Master Thesis

Robust detection of heart rate using facial analysis

J. C. Bukala

Master Thesis DKE 16-09

Thesis submitted in partial fulfillment
of the requirements for the degree of Master of Science
of Operations Research at the Department of
Data Science and Knowledge Engineering
of the Maastricht University

Thesis Committee:

Dr. S. Asteriadis
Dr. J. Karel

Maastricht University
Faculty of Humanities and Sciences
Department of Data Science and Knowledge Engineering
Master Operations Research

July 1, 2016

ABSTRACT

Heart rate is an important vital sign when tracking a person's physical and emotional state. However, traditional methods for long-term measurement are often seen as obtrusive and uncomfortable. This thesis will research non-contact techniques to measure heart rate and will advance current methods relying on the tracking of head-motions caused by cardiac activity. A series of techniques have been tested in this thesis, in an attempt to advance the state-of-the-art. The most promising direction to look into, when it comes to spontaneous head motions, is to detect and filter out noisy motions, in an unsupervised manner, especially when no prior knowledge is given regarding the subject or an initial estimate of the heart rate. Several methods have been tested on a publicly available dataset, as well as a dataset recorded for the needs of this work.

Contents

1 Introduction **4**
- 1.1 The cardiac cycle . . . 4
- 1.2 Aim of this thesis . . . 5

2 Related work **7**
- 2.1 Non-vision based . . . 7
- 2.2 Vision based . . . 7
  - 2.2.1 Color-based . . . 8
  - 2.2.2 Motion based . . . 8
  - 2.2.3 Combination . . . 9

3 Methods **10**
- 3.1 Base algorithm . . . 10
  - 3.1.1 Tracking . . . 10
  - 3.1.2 Discarding points . . . 11
  - 3.1.3 Filtering . . . 12
  - 3.1.4 HR-signal selection . . . 12
  - 3.1.5 HR calculation . . . 12
- 3.2 Variations . . . 13
  - 3.2.1 Tracking . . . 14
  - 3.2.2 K-means signal clustering (minimal-norm) . . . 14
  - 3.2.3 Signal selection & HR-calculation . . . 16

4 Theoretical background **18**
- 4.1 Tracking . . . 18
  - 4.1.1 Face detection . . . 18
  - 4.1.2 Tracking points in video . . . 20
  - 4.1.3 Finding good features to track . . . 21
- 4.2 Digital filtering . . . 21
- 4.3 Self-Organizing maps . . . 22
- 4.4 Principal Component Analysis . . . 22

5 Experiments & results **24**
5.1 MAHNOB-HCI 24
5.2 Database created for the purposes of this thesis 25
5.3 Results 26

6 Discussion & Conclusion **31**
6.1 Discussion 31
6.1.1 Recommendations for future work 33
6.2 Conclusion 33

A Code documentation **35**
A.1 C++ code 35
A.2 Unused videos 36

Chapter 1

Introduction

An individual's heart rate (HR) is an important indicator of their emotional and physical condition [1, 2]. A low resting rate can indicate a physically well-trained person, whilst a (too) high rate can even be a risk factor for mortality [3]. In (cognitive or emotionally) stressful situations, HR will increase as well [1]. Thus, the information retrieved from HR measurements can be used for numerous applications, ranging from medicine to research and marketing. For some of these applications, it is vital that the measurement is performed in the most non-invasive way possible, so as not to interfere with the person's actions or mental state. The most commonly used HR measurement devices are an ECG device with stickers and cables or a finger-clip which can be seen as annoying and obtrusive during daily activities. Sometimes a barrier for measurement is the required hardware or measurement device, which can be very expensive to obtain.

1.1 The cardiac cycle

The heart rate is determined by how many times per minute the ventricles contract. For a normal human in rest this can range from 60 - 100 beats per minute [4]. A sinus rhythm (necessary for normal electrical activity within the heart) starts with an electrical impulse generated by the sino-atrial node, located in the wall of the right atrium. The impulses travel to the muscle cells in the atria making them contract. Then they arrive at the atrioventricular node that effectuates a delay, after which they travel through the His-bundle. After this, the electrical impulse is conducted through the Purkinje fibre system arriving at the ventricles, making them contract as well [5]. This (powerful) contraction of the left and right ventricle pumps blood towards the lungs and the aorta respectively. (Of particular interest for this thesis is the blood flowing from the aorta through the carotid artery supplying the head with oxygenated blood.) The contraction of the muscle-cells is caused by (and causes) an electrical impulse that can be measured when measuring the electrical potential over different parts of

the body [5] . This constitutes an electrocardiogram(ECG). In Figure 1.1 the typical form of an ECG for an individual in sinus rhythm is shown. Usually, several parts are discerned: the P-peak which corresponds to the depolarization of the atria which causes them to contract, the QRS-complex which is caused by the ventricles contracting and the T-peak caused by the electric re-polarisation of the ventricles.

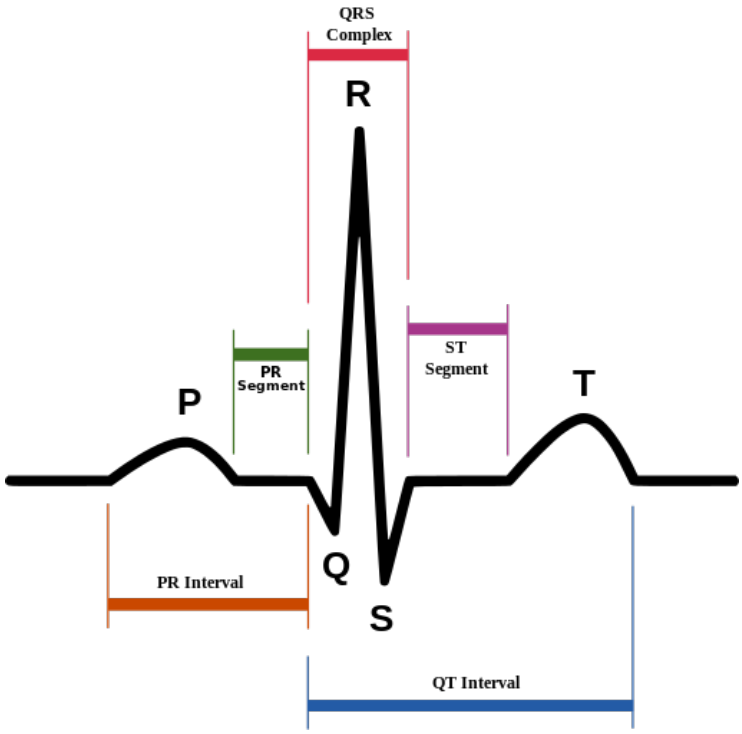

Figure 1.1: The typical form of an ECG during a heartbeat. Picture taken from [6].

1.2 Aim of this thesis

In this thesis a method will be explored and improved to measure HR using only visual information. Approaches utilizing computer vision have been proposed in literature, with as typical example the work presented in [7]. The method used in this thesis will however rely on observing head motions (caused by cardiac activity) with a regular web-cam, and will not depend on color fluctuations of occlusions, since only the motion is needed for HR estimates. This type of approach was first done in [8]. The head motions caused by the blood pumping into the carotid artery will be isolated and further used for this. It is the central aim in this thesis to implement the work in [8] and further optimize it using

machine intelligence and signal processing, so that it can also function in contexts of spontaneous head motions, as well as under low quality video capturing conditions. The research question of this thesis is: **"What is the best way to measure heart rate in realistic human-computer interaction scenarios using only head motions?"**.
First an overview of past literature on both contact and non-contact HR measurement techniques will be given. Then several methods will be proposed for improvement of motion-based non-contact HR measurement and their techniques will be explained. Afterwards, the theoretical background will be given, covering the issues from all different disciplines involved. Then the proposed methods' performance will be tested on both a database constructed for the purposes of this research, as well as the MAHNOB-HCI database [9], a collection of videos which can be seen as the standard to test these methods against [10]. Using the results of these tests, a conclusion is drawn as to the usability of this method in future research and applications. In the appendix there is described how to retrieve and use the code implemented for this thesis.

Chapter 2

Related work

2.1 Non-vision based

An ECG is used for the most common form of HR measurement by calculating the average time between the points of deepest descent of consecutive R-peaks (that correspond to the contraction of the ventricles). This method of HR-measurement is widely used by medical professionals because it is highly reliable and the ECG-measurement can be used to spot a whole range of heart-conditions [5]. Another popular way to measure HR uses the fact that for every heartbeat, the amount of blood in the skin goes through a cyclic movement. A so-called pulse oximeter can measure the amount of blood in the skin of a fingertip by measuring the reflectance of infrared light. This way of using light to measure the volume of blood in an organ is called a photoplethysmogram (PPG). It requires a measurement device to be very close to (or in contact with) the subject's skin. A different (less frequently used) method to measure HR is by measuring the shock wave the beating heart sends through the body. The measurement of movement caused by cardiac activity is called ballistocardiography (BCG). A system that monitors HR and respiration using BCG has already been demonstrated in [11] by placing a pressure sensor under a pillow during sleep.

All these methods require expensive equipment and/or contact with the subject for HR-measurement, which can be unwanted or limit its applications.

2.2 Vision based

It is possible to measure HR using thermal imaging of blood-vessels on the forehead as done in [12], which reported an accuracy of 85%. This was done for subjects in rest, in pain or holding a weight of 20 - 40 lbs in the air. The basic idea behind this method is that the amount of blood in the vessel influences the temperature, making the temperature have a cyclic character with a frequency equal to the HR. This has the obvious downside of requiring thermal imaging

equipment, which is not all that ubiquitous. Hereby an overview is given of works using RGB/greyscale video information to infer heart rates. A more exhaustive analysis, involving thermal imaging as well, can be found in [13].

2.2.1 Color-based

The amount of blood in the skin can not only be measured using infrared light, it can also be seen using light within the visible spectrum. This opens up a way of contact-less heart rate measurement employing off-the-shelf equipment, such as ordinary web-cams, which are today embedded in almost every portable computing device. It is possible to track an individuals' skin-color over time and from the cyclic change in this color deduce the HR [7] [10] [14]. This cyclic color-change has even been used for a form of biometric recognition [15]. It has also been utilized to monitor stress-levels through heart rate variability (HRV) [16]. This method requires color-video to track the color of a patch of uncovered skin and is sensitive to changes in illumination throughout the video, decreasing in accuracy outdoors or in front of a window. The reason for this is that when measuring the color emitted from or reflected off the skin, in can be unclear whether a change in this color has to be attributed to a change in the skin itself or a change in the illumination from the environment (e.g. a lamp turning on or the sun disappearing behind the clouds can both influence the color of the light received by the camera). Another issue with this approach is that skin-color is dependent on the location on which it is measured, so that moving subjects can appear to change skin-color if not tracked accurately. This leads to interference with the actual skin-color change due to the cardiac activity.

2.2.2 Motion based

The aorta splits into multiple arteries, one of which is the carotid artery. This artery goes up through the neck to supply the head with blood. This blood rushing into the head with each heartbeat causes a Newtonian reaction causing the head to move by a slight amount. This movement can be detected by computer vision techniques and it is possible to use this movement to determine a persons HR and HRV. In [8] this is done by tracking several points on the face over time, and discarding the points that move around too much. Then the time-series is bandpass-filtered to leave only the frequencies that could correspond to a HR-signal and their first harmonics. After this principal component analysis(PCA) is applied to the signals and the most periodic signal is selected, defining periodicity as the amount of power in the signal in the frequency with maximal power and its first harmonic. This signals frequency of maximal power is taken to be the HR, and HR-variability can be determined by looking at the standard deviation of the peaks in this signal. This gives decent results for static subjects with neutral facial expressions. In [17] this method is altered by applying a moving-average filter after the bandpass-filter and choosing the most periodic signal using a discrete cosine transform(DCT) instead of a Fourier-transform based method. This is claimed to give good results even when the subjects are

moving, talking or making facial expressions. [18] takes this method and adds to it the tracking of certain specific points on the face. However, [19] found no improvement implementing the DCT-method in [17], so a motion based method to measure HR for realistic scenario's (subjects that are moving, laughing, etc.) is still missing. On a side note, it is also possible to use a motion-tracking technique like this to track the chest in order to determine a subjects' respiration rate [20], yet an acoustic method seems to be the preferred method so far [21]. The method of [8] shall be used in this thesis as a starting point for further improvement.

2.2.3 Combination

A combination of these two vision-based methods to measure HRV has already been tested, as well as a combination of these two and a ballistocardiogram (BCG) measured by the pressure on a chair on which the subject is sitting [19]. The results of these methods are then fused using a Bayesian approach, interpreting each methods' result as an a-posteriori probability density function, and calculating the end-result as in Equation 2.1 (implying a uniform prior distribution). Here η is the HR, and S_i is the method of HR-determination using information from i. This resulted in a mean absolute error in the beat-time of 24.4ms.

$$p(\eta|S_{color}, S_{motion}, S_{BCG}) \propto p(\eta|S_{color})p(\eta|S_{motion})p(\eta|S_{BCG}) \tag{2.1}$$

Chapter 3

Methods

Firstly, the method used in [8] (used as a starting-point) will be explained in detail and in the next section numerous variations on and changes to this method will follow. The methods are mostly implemented in C++ using the OpenCV library, a well-known library in the field of computer vision [22]. Some of the calculations are done in MATLAB in specific variations of the method.

3.1 Base algorithm

This algorithm is using frame sequences of a person sitting/standing in front of a video camera. Several points on the face and head are tracked throughout the video, creating a time-series of their position. After removing erratically moving points, the time series is temporally filtered such that only frequencies which could correspond to a heart rate remain in the signal. On these non-noisy pieces, principal component analysis (PCA) is performed in order to isolate the HR-signal. Now the problem is selecting the principal component corresponding most to the HR-signal. This is taken to be the most-periodic signal. From this selected principal component the HR-frequency is determined. The algorithms' steps will now be discussed in further detail.

3.1.1 Tracking

Tracking is defined here as following the location of certain points between frames of a movie. Because only points on the face are to be tracked, it is important to first find the location of the face in a frame. This is done with OpenCV's implementation of the Viola-Jones face-detector [23]. The points that are to be tracked are found by two methods. One method is by using OpenCV's GoodFeaturesToTrack(GFT) function, which selects points using the Shi-Tomasi corner detector [24]. The GFT-function is applied to the first frame of a video in an area 80% of the height and 50% of the width of the area returned by the cascade classifier, to make sure no points are placed outside

of the face. The area from 45% to 55% from the top height-wise is left out as well, to remove the eye-region from the face. This is done because the frequency at which a person blinks can be misinterpreted as a heart rate frequency later on in the analysis. An example of these points in a video still can be seen in red in Figure 3.1. These points selected by the GFT-function are tracked by means of the Lukas-Kanade optical flow algorithm [25]. The videos used in [8] have a frame-rate of 30Hz. After having obtained the time-series of the points' position, only the y-coordinate is used in the further analysis, because most of the head movement caused by cardiac activity is in the vertical direction due to the blood of the carotid artery flowing upwards.

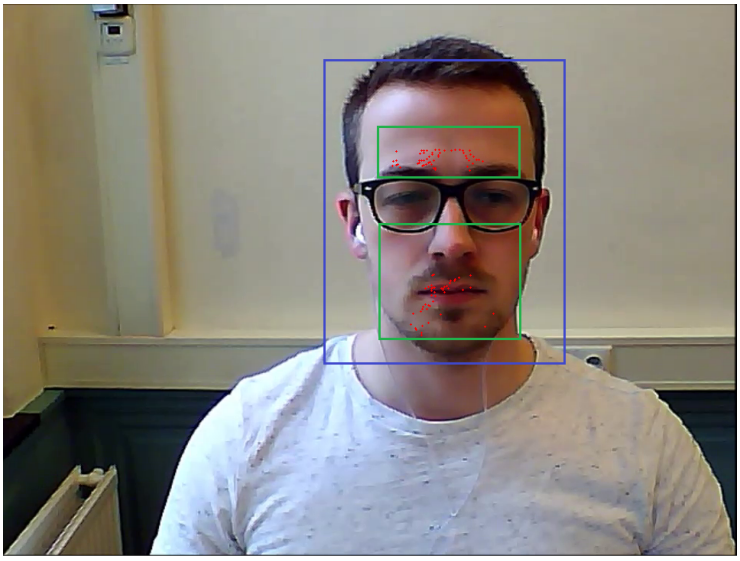

Figure 3.1: Example of the points GFT selects for tracking in a video frame. The points selected by the GFT-function can be seen in red while the bounding box of the facial-area can be seen in blue and the box classified as the region of interest in green.

3.1.2 Discarding points

After tracking, features whose position over time is noisy and not smooth should be discarded, in order to only keep those that have higher chances of following the actual movement of their corresponding facial area. For this, the amount of pixels moved between each frame is calculated for each point. If the maximum movement between frames of a tracked point is bigger than the mode of its movement between frames, the entire point is discarded. This has the effect of disregarding points that have sudden jumps or large movements, where either a large movement took place or where something went wrong during tracking. The points that are left over after this step are used in further steps.

3.1.3 Filtering

Each time-series is then centered around its mean. Because all persons are assumed to have a (resting) HR higher than 45 beats/min and lower than 200 beats/min, a lot of noise can be filtered out by means of a temporal digital filter. A 5th-order Butterworth bandpass filter is applied to the 30Hz signal, with a passband of [0.75, 5]Hz. This choice of frequency range also conserves the first harmonics of the HR-signal, which will help to recognize the signal. At the same time this removes low frequency periodic movement, like the movement caused by respiration which is of a larger amplitude and would otherwise interfere with the detection of the HR-signal. After filtering, the signals are up-sampled using cubic spline interpolation to 250Hz. This is done to replicate the sampling rate of a modern ECG device that is used to calculate HR-variability. In [8] the up-sampling step is done before the filtering step, however this results in an unstable filter for the required filter settings so the order of these steps are switched.

3.1.4 HR-signal selection

Now the HR-signal has to be isolated from the other sources of movement. This is done by PCA, which is expected to isolate signal components, according to their frequency content. During this step, the feature space is defined by the y-location of each point being a separate dimension, and so all the points' locations are represented by a point in this space for every frame. However, when calculating the principal components, 25% of the points in this space which have the biggest L2-norm are not taken into account. This is done to not let a few outliers or big movements in the video influence the result of the PCA decomposition. Then, the entire signals (including points not taken into account during computation of the principal components) are projected onto the principal components to retrieve the signals in their new basis. Afterwards, the signal corresponding to the HR-signal has to be found. This signal characterizes itself by having a high periodicity, so the most periodic signal has to be selected as the HR-signal. The periodicity Q_S of a signal $S(t)$ with Fourier-transform $\mathcal{S}(f)$ is defined in the paper according to equation 3.1. f_{max} is the frequency with maximal power in the signal, which can be easily found as it is the same frequency at which $\mathcal{S}(f)$ is at its maximum. Here P_S is the total power of the signal S. In [8] it was found that it was not necessary to calculate this for more than the first 5 principal components.

$$Q_S = \frac{\mathcal{S}(f_{max})^2 + \mathcal{S}(2 \cdot f_{max})^2}{P_S} \tag{3.1}$$

3.1.5 HR calculation

When this most-periodic signal is selected, the HR-frequency is calculated simply as the frequency of maximum power in the Fourier-transform of this signal, HR$= 60 \cdot f_{max}$ beats per minute.

3.2 Variations

The method described above has the drawback that it requires subjects to remain static over the course of the tracking procedure, in order to allow for the HR signal to be recovered easily. However, in real-life circumstances people move their heads and could change their facial expressions. For this reason, a large part of this thesis implemented and tested a variety of techniques to alleviate this problem. Towards this direction, we employed the following steps: The main differences from [8] consist of adding a clustering step after the filtering step and using a different method of selecting the HR-signal after PCA as well as a change in the calculation of the HR from this signal. The point-discarding step is skipped for all of these variations, because the discarding-rule is found to be too strict for video's in which there is a bit of movement. In these video's the rule causes all the points to be discarded, making it useless. The up-sampling to 250Hz is skipped as well, seeing as the goal of this thesis is not determination of HR variability.
To improve the stability of the tracking, so-called landmark (LM) points can be tracked in addition to the GFT-points used in [8].
The following different methods are tested for the clustering step:

- K-means clustering selecting the cluster with minimal norm: Due to the shape of the k-means feature space, this should correspond to the least noisy cluster
- K-means clustering selecting the largest cluster: The largest cluster should contain the most frequently occurring type of signal, the stable signal.
- SOM + K-means clustering selecting the cluster with minimal norm: SOMs are used for pre-clustering to achieve a better clustering result.

All of the clustering methods can result in either selecting the whole trajectories of a subset of points (called *no-cut*) or selecting all points but only using a 10-sec time-interval (called *time-slice*). To improve upon the method of isolating the HR-signal and calculating the HR from it, the following methods are used:

- Peak-detection based: Robustly detecting local maxima in the signals, and using the regularity of these peaks to detect the HR-signal. Then using the mean time between these peaks to calculate HR.
- Peak-detection based using 'chunks': the HR-signal is partitioned into 10-second windows, and the windows that give the closest result to each other are used in the calculation of the HR. This should make the peak-detection based method more robust.
- Discrete cosine transform based approach: To increase accuracy of method. This method performed well in [17].

3.2.1 Tracking

Another way to define points to be tracked is by using the IntraFace C++-library, which finds certain specific points around the eyes, nose, eyebrows and lips [26]. This library recognizes certain landmark (LM) points on a face using the supervised descent method (SDM). This method minimizes a non-linear least squares function to align the face without computing the Jacobian or the Hessian, by being trained on images of faces to pick descent directions. This greatly decreases the calculation-time, making it useful for even real-time tracking. In each frame, SDM is used to detect the LM-points, initializing the points' location as their location in the previous frame. An example of these points can be seen in green in Figure 3.2. The thought behind using these points is that they should be more stable in the case of moving subjects, as their position is re-calculated separately each frame. They should be more robust, even returning to their intended position after a subject for example moves a hand across his/her face, something that would cause the GFT-points to be completely displaced. By using both the GFT-points and the LM-points, the result should only get better. The tracking of LM-points for HR-detection using head motions was first done in [17].

3.2.2 K-means signal clustering (minimal-norm)

A way to remove noise (caused by rigid head motions and facial expressions) is to disregard the signals at those points in time where there is lots of movement. This step is performed directly after the filtering-step. In the *time-slice* variation, this noise-removal is done by cutting each signal into pieces with a length of 10 seconds (the *no-cut* variation skips this step). Then, a few features are calculated from each signal-segment; the mean value, standard deviation and entropy. Entropy of a signal is defined in Equation 3.2, where p_i is the relative frequency of number x_i. To calculate this, first a histogram of the signal is calculated with N bins, in our case $N = 256$. Then p_i is the value of the histogram at x_i divided by the length of the signal.

$$S = -\sum_{i \in N} p_i \log p_i \tag{3.2}$$

The features are chosen because it is expected that noisy signals will tend to have a high standard deviation, mean value and entropy. This would make it possible to separate noisy from non-noisy signals using these features. The features are used to cluster each signal by using the k-means algorithm (with $k = 3$). After clustering, the segments that are put in the cluster with the cluster-center of minimal norm are used in the further analysis. In the *no-cut* method, points that have a lot of noisy segments in them are discarded and in the *time-slice* method only the segments that are in the least noisy time-slice are used. An example of where this would be useful is in the case of movement as in Figure 3.3. Only the middle of this signal would be used in further analysis, discarding the frames with a lot of movement in the beginning and the end of

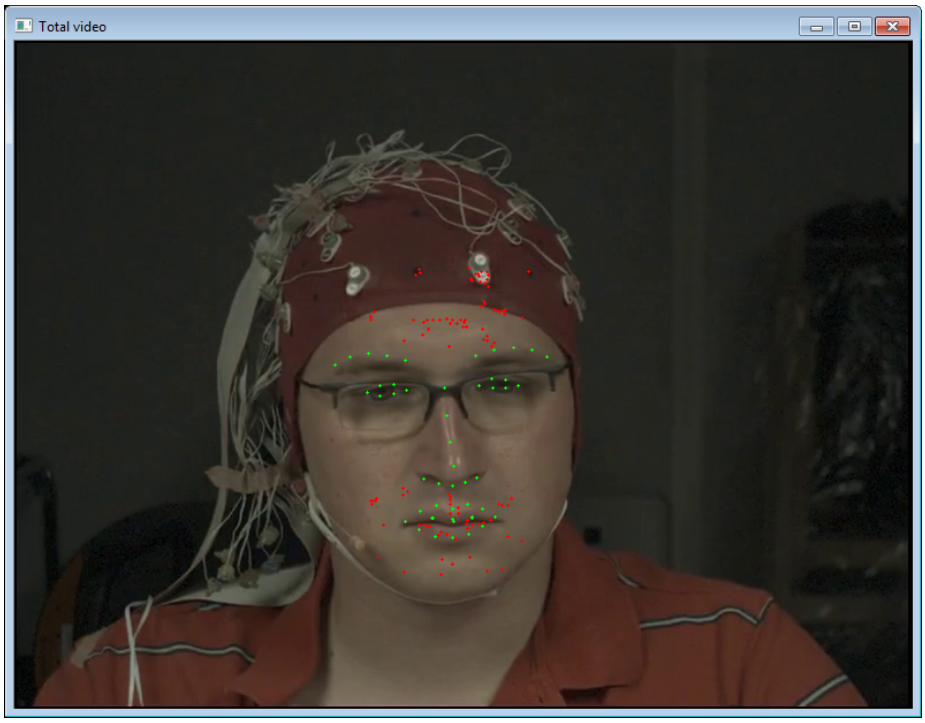

Figure 3.2: Example of the points selected for tracking in a video frame. The points selected by the GFT-function can be seen in red while the IntraFace-points are depicted in green.

the video. In a sense, this can be seen as a (more versatile) alternative for the point-discarding in [8].

Signal clustering (largest cluster)

Instead of selecting the pieces in the cluster which has the center of minimal norm during the clustering step, the pieces of the most-populated cluster are chosen in this variation. The thought behind this is that during most of the video, the most signals should be stable, so the largest cluster should also be the cluster containing the cleanest signals.

Clustering using self-organizing maps

Instead of using only k-means in the noisy segment removal, it is possible to use SOMs to pre-cluster the data into a relatively large amount of clusters (because a large amount of neurons can better approximate the distribution), and then take these clusters as input for the k-means algorithm. The idea is that this will result in a better clustering result, because the clustering is done 'gradually',

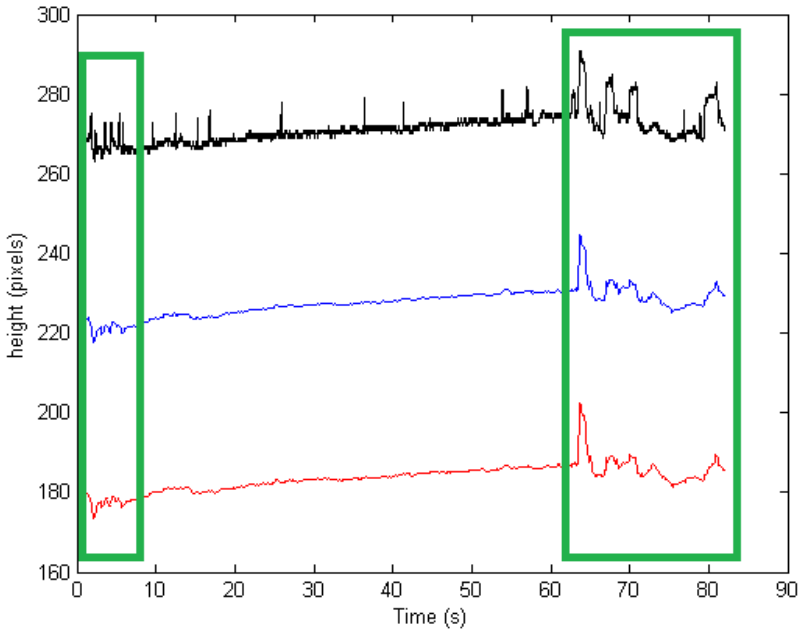

Figure 3.3: Time-series of a few points tracked throughout a video. The problematic segments are depicted within the green rectangles.

and the SOMs are able to fit to more complex cluster shapes. SOMs can also give an indication of the amount of clusters present in the data by manual observation of the U-matrix [27]. Doing this for several videos gave the value $k = 3$ for all of the methods using the k-means algorithm.

3.2.3 Signal selection & HR-calculation

In all methods, PCA is still first used to isolate the HR-signal. However, now all of the first fifteen principal components are used in the search for the HR-signal, due to the fact that the videos are expected to be more noisy, resulting in the HR-signal possibly being contained in a component with relatively little power. All signals are projected unto the first fifteen components and amongst these components the HR-signal will be searched.

Peak-detection based

In this variation, component selection is done by determining the locations of local maxima for each signal, whilst constraining the maxima to have a minimum temporal separation corresponding to a HR of 150 beats per minute ($\Delta t > \frac{60}{150}$ sec). To remove small peaks that do not correspond to a heart beat, a height

threshold is set, under which peaks are discarded. By experimentation, it is found that a peak height of 0.6 times the average peak height for the signal is a good threshold. Afterwards, for each component the standard deviation of the time-lengths between the local maxima is determined and the component with the lowest value for this standard deviation is chosen as the most periodic one and thus most likely to be the HR-signal. The HR can now be calculated easily given the chosen component and its local maxima. The mean of the time between the local maxima is the heartrate-period T, thus the HR-frequency is given as $f = \frac{60}{T}$ beats/min.

Peak-detection based using signal 'chunks'

This method first uses the same method as the previous one to select the HR-signal. Then, this signal is cut into three 'chunks' of the same size. On each of these chunks, the local maxima are determined and the HR-frequency is calculated from it as in the previous method. Then, the mean of the two HR-frequencies that are closest together is taken as the HR. This prevents noise in only one part of the signal to influence the end-result, making it more robust.

Discrete cosine transform based

Instead of using a Fourier-transform based method to select the signal containing the heart beat, a method based on the discrete cosine transform (DCT) is used. This method is taken from [17], where it is said to improve upon the Fourier-based method.

Chapter 4

Theoretical background

4.1 Tracking

There are multiple steps needed to track the location of a head throughout a video, see Figure 4.1. First, the general area where the head is present in the frame of the video should be recognized. To do this, the face is detected. Then, points on the head need to be selected such that they will be relatively easy to track in between frames. Only then can the actual tracking begin. In the next subsections, face recognition will be explained in further detail. Then tracking will follow, and using the knowledge about tracking, the theory behind selecting good points to track will be treated.

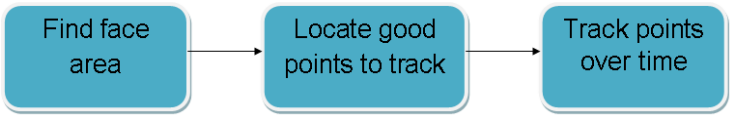

Figure 4.1: Steps needed to track points on a person's head.

4.1.1 Face detection

The currently most popular way of detecting the location of faces in an image is by usage of the Viola-Jones face detector [23]. It works on a gray-scale image using the intensity value of each pixel. The detector uses Haar-like features combined with an integral representation of the image to quickly calculate them. In the integral representation, the value of each pixel is equal to the values of all the pixels in the original picture to the top and left of it (inclusive) summed. Once calculated, the integral image is used to calculate the sum of pixels in rectangular subsets of the image very rapidly, greatly speeding up the calculation of the features. It uses the AdaBoost algorithm to train a lot of weak classifiers, each weak classifier responding to only one feature [28]. From equation 4.1 the

AdaBoost classifier can be seen to be a weighted linear sum of several 'weak' classifiers $h_j(\mathbf{x})$ defined in Equation 4.2.

$$h(\mathbf{x}) = \text{sign}\left(\sum_{j=1}^{M} \alpha_j h_j(\mathbf{x})\right) \tag{4.1}$$

$$h_j(\mathbf{x}) = \begin{cases} -s_j & \text{if } f_j < \theta_j \\ s_j & \text{otherwise} \end{cases} \tag{4.2}$$

Here the f_j is the value of the feature and α_j, θ_j and $s_j \in \pm 1$ are determined during training on a set of images. After training these weak classifiers about 1000 of the features are kept and the rest are disregarded. To save computation time, each sub-window of the image can be classified quickly with both a high detection rate and a high false-positive (the system sees a face where there is none) rate so that parts of the image where there is almost surely no face can be disregarded in later, more refined searches, using smaller sub-windows. For example, with appropriate threshold values, only two features are needed to get a detection rate of about 100% with a false-positive rate of about 40%. For an example of these features applied on an image, see figure 4.2. Then a cascade of

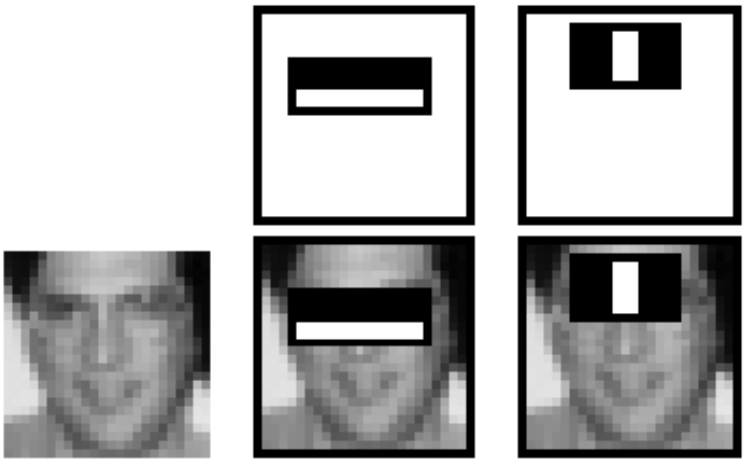

Figure 4.2: The two most important features in the Viola-Jones face detector together with their application to a face-image. The value of each feature is the intensity of the pixels within the white box minus the intensities of the pixels within the black box. These are so successful because the upper cheeks are brighter than the eye-area (first feature) and the nose-bridge is brighter than the eyes (second feature). Picture taken from [23].

more refined classifications can be performed on each remaining sub-window, at

each step classifying smaller sub-windows and discarding sub-windows without faces. The full algorithm has 38 of these stages with over 6000 features in total, nevertheless detecting faces very quickly.

4.1.2 Tracking points in video

The most popular method of tracking points in a video is by using the Lucas-Kanade tracking algorithm [25]. This algorithm assumes that when tracking a point $p = (x_p, y_p)$ the image intensity in between frames of the points in a small neighbourhood around p is constant. This gives rise to Equation 4.3 which can be rewritten as Equation 4.4, the optical flow equation, in which $I(x, y, t)$ is the intensity of the video at position (x, y) and time t. $\vec{V} = [V_x, V_y]^T$ is the optical flow of $I(x, y, t)$, which approximates the movement speed of this pixel.

$$\frac{\partial I(p,t)}{\partial x}\Delta x + \frac{\partial I(p,t)}{\partial y}\Delta y + \frac{\partial I(p,t)}{\partial t}\Delta t = 0 \tag{4.3}$$

$$\nabla I(p,t)^T \cdot \vec{V} = -\frac{\partial I(p,t)}{\partial t} \tag{4.4}$$

When assuming that the flow is constant in this small neighbourhood around p, the equation holds for all the points $n_i = (x_i, y_i)$ in the neighbourhood of p. This gives an overdetermined set of equations, which can be solved for $\vec{V}$ using the least-squares solution which can be calculated as in Equation 4.5.

$$\vec{V} = -\left(\sum_i \nabla I(n_i) \cdot (\nabla I(n_i))^T\right)^{-1} \cdot \sum_i \nabla I(n_i) \cdot \frac{\partial I(n_i)}{\partial t} \tag{4.5}$$

However, this method disregards the fact that the closer n_i is to the tracked pixel p, the more important that pixel is to estimate $\vec{V}$. That is why each pixel can be given a weight w_i equal to a Gaussian function of the distance between it and p. After defining the matrix G in Equation 4.6, the solution can then be calculated as in Equation 4.7.

$$G := \sum_i w_i \cdot \nabla I(n_i) \cdot (\nabla I(n_i))^T \tag{4.6}$$

$$\vec{V} = -G^{-1} \cdot \sum_i w_i \cdot \nabla I(n_i) \cdot \frac{\partial I(n_i)}{\partial t} \tag{4.7}$$

Using $\vec{V}$ it is possible to identify the pixel p in a subsequent frame with Equation 4.8, where Δt is the time between subsequent frames.

$$p^{t+1} = p^t + \vec{V} \cdot \Delta t \tag{4.8}$$

4.1.3 Finding good features to track

Of course, some pixels are easier to track than others: a point in a homogeneous area is impossible to track because optical flow can not be derived from the neighbourhood, while it is simpler for an area with a good texture. This is why often before the tracking of an object starts, points that will be easier to track throughout the video are identified in the first frame. The method for finding these points is described in [29]. Starting from the matrix G (as defined in Equation 4.6), its eigenvalues λ_1 and λ_2 are determined. If $\min(\lambda_1, \lambda_2) > \lambda$ for a pre-defined quality-threshold λ, the point is selected for tracking. (Whilst being outside the scope of this thesis, there are also methods of checking the tracking quality during the tracking itself using affine transformations between subsequent frames, see [24].)

4.2 Digital filtering

A filter is a system that modifies a signal in order to enhance or discard certain features of it. Filtering can alter the signal in such a way that certain frequencies are suppressed or enhanced. Digital filtering is the term used in the cases where time is discrete (this means that there is a finite sampling frequency). The difference equation representing the general form of a (causal) digital filter can be seen in Equation 4.9.

$$y[n] = -\sum_{k=1}^{M} a_k y[n-k] + \sum_{k=0}^{N} b_k x[n-k] \tag{4.9}$$

Here $x[i]$ and $y[i]$ are the values of the original signal and the filtered signal respectively at time i. $a_k \in \mathbb{R}$ are called the filters' feed-backward coefficients and $b_k \in \mathbb{R}$ the feed-forward coefficients. In designing a filter, the filter lengths M and N have to be determined, together with the coefficients a_k and b_k. In the case of a proper filter, M is also commonly referred to as the order of the filter. When using a filter with a long length, certain features that are very localized in the original signal x can get spread out over a distance equal to the filter length. This effect is stronger in the case in the case of Infinite Impulse Response (IIR) filters, which are filters where at least one of the $a_k \neq 0$. The dependency of the output on the previous values of y can cause the input value at the beginning of the signal x to have an impact on the output value y for an arbitrary time after. At the same time, other desired properties of the filter can force this design. For example a bandpass filter, which only lets a certain band of frequencies through, can get longer and longer as the frequency band gets narrower. This has to do with the uncertainty principle, according to which the product of the width of the frequency band, and the distance by which something can get spread out after the filter, cannot be smaller than a specific constant, called the Heisenberg-Gabor limit.
The function that describes how much each frequency is suppressed or enhanced

by a certain filter is called the gain $G(\omega) = |H(i\omega)|$, where ω is the angular frequency and $H(z)$ is the complex-valued function defined by Equation 4.10.

$$H(z) = \frac{\sum_{k=0}^{N} b_k z^{-k}}{1 + \sum_{k=1}^{M} a_k z^{-k}} \tag{4.10}$$

$H(i\omega)$ is commonly referred to as the frequency response, whilst $H(z)$ is called the transfer function. A commonly chosen type of filter is the Butterworth filter. It is useful for its property that $G(\omega)$ is maximally flat within the required frequency band whilst going to zero quite fast outside this band [30]. More information on the exact calculation of the Butterworth filter coefficients can be found in [31] and information on filtering in general can be found in [32].

4.3 Self-Organizing maps

Self-organizing maps (SOMs) are neural networks (often organized in a grid) which can be trained by competitive learning to approximate a distribution [33]. A SOM defines a mapping from a space ξ to the set of neurons $\mathcal{S}$, which each also have a location in the space ξ, called the 'weight'. For each point $x \in \xi$, its image is the neuron which is the closest (using some metric, often Euclidean) to it in ξ. To be useful in approximating a distribution in ξ, the locations of the neurons have to be determined during a training period using samples from the desired distribution. During this training, the weight $\mathbf{W_v}$ of neuron $\mathbf{v}$ is updated according to Equation 4.11.

$$\mathbf{W_v}(t+1) = \mathbf{W_v}(t) + \Theta(u, v, t)\alpha(t)(\mathbf{D} - \mathbf{W_v}(t)) \tag{4.11}$$

In this update step, $\mathbf{D} \in \xi$ is the samples' location, t denotes the updating step, $\alpha(t)$ is a learning coefficient that decreases with each step, and $\Theta(u, v, t)$ is the neighbourhood function. This function gives the distance between the neuron $\mathbf{v}$ and the neuron $\mathbf{u}$ which is defined to be the closest neuron to the sample during step t. For fast convergence, initial neuron weights are often sampled from the first principal components of the distribution that has to be approximated. After this training, the locations of the neurons in $\mathcal{S}$ should approximate the distribution. It is then possible to use the inter-neuron distance in a plot to visualize high-dimensional data in a lower-dimensional form. Often a two-dimensional grid is used, so a 2-D plot image can be made to visualize the distances between the neutron. This is called a U-Matrix [27]. The trained SOM can also be used for clustering by using the locations of each neuron as cluster centers and assigning each sample to its nearest neuron.

4.4 Principal Component Analysis

Principal Component Analysis (PCA) is a method of calculating a new (orthogonal) basis for a set of measurements [34]. This new basis is chosen in such a

way that the first basis vector points in the direction of highest variance of the set of measurements. The next basis vectors point in the direction of the highest variance with the constraint that they must be perpendicular to the preceding basis vectors.
The principal components can be calculated in the following way (using the so-called covariance method) [35]: Organize the data in the matrix $\mathbf{X}$, where each of the n observations occupies a row of length p of this matrix. Then calculate the mean of these data points, construct the matrix $\mathbf{M}$ which has this mean vector on each of its n rows and subtract it from $\mathbf{X}$ to obtain $\mathbf{B}$ (see Equation 4.12).

$$\mathbf{B} = \mathbf{X} - \mathbf{M} \tag{4.12}$$

Now determine the empirical covariance matrix $\mathbf{C}$ through Equation 4.13.

$$\mathbf{C} = \frac{1}{n-1}\mathbf{B}^T\mathbf{B} \tag{4.13}$$

Now the eigenvalues and eigenvectors of $\mathbf{C}$ have to be determined, by using Equation 4.14.

$$\mathbf{V}^{-1}\mathbf{C}\mathbf{V} = \mathbf{D} \tag{4.14}$$

Here $\mathbf{V}$ is a matrix with an eigenvectors of $\mathbf{C}$ in each of its columns and $\mathbf{D}$ is a diagonal matrix with all the eigenvalues on its diagonal. This equation is commonly solved using a singular value decomposition (SVD). The columns of $\mathbf{V}$ (the eigenvectors) are then ordered according to their eigenvalues, the eigenvector with the highest eigenvalue ending up in the leftmost column, giving $\mathbf{V_o}$. This puts the vectors explaining the most of the data's variance first. Now each value can be projected onto this new basis of eigenvectors (see Equation 4.15), and the mean value of the data points can be added to each row again to obtain the measurements in their new basis as in Equation 4.16.

$$\mathbf{B}' = \mathbf{V_o}^T\mathbf{B} \tag{4.15}$$

$$\mathbf{X}' = \mathbf{B} + \mathbf{M} \tag{4.16}$$

This has several consequences: Firstly, a lower-dimensional projection of the measurements can be made by projecting them onto the first few basis vectors. This works because the first vectors typically capture most of the variance present in the data, so not much information is lost using this projection. Secondly and most importantly for this thesis, this removes most correlation between the basis vectors, describing different sources of variation by different basis vectors.

Chapter 5

Experiments & results

To test the basic method and its variants proposed in this thesis, videos from two sources will be tested. One of these sources is MAHNOB-HCI, a publicly available database. Another source is a database of videos recorded specifically for the purpose of this thesis.

5.1 MAHNOB-HCI

The MAHNOB-HCI database is a database of realistic human-computer interactions. Participants are shown both neutral and emotion-eliciting videos. For representative examples of these, see Figure 5.1. Among other modes of measurement, the 30 participants are filmed from the front at 61 frames/sec with a resolution of 780x580 pixels while their electrocardiogram(ECG) is measured. The ground truth HR can be obtained from the ECG by means of a MATLAB implementation of the Pan-Tompkins algorithm [36]. However, using this algorithm an inaccurate HR is calculated for at least one person, of which an ECG2-signal can be seen in Figure 5.2. This is due to the T-peak of the ECG-signal, which has a relatively high amplitude. To determine the HR for this person the beats are just counted manually. Only videos where the recorded individual has consented to publication of their data are taken into account. This leaves 25 out of the 30 persons in the database. Of these individuals, subject 12 has no ECG-data so the ground truth HR cannot be calculated. For subject 26, the color-video is missing so the black and white video is used. This is not seen to be a problem because the camera-angle of the gray-scale videos is almost the same and each color-frame is converted to gray-scale for tracking anyways. The method is tested on the 491 videos that remain when selecting videos that are 30 seconds or longer and where the subjects are shown emotion-eliciting videos. Following [10] and [18] only frames 306 to 2135 will be used for the analysis. More information on the MAHNOB-HCI database can be found in [9]. An example still of a video from the database can be seen in Figure 3.2 (sans red and green dots). In Figure 5.3 the results of analyzing the ECG2 leads of the

used sessions is shown in a histogram to give an idea of the range of HR of the subjects in this database. In Figure 5.4 a representative part of the ECG2 signal accompanying the videos in the database is shown.

Figure 5.1: An example of an emotion-eliciting video to the left (a scene from a horror-movie) and a neutral video to the right (static color-bars) used during the creation of the MAHNOB-HCI dataset.

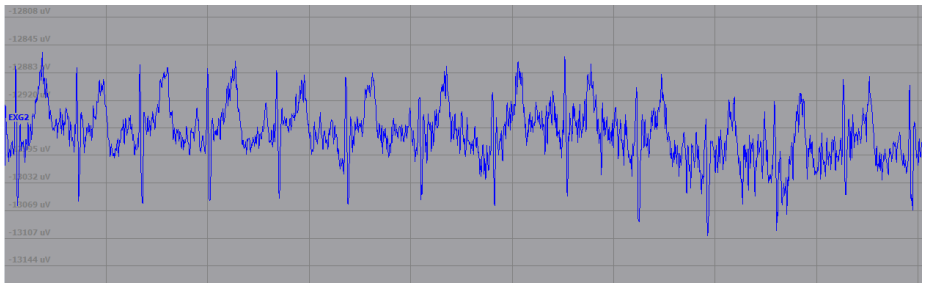

Figure 5.2: A plot of the ECG2-lead of subject 23 in the MAHNOB-HCI database recorded during session 2884. A clear T-elevation can be seen, causing the Pan-Tompkins algorithm to give an erroneous HR.

5.2 Database created for the purposes of this thesis

This is a collection of 21 videos recorded with a web-cam of several persons sitting in front of computer with either a neutral expression, making facial expressions or speaking. They are shot at 30 frames/sec with a resolution of 640x480 pixels. For a representative still from a video, see Figure 5.5. The ground-truth HR is measured simply by feeling the wrist of the subject throughout the recording and counting the number of pulsations of the artery felt in one minute. This database was developed to test the feasibility and flexibility of the methods, and is designed to test the effect of several specific types of behaviour on the result, sometimes containing more movement or facial activity than the recordings of

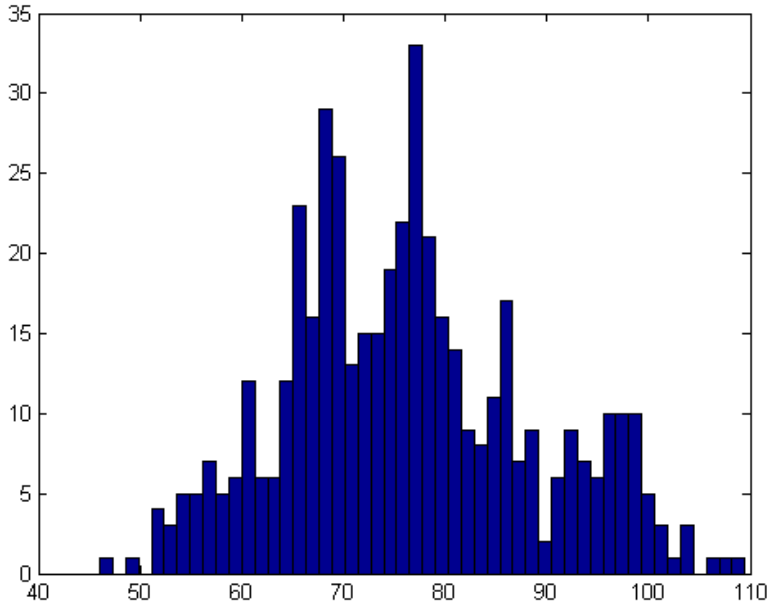

Figure 5.3: A histogram of the HR measured from the ECG2-lead using the Pan-Tompkins algorithm on the selected subset of videos from the MAHNOB-HCI database.

the MAHNOB-HCI dataset. Four extra video's were made in addition to the 21 aforementioned video's. These video's consisted of subjects bobbing their head in a specific frequency. These were made to test the effects of periodic movement with a large amplitude on the method, and the results of these are not incorporated in the rest of the results on the database created for the purposes of this thesis.

5.3 Results

An example of a time-series obtained by tracking points during a video can be seen in Figure 5.6. The black plot is from a landmark point tracked using the IntraFace library. This tracking library does not support sub-pixel tracking, so the value is rounded to entire pixels. The times at which bigger movements are made, in the first 10 and last 20 seconds, are clearly visible. When zooming in on one of the signals tracked using the Lukas-Kanade algorithm, Figure 5.7 is obtained. Here some periodic behaviour that could be attributed to cardiac activity can already be seen, although it is still mixed with noise and other movements. A signal from the same video after applying clustering and PCA

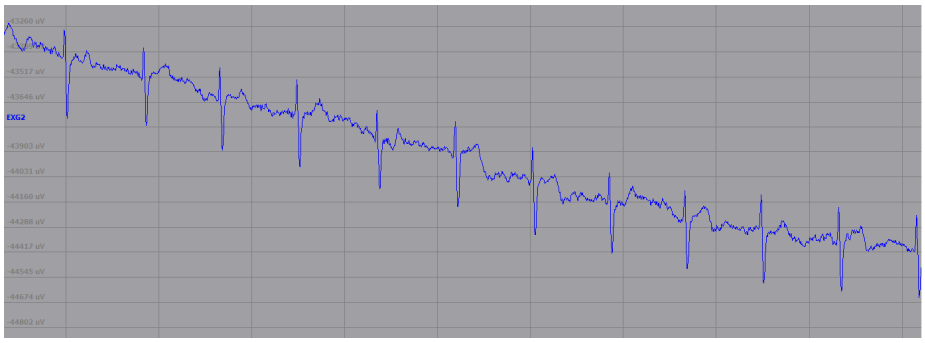

Figure 5.4: A plot of the ECG2-lead of one of the subjects in the MAHNOB-HCI database recorded during a session.

Figure 5.5: Still from a video in the self-made database.

can be seen in Figure 5.8.

In Table 5.1 the results of different methods on the MAHNOB-HCI database are shown. These methods differ in the usage of the k-means clustering step (and the choice of cluster), the usage of SOMs before k-means as a way of clustering, as well as the method of detecting the HR from the processed signal (peak-detection, DCT or Fourier). The usage of the different types of points, GFT points and/or IntraFace landmark (LM) points, is also varied. For each method the mean absolute error is calculated, as well as the percentage of predictions that are not further than 10% from the ground truth, called the 'Accuracy'. The 10% is chosen as a reasonable margin where the measurement is still useful in a medical setting. The last column gives the value of the Pearson correlation coefficient ρ, a measure of the linear correlation between the ground truth HR and the given methods' HR prediction [37]. Statistical significance at the $p < 0.01$ level (calculated using a Student's t distribution) is indicated by the presence of a '*' marker. Table 5.2 contains the same results for our own dataset.

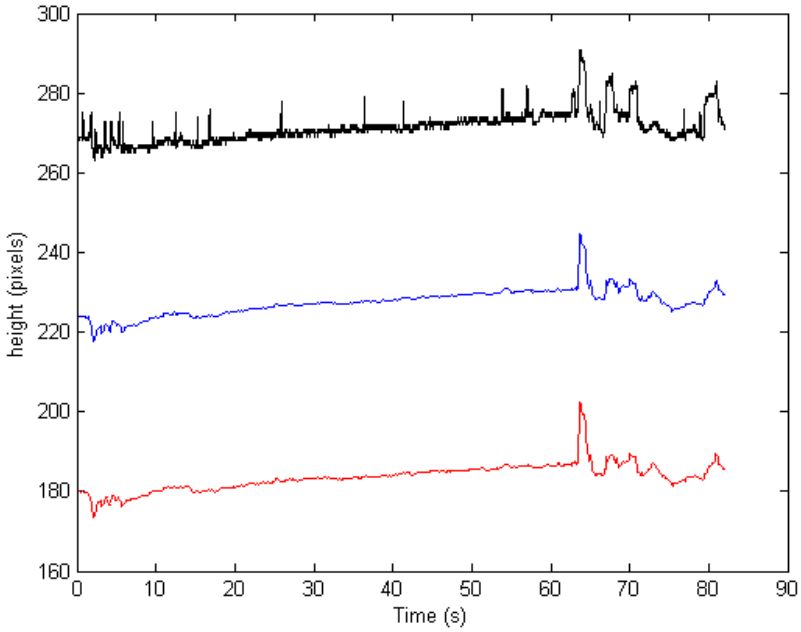

Figure 5.6: Time-series of a few points tracked throughout a video of the MAHNOB-HCI database. The location of the point plotted in black was tracked using the IntraFace library, whilst the other two points use GoodFeaturesToTrack with the Lukas-Kanade algorithm.

Method	HR-detection	Mean absolute error	Accuracy	ρ
Plain as in [8]	Fourier	26.2785%	17.62%	0.13*
Plain as in [8]	Peak-detection	15.116%	41.61%	0.02
As in [8] + LM-points	Peak-detection	15.091%	42.46%	-0.04
Plain as in [8]	DCT	31.37%	10.19%	0.05
K-means (no-cut, min-norm)	Fourier	26.5%	18.05%	0.00
K-means (no-cut, min-norm)	Peak-detection	14.83%	42.25%	0.00
K-means (time-slice, min-norm)	Fourier	25.38%	20.81%	-0.02
K-means (time-slice, min-norm)	Peak-detection	18.77%	34.18%	0.03
Only GFT + K-means (no-cut, min-norm)	Peak-detection	15.04%	41.61%	-0.01
Only LM + K-means (no-cut, min-norm)	Peak-detection	15.66%	40.13%	0.02
SOM+K-means (min-norm)	Peak-detection	21.12%	33.76%	-0.01
K-means (no-cut, largest-cluster)	Fourier	26.43%	18.47%	0.15*
K-means (no-cut, largest-cluster)	Peak-detection	14.97%	41.4%	-0.01

Table 5.1: Results of different methods on the MAHNOB-HCI database.

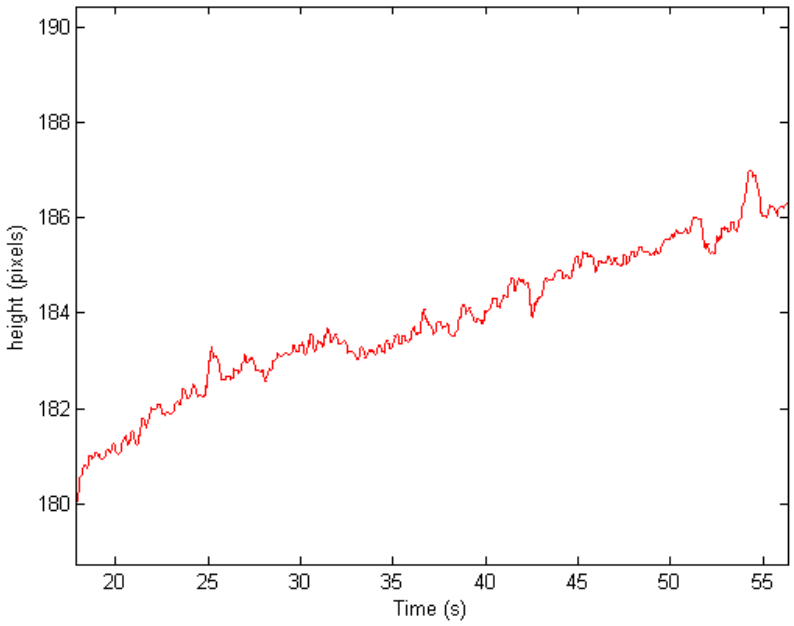

Figure 5.7: Time-series of a point tracked throughout a video of the MAHNOB-HCI database using the Lukas-Kanade algorithm.

Method	HR-detection	Mean absolute error	Accuracy	ρ
Plain as in [8]	Fourier	15.53%	45%	-0.03
Plain as in [8]	Peak-detection	13.32%	50%	0.22
Plain as in [8]	Peak-detection(chunks)	15.88%	40%	-0.03
K-means (no cut, min-norm)	Fourier	17.02%	35%	0.55*
K-means (no cut, min-norm)	DCT	17.18%	40%	-0.23
K-means (no cut, min-norm)	Peak-detection	24.01%	15%	-0.18
K-means (no cut, min-norm)	Peak-detection(chunks)	18.85%	25%	0.24
K-means (no cut, largest-cluster)	Fourier	14.24%	45%	0.14

Table 5.2: Results of different methods on the dataset make for the purpose of this thesis.

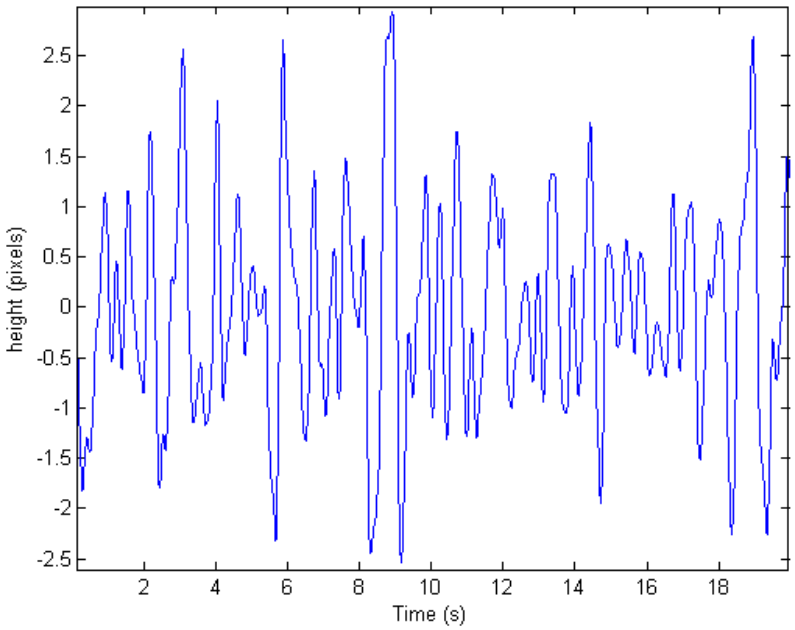

Figure 5.8: Plot of the first 20 seconds of a signal after clustering and PCA. This is the kind of signal the HR-detection takes as input.

Chapter 6

Discussion & Conclusion

6.1 Discussion

This method managed to reduce the mean error rate on the MAHNOB-HCI dataset from 26.28% to 14.97%. For the self-made database, the results changed from 15.53% to 13.32% (depending on the chosen methods). However, by looking at the correlation coefficient the effectiveness of these methods on the MAHNOB-HCI database is called into doubt. Only the k-means (no-cut, largest cluster) method managed to get better results than [8] with a statistically significant coefficient of 0.15, sign of a (weak) correlation. The dataset made for this thesis resulted in higher correlation coefficients, with the k-means (no-cut, min-norm) method with Fourier HR-detection giving a statistically significant correlation coefficient of 0.55. In addition to this, the method in [8] combined with peak-detection resulted in a correlation of 0.22, although with $p > 0.01$. Better results on this dataset are to be expected, as the self-made database can be seen (on average) as easier and more restrictive than the MAHNOB-HCI, due to a big part of the database consisting of videos where subjects are sitting still.
In [10] the result of the method described in [8] on the MAHNOB-HCI dataset resulted in a somewhat lower mean absolute error but a similar value for the correlation. This is thought to be attributed to the fact that in this thesis a somewhat smaller subset of videos of the MAHNOB-HCI dataset is used than in [10], where they do not filter out the video's containing a subject that did not consent for their recorded data being published, resulting in 527 video's. Another source for the discrepancy could be that the implementation for this thesis performed the up-sampling to 250Hz step after the filtering step, whereas it happens before the filtering in [8]. The reason this was done is that the described filter applied to a 250Hz signal (designed with either the MATLAB butter()-function or C++-code) would be unstable (resulting in a diverging output). In [18] a mean error rate of 6.61% is reported for the DCT-based method on the MAHNOB-HCI dataset, and a mean error rate of 4.65% when also track-

ing landmark points. This study has been unable to reproduce these results and the authors were not available for further explanation of their methods. Another paper [19] also reported no improvement when implementing this method with respect to the method in [8].
This study implemented and tested the performance of several published methods of motion-based HR measurement. In addition, several new methods are implemented and tested as well. These methods are new in their approach of recognizing and disregarding noisy signals using a clustering-approach of (parts of) the signals. This clustering approach allows a subject moving at some points in time or improper tracking of a few points throughout the video to be recognized and the HR determination to take this into account. Another area of innovation of these new methods lies in the calculation of the heart rate given a certain signal by using its peaks.
The Pan-Tompkins algorithm might not have given fully accurate results for the ground truth HR, as can be seen for example in the results on subject 23's ECG (due to the subjects' T-elevation) but other errors might have been missed. This uncertainty in the ground truth HR obviously translates itself into uncertainty in the eventual results of the methods. A Lilliefors test revealed the ground truth HR of both datasets to not be normally distributed. However, the Pearson correlation coefficient was still calculated using the student's t distribution, as this correlation coefficient was mainly used in order to be able to compare results to [10]. In the peak-detection method, the HR-period is calculated as the mean time between the peaks of the selected signal. A method that would perhaps give more robust results is using the mode of the time between peaks (with an appropriate bin size). This would make the result insensitive to a few peaks that are cause by noise. The results of the k-means (no-cut, min-norm) method with Fourier HR-detection results in the best performance of all the tested methods, suggesting that the method of noise-reduction through recognition and discarding of noisy parts of a signal might be a useful approach to the problem of HR recognition using head-motions in realistic human-computer interaction scenarios. However, the performance of motion-based methods remains worse than that of color-based methods [10]. The results suggest that decoupling motion attributed to cardiac activity from the rest of the head movement remains a very challenging issue. In the raw tracked signals, there is often no clear HR-movement discernable. This does not prove there is none present, of course, but at the least it shows that it could be quite difficult to actually measure this signal using a one-size-fits-all methodology. During development, several videos of subjects purposely moving their head up and down with a certain frequency were used to verify that the algorithm picked up on periodical movement like this. The accurate results on these video's give rise to the thought that perhaps the video quality or imperfect tracking are the main limiting factors and that the tested videos' quality is too low to pick up on the small amplitude of the movement caused by the heart.

6.1.1 Recommendations for future work

The influence of the quality of the video and the tracking used on the accuracy of motion-based HR detection methods is not yet known but would be important in eventual applications. An interesting application for future research is using this method to better determine the relation between HR, HR variability (HRV) and emotional responses (when the subjects are uninhibited by HR measurement devices). Then, this could be used to improve the determination of either one. The MAHNOB-HCI database could also be used for this, because after each session the subjects answered questions about which emotion they felt, the arousal, valence and control they felt and the predictability that was experienced. These things were all graded on a scale of 1 to 9, 9 being the most intense.
After advances in color-based and motion-based HR-detection, it might be interesting to repeat an experiment like [19], in which both methods are used and their results are combined using a Bayesian approach. If the error of these methods is low enough and results are consistent, it can be applied in numerous ways: non-invasive long-term monitoring of patients health at home, monitoring of HRV to identify suspicious people at airports or other public places to enhance security (however, the MAHNOB-HCI EULA [38] contains a clause forbidding the use of it to develop governmental systems used in public spaces) or using HRV as a measure of excitement during experiments or (computer) games [39].
Another interesting avenue of research would be to train a method to recognize the instantaneous times at which the heart beats take place, using only the visual information. Because the MAHNOB-HCI database contains ECG-information, the times of the QRS-complexes could be determined from them to research the 'lag' between a heart beat and the corresponding head movement. Together with more research on the morphology of the movement signal caused by cardiac activity, this could be viable.

6.2 Conclusion

In this thesis, novel methods were proposed to estimate a subject's HR using only the movement of the head in a video. Of all the methods tested, a method relying on unsupervised detection and filtering of noisy motions (the no-cut k-means method) using a Fourier-transform based HR-detection on the resulting signal seems the most promising method to estimate HR from head motions. Its performance on the MAHNOB-HCI dataset modestly improved upon [8], but on the dataset made for this thesis, promising results are obtained. The filtering out of the most noisy points seems like an effective approach. The peak-detection method is interesting theoretically, but there remain some changes to be done to it for it to outperform the Fourier method. These novel methods are more robust to movement and facial expressions than other systems that measure heart rate based on head motions. This opens up the possibility of more accurate heart

rate measurements in realistic human-computer interaction scenarios.

Appendix A

Code documentation

The code (released under the GNU license) can be found on GitHub [40]. The scripts to work with the MAHNOB-HCI dataset require MATLAB/Octave. To run the MATLAB scripts, the biosig-toolbox (included in the repository) has to be installed by running biosig_install.m. This toolbox is used to read data from the bdf-files containing the ECG measurements in het MAHNOB-HCI dataset. After this, determine_MAHNOB_HR.m can be used to run the algorithm on all MAHNOB-HCI videos that are stored in a folder defined near the top of the script. The folder should contain a subfolder for each recording-session that has to be analyzed and the subfolders' title has to be a number (ideally the MAHNOB-HCI session number). No other folders or files should be present in this folder. In the subfolders there should be a bdf-file containing the ECG recording and a color or grayscale avi or mp4-file of a recording of the front of a subjects' head. After running the script, the 'results'-variable stores the results of each video in a separate row containing the session no., ECG-HR and results for several types of algorithms. To run the algorithm for each video in the database made for this thesis, process_selfmade_DB.m can be used.

A.1 C++ code

It is also possible to run the C++-executable for a single video-file. This executable is located in pulsefromheadmotion-master/bin/Release/pfhmain.exe, with a shortcut in pulsefromheadmotion-master/pfhmain.lnk. This file can be called from the pulsefromheadmotion-master folder with the video-file as a first argument, a video-id as an optional second argument and an optional flag as a third argument that should be set to 1 if the tracking should not be performed again but only the signal-processing is required. This will use the tracked signals that are stored in output/rawSignals.csv after each time the tracking is performed. So in effect this will shorten processing time when running the algorithm on the same video multiple times. After execution, a line containing the session-id, Fourier results and DCT results will be appended to /output/cpp_results.csv.

Retrieving the peak-detection results requires running the MATLAB script, which will calculate it starting from the processed signals (which are made by the C++-code) in output/PCA_signals.csv. After tracking, the executable will save the time-series of the GFT-points in output/GFT_signals.csv, the time-series of the LM-points in output/landmarkpoints.csv and the time-series of both points in output/rawSignals.csv. This makes it easier to test the effect of only using one of the type of points.
Please keep in mind there can be problems with OpenCV automatically recognizing the videos' frame-rate (giving erroneous results), so it is currently required to set this manually near the top of pfhmain.cpp and re-build the C++-code. The C++ code is currently built with Microsoft Visual Studio 2015 Community and the solution consists of three projects:

- pfhmain: containing the main()-function and the tracking code
- pfhlib: containing the code to process the signals and calculate the HR
- Intraface: containing the code for the IntraFace tracking function

A.2 Unused videos

In the dataset made for this thesis, some of the video-files are not used when determining the effectiveness of the methods. The reasons for these range from very blurry video, to videos in which persons are deliberately bobbing their head in a certain frequency for testing purposes. These are the video's with id numbers: 5, 8, 9, 10, 20, 21 and 22.

Bibliography

[1] N. Hjortskov, D. Rissén, A. K. Blangsted, N. Fallentin, U. Lundberg, and K. Søgaard, "The effect of mental stress on heart rate variability and blood pressure during computer work," *European journal of applied physiology*, vol. 92, no. 1-2, pp. 84–89, 2004.

[2] E. Kristal-Boneh, M. Raifel, P. Froom, and J. Ribak, "Heart rate variability in health and disease," *Scandinavian journal of work, environment & health*, pp. 85–95, 1995.

[3] M. T. Jensen, P. Suadicani, H. O. Hein, and F. Gyntelberg, "Elevated resting heart rate, physical fitness and all-cause mortality: a 16-year follow-up in the copenhagen male study," *Heart*, vol. 99, no. 12, pp. 882–887, 2013.

[4] "Target heart rates." `www.heart.org`. Accessed: 2016-06-07.

[5] A. C. Guyton and J. E. Hall, *Textbook of Medical Physiology*. Elsevier, 11 ed., 2006.

[6] "Wikipedia - Electrocardiography." `https://upload.wikimedia.org/wikipedia/commons/9/9e/SinusRhythmLabels.svg`. Accessed: 2016-06-07.

[7] M.-Z. Poh, D. J. McDuff, and R. W. Picard, "Non-contact, automated cardiac pulse measurements using video imaging and blind source separation.," *Optics express*, vol. 18, no. 10, pp. 10762–10774, 2010.

[8] G. Balakrishnan, F. Durand, and J. Guttag, "Detecting pulse from head motions in video," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3430–3437, 2013.

[9] M. Soleymani, J. Lichtenauer, T. Pun, and M. Pantic, "A multimodal database for affect recognition and implicit tagging," *Affective Computing, IEEE Transactions on*, vol. 3, no. 1, pp. 42–55, 2012.

[10] X. Li, J. Chen, G. Zhao, and M. Pietikainen, "Remote heart rate measurement from face videos under realistic situations," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4264–4271, 2014.

[11] X. Zhu, W. Chen, T. Nemoto, Y. Kanemitsu, K. Kitamura, K.-i. Yamakoshi, and D. Wei, "Real-time monitoring of respiration rhythm and pulse rate during sleep," *Biomedical Engineering, IEEE Transactions on*, vol. 53, no. 12, pp. 2553–2563, 2006.

[12] T. Gault and A. Farag, "A fully automatic method to extract the heart rate from thermal video," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 336–341, 2013.

[13] S. Sathyanarayana, R. K. Satzoda, S. Sathyanarayana, and S. Thambipillai, "Vision-based patient monitoring: a comprehensive review of algorithms and technologies," *Journal of Ambient Intelligence and Humanized Computing*, pp. 1–27, 2015.

[14] W. Wang, S. Stuijk, and G. de Haan, "Exploiting spatial redundancy of image sensor for motion robust rppg," *Biomedical Engineering, IEEE Transactions on*, vol. 62, no. 2, pp. 415–425, 2015.

[15] M. A. Haque, K. Nasrollahi, and T. B. Moeslund, "Heartbeat signal from facial video for biometric recognition," in *Image Analysis*, pp. 165–174, Springer, 2015.

[16] D. McDuff, S. Gontarek, and R. Picard, "Remote measurement of cognitive stress via heart rate variability," in *Engineering in Medicine and Biology Society (EMBC), 2014 36th Annual International Conference of the IEEE*, pp. 2957–2960, IEEE, 2014.

[17] R. Irani, K. Nasrollahi, and T. B. Moeslund, "Improved pulse detection from head motions using dct," in *Computer Vision Theory and Applications (VISAPP), 2014 International Conference on*, vol. 3, pp. 118–124, IEEE, 2014.

[18] M. A. Haque, R. Irani, K. Nasrollahi, and T. B. Moeslund, "Heartbeat rate measurement from facial video," *IEEE Intelligent Systems*, 2016.

[19] C. H. Antink, H. Gao, C. Brüser, and S. Leonhardt, "Beat-to-beat heart rate estimation fusing multimodal video and sensor data," *Biomedical optics express*, vol. 6, no. 8, pp. 2895–2907, 2015.

[20] K. S. Tan, R. Saatchi, H. Elphick, and D. Burke, "Real-time vision based respiration monitoring system," in *Communication Systems Networks and Digital Signal Processing (CSNDSP), 2010 7th International Symposium on*, pp. 770–774, IEEE, 2010.

[21] J. Kroutil, A. Laposa, and M. Husak, "Respiration monitoring during sleeping," in *Proceedings of the 4th International Symposium on Applied Sciences in Biomedical and Communication Technologies*, p. 33, ACM, 2011.

[22] G. Bradski, "OpenCV library," *Dr. Dobb's Journal of Software Tools*, 2000.

[23] 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*, vol. 1, pp. I–511, IEEE, 2001.

[24] J. Shi and C. Tomasi, "Good features to track," in *Computer Vision and Pattern Recognition, 1994. Proceedings CVPR'94., 1994 IEEE Computer Society Conference on*, pp. 593–600, IEEE, 1994.

[25] B. D. Lucas, T. Kanade, *et al.*, "An iterative image registration technique with an application to stereo vision.," in *IJCAI*, vol. 81, pp. 674–679, 1981.

[26] X. Xiong and F. Torre, "Supervised descent method and its applications to face alignment," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 532–539, 2013.

[27] A. Ultsch, *U*-matrix: a tool to visualize clusters in high dimensional data*. Fachbereich Mathematik und Informatik Marburg, 2003.

[28] Y. Freund and R. E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," *Journal of computer and system sciences*, vol. 55, no. 1, pp. 119–139, 1997.

[29] C. Tomasi and T. Kanade, *Detection and tracking of point features*. School of Computer Science, Carnegie Mellon Univ. Pittsburgh, 1991.

[30] S. Butterworth, "On the theory of filter amplifiers," *Wireless Engineer*, vol. 7, no. 6, pp. 536–541, 1930.

[31] G. L. Matthaei, L. Young, and E. M. T. Jones, *Microwave filters, impedance-matching networks, and coupling structures*. Artech house, 1980.

[32] P. Van Fleet, *Discrete Wavelet Transformations: An Elementary Approach with Applications*. Wiley, 2008.

[33] T. Kohonen, "Self-organized formation of topologically correct feature maps," *Biological cybernetics*, vol. 43, no. 1, pp. 59–69, 1982.

[34] K. Person, "On lines and planes of closest fit to system of points in space. philiosophical magazine, 2, 559-572," 1901.

[35] I. Jolliffe, *Principal Component Analysis*. Springer, 2 ed., 2002.

[36] J. Pan and W. J. Tompkins, "A real-time QRS detection algorithm," *Biomedical Engineering, IEEE Transactions on*, no. 3, pp. 230–236, 1985.

[37] K. Pearson, "Note on regression and inheritance in the case of two parents," *Proceedings of the Royal Society of London*, vol. 58, pp. 240–242, 1895.

[38] "MAHNOB-HCI End User License Agreement." `http://mahnob-db.eu/hci-tagging/media/doc/eula.pdf`. Accessed: 2016-05-27.

[39] R. D. Lane, K. McRae, E. M. Reiman, K. Chen, G. L. Ahern, and J. F. Thayer, "Neural correlates of heart rate variability during emotion," *Neuroimage*, vol. 44, no. 1, pp. 213–222, 2009.

[40] "GitHub - Pulse from Head Motions." `https://github.com/jbukala/pulse-head-motions`. Accessed: 2016-06-07.