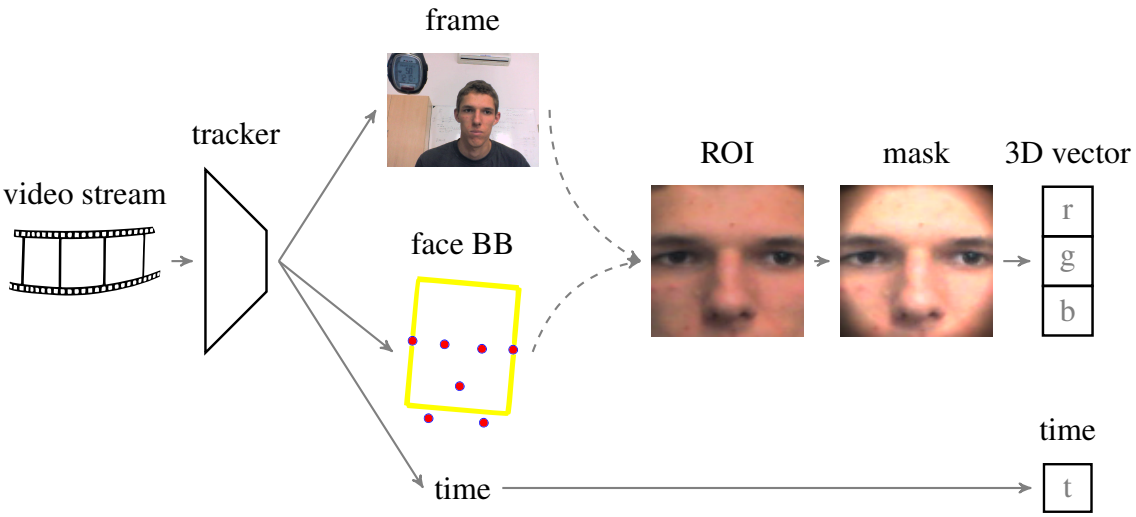


Figure 2.1 Video Processing

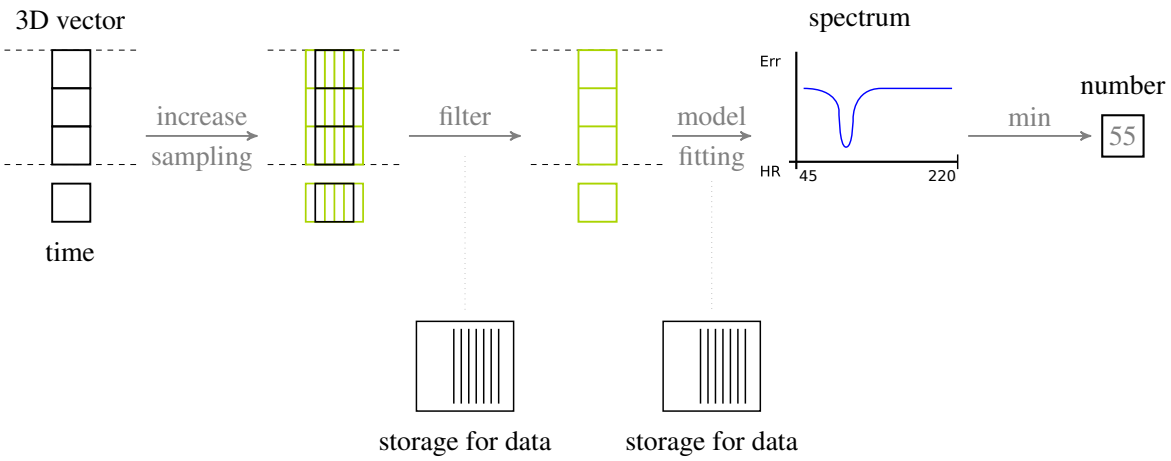
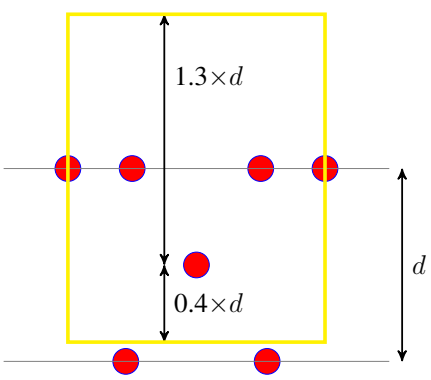
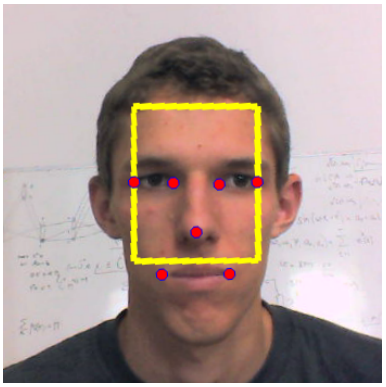


Figure 2.2 Low dimensional signal processing



(a) selecting ROI from landmarks



(b) example of ROI on face

Figure 2.3 ROI selection

2.1.2 Tracker of Facial Landmarks

The video-stream is directed into the tracker. We use a recently published 3D Landmark tracker [5] capable to track 7 distinctive points on a face, namely: the outer and the inner corner of both the eyes, the nose and the left and the right corner of the mouth. We call these points the landmarks. The tracker is also able to estimate the position of the head of the tracked person in 3D, but this information is not used in the present version of the HR estimator.

2.1.3 Determining the ROI

The ROI is computed from positions of the facial landmarks returned by the tracker. The ROI is defined to cover mainly the skin region above the mouth to allow speaking of observed person with obtaining as low amount of signal noise from mouth movement as possible.

The ROI is a rectangular area and is computed as follows: First, the position of the eyes is approximated by a straight line in the sense of minimising the sum of the square distance of the landmarks from the line. A mean distance d from the distances between the eye line and the positions of the left and right mouth-landmarks is computed.

The upper and the lower edge of the ROI are defined by parallel lines with the eye-line in the distance of $1.3 \times d$ above the eye-line and $0.4 \times d$ below the eye-line. The left and the right edges of the ROI are defined by a vertical line passing through the position of the outer landmarks of the left and the right eye. Figure 2.3a shows the definition of the ROI.

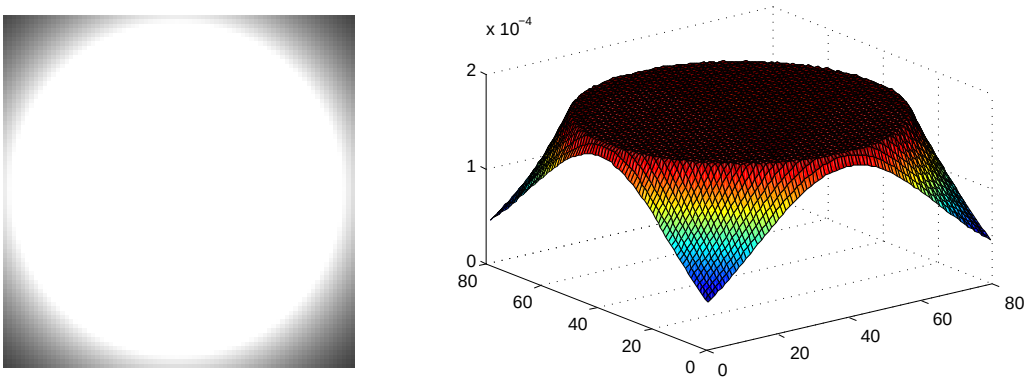


Figure 2.4 Mask matrix in 2D and 3D visualisation

2.1.4 Converting The Pixel Values to the Low Dimensional Signal

The image contained in the ROI is processed next. The ROI is transformed into an image 80×80 px.

The pixels close to the edges of the ROI can fluctuate as a result of inaccuracy of the tracker and the movement of the head. E.g. when the face is being rotated from left to right, the area on the right side of the face disappear and the left one appears. On the other hand the area near the center of the ROI is supposed to be always contained inside the ROI.

To suppress the influence of the changes in the edge pixels, we apply a mask $\mathbf{M}'$ to the transformed ROI. We use the width W and the height H of the ROI and define parameters

$$[\sigma_w,\, \sigma_h] = [W,\, H] \cdot 0.45$$
$$[\mu_w,\, \mu_h] = [W,\, H] \cdot 0.5$$

and compute a mask $\mathbf{M}$ (respectively $\mathbf{M}'$) which is shown in figure 2.4 and is computed as

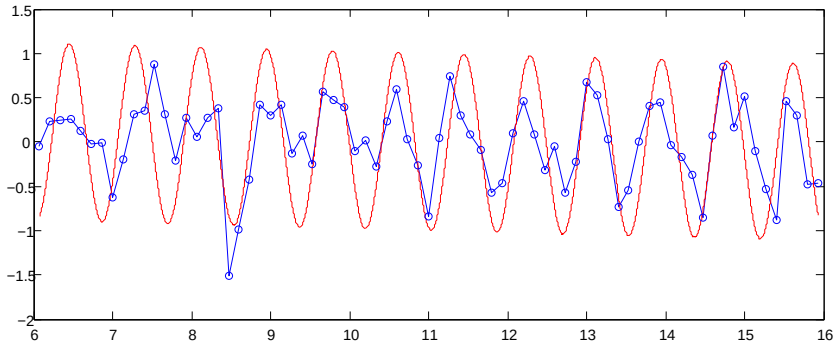
$$\mathbf{M}_{i,j} = \min(1, 3 \cdot \exp -(\frac{i-\mu_h{}^2}{\sigma_w{}^2} + \frac{j-\mu_h{}^2}{\sigma_h{}^2})) \tag{2.1}$$

$$\mathbf{M'}_{i,j} = \frac{\mathbf{M}_{i,j}}{\sum_{i=1}^{W}\sum_{j=1}^{H}\mathbf{M}_{i,j}} \tag{2.2}$$

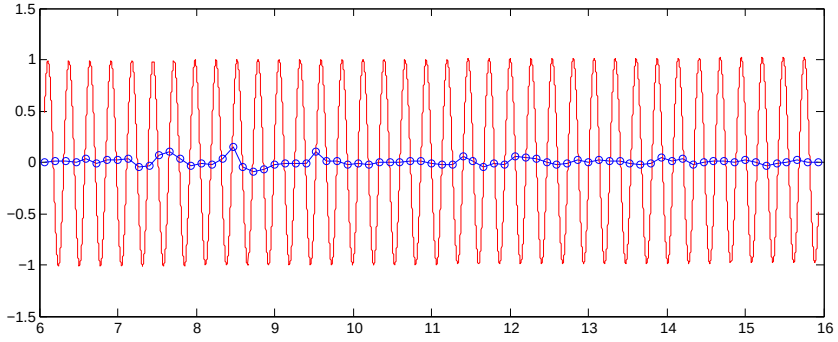
The shape of the mask $\mathbf{M}$ and the parameters σ_w, σ_h, μ_w, μ_h were determined experimentally.

The weighted average of each color in the sub-image I_k defined by the ROI in the measurement k is computed as

$$\begin{bmatrix} \overline{r}_k \\ \overline{g}_k \\ \overline{b}_k \end{bmatrix} = \sum_{i=1}^{W}\sum_{j=1}^{H} \mathbf{M}_{i,j} \cdot \mathbf{I}_k(i,j) \tag{2.3}$$



(a) HR $= 75$bpm, $\epsilon = 22.15$



(b) HR $= 225$bpm, $\epsilon = 6.37$

Figure 2.5 Example of an over-fit of model on the signal. Dependence of the signal at time (s). The annotated HR $= 75$bpm, fps $= 7.5$ and the sum of the square error difference is ϵ

This way we obtain the weighted average giving the pixels on the edges lower weight than to the pixels in the center. The result of this stage is a vector $(\mathbf{c}_k,\ t_k) \in \mathbb{R}^4$ where

$$\mathbf{c}_k = \begin{bmatrix} \overline{r}_k \\ \overline{g}_k \\ \overline{b}_k \end{bmatrix}$$

and t_k is a timestamp corresponding to the time of capturing of the current frame.

2.1.5 Sampling Rate Increasing

We propose computing a sum of the square error of the difference between the estimated projection of the signal and the model. The computation is done only for discrete time intervals which can lead to over-fit (see figure 2.5).

In order to prevent the over-fitting the signal $\mathbf{c_k}$ is interpolated. New values of the signal $\mathbf{c_{k+r}}$ are estimated to achieve sampling frequency

$$f_s = 40 \text{ fps}.$$

The timestamp t_k of the current measurement is compared to the timestamp t_{k-1} of the previous measurement. If the distance between them is higher than the sampling period of the desired frequency

$$t_k - t_{k-1} > \frac{1}{f_s}$$

we compute the minimum number n of equidistant timestamps between t_k and t_{k-1} and using linear interpolation of the current $\mathbf{c_k}$ and previous measurement $\mathbf{c_{k-1}}$ to obtain the new values for $i = 1, 2, .., n$ as follows

$$t_{k+\frac{i}{n+1}} = t_{k-1} + i \cdot \frac{t_k - t_{k-1}}{n+1} \tag{2.4}$$

$$\tilde{\mathbf{c}}_{k+\frac{i}{n+1}} = \mathbf{c}_{k-1} + \frac{[\mathbf{c}_k - \mathbf{c}_{k-1}] \cdot [t_{k+\frac{i}{n+1}} - t_{k-1}]}{[t_k - t_{k-1}]} \tag{2.5}$$

2.1.6 Data Filtering

The signal with the enhanced sampling frequency $(\mathbf{c}_j, t_j) \in \mathbb{R}^4$ is preprocessed to $(\mathbf{x}_j, t_j) \in \mathbb{R}^4$ by a hi-pass filter. The filter is implemented by subtracting the moving average computed on window with N frames.

$$\mathbf{x}_j = \mathbf{c}_j - \frac{\sum_{i=0}^{N-1} \mathbf{c}_{j-i}}{N} \tag{2.6}$$

The length $N = \lfloor 7/15 \cdot f_s \rfloor$ was determined experimentally. The filter removes the noise with low frequencies and it translates the signal to oscillate around zero instead of the original value. The comparison of an input and the output of the filtering step is in the figure 2.6.

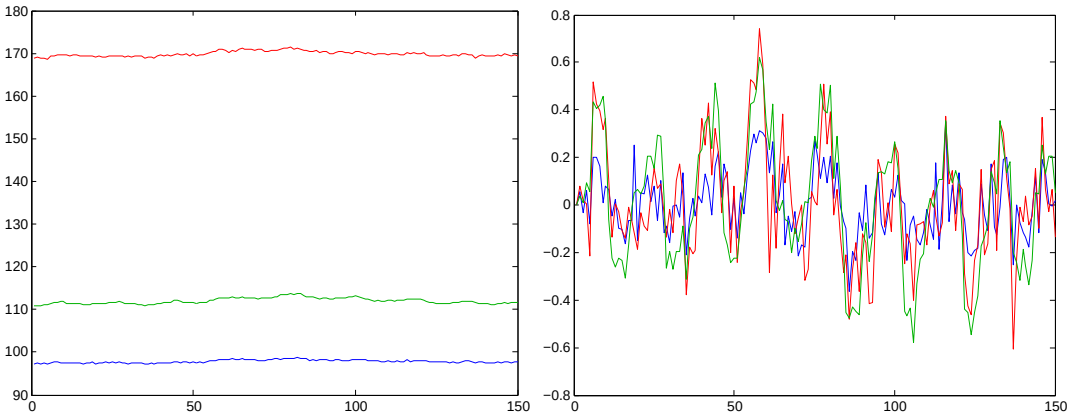
2.1.7 Model Fitting to the Signal and the HR Estimating

Our model assumes, that there exits a linear combination $s(j)$ of the three components of the signal $\mathbf{x}_j \in \mathbb{R}^3$, computed by the previous steps, such that the signal $\mathbf{x}_j$ projected by the linear combination with coefficients $\mathbf{w}_f$ for the given window f

$$s(j) = \mathbf{w}_f^T \cdot \mathbf{x}_j \tag{2.7}$$

contains a lot of the PPG information and therefore can be represented as a sinusoid curve superimposed on a line

$$\hat{s}(j) = \sin(\omega \cdot j + \phi) + a_1 \cdot j + a_0 \tag{2.8}$$



(a) raw signal - $\mathbf{x}(t)$ **(b)** hi-passed signal - $\mathbf{x}(t)$

Figure 2.6 Comparsion of original and hi-passed signal

The unknown parameters $(\mathbf{w}_f, \omega, \phi, a_1, a_0)$ are estimated from a time-window of a specified length (e.g. 10s) by an algorithm proposed in section 2.2 and 2.3. Unlike the continuous parameters $(\mathbf{w}_f, \omega, \phi, a_1, a_0)$, the angular speed ω is selected from a set of admissible discrete values

$$\omega = \frac{2\pi \cdot f}{60}, \; f \in \{45, 46, .., 220 \text{ bmp}\} \tag{2.9}$$

2.1.8 Smoothing of the Estimates

The sequence of the obtained estimates of the HR can fluctuate around the correct value. To smooth the estimation we apply the exponential moving average filter with $\alpha \in (0, 1)$

$$\begin{aligned} f'(t_0) &= f(t_0) \\ f'(t_i) &= (1 - \alpha) \cdot f'(t_{i-1}) + \alpha \cdot f(t_i) \end{aligned}$$

The value of $f'(t)$ is the final output of the program. The value of α was experimentally set to 0.3.

2.2 Model of the Low Dimensional Signal

The estimation is based on the algorithm which takes all the measurement vectors $(\mathbf{x}, t) \in \mathbb{R}^4$ for the given time-frame and tries to find an optimal projection to $(s, t) \in \mathbb{R}^2$, where

$$\mathbf{x} = \begin{bmatrix} r \\ g \\ b \end{bmatrix}$$

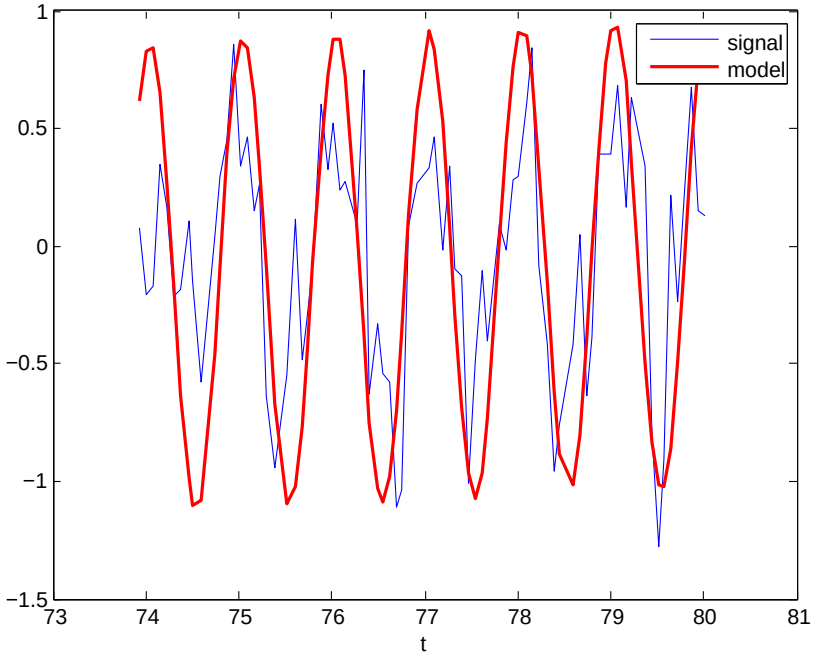


Figure 2.7 The projection of signal $s(t)$ with the estimated model $\hat{s}(t)$

are the values of the RGB color channels and s is the projected signal best matching a sinusoid.

The input of the algorithm is a matrix $X \in \mathbb{R}^{3\times n}$ with n measurements $\mathbf{x}$ of the RGB values and vector $t \in \mathbb{R}^n$ of the corresponding time-stamps.

$$X = \begin{bmatrix} \mathbf{x}_1 & \mathbf{x}_2 & \mathbf{x}_3 & \dots & \mathbf{x}_n \end{bmatrix}$$
$$t^T = \begin{bmatrix} t_1 & t_2 & t_3 & \dots & t_n \end{bmatrix}$$

We want to find a projection of the heart signal from the color channels. We have to find the coefficients of weights $\mathbf{w}$ for each color channel, $\mathbf{w} = \begin{bmatrix} w_r & w_g & w_b \end{bmatrix}$. We can express the projected curve of the HR signal as

$$s(t) = \mathbf{x}_t^T \cdot \mathbf{w} \tag{2.10}$$

and our estimated model as a sinusoid superimposed on a line

$$\hat{s}(t) = \sin(\omega \cdot t + \phi) + a_1 \cdot t + a_0 \tag{2.11}$$

where ω is a phase, ϕ is a phase shift, a_1 is a coefficient of linear translation and a_0 is a constant.

From these two expressions we can compute the error as difference of these two curves:

$$\epsilon(t) = s(t) - \hat{s}(t) \tag{2.12}$$
$$\epsilon(t) = \mathbf{x}_t^T \cdot \mathbf{w} - a_0 - a_1 \cdot t - \sin(\omega \cdot t + \phi) \tag{2.13}$$

The value of estimated HR frequency f is bounded to the value of optimal ω

$$\omega = \frac{2\pi \cdot f}{60} \tag{2.14}$$

We can fix the parameter ω and compute the rest of the parameters $(\mathbf{w}_f, \phi, a_1, a_0)$ analytically.

Let us define an error function $F(\mathbf{w}, a_0, a_1, \phi|\omega)$ representing the sum of the square error between the model and the projected signal in the time-window T

$$F(\mathbf{w}, a_0, a_1, \phi|\omega) = \sum_{t \in T} \epsilon^2(t) \tag{2.15}$$

For a fixed ω we look for the parameters $(\mathbf{w}_\omega^*, a_{0\omega}^*, a_{1\omega}^*, \phi_\omega^*)$ by minimizing the value of $F(\mathbf{w}, a_0, a_1, \phi|\omega)$

$$\begin{aligned}
(\mathbf{w}_\omega^*, a_{0\omega}^*, a_{1\omega}^*, \phi_\omega^*) &= \underset{\mathbf{w}, a_0, a_1, \phi}{\arg\min} \sum_{t \in T} \epsilon^2(t) \\
&= \underset{\mathbf{w}, a_0, a_1, \phi}{\arg\min} \sum_{t \in T} [\mathbf{x}_t^T \cdot \mathbf{w} - a_0 - a_1 \cdot t - \sin(\omega \cdot t + \phi)]^2 \\
&= \underset{\mathbf{w}, a_0, a_1, \phi}{\arg\min} \sum_{t \in T} [\mathbf{x}_t^T \cdot \mathbf{w} - a_0 - a_1 \cdot t - \sin(\omega \cdot t) \cdot \underbrace{\cos\phi}_{\text{A}} - \cos(\omega \cdot t) \cdot \underbrace{\sin\phi}_{\text{B}}]^2 \\
&= \underset{\substack{\mathbf{w}, a_0, a_1 \\ \text{s. t. } A^2+B^2=1}}{\arg\min} \sum_{t \in T} [\mathbf{x}_t^T \cdot \mathbf{w} - a_0 - a_1 \cdot t - \sin(\omega \cdot t) \cdot A - \cos(\omega \cdot t) \cdot B]^2
\end{aligned} \tag{2.16}$$

The error function 2.16 for all input data $\mathbf{X}$ and $\mathbf{t}$ can be rewritten into a matrix form:

$$\begin{bmatrix} \mathbf{x}_1^T & -1 & -t_1 \\ \mathbf{x}_2^T & -1 & -t_2 \\ \vdots & \vdots & \vdots \\ \mathbf{x}_n^T & -1 & -t_n \end{bmatrix} \cdot \begin{bmatrix} \mathbf{w} \\ a_0 \\ a_1 \end{bmatrix} - \begin{bmatrix} \sin\omega t_1 & \cos\omega t_1 \\ \sin\omega t_2 & \cos\omega t_2 \\ \vdots & \vdots \\ \sin\omega t_n & \cos\omega t_n \end{bmatrix} \cdot \begin{bmatrix} A \\ B \end{bmatrix} = \begin{bmatrix} \epsilon(t_1) \\ \epsilon(t_2) \\ \vdots \\ \epsilon(t_n) \end{bmatrix}$$
$$\mathbf{P} \cdot \mathbf{u} - \mathbf{Q} \cdot \mathbf{v} = \epsilon(t) \tag{2.17}$$

The optimal values are then

$$(\mathbf{w}^*_\omega, a^*_{0\omega}, a^*_{1\omega}, \phi^*_\omega) = \underset{\substack{\mathbf{u},\mathbf{v} \\ \text{s. t. } \|\mathbf{v}\|^2=1}}{\arg\min} \|\mathbf{P}\cdot\mathbf{u} - \mathbf{Q}\cdot\mathbf{v}\|^2 \tag{2.18}$$

We use the Lagrange multiplier method to find the optimal parameters

$$\begin{aligned} L(u,v,\lambda) &= \|\mathbf{P}\cdot\mathbf{u} - \mathbf{Q}\cdot\mathbf{v}\|^2 - \lambda\cdot(\|\mathbf{v}\|^2 - 1) \\ &= (\mathbf{P}\cdot\mathbf{u} - \mathbf{Q}\cdot\mathbf{v})^T(\mathbf{P}\cdot\mathbf{u} - \mathbf{Q}\cdot\mathbf{v}) - \lambda\,(\mathbf{v}^T\mathbf{v} - 1) \\ &= \mathbf{u}^T\mathbf{P}^T\mathbf{P}\mathbf{u} - 2\mathbf{u}^T\mathbf{P}^T\mathbf{Q}\mathbf{v} + \mathbf{v}^T\mathbf{Q}^T\mathbf{Q}\mathbf{v} - \lambda\mathbf{v}^T\mathbf{v} + \lambda \end{aligned} \tag{2.19}$$

We derive the equation and set it equal to zero

$$\begin{aligned} \tfrac{\partial L}{\partial \mathbf{u}} &= & 2\mathbf{u}^T\mathbf{P}^T\mathbf{P} - 2\mathbf{v}^T\mathbf{Q}^T\mathbf{P} &= 0 \quad \Rightarrow \quad \mathbf{u}^T = \mathbf{v}^T\mathbf{Q}^T\mathbf{P}(\mathbf{P}^T\mathbf{P})^{-1} \\ \tfrac{\partial L}{\partial \mathbf{v}} &= & -2\mathbf{u}^T\mathbf{P}^T\mathbf{Q} + 2\mathbf{v}^T\mathbf{Q}^T\mathbf{Q} - 2\lambda\mathbf{v}^T &= 0 \\ & & -\mathbf{u}^T\mathbf{P}^T\mathbf{Q} + \mathbf{v}^T\mathbf{Q}^T\mathbf{Q} &= \lambda\mathbf{v}^T \\ & & \mathbf{v}^T\mathbf{Q}^T\mathbf{Q} - \mathbf{v}^T\mathbf{Q}^T\mathbf{P}(\mathbf{P}^T\mathbf{P})^{-1}\mathbf{P}^T\mathbf{Q} &= \mathbf{v}^T\lambda \qquad \text{// substituting } \mathbf{u}^T \\ & & \mathbf{v}^T(\mathbf{Q}^T\mathbf{Q} - \mathbf{Q}^T\mathbf{P}(\mathbf{P}^T\mathbf{P})^{-1}\mathbf{P}^T\mathbf{Q}) &= \mathbf{v}^T\lambda \end{aligned} \tag{2.20}$$

From the previous equation 2.20 we see that $\mathbf{v}$ is an eigenvector of matrix $(\mathbf{Q}^T\mathbf{Q} - \mathbf{Q}^T\mathbf{P}(\mathbf{P}^T\mathbf{P})^{-1}\mathbf{P}^T\mathbf{Q})$. We know that $\|\mathbf{v}^T\mathbf{v}\|^2 = 1$. We can multiply both sides of the equation 2.20 by vector $\mathbf{v}$:

$$\begin{aligned} \mathbf{v}^T\lambda\mathbf{v} &= \mathbf{v}^T(\mathbf{Q}^T\mathbf{Q} - \mathbf{Q}^T\mathbf{P}(\mathbf{P}^T\mathbf{P})^{-1}\mathbf{P}^T\mathbf{Q})\mathbf{v} \\ 1\cdot\lambda &= \mathbf{v}^T\mathbf{Q}^T\mathbf{Q}\mathbf{v} - \mathbf{v}^T\mathbf{Q}^T\mathbf{P}(\mathbf{P}^T\mathbf{P})^{-1}\mathbf{P}^T\mathbf{Q}\mathbf{v} \\ \lambda &= \mathbf{v}^T\mathbf{Q}^T\mathbf{Q}\mathbf{v} - \mathbf{u}^T\mathbf{P}^T\mathbf{Q}\mathbf{v} \\ \lambda &= \mathbf{v}^T\mathbf{Q}^T\mathbf{Q}\mathbf{v} - 2\mathbf{u}^T\mathbf{P}^T\mathbf{Q}\mathbf{v} + \mathbf{u}^T\mathbf{P}^T\mathbf{P}\mathbf{u} \qquad (2.21) \\ \lambda &= \|\mathbf{P}\cdot\mathbf{u} - \mathbf{Q}\cdot\mathbf{v}\|^2 \qquad (2.22) \end{aligned}$$

Note: we obtain 2.21 by adding the optimum condition $\frac{\partial L}{\partial \mathbf{u}} = 0$.

The equation 2.22 proves that the minimum of $\|\mathbf{P}\cdot\mathbf{u} - \mathbf{Q}\cdot\mathbf{v}\|^2$ is equal to the minimal eigenvalue λ^* of matrix $\mathbf{Q}^T\mathbf{Q} - \mathbf{Q}^T\mathbf{P}(\mathbf{P}^T\mathbf{P})^{-1}\mathbf{P}^T\mathbf{Q}$.

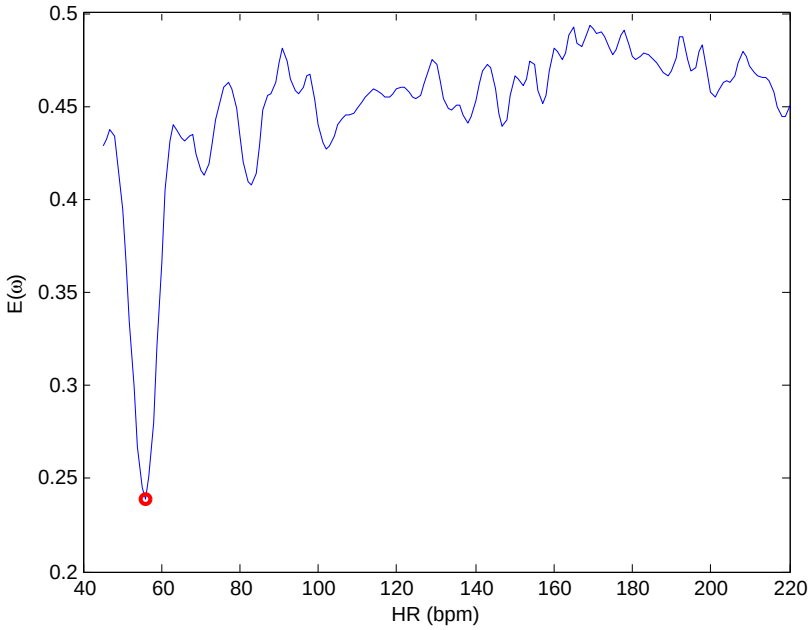


Figure 2.8 Fitness function with a peak detected at 56 bmp

Model Parameters

To estimate the parameters $(\mathbf{w}^*_\omega, a^*_{0\omega}, a^*_{1\omega}, \phi^*_\omega)$ we search for the eigenvector $\mathbf{v}$ with lowest eigenvalue λ^*:

1. Finding eigenvector $\mathbf{v}$ corresponding to the λ^* of matrix $\mathbf{Q}^T\mathbf{Q} - \mathbf{Q}^T\mathbf{P}(\mathbf{P}^T\mathbf{P})^{-1}\mathbf{P}^T\mathbf{Q}$.
2. Computing $[\mathbf{w}^T\, a_0\, a_1] = \mathbf{u}^T = \mathbf{v}^T\mathbf{Q}^T\mathbf{P}(\mathbf{P}^T\mathbf{P})^{-1}$.
3. Computing ϕ from the values in vector $\mathbf{v} = [\cos(\phi)\ sin(\phi)]$.

2.3 Fitness Function

In the previous section we demonstrated how we can obtain the optimal parameters $(\mathbf{w}^*_\omega, a^*_{0\omega}, a^*_{1\omega}, \phi^*_\omega)$ and compute the value of the error function $F(\mathbf{w}^*_\omega, a^*_{0\omega}, a^*_{1\omega}, \phi^*_\omega|\omega)$ for the given HR frequency ω.

We can construct a fitness function $E : \Omega \to \mathbb{R}$ mapping each ω in considered range of HR $\Omega \approx \{45, 46, .., 220 \text{ bmp }\}$ to the error of the best fit of the estimated model. The error of the fit is represented by the value of the error function $F(\mathbf{w}^*_\omega, a^*_{0\omega}, a^*_{1\omega}, \phi^*_\omega|\omega)$ for each ω.

We take the HR corresponding to the ω with the the best fit and declare it as the estimated value

$$\omega^* = \arg\min_{\omega} E(\omega) \tag{2.23}$$

Figure 2.8 shows an example of the fitness function E.

3 Experiments

3.1 Database with Annotated Grand Truth

We collected a video database to validate the precision of our proposed method. The videos were also used in the development stage and for tuning the parameters of the method. Therefore the results can be slightly positively biased.

Setup

The Database was recorded inside of an office with 8 participants of different age, one female, seven males. The scene was captured by two devices at the same time. We used a built-in laptop web-camera (Lenovo ThinkPad E420) and much better USB camera (Logitech QuickCam Pro 9000). The quality of the video was set to 640 $\times$ 480 px at 15 fps. The measuring of the HR was made by Polar RS300 sport-tester using watches and a chest band. The display of the watches was captured together with the face of the participant into the video. See figure 3.1 showing an example from the database.

We asked our participants to act in three different scenarios:

1. Sitting still near the camera. (cca 50 cm)
2. Sitting still far from the camera. (cca 1.5 m)
3. Reading a text aloud near the camera. (cca 50 cm)

The videos from these three scenarios were divided in three corresponding sets.

1. In the first set the head covers about $\frac{1}{8}$ of the screen.
2. In the second set the head covers about $\frac{1}{20}$ of the screen.
3. In the third set the head covers about $\frac{1}{6}$ of the screen, the participants has to read some random text aloud or talk to somebody else. They are allowed to move freely and laugh as natural by such for them unusual funny activity.

Annotation

The annotation ground truth of the HR is done by reading out each 5 seconds the numerical value from the display of the watches captured in the video. The values for each second for the video are computed by application of linear interpolation on the beginning and end of each five-second interval.

The database does not contain videos suffering from a permanent changing of the brightness caused by the auto-white-balance. Such videos containing obviously wrong graphical

Figure 3.1 Example frame from the video database. In the top-left corner is displayed the current HR measured by the Polar RS300 with a chest band

information were removed. The important informations about the database are summed up in the table 3.1.

Table 3.1 Database - specifications

resolution	640×480
frame rate	15
cam 1	built-in Lenovo E420
cam 2	Logitech QuickCam Pro 9000
codec	JPEG
container	AVI
number of person	8
length	27-34s
number of videos	43

number of videos by categories

sitting near		**sitting far**		**talking**	
18		15		10	
cam 1	**cam 2**	**cam 1**	**cam 2**	**cam 1**	**cam 2**
8	10	7	8	4	6

3.2 Benchmarks of Our Implementation

We ran a benchmark test on the whole video database. We estimate the precision of our method for each video subset (sitting near, sitting far, talking) independently as well as for all the videos at once. The benchmarking method computes estimates of the HR every 10th frame of the video. The obtained estimates are linearly interpolated and transformed to the estimates for every second. For each estimate we compute its absolute deviation from the ground truth. We report the average absolute deviation, the median value of abs. dev., the standard deviation (STD) of the abs. dev. and the maximal abs. dev. of all the estimations.

To obtain valid estimates of the HR, the benchmarking started after 10 seconds of the video such that the signal from the whole length of the time-window can be used. The parameters of the test are listed in the table 3.2. We made a statistics for the pure estimates and also for the values computed by the smoothing filter (see section 2.1.6).

Table 3.2 Summary info of the created benchmark

parameter	value
frame rate (fps)	15
sampling rate (sps)	40
hipass frame length (frames)	$7/15\times$sampling rate
estimation frame length (frames)	$10\times$frame rate
α	0.3

The results of the test are summed up in the table 3.3. By the abbreviation "dev" in the table we mean the absolute deviation. The graphic summary of the test is in the figure 3.2 showing the percentage of videos with average absolute deviation lower than the given value.

Table 3.3 Summary of the results of the benchmark

	statistics	**near**	**far**	**talking & near**	all
	no. of estimates	335	284	187	806
pure	average dev (bpm)	1.49	3.40	15.11	5.25
	median dev (bpm)	1.00	2.00	9.22	1.40
	STD of dev (bpm)	2.10	4.90	15.10	9.86
	maximal dev (bpm)	18.00	30.60	55.11	56.20
smoothed	average dev (bpm)	1.51	3.30	14.82	5.30
	median dev (bpm)	1.00	1.96	5.67	1.73
	STD of dev (bpm)	1.70	4.30	15.94	9.49
	maximal dev (bpm)	10.10	24.82	56.20	55.11

Smoothing filter decreases the maximal deviation and it also makes the estimates more stable as seen from the STD deviation. In the case there is a consequential failure in the

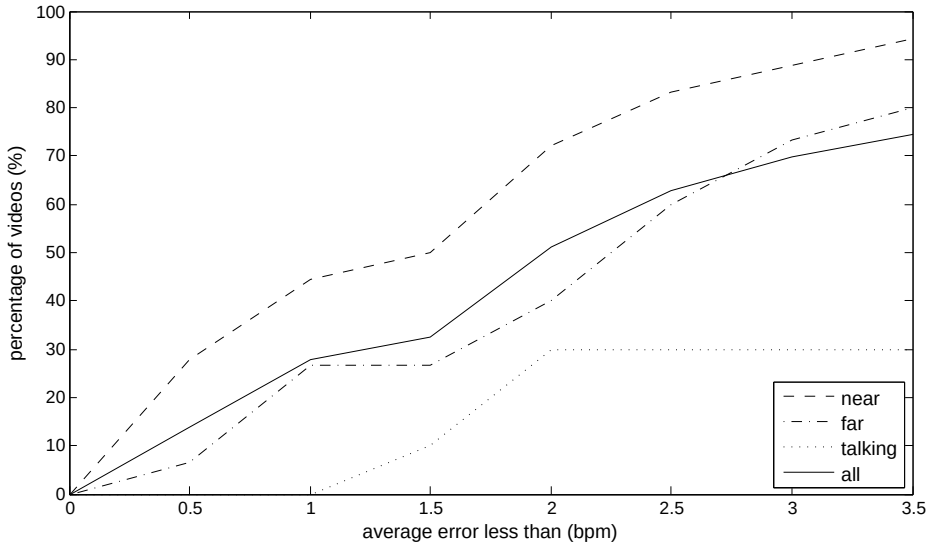


Figure 3.2 Benchmarks - the percentage of videos from the database with lower average error than the given value

estimation caused by some noisy frequency followed by a correct estimation the filter needs a long time for recovering. In the case the wrong estimation appears only rarely the filter suppress the influence of such wrong estimation.

The dependence on the distance from the camera has the expected influence on the quality of the estimating. When the measured subject is far the estimation is less precise, because the face is small and there is less information to work with.

The talking and movements of the subjects have a significant influence on the precision. The estimations are mostly not correct, average deviation is quite big.

3.3 Influence of the Video Frame Rate on the Estimating

The implementation is supposed to estimate the HR in real time when the frame rate can vary depending on the computer performance and load. We tested the dependence of the quality of estimation on the frame rate. We tested this aspect on the set of the database, where the people are not moving and theirs faces cover about $\frac{1}{6}$ of the screen. We removed some frames from the video to simulate lower fps. We removed every 3rd, 5th and 2nd frame to achieve 12, 10 and 7.5 fps from the original 15 fps. The comparison of the results is in table 3.4. The parameters (table 3.2) were the same as for the first benchmark (section 3.2). The statistics were done in the same manner as for first benchmark.

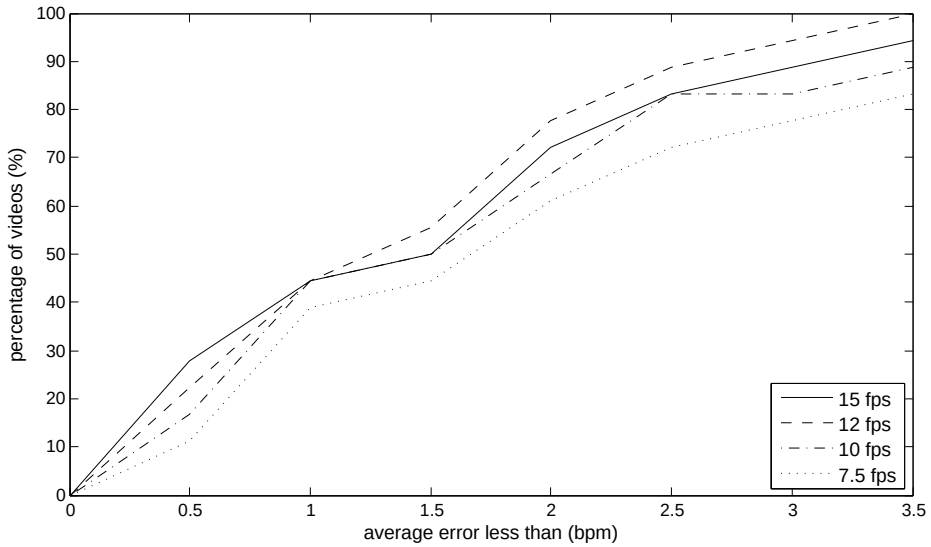


Figure 3.3 Influence of fps - the percentage of videos from the database with lower average error than the given value

Table 3.4 Influence of the fps of the video tested on the set "near"

	statistics	**15 fps**	**12 fps**	**10 fps**	**7.5 fps**
	no. of estimates	335	335	335	335
pure	average dev (bpm)	1.49	1.40	1.95	6.31
	median dev (bpm)	1.00	1.00	1.00	1.00
	STD of dev (bpm)	2.10	1.62	2.96	25.88
	maximal dev (bpm)	18.00	11	25.60	177.60
smoothed	average dev (bpm)	1.51	1.41	1.73	7.52
	median dev (bpm)	1.00	1.00	1.07	1.00
	STD of dev (bpm)	1.70	1.44	2.04	24.09
	maximal dev (bpm)	10.10	9.10	13.19	144.60

This test proved that the PPG signal can be extracted even from low frame rate as 7.5 fps is. However on such low frame rates the probability of the estimation of some over-fitted frequency is increased as shows the large maximal absolute deviation.

Using the increased sampling frequency described in section 2.1.5 improves preventing this error on the frame rates above 10 fps but for the lower values of frame rate it is not so powerful.

3.4 Real Time Implementation

We implemented a real time application in MATLAB [8]. The application uses the OpenCV [2] library for reading from a web-cam. Another used libraries are the 3D Landmark Tracker [5] with flandmark detector [14] and commercial version of EyeDea Face Detector [1] used for the initialization of the 3D Tracker. The user can select the source of the video by providing the path to the video and it's frame rate or the program uses a web-cam connected to the computer.

The applications shows all the important information in the graphical interface (GI) (see picture 3.4). The window of the GI shows the actual frame from the video with info boxes displayed on it. The seven tracked facial landmarks are highlighted by red circles. The ROI from which the RGB measurements are extracted is drawn by a yellow rectangle. There is displayed the projection of the signal with the estimated sinusoid-model, the actual estimated HR and the smoothed value enclosed in brackets (computed by the filter described in section 2.1.6).

There is displayed a color-graph showing the estimation fit for all possible HR frequencies. It uses the following color spectrum to represent the quality of fit: spectrum from red (worst fit) to dark blue (best fit). The column on the right side of the graph is the actual estimation, the columns more to the left are the older information. The rows of the graph corresponds to the HR frequencies. The white line drawn over the color-graph is showing the estimated values of the HR. The black line is showing the smoothed estimates by the filter.

The implementation run on a common laptop with dual core computational unit Intel® i3 @ 2.10GHz at about 12 fps. On a faster hardware we were able to achieve about 25 fps and the estimating was slightly better.

The user is recommended to keep in still position to obtain a really good and valid estimates. The quality of estimation can be assessed from the graph showing the signal projection and the estimated model and from the fit color-graph. A long blue stripe in many consecutive estimations shows the fit is correct. In case the color graph does not have any trend we can not be sure about the estimated value.

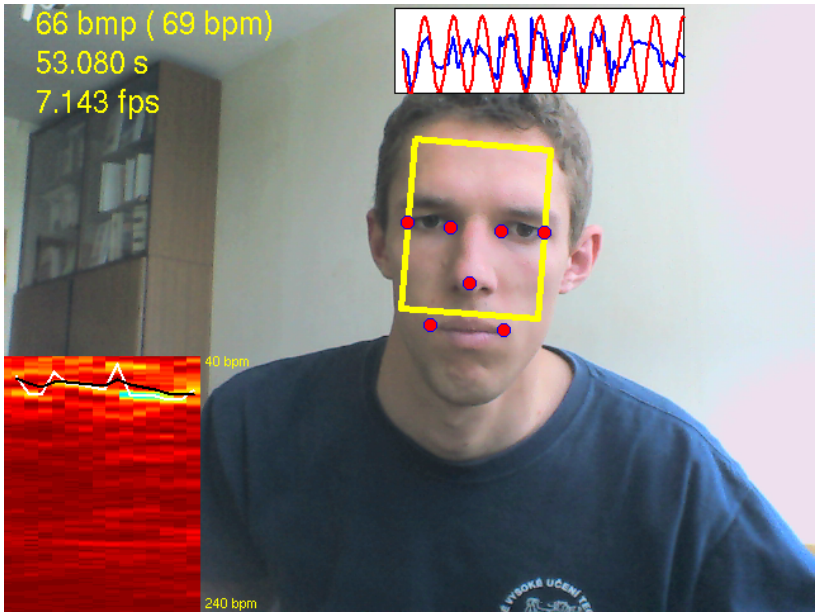


Figure 3.4 Graphical interface of the implementation showing the estimation (top-left), the model and the signal projection (top), history of the estimating(bottom-left) and the used ROI

4 Conclusions

In this thesis we have implemented an algorithm for estimation of the heart beat rate using the changes in skin color in a video captured by a standard web-camera. The algorithm uses a simple sinusoid curve to model a linear projection of the measured color signal. The model parameters are estimated by the Least Square method for which we provide an analytical solution. Our implemented solution runs real time on a common PC with the recording frequency 12 fps or better. The algorithm allows the extraction of the measured signal even when the measured subject slightly moves or talks during the recording. We have collected a database of 43 videos of 8 persons with various positions in front of a camera. The videos are annotated with the ground truth heart beat rate information obtained by a precise contact device. We ran the tests on our database and validated the ability of using the algorithm in real conditions including speaking persons far from the camera achieving an average deviation of 5.25 bpm from the ground truth.

On the other hand the robustness of our method against face movements and talking is not as good as we expected. There is a space for improving the selection of the region of interest. The method is built on the 3D tracker but uses only the 2D projections of the facial landmarks. The method uses the landmarks of the mouth, but they fluctuate due to speech. The future extensions of this software can also focus on better preprocessing of low dimensional signal. We suppose a better filter of large smooth changes in color caused by significant movements of the subject can be designed. Our implementation allows visual inspection of the quality of estimations but for the software it can be a challenging task requiring further study. Hence another step to be done is a reliable confidence measure of the estimate.

Bibliography

[1] *EyeFace SDK*. http://www.eyedea.cz. 23

[2] *The OpenCV Library*. http://opencv.org/. 4, 23

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

[4] G. Balakrishnan, F. Durand, and J. Guttag. Detecting pulse from head motions in video. In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, pages 3430–3437, June 2013. 6

[5] Jan Čech, Vojtěch Franc, and Jiří Matas. A 3d approach to facial landmarks: Detection, refinement, and tracking. In *ICPR '14: Proceedings of 23st International Conference on Pattern Recognition*. IAPR, IEEE, 2014. To appear. 6, 9, 23

[6] P. Comon. Independent component analysis – a new concept? *Signal Processing*, 36(3):287–314, 1994. 5

[7] M. Lewandowska, J. Ruminski, T. Kocejko, and J. Nowak. Measuring pulse rate with a webcam - a non-contact method for evaluating cardiac activity. In *Computer Science and Information Systems (FedCSIS), 2011 Federated Conference on*, pages 405–410, Sept 2011. 5

[8] MATLAB. *8.1.0.604 (R2013a)*. The MathWorks Inc., Natick, Massachusetts, 2013. 4, 23

[9] H. Monkaresi, R. Calvo, and H. Yan. A machine learning approach to improve contactless heart rate monitoring using a webcam. *Biomedical and Health Informatics, IEEE Journal of*, PP(99):1–1, 2013. 5

[10] Dany Obeid, Gheorghe Zaharia, Sawsan Sadek, and Ghais El Zein. Cardiopulmonary activity monitoring with contactless microwave sensor. In *Mediterranean Microwave Symposium 2012, Istanbul : Turquie (2012)*, Sept 2012. 6

[11] Jan Plesek. Bc. Master's thesis, CVUT, 2013. 5

[12] Ming-Zher Poh, Daniel J. McDuff, and Rosalind W. Picard. Non-contact, automated cardiac pulse measurements using video imaging and blind source separation. *Opt. Express*, 18:10762–10774, 2010. http://www.opticsinfobase.org/oe/abstract.cfm?URI=oe-18-10-10762. 4

[13] Carlo Tomasi and Takeo Kanade. Detection and tracking of point features. Technical report, International Journal of Computer Vision, 1991. 6

[14] Michal Uřičář, Vojtěch Franc, and Václav Hlaváč. Detector of facial landmarks learned by the structured output SVM. In Gabriela Csurka and José Braz, editors, *VISAPP '12: Proceedings of the 7th International Conference on Computer Vision Theory and Applications*, volume 1, pages 547–556, Porto, Portugal, February 2012. SciTePress - Science and Technology Publications. 23