Adaptive Gaussian Mixture Model Driven Level Set Segmentation for Remote Pulse Rate Detection

Alexander Woyczyk, Vincent Fleischhauer, and Sebastian Zaunseder

***Abstract*—This paper presents an approach for pulse rate extraction from videos. The core of the presented approach is a novel method to segment and track a suitable region of interest (ROI). The proposed method combines level sets with subject-individual Gaussian Mixture Models to yield a time varying ROI. The ROI builds up from multiple homogeneous skin areas under constraints regarding the area and contour length of the ROI. Together with state of the art signal processing methods our approach yields an Mean Average Error (MAE) of 2.3 bpm, 1.4 bpm and 2.7 bpm on own data, the PURE database and the UBFC-rPPG database, respectively. Therewith, our method performs equal or better compared to widely used approaches (e.g. the KLT tracker instead of the proposed image processing yields an MAE of 2.6 bpm, 2.6 bpm and 4.4 bpm). Such results and the 2nd place with a MAE of 7.92 bpm in the 1st Challenge on Remote Physiological Signal Sensing prove the applicability of the proposed method. The taken approach, however, bears further potential for optimization in the context of photoplethysmography imaging and should be transferable to other segmentation tasks as well.**

***Index Terms*—Active contours, bayesian, biomedical informatics, biomedical monitoring, biomedical signal processing, heart rate, image segmentation, level set, object detection, object segmentation, PPGI, pulse rate.**

I. Introduction

PHOTOPLETHYSMOGRAPHY imaging is a non-contact method for acquisition of photophlethysmography (PPG) signals. As conventional PPG, photoplethysmography imaging (PPGI) relies on the absorption of light within tissue and blood vessels. Light encountering skin tissue is partly reflected at the surface and partially transmits through the tissue, where it is scattered and absorbed [1]. A part of the scattered light is re-emitted and using a photo-detector, it is possible to measure the amount of light originating from the skin surface (i.e. reflected and scattered light). The recorded light intensity features slight variations due to local blood volume changes induced by the heart activity [2]. PPGI omits the use of contact equipment (consisting of light source and photo-detector) in favour of cameras located at a distance to the patient. The remote acquisition has the advantage of avoiding bruises from ear or finger clips and can be used despite of injured skin. Also it requires less disinfecting and grants the patient more freedom of movement.

While PPGI features multiple physiological information, the pulse rate and pulse rate extraction is most often addressed [3]. Early work of Humphreys *et al.* demonstrated the extraction of physiological signals via camera using discrete light sources [4]. Verkruysse *et al.* laid another milestone regarding the remote measurement of PPG signals, showing that pulse signals can be obtained using ambient light as well [5]. Their observations confirmed that intensity variations in videos correspond to the PPG. Since such early works, several modifications and improvements have been presented. Today's common procedure to extract the pulse rate from videos consists of four steps, namely image processing, i.e. segmentation of a region of interest (ROI) and its tracking, pulse signal formation, e.g. the combination of color channels or fusion of spatial information, signal processing, i.e. filtering, and pulse rate extraction. All steps are relevant and only a proper combination of them guarantees a high performance. During recent years, huge progress has been made in pulse signal formation, signal processing and pulse rate extraction. Particularly, the combination of color channels, e.g. by independent component analysis (ICA) [6] or by methods like CHROM and POS [7], [8], can have enormous (positive) impact on the performance [9].

As fewer works direct at image and video processing, our work has its focus on such aspects. We propose a novel method to segment and track a suitable (ROI). The proposed method combines subject individual Gaussian mixture model (GMM) with a level set formulation to yield a time varying ROI. This work builds upon the ideas of our contribution that won the second place at the 1st RePSS challenge [10]. We refined our initial approach and conducted intensive tests on different data.

The remainder of the work is structured as follows. Section II gives an overview on the most widespread image processing approaches for pulse rate extraction from videos. Section III provides some background to Gaussian Mixture Models and Level sets. Section IV describes the novel method for

Manuscript received July 28, 2020; revised December 16, 2020; accepted January 18, 2021. Date of publication January 26, 2021; date of current version May 11, 2021. This work was funded by Deutsche Forschungsgemeinschaft (DFG, German Research Foundation), project 401786308. *(Corresponding author: Alexander Woyczyk.)*

The authors are with the Faculty of Information Technology, University of Applied Scieneces and Arts Dortmund, 44139 Dortmund, Germany (e-mail: alexander.woyczyk@fh-dortmund.de; vincent.fleischhauer@fh-dortmund.de; sebastian.zaunseder@fh-dortmund.de).

Digital Object Identifier 10.1109/JBHI.2021.3054779

segmentation and tracking of a ROI. Section V describes the used data and evaluation strategy. Sections VI to VIII give results, discuss them and draw final conclusions.

II. Overview on Image and Video Processing for PPGI

As stated before, PPGI invokes multiple steps. In the following we provide a short overview on the most important approaches to image processing for PPGI.

Though PPGI applies to all skin regions, the vast majority of works rely on the face as it features several advantages over other skin areas. Firstly, most of the time it is not covered by clothes. Secondly, it has a large skin area and good perfusion, which is essential to the extraction of a pulse signal. And thirdly, it features prominent landmarks and other features, which facilitate automatic detection [11]. One basic approach (but the most often employed) is the use of a cascade classifier to detect the face as described by Viola and Jones [12]. The algorithm uses Haar-like image filters to recognize facial features and results in a bounding box indicating its position and size. Pulse signals for every color channel are then extracted by averaging over the image data of the face bounding box [13]. A major drawback of this method is its limitation to predefined viewing angles on the face. In order to deal with rotation as well as movement of the subject, a very common solution is the usage of a point tracking algorithm, e.g. Kanade-Lucas-Tomasi Tracker [14], to subsequently move the bounding box in accordance to subject movement [15], [16]. A more detailed ROI can be achieved by the usage of landmark detectors. A set of facial landmarks, e.g. eyes, nose, mouth and jaw line, are detected, and those landmarks are either used as anchor points for an ellipsoid ROI [17] or used to define the outer corners of a polygon which serves as ROI [18].

Face detection methods allow to focus on relevant areas and reduce the number of background pixels. Additionally using re-detection or tracking enables reduction of movement related signal distortions. However, those methods are incapable of detecting occluded skin areas, e.g. by hair or glasses. Therefore, a combination of the above methods with skin detection can be a solution. Skin detection can be achieved by using color thresholds [19] or using more generic methods like Bayesian skin detectors based on a probability matrix for all RGB values [20]. Labelling detected skin pixels, this procedure further reduces the number of pixels used for signal extraction. Its disadvantage is to rely on a skin model, which has to be general enough to detect all kinds of skin tones under varying lighting conditions, while at the same time being specific in order to limit false positives on non-skin areas. Another approach to a refined ROI uses so-called superpixels, i.e. subregions of an image. Superpixels can be formed within a previously defined ROI, e.g. a bounding box, or applied to a full frame. The selection or weighting of those superpixels to be used typically invokes some signal processing, e.g. the evaluation of its pulsatile character [21]. Po *et al.* use block division of the face bounding box and a SNR based quality measure for each block to form an adaptive ROI [22].

Although the aforementioned methods and their combination can yield good results, they have some limitations. As stated before, using bounding boxes includes non-skin areas. An additional skin classification is helpful. Skin classifiers, however, typically do not make use of spatial dependencies and discard available information, though recent research targeted this limitation through the usage of convolutional neural network (CNN) for skin segmentation [23].

In recent development CNN are also used to directly derive physiological signals, e.g. Špetlík *et al.* use a CNN to derive the heart rate [24] from video data. Other techniques suffer from (partial) occlusion. Owing to such limitations, Trumpp *et al.* [25] proposed a method, which exploits color information and spatial information. The method uses a level set formulation to segment the skin area. The level set is evolved using a Gaussian distribution for the skin and background RGB values. The method shows good results but also has limitations. Firstly, it relies on univariate models, i.e. considers each color channel independently. The interaction of color channels, and therewith the skin signature, is not used. Secondly, modelling of foreground and background relies on single Gaussians. In case of mixed backgrounds (which might often be the case) or foregrounds (e.g. in case of inhomogeneous illumination) this can, but does not have to, lead to faulty segmentations.

This contribution aims to overcome such limitations. Compared to [25] we adhere to the idea of using level sets but we introduce a novel formulation, which allows to include multivariate modelling through mixtures of Gaussians. To the best of our knowledge the method of combining level sets with Gaussian Mixture Models in RGB space has not been explored widely. Though Soffientini *et al.* use a GMM driven level set on grayscale PET data [26], this is the first time, that a GMM based level set is formulated in RGB space in order to segment patches of similar textures, e.g. skin areas.

III. Methods

For our method we combine GMM and level set/active contour for adaptive ROI segmentation. We support the level set with probabilistic foreground and background models (i.e. GMM). Our approach therefore contains four components: First, we use unsupervised training of the GMM to model foreground (skin) and background by using the starting frame of a video sequence. Second, GMM are used to assign fore- and background probabilities to each pixel. Third, using level set, we combine these probabilities with additional constraints regarding area and contour length to yield a smooth area, which is used as ROI. Fourth, we initialize the ROI for the next frame with the previously ROI and repeat steps two and three for the new frame. The following description thus will provide basic information on GMM and level set and afterwards goes into detail of the concrete algorithm.

A. Gaussian Mixture Models

GMM are used to approximate unknown distributions, e.g. histograms of color distributions. Using these models as class description it is possible to predict the probability of class affiliation of an observation, e.g. color value of a pixel. GMM model the data distribution by the combination of multiple

Gaussian distributions. The single Gaussians are entitled as components of the GMM. Using the notation $\mathcal{N}(\mu, \sigma^2)$ for a Gaussian distribution with mean μ and standard deviation σ, the approximated data distribution is

$$\sum_{k=1}^{K} \pi_k \mathcal{N}(\mu_k, \sigma_k^2) \tag{1}$$

where K denotes the number of Gaussians and π_k each Gaussian's weight, according to its prior probability. For use in higher dimensional spaces, as the utilized RGB color space, it is necessary to expand the Gaussians to m dimensions (i.e. 3-dimensional space for RGB), making $\boldsymbol{\mu}$ a point in m-dimensional space and σ^2 expanding to the $m \times m$ covariance matrix $\boldsymbol{\Sigma}$. Using the above definitions and Bayes' theorem, the posterior probability of an observation $\mathbf{u}$, e.g. the observed pixel color in RGB space, belonging to any component k can be calculated by

$$p(z_k = 1|\mathbf{u}) = \pi_k \cdot \frac{\mathcal{N}(\mathbf{u}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_{o=1}^{K} \pi_o \cdot \mathcal{N}(\mathbf{u}|\boldsymbol{\mu}_o, \boldsymbol{\Sigma}_o)} \tag{2}$$

where $z_k = 1$ indicates $\mathbf{u}$ belonging to component k.

To use a GMM, its parameters first have to be determined by fitting the model to a sample distribution. Therefore, the number of components is set to a fixed size, either by prior knowledge or an assumption about the expected distribution. The GMM itself is fitted to the sample data $\mathrm{U} = \{\mathbf{u}_1 \ ... \ \mathbf{u}_L\}$ using the expectation-maximization (EM) algorithm. The EM algorithm alternates between an expectation and a maximization step. The expectation step is used to categorize the sample data, whereas the maximization step modifies the models parameters. A practical way to get an initial estimation for each GMM component is achieved by the K-means algorithm [27]. Using the estimations for each component and (2), each observation is assigned an expectation for each component. Using the expectations, the maximization step modifies the parameters for every component to better fit the observed data. For better readability, the posterior probability $p(z_k = 1|\mathbf{u}_l)$ will further be substituted by $\gamma_{l,k}$. The weights π_k, means $\boldsymbol{\mu}_k$ and covariance $\boldsymbol{\Sigma}_k$, respectively, are updated by

$$\pi_k{}' = \frac{1}{L} \sum_{l=1}^{L} \gamma_{l,k} \tag{3}$$

$$\boldsymbol{\mu}_k{}' = \frac{\sum_{l=1}^{L} \gamma_{l,k} \mathbf{u}_l}{\sum_{l=1}^{L} \gamma_{l,k}} \tag{4}$$

$$\boldsymbol{\Sigma}_k{}' = \frac{\sum_{l=1}^{L} \gamma_{l,k} (\mathbf{u}_l - \boldsymbol{\mu}_l{}')(\mathbf{u}_l - \boldsymbol{\mu}_l{}')^T}{\sum_{l=1}^{L} \gamma_{l,k}}. \tag{5}$$

This process leads to a local maximum of the log-likelihood function of the entire model, which is given by

$$\ln p(U) = \sum_{l=1}^{L} \ln \sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{u}_l|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k). \tag{6}$$

The posterior probability of a sample $\mathbf{u}$ belonging to the model can then be calculated using the following formula

$$p(\mathbf{u}) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{u}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_K). \tag{7}$$

B. Level Set Contour Description

Level sets are able to produce a segmentation based on a contour description. The contour is described by an energy function, which is approximated based on different aspects, as image data, size or shape of the contour. They hereby implicitly represent a contour in the two dimensional image space [28]. This is achieved by defining a continuous function over the image space, which integrates image data and smoothness. Using a threshold plane the contour is described by the intersection of the plane with the energy function so that each point can be distinctly assigned to the outer or inner side of the contour based on its energy value. Combining the level set method with an active contours approach, the contour can be evolved to satisfy smoothness or image data related constraints. The contour is therefore evolved over a series of iterations or time steps t_s. Since the level set function ϕ is defined for every position $\mathbf{x}$ within the image's x-y plane, the contour at each time step is given with all points where $\phi(\mathbf{x}, t_s) = \tau$ for threshold τ [29]. The inside of the contour is then implicitly defined by all points where $\phi(\mathbf{x}, t_s) > \tau$. Every point is either inside, outside or part of the boundary according to its current ϕ value. This definition of a contour is beneficial as it does not have to deal with self intersections of the contour.

Chan and Vese presented one fundamental approach of a level set function in 1999 by describing the evolution of the contour with an energy minimization problem [30]. They formulate the total energy F of a contour C and its enclosed area $\tilde{C}$ within the image Ω by

$$\begin{aligned} F(C, c_1, c_2) = \kappa \cdot \oint_C \mathrm{d}r + \nu \cdot \iint_{\tilde{C}} \mathrm{d}x\mathrm{d}y \\ + \lambda_1 \cdot \iint_{\tilde{C}} |\mathbf{u}(x,y) - c_1|^2 dxdy \\ + \lambda_2 \cdot \iint_{\Omega \setminus \tilde{C}} |\mathbf{u}(x,y) - c_2|^2 dxdy. \end{aligned} \tag{8}$$

The energy consists of the length of the contour, the area it encloses and as image data property the distance of every pixels intensity $\mathbf{u}(x, y)$ to its regions average intensity c_1 (for the inside) or c_2 (for the outside). The factors κ, ν, λ_1, and λ_2 are weights for the different aspects of the total energy. A large κ penalizes high curvature and thus results in a smoother contour. ν is used to penalize the area enclosed by the contour. Additionally ν determines if the enclosed area has a stronger force to shrink (ν used with positive sign) or to grow (ν used with negative sign). Chan and Vese derive, through discretisation and linearisation, a numeric formulation to update the ϕ function at position (i, j) with i, j being the discrete pixel positions, fictional space and time steps h and Δt, n being the index of fictional time steps

and δ_h an approximation of the dirac function [31].

$$\frac{\phi_{i,j}^{n+1}-\phi_{i,j}^{n}}{\Delta t} = \delta_h(\phi_{i,j}^n)\cdot\left[\frac{\kappa}{h^2}\Delta_-^x\left(\frac{\Delta_+^x\phi_{i,j}^{n+1}}{\sqrt{\frac{(\Delta_+^x\phi_{i,j}^n)^2}{h^2}+\frac{(\phi_{i,j+1}^n-\phi_{i,j-1}^n)^2}{(2\tilde{}h)^2}}}\right)+\frac{\kappa}{h^2}\Delta_-^y\left(\frac{\Delta_+^y\phi_{i,j}^{n+1}}{\sqrt{\frac{(\phi_{i+1,j}^n-\phi_{i-1,j}^n)^2}{(2\tilde{}h)^2}+\frac{(\Delta_+^y\phi_{i,j}^n)^2}{h^2}}}\right)-\nu-\lambda_1(\mathbf{u}(i,j)-c_1(\phi^n))^2+\lambda_2(\mathbf{u}(i,j)-c_2(\phi^n))^2\right] \quad (9)$$

Because of computation time and discrete space steps based on the pixel level, gradients in (9) are only calculated on a local neighbourhood and therefore are substituted by the finite differences given in (10).

$$\Delta_-^x\phi_{i,j}=\phi_{i,j}-\phi_{i-1,j},\quad \Delta_+^x\phi_{i,j}=\phi_{i+1,j}-\phi_{i,j},$$
$$\Delta_-^y\phi_{i,j}=\phi_{i,j}-\phi_{i,j-1},\quad \Delta_+^y\phi_{i,j}=\phi_{i,j+1}-\phi_{i,j} \quad (10)$$

IV. ROI Segmentation by Level Set and GMM

A. Algorithmic Approach

The proposed algorithm for ROI segmentation combines level set and GMM. The original level set approach by Chan and Vese is limited to separation between two mean values in monochromatic images (i.e. mean foreground intensity and mean background intensity). This limitation may lead to difficulties when the background's intensity distribution overlaps or envelopes the foreground's distribution. In order to deal with colored images and heterogeneous backgrounds, we employ GMMs that model the color distribution of fore- and background (i.e. skin and non skin) instead of a mean intensity values. These GMMs are formed of multiple kernels in order to be able to model various color distributions. Since color distributions for the face region often show multi-variate distributions (e.g. due to shadows or non-uniform illumination), the skin-GMM was initially set to three kernels. Within this contribution the GMMs are trained for each video sequence individually and the non-skin kernel number remains fixed for all sequences and data sets, whereas the number of skin kernels is automatically adapted to each subject or video sequence. The proposed algorithm thus features two steps, namely **GMM initialization** and **ROI segmentation**, which are detailed in the next sections. The algorithm was implemented in C++ using OpenCV [27].

B. Initialization of GMMs

The initialization features three stages, face detection, setting up model parameters and model training. Using the first frame, the subject's face is detected by a cascade classifier. The presented algorithm uses the pre-trained Haar cascade distributed with OpenCV to detect faces [27]. Pixels from the resulting bounding box are then used to train the skin GMM. In order to eliminate kernels representing non-skin color from the skin GMM, each kernel is evaluated by combining the kernel's probability distribution with the probability distribution of the skin classifier, p_{SkinCL}, described by Jones and Rehg's skin model [32]. The combined weight s_k of each kernel k is given by

$$s_k=\frac{\sum_{r=0}^{R}\sum_{g=0}^{G}\sum_{b=0}^{B}p_{\text{SkinCL}}\left(\begin{pmatrix}r\\g\\b\end{pmatrix}\right)\cdot\mathcal{N}\left(\begin{pmatrix}r\\g\\b\end{pmatrix}|\boldsymbol{\mu}_k,\boldsymbol{\Sigma}_k\right)}{\sum_{r=0}^{R}\sum_{g=0}^{G}\sum_{b=0}^{B}\mathcal{N}\left(\begin{pmatrix}r\\g\\b\end{pmatrix}|\boldsymbol{\mu}_k,\boldsymbol{\Sigma}_k\right)}. \quad (11)$$

Because of the 3-dimensional RGB space, pixel intensities are denoted as 3-dimensional vectors $\mathbf{u}=(rgb)$, holding intensity values for every color channel. Using a decision threshold Θ, a kernel k is deactivated, if $s_k<\Theta\cdot\max(s_1,\ldots,s_K)$. This step dynamically reduces the number of actual used skin kernels based on the image data. Practically, if a certain kernel is far from the skin model, this will lead to exclusion of that kernel. For our tests, Θ was set to 0.3 (as general purpose parameter). Lastly, the weights of the remaining kernels are adjusted to reflect the new model with reduced kernels, i.e. weights of the remaining kernels are normalized to sum up to 1. After setting up the skin GMM, the pixels outside of the face bounding box are used to train the non-skin GMM. Training of both GMMs is performed by using the EM algorithm as described in Section III-A.

C. Segmentation of ROI

To segment the ROI, we modified the level set function as described by Chan and Vese in order to incorporate both skin and non-skin GMMs. To that end, the data terms $(\mathbf{u}(i,j)-c_1(\phi^n))^2$ and $(\mathbf{u}(i,j)-c_2(\phi^n))^2$ in (9) are substituted by data terms based on both GMM models. Instead of the Euclidean distance between a pixel and the average intensity of its class, we use GMM based distances for the pixels RGB value. They are defined by

$$dist_{\text{skin}}(\mathbf{u})=\frac{p_{\text{skin}}(\mathbf{u})}{p_{\text{skin}}(\mathbf{u})+p_{\text{nonSkin}}(\mathbf{u})} \quad (12)$$

for all pixels assigned to the skin region and

$$dist_{\text{nonSkin}}(\mathbf{u})=\frac{p_{\text{nonSkin}}(\mathbf{u})}{p_{\text{skin}}(\mathbf{u})+p_{\text{nonSkin}}(\mathbf{u})} \quad (13)$$

for all background pixels. As described in (7), the posterior probability p_{skin} (and p_{nonSkin}) is the cumulative posterior over all components of the according GMM.

Since the distance function might yield high values when both $p_{\text{skin}}(\mathbf{u})$ and $p_{\text{nonSkin}}(\mathbf{u})$ are small, an additional scaling parameter is added to each distance measure according to

$$w_{\text{skin}}(\mathbf{u})=-\frac{1}{\log(p_{\text{skin}}(\mathbf{u}))} \quad (14)$$

and for the non-skin distance

$$w_{\text{nonSkin}}(\mathbf{u}) = -\frac{1}{\log(p_{\text{nonSkin}}(\mathbf{u}))}. \tag{15}$$

Combining the above distance measures with the level set approach from (9) our iterative solution to propagate the ϕ-function reads as follows

$$\begin{aligned}\phi_{i,j}^{n+1} = \Big[&\phi_{i,j}^n + \delta_h \cdot (\kappa \cdot \Big(\phi_{i+1,j}^n \cdot divR \\ &+ \phi_{i-1,j}^n \cdot divL + \phi_{i,j+1}^n \cdot divU + \phi_{i,j-1}^n \cdot divD\Big) \\ &- \nu - \lambda_1 \cdot w_{\text{skin}}(\mathbf{u}(i,j)) \cdot dist_{\text{skin}}(\mathbf{u}(i,j)) \\ &+ \lambda_2 \cdot w_{\text{nonSkin}}(\mathbf{u}(i,j)) \cdot dist_{\text{nonSkin}}(\mathbf{u}(i,j)))\Big] \\ &\cdot \frac{1}{1 + \delta_h \cdot \kappa \cdot (divR + divL + divU + divD)}\end{aligned} \tag{16}$$

with the inverse gradients

$$\begin{aligned} divR &= \frac{1}{\sqrt{(\Delta_+^x \phi_{i,j}^n)^2 + (\frac{\phi_{i,j+1}^n - \phi_{i,j-1}^n}{2})^2}} \\ divL &= \frac{1}{\sqrt{(\Delta_-^x \phi_{i,j}^n)^2 + (\frac{\phi_{i,j+1}^n - \phi_{i,j-1}^n}{2})^2}} \\ divU &= \frac{1}{\sqrt{(\Delta_+^y \phi_{i,j}^n)^2 + (\frac{\phi_{i+1,j}^n - \phi_{i-1,j}^n}{2})^2}} \\ divD &= \frac{1}{\sqrt{(\Delta_-^y \phi_{i,j}^n)^2 + (\frac{\phi_{i+1,j}^n - \phi_{i-1,j}^n}{2})^2}}. \end{aligned} \tag{17}$$

In order to limit calculation time, two stopping criteria are formulated. The first triggers if the solution converges to a stable segmentation, whereas the second is defined as a maximum number of iteration steps. The computed segmentation is afterwards used to initialize the ϕ-function in the subsequent frame, where the iteration loop is restarted. This way the contour is allowed to adapt to the new image data.

V. Data and Evaluation Strategy

A. Video Data

The used data is composed of different, partially publicly available, datasets to include a substantial number of subjects and reflect a wide variety of experimental settings. Our analyses primarily focus on uncompressed video data. Overall, we include 145 video sequences of 74 subjects from uncompressed databases. Details for such data are given below. For our discussion, we further considered 673 video sequences of 67 subjects in compressed databases, which will be shortly presented in Section VII-B.

Cold pressure test data (CPT data): the data originates from a custom cold pressure study. Video recordings captured the subjects' faces in frontal view at a distance of approximately 1 m by an RGB camera (UI-3370CP-C-HQ, IDS). Recordings were done at a color depth of 12bit, a frame rate of 100 fps and a resolution of 420×320 pixels. Data was stored without compression but reduced to 8bit color depth and a frame rate of 25 fps before processing. The experimental setup was illuminated by ambient and a fluorescent ceiling light. The reference pulse rate was extracted from an electrocardiogram in a semi-automated manner. The experimental protocol of the study consisted of an initial resting phase followed by the CPT and another resting phase [33]. During CPT, subjects immersed their hand into cold water (4 °C). Within this contribution, we use a data segment 30 s before to 30 s after immersion of the hand into cold water. This segment is challenging because it contains subject motion together with a typical physiological reaction in pulse rate. 22 healthy subjects (age 25.5 ± 3.73 years, 10 female) participated in the study. Each participant took part twice, one time in supine position and one time in sitting position. One recording had to be discarded due to technical problems resulting in 43 video sequences of 60 s each.

PURE dataset (PURE data)[34]: the data originates from the Technical University Ilmenau, Germany and is available upon request via http://www.tu-ilmenau.de/neurob/data-sets/pulse. Video recordings captured the subjects' faces in a frontal view at a distance of about 1.1 m by a RGB camera (eco274CVGE, SVS-Vistek GmbH). Recodings where done at a color depth of 8bit, a frame rate of 30 fps and a resolution of 640×480 pixels. Each frame was stored as uncompressed jpg file. The reference pulse rate was extracted from a pulse oximeter (pulox CMS50E) in a semi-automated manner. The recordings consist of 10 subjects (8 male, 2 female) performing different instructed tasks as talking, head translation and head rotation. Each task was performed for 60 s, resulting in six 1-minute sequences per subject.

UBFC-rPPG dataset (UBFC data)[21]: the data originates from the University of Burgundy - Franche-Comté, and is available upon request [21]. Video recordings captured the subjects' faces at a distance of about 1m in frontal view by a RGB camera (Logitech C920 HD Pro). Recordings were done at a color depth of 8bit, a frame rate of 30 fps and a resolution of 640×480 pixels. Videos were stored as uncompressed files. Illumination invoked varying amounts of sunlight and indoor illumination. The reference pulse rate was extracted from a transmissive pulse oximeter in a semi-automated manner. The dataset consists of two parts, part one (labeled SIMPLE) features videos, where the participants were instructed not to move during the recording. Videos of the second part (labeled REALISTIC) show a more realistic scenario, where participants conducted an experiment. During the experiment, participants played a time sensitive mathematical game that aimed at augmenting their pulse rate while simultaneously emulating a normal human-computer interaction scenario. The experiment lasted 60 s. 49 subjects participated in the study. Three recordings were not usable due to technical problems. Of the remaining subjects, 42 agreed to provide their data for research purposes. In this publication only videos of the second, realistic, part were used.

B. Reference Methods for ROI Segmentation

We tested our method against four methods, which were frequently used in the literature. The first one, further denoted as **VJ static**, uses the face detection system described in [12], to

(a) ROI by LS-GMM (b) ROI by KLT-Skin (c) ROI by KLT (d) ROI by LM

Fig. 1. ROI of participants 37 (top) and 45 (bottom) from the UBFC-rPPG dataset. Images show, from left to right, our own LS-GMM method, the KLT tracked bounding box with skin classifier, the tracked Viola Jones bounding box and the ROI defined by landmarks.

obtain a squared bounding box containing the face. This initial ROI is kept static over the complete video sequence. In contrast to other methods, as [13], we do not reduce the horizontal size of the bounding box. The shrinking is usually applied in order to better approximate the oval shape of the face, but since we are expecting head movement and rotation in our videos we benefit from additional space to the left and right of the face.

The second reference method, denoted as **KLT**, uses the initial ROI from the VJ static algorithm. Instead of keeping the ROI at a fixed position, it is tracked over time using the Kanade-Lucas-Tomasi algorithm. Using feature points, i.e. Good-Features-to-Track [35], found within the ROI of a frame, the algorithm calculates the new position of each feature point in the next frame. A set of displacement vectors can then be calculated from each feature's old and new position. The shift of the ROI is then calculated by averaging over all displacement vectors.

The third method, denoted as **KLT-Skin**, builds upon the tracked ROI of KLT. A Bayesian skin classifier is used to label each pixel within the KLT ROI. The classifier, presented in [32], uses probability density functions for both skin and non-skin class in RGB space. Comparing the proportion of posterior skin vs. posterior non-skin probability against a threshold value results in the classification as skin or non-skin pixel. In order to gain a smoother ROI and a less noisy signal the resulting binary mask was dilated and spatial averaging over all mask pixels was applied to extract the signal value.

The fourth method, denoted as **LM** makes use of facial landmarks to define a ROI. Similar to Li *et al.* we used landmarks to segment the cheeks [36]. Landmark detection is based on the pre trained Dlib 68 landmark shape predictor [37]. The pulse signal is generated by averaging the intensity values of every pixel in both left and right cheek ROI.

Fig. 1 exemplarily shows the ROIs from all presented methods (except **VJ**, since it is a stationary version of the **KLT** bounding box). All ROIs, our own and the four reference methods, undergo the same signal processing steps to yield a pulse signal and subsequently the pulse rate from it.

C. Signal Processing

The ROI is used as base for the subsequent processing steps, namely signal extraction and pulse rate estimation.

For signal extraction multiple strategies have been described in the literature, e.g. blind source separation or model based approaches [8], [13]. We have employed various of them to test the impact of using single or combining multiple color channels. Within this contribution, we report results for the green channel and CHROM [7]. The green channel was shown to yield the highest signal quality as single channel [5], does not require further computations and is widely used. Similarly, CHROM [7] is a widely used and powerful method that serves as representative of color combination approaches. Within our tests we employed multiple methods for channel combination like CHROM and POS. Despite producing very similar results CHROM yielded the overall best performance leading to the decision to only present CHROM (using the proposed parameter setting) in our results.

Pulse rate estimation is done by a sliding window with a window length of 10 s and a step size of 1 s. The found pulse rate is assigned to the center point of the window. The extraction applies to the whole record except the first and the last 5 s to account for the used window length. We estimate the pulse rate for each window independently in order to assure that stable pulse rates do not impose a positive bias to the performance. The estimation procedure thus applies to each single window of 10 s and reads as follows. First, in order to remove trends and high frequency components originating from image noise or artefacts, the signal is band pass filtered. Filtering is implemented by applying two fifth order butterworth low-pass filters in forward and reverse direction and subtracting their outputs. The cut-off frequencies are set to 0.8 Hz and 3 Hz. Secondly, we transform the filtered signal to frequency domain using fast Fourier transform. Lastly, the frequency component having the highest amplitude within the range of 40 bpm to 180 bpm is considered as pulse rate. This procedure is equally applied to all methods of signal extraction, i.e. the green channel or CHROM.

TABLE I
PARAMETER CONFIGURATION OF THE LS-GMM METHOD USED FOR EXPERIMETS

$\lambda_1 = \lambda_2$	κ	ν	max. iterations	Θ	non-skin kernel number
10.0	0.1	0.3	100	0.3	5

VI. RESULTS

A. Evaluation Metrics

The evaluation is inspired by [38]. We use Availability, Root Mean Square Error (RMSE) and Mean Absolute Error (MAE) to assess the performance of the proposed method. The Availability $A(e)$ describes the percentage of correct pulse rate estimates in dependency on the error margin e within a measurement is considered as correct. For all N pulse rates within a dataset $A(e)$ is defined by

$$A(e) = \frac{\sum_{n=1}^{N} b_n(e)}{N} \tag{18}$$

with

$$b_n(e) = \begin{cases} 1, & \text{if } |\Delta_n| < e \\ 0, & \text{otherwise} \end{cases} \tag{19}$$

where Δ_n denotes the difference between the belonging pulse rates from the algorithm and a reference measurement according to

$$\Delta_n = \mathrm{HR}_n^{(Alg)} - \mathrm{HR}_n^{(Ref)}. \tag{20}$$

The RMSE assesses the squared differences between algorithm and reference considering the available estimates only. It is defined by

$$\mathrm{RMSE}(e) = \sqrt{\frac{\sum_{n=1}^{N} b_n(e) \cdot \Delta_n^2}{\sum_{n=1}^{N} b_n(e)}}. \tag{21}$$

The depicted measure can be used in variable ways. $A(5\text{ bpm})$, i.e. the Availability at an error margin of e = 5 bpm assesses the percentage of measurements within a margin of 5 bpm, which often serves as measure of performance. The RMSE at Availability of 100% can be considered as overall accuracy. RMSE shown over availability yields a receiver operator characteristic (ROC) curve that illustrates the interplay between both measures graphically. The ROC curve can be used to get an idea on possible tradeoffs between Availability and RMSE. Additionally we calculate the MAE at Availability of 100% based on the following definition.

$$\mathrm{MAE}(e) = \frac{\sum_{n=1}^{N} b_n(e) \cdot |\Delta_n|}{\sum_{n=1}^{N} b_n(e)} \tag{22}$$

B. Results

Fig. 2 shows the ROC curves using CPT, PURE and UBFC datasets for KLT-Skin and the proposed method. As expected, an increasing availability increases RMSE. Though the general behaviour can be found in all databases, there are large absolute differences between databases. Table II and Table III highlight absolute differences between databases by giving mean values on $A(5\text{ bpm})$ and RMSE at Availability of 100% for different methods. Fig. 3 shows Bland Altman plots and correlation coefficients of our method and KLT-Skin in comparison. Our method not only results in lower standard deviation, but also has a higher correlation coefficient in every database. All results for our method were obtained using the parameter configuration introduced in Table I, if not specified differently.

TABLE II
RESULTS ON $A(5\text{ bpm})$, i.e. PERCENTAGE OF MEASUREMENTS WITH AN ERROR LOWER THAN 5 bpm, FOR DIFFERENT DATABASES AND METHODS. BOLD LETTERS MARK THE BEST RESULTS PER COLOR CHANNEL AND DATABASE

	CPT data		PURE data		UBFC data	
	Green	CHROM	Green	CHROM	Green	CHROM
LS-GMM	**73.8**	**89.4**	**88.8**	**96.9**	78.1	**91.6**
LM	49.1	80.3	48.9	69.4	56.9	65.9
KLT-Skin	65.8	88.4	73.1	96.2	63.3	84.4
KLT	64.5	87.3	62.3	92.0	72.1	87.0
VJ static	63.7	88.2	66.7	84.5	**79.5**	87.9

TABLE III
RMSE AT $A(\infty\text{ bpm})$, i.e. 100% AVAILABILTY, FOR DIFFERENT DATABASES AND METHODS. BOLD LETTERS MARK THE BEST RESULTS PER COLOR CHANNEL AND DATABASE

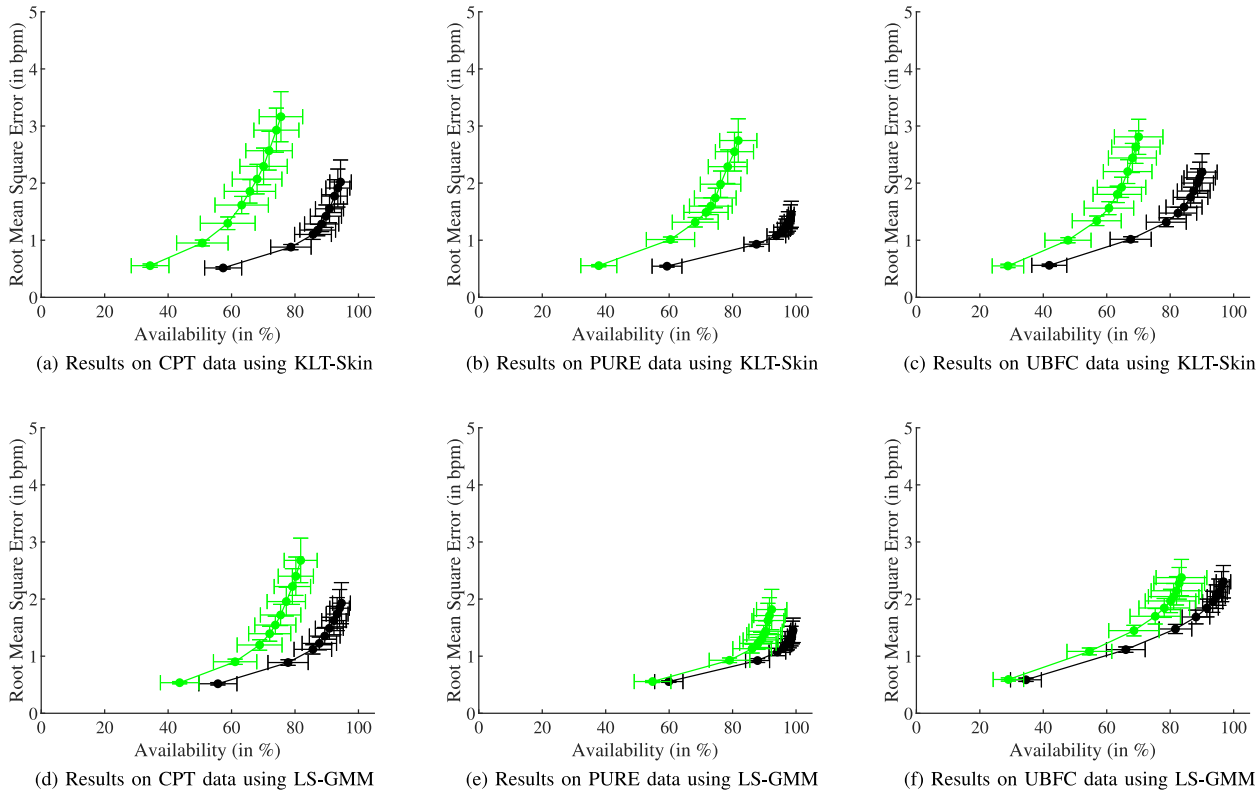
	CPT data		PURE data		UBFC data	
	Green	CHROM	Green	CHROM	Green	CHROM
LS-GMM	**8.9**	**3.9**	**6.0**	**2.3**	12.2	**4.0**
LM	15.9	7.2	18.9	11.9	19.4	14.5
KLT-Skin	10.6	4.1	11.5	2.9	16.8	6.4
KLT	10.8	4.1	12.3	4.7	14.2	6.9
VJ static	11.1	4.7	12.5	6.3	**9.6**	6.4

VII. DISCUSSION

A. Discussion on Results

Overall, our results give a highly consistent picture on the performance of the proposed method. LS-GMM performs best in all databases, selected color channels or color channel combination and according to all quality measures with only one exception, namely using the green channel in UBFC dataset (details below).

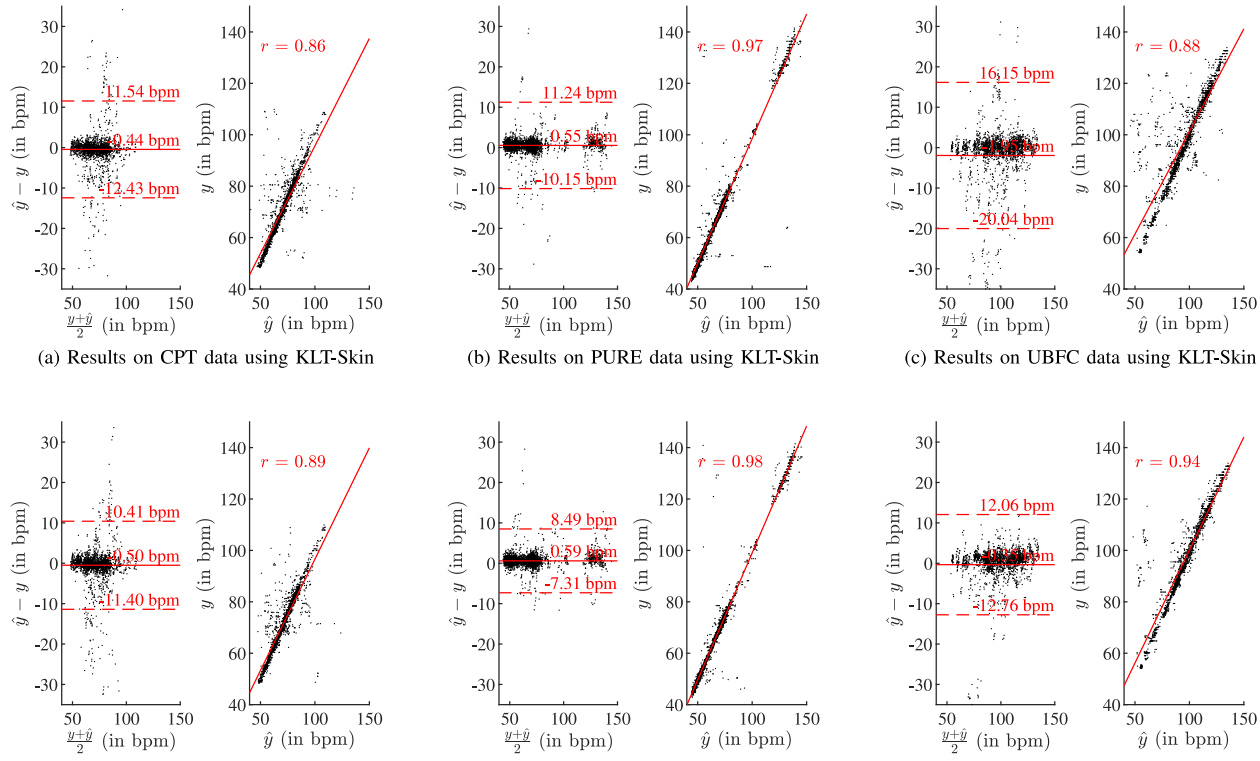
As expected, CHROM causes a significant boost of the results over the green channel. LS-GMM on average slightly improves the results compared to the other methods. The positive effect is much more pronounced on the green channel. This finding is a strong hint at the proper function of the segmentation by LS-GMM. Considering CHROM, the effect of LS-GMM is much less pronounced but still existing. This can be attributed, most importantly, to the strength of CHROM, which is able to remove distortions resulting from inaccurate ROI segmentation or tracking, respectively. Considering the UBFC-rPPG dataset, LS-GMM yields 91.6% availability at 5 bpm and outperforms results reported in other publications. Using the realistic part of UBFC-rPPG, e.g. Macwan *et al.* yield 87% precision at 5 bpm tolerance) using constrained ICA [39], Li *et al.* yield 87.6% precision at 5 bpm tolerance using weighted mask model [40] and Bobbia *et al.* report 89% using weighted superpixels [21].



(a) Results on CPT data using KLT-Skin (b) Results on PURE data using KLT-Skin (c) Results on UBFC data using KLT-Skin

(d) Results on CPT data using LS-GMM (e) Results on PURE data using LS-GMM (f) Results on UBFC data using LS-GMM

Fig. 2. ROC curves for all databases. The plots show results for the green channel (green line) and CHROM (black line) using KLT-Skin (upper row) and the LS-GMM (lower row).



(a) Results on CPT data using KLT-Skin (b) Results on PURE data using KLT-Skin (c) Results on UBFC data using KLT-Skin

(d) Results on CPT data using LS-GMM (e) Results on PURE data using LS-GMM (f) Results on UBFC data using LS-GMM

Fig. 3. Bland Altman plots and illustration of correlation for all databases. The plots show results for CHROM using KLT-Skin (upper row) and the LS-GMM (lower row).

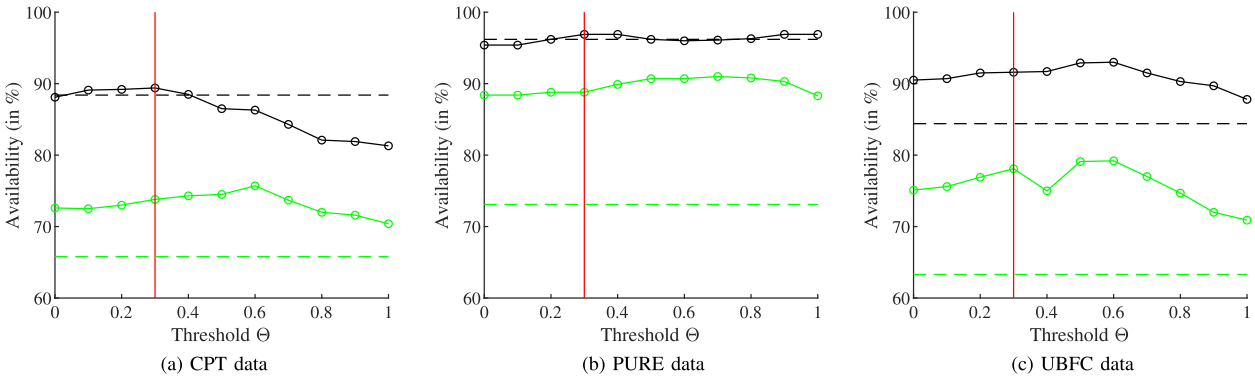


Fig. 4. Effect of the threshold Θ on the availability $A(5\text{ bpm})$ using the LS-GMM method (solid lines). Dashed lines indicate the perfomance of using KLT. Black lines indicate usage of CHROM, green lines indicate the usage of the green color channel. The red vertical line indicate the used general purpose parameters. Overall, the LS-GMM method behaves stable but has maxima at varying values. Using $\Theta = 0.6$ or higher values generally result in a slight decrease.

TABLE IV
MAE AT $A(\infty$ bpm), i.e. 100% AVAILABILTY, FOR DIFFERENT DATABASES AND METHODS. BOLD LETTERS MARK THE BEST RESULTS PER COLOR CHANNEL AND DATABASE

	CPT data		PURE data		UBFC data	
	Green	CHROM	Green	CHROM	Green	CHROM
LS-GMM	**5.5**	**2.3**	**4.1**	**1.4**	8.4	**2.7**
LM	11.2	4.4	14.7	8.9	14.3	10.8
KLT-Skin	7.1	2.6	8.0	1.6	11.4	4.0
KLT	7.3	2.6	9.5	2.6	10.0	4.4
VJ static	7.5	3.0	10.5	4.9	**7.0**	4.3

Such findings underline the efficiency of LS-GMM. When interpreting the results one has to keep in mind that the UBFC data shows only small movement, accordingly using a static ROI as VJ yields nearly as good results as other methods or even outperforms them. This may be caused by artifacts introduced by skin classification or non stable tracking points. In the CPT data movement mostly consists of short, fast head movements, while subjects are looking at their arm. Here the available area of skin changes significantly, which may explain the worse results using ROIs without skin detection as in KLT or VJ static. Lastly, the strength of LS-GMM method, being able to maintain a qualitatively superior ROI during motion and mimic, is evident, when evaluating the results on the PURE dataset, where $2/3$ of the videos show translational or rotational head motion. Our method achieves an availability of 96.9% at 5 bpm and surpasses all reference methods. With an RMSE of 2.3 it is at scale with recent CNN based approach [24], though differences in the results of CHROM in our paper and [24] (2.9 in ours vs. 2.5 in Špetlík *et al.*) suggest differences in HR evaluation.

Remarkably, we obtained the results from Table II to Table IV using *general purpose parameters* (i.e. using the fixed parameter setting from Table I for all datasets). Most importantly, this relates to the threshold Θ, which controls the number of Gaussian kernels that is actually used to model the foreground per video. It balances between using all three kernels (probably adding facial hair, eyes or mouth to the model) and using only one kernel (omitting different lighted skin patches). For the evaluation, we globally selected $\Theta = 0.3$. At $\Theta = 0$ all kernels would remain part of the GMM while $\Theta = 1$ would use a single kernel, the highest weighted, only. In order to determine the influence of an individually adjusted number of kernels for the skin GMM, the decision factor Θ was tested by gradually increasing its value by 0.1 from 0.0 to 1.0 for each dataset. The resulting ROIs were evaluated based on the availability at 5 bpm heart rate (HR). The results for different data sets can be found in fig. 4. According to fig. 4, all data sets have in common that combining multiple kernels yields better results than using only a single one. It further turns out that, though $\Theta = 0.3$ being a good tradeoff, all databases have individual maxima. In other words: LS-GMM can yield even better results than reported in Table II underlining the high potential of the method. Results should further improve if a patient-specific threshold is used. However, of course this would require a data-driven criterion to determine Θ, which we currently do not have.

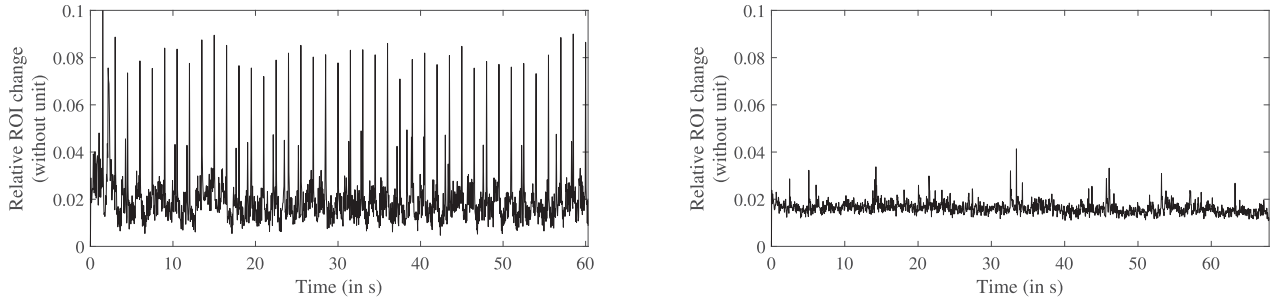
B. Compressed Datasets

Beyond the data we introduced in detail before, we considered two more datasets, namely COHFACE dataset and MANHOB-HCI Tagging dataset.

The COHFACE dataset from Idiap Research Institute (available upon request via https://www.idiap.ch/dataset/cohface) features 160 videos from 40 subjects [41]. Subjects were instructed not to move or talk during the recording. The database contains 4 videos of every subject, two under controlled lighting and two under natural conditions with ambient light.

The MANHOB-HCI Tagging dataset (available upon request at https://mahnob-db.eu/hci-tagging/) from the Imperial College London features 513 videos from 27 subjects [42]. Subjects were presented various images or short videos in order to elicit emotional responses. The recordings took place in a controlled environment.

Considering both datasets, the results, as presented in Table V to Table VII, are substantially worse compared to the results shown before. As both datasets contain compressed videos and previous research has shown that video compression degrades the results [43], [44], some deterioration was expected. The extent of deterioration, particularly regarding LS-GMM which performs best on other data, can be explained by details of the employed compression algorithm. Both datasets



(a) Relative ROI change on participant 2 - video 2 of the COHFACE dataset

(b) Relative ROI change on participant 33 of the UBFC dataset

Fig. 5. Comparison on temporal ROI changes on samples from the COHFACE and UBFC-rPPG dataset. The black line indicates the amount of different pixels between the ROIs of consecutive frames relative to the ROI size.

use the MPEG-4 file format, which omits color information for compression (Chroma subsampling at 4:2:0 Y'CbCr) and additionally uses (temporal) forward and backward prediction of image data [45].

As already described in [8], CHROM is strongly affected by the loss of color information due to compression as well. Similarly, all algorithms that rely on color information for skin classification, which applies here to LS-GMM and KLT-Skin, are heavily affected by such compression. Another problem applies specifically to the LS-GMM method. Fig. 5 shows an exemplary recording of COHFACE compared to UBFC data. The figure details the size of the adaptive ROI from LS-GMM. The first signal reveals a rhythmic adaptation of ROI. This behaviour is likely to originate from forward and backward prediction of image data and may lead to temporal static color information as well as artificial image features (i.e. edges) due to block motion compensation. As a result, the ROI is kept statically and adapted rhythmically. This rhythmicity can interfere with the pulsation and negatively affects the results. Despite such unfavorable findings, we decided to present results from COHFACE and HCI database to underline the effect of compression. Importantly, while many compression algorithms introduce mild errors, the MPEG-4 compression is critical (see also [46]). This should be kept in mind when choosing an appropriate processing method.

C. Execution Speed

With respect to real time performance our method is not yet able to process the typical 25 to 30 fps. Additionally our algorithm needs a initialization phase for each video sequence to train the subject specific GMM. This process takes an average of 21.22s on a framesize of 640×480 pixels. Table VIII shows single threaded execution times of all compared methods without initialization phase on a Intel Core i-10940X. Methods denoted with * were executed with the help of a GPU, namely NVidia Geforce GTX 1660. The LM method used the parallel implementation of the Dlib library, whereas we implemented LS-GMM ourselves.

D. Limitations

By using publicly available data and implementing previously described algorithms as reference, we want to make our research comparable to other works. In many cases we obtained even better results than previously reported using said reference algorithms. Anyway, regarding the reference algorithms there might be details, which our implementation does not capture. Particularly using facial landmarks (LM), more sophisticated schemes for their usage are feasible but out of scope of this work. Particularly the results on LM thus must be taken with caution.

TABLE V
RESULTS ON A (5 bpm), i.e. PERCENTAGE OF MEASUREMENTS WITH AN ERROR LOWER THAN 5 bpm, FOR DIFFERENT DATABASES AND METHODS. BOLD LETTERS MARK THE BEST RESULTS PER COLOR CHANNEL AND DATABASE

	COHFACE data		HCI data	
	Green	CHROM	Green	CHROM
LS-GMM	61.7	47.1	51.7	44.2
LM	41.9	42.7	41.3	42.7
KLT-Skin	26.2	34.8	23.9	26.3
KLT	74.1	53.0	59.0	47.9
VJ static	**76.9**	**60.9**	**60.4**	**51.6**

TABLE VI
RMSE AT $A(\infty$ bpm), i.e. 100% AVAILABILTY, FOR DIFFERENT DATABASES AND METHODS. BOLD LETTERS MARK THE BEST RESULTS PER COLOR CHANNEL AND DATABASE

	COHFACE data		HCI data	
	Green	CHROM	Green	CHROM
LS-GMM	10.5	13.2	12.7	15.0
LM	21.1	16.9	17.9	16.1
KLT-Skin	18.3	17.8	21.1	20.9
KLT	8.4	12.7	11.8	15.4
VJ static	**7.3**	**11.5**	**11.5**	**14.2**

TABLE VII
MAE AT $A(\infty$ bpm), i.e. 100% AVAILABILTY, FOR DIFFERENT DATABASES AND METHODS. BOLD LETTERS MARK THE BEST RESULTS PER COLOR CHANNEL AND DATABASE

	COHFACE data		HCI data	
	Green	CHROM	Green	CHROM
LS-GMM	7.6	9.8	9.9	11.6
LM	15.3	11.9	13.5	12.2
KLT-Skin	14.5	13.5	17.3	17.0
KLT	6.0	9.4	**9.0**	11.8
VJ static	**5.1**	**8.0**	**9.0**	**10.8**

TABLE VIII
AVERAGE EXECUTION TIME OF THE EVALUATED ALGORITHMS ON A 640×480 VIDEO FRAME

*LS-GMM	LS-GMM	*LM	KLT-Skin	KLT	VJ static
73.2 ms	2322.2 ms	7.6 ms	77.7 ms	22.0 ms	5.6 ms

VIII. CONCLUSION

The presented approach yields a ROI that is able to improve pulse rate extraction in PPGI compared to commonly used procedures. By successfully applying the method to different data sets, we could prove its applicability to some extent. Additionally, a preceding version of the presented algorithm participated in the 1st challenge on Remote Physiological Signal Sensing (RePSS), where it achieved the second rank with a MAE of 7.92 bpm [10].

The presented algorithmic approach thereby leaves room for improvement. We have decided to use a varying number of kernels for the foreground and a fixed number of kernels for the background GMM. Adapting the number of skin kernels reflects our own experiences on often occurring distributions in video data of variable origin. Though the obtained results prove the effectiveness of our current approach, there is still potential for optimization. First, an adaptive choice of Θ can improve the results as discussed before. Second, based on the multivariate models all pixels of the ROI have a strongest supporting kernel of the GMM. This information can be used to generate multiple separated time signals that allow stable pulse rate extraction although some parts of the ROI are not useful or introduce distortions. Third, updating GMM parameters, as kernel mean, covariances and weights, during the segmentation process may lead to less constrained skin and non-skin models, which may improve signal quality under changing illumination situations, e.g. changing shadows during head movement or gradually changing illumination conditions. Fourth, changing the initialization method from a frontal face detector to a more general face detection or color based subject detection has the potential to increase the coverage of video sequences in which a ROI can be established.

Overall, we are convinced that the proposed method bears large potential for PPGI. But even beyond PPGI the combination of GMM and LS might be beneficial for segmentations tasks, e.g. automated skin lesion segmentation or satellite image classification. Since the presented method can be expanded to an arbitrary number of image channels, e.g. MRT and CT channels instead of RGB, it could be applied in tasks of image segmentation in the context of multimodal image fusion.

ACKNOWLEDGMENT

Portions of the research in this paper used the COHFACE Dataset made available by the Idiap Research Institute, Martigny, Switzerland.

Portions of the research in this paper used the MANHOB-HCI Dataset made available by the Imperial College London, London, United Kingdom.

Portions of the research in this paper used the PURE Dataset made available by the Technical University Ilmenau, Ilmenau, Germany.

Portions of the research in this paper used the UBFC-rPPG dataset made available by the University of Burgundy - Franche-Comté, Bourgogne-Franche-Comté, France.

REFERENCES

[1] T. Lister, P. A. Wright, and P. H. Chappell, "Optical properties of human skin," *J. Biomed. Opt.*, vol. 17, no. 9, 2012, Art. no. 0909011.

[2] Y. Sun and N. Thakor, "Photoplethysmography revisited: From contact to noncontact, from point to imaging," *IEEE Trans. Biomed. Eng.*, vol. 63, no. 3, pp. 463–477, Mar. 2016.

[3] S. Zaunseder, A. Trumpp, D. Wedekind, and H. Malberg, "Cardiovascular assessment by imaging photoplethysmography-a review," *Biomedizinische Technik*, vol. 63, no. 5, pp. 529–535, 2018.

[4] K. Humphreys, T. Ward, and C. Markham, "Noncontact simultaneous dual wavelength photoplethysmography: A further step toward noncontact pulse oximetry," *Rev. Sci. Instruments*, vol. 78, no. 4, pp. 1–6, 2007.

[5] W. Verkruysse, L. O. Svaasand, and J. S. Nelson, "Remote plethysmographic imaging using ambient light," *Opt. Exp.*, vol. 16, no. 26, pp. 21 434–21 445, 2008.

[6] D. McDuff, S. Gontarek, and R. W. Picard, "Remote detection of photoplethysmographic systolic and diastolic peaks using a digital camera," *IEEE Trans. Biomed. Eng.*, vol. 61, no. 12, pp. 2948–2954, Dec. 2014.

[7] G. De Haan, V. Jeanne, G. D. Haan, and V. Jeanne, "Robust pulse-rate from chrominance-based rPPG," *IEEE Trans. Biomed. Eng.*, vol. 60, no. 10, pp. 2878–2886, Oct. 2013.

[8] W. Wang, A. C. den Brinker, S. Stuijk, and G. de Haan, "Algorithmic principles of remote PPG," *IEEE Trans. Biomed. Eng.*, vol. 64, no. 7, pp. 1479–1491, Jul. 2017.

[9] D. Wedekind *et al.*, "Assessment of blind source separation techniques for video-based cardiac pulse extraction," *J. Biomed. Opt.*, vol. 22, no. 3, 2017, Art. no. 0 35002.

[10] X. Li, H. Han, H. Lu, X. Niu, Z. Yu, A. Dantcheva, G. Zhao, andS. Shan, "The 1st challenge on remote physiological signal sensing (RePSS)," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops*, 2020, pp. 314–315.

[11] T. Baltruvśaitis, P. Robinson, and L. P. Morency, "Constrained local neural fields for robust facial landmark detection in the wild," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2013, pp. 354–361.

[12] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 1, 2001, pp. *I-511–I-518*.

[13] M.-Z. Poh, D. J. McDuff, and R. W. Picard, "Non-contact, automated cardiac pulse measurements using video imaging and blind source separation," *Opt. Exp.*, vol. 18, no. 10, 2010, Art. no. 10762.

[14] C. Tomasi and T. Kanade, "Shape and motion from image streams: A factorization method—Part 3 detection and tracking of point features," Carnegie Mellon Univ., Pittsburgh, PA, USA, Tech. Rep. CMU-CS-91-132, Apr. 1991.

[15] L. Tarassenko, M. Villarroel, A. Guazzi, J. Jorge, D. A. Clifton, and C. Pugh, "Non-contact video-based vital sign monitoring using ambient light and auto-regressive models," *Physiol. Meas.*, vol. 35, no. 5, pp. 807–831, 2014.

[16] L. Iozzia, L. Cerina, and L. Mainardi, "Relationships between heart-rate variability and pulse-rate variability obtained from video-PPG signal using ZCA," *Physiol. Meas.*, vol. 37, no. 11, pp. 1934–1944, 2016.

[17] F. Bousefsaf, C. Maaoui, and A. Pruski, "Automatic selection of webcam photoplethysmographic pixels based on lightness criteria," *J. Med. Biol. Eng.*, vol. 37, no. 3, pp. 374–385, 2017.

[18] M. Kumar, A. Veeraraghavan, and A. Sabharval, "DistancePPG: Robust non-contact vital signs monitoring using a camera," *Biomed. Opt. Exp.*, vol. 6, no. 5, pp. 200–215, 2015.

[19] F. Bousefsaf, C. Maaoui, and A. Pruski, "Continuous wavelet filtering on webcam photoplethysmographic signals to remotely assess the instantaneous heart rate," *Biomed. Signal Process. Control*, vol. 8, no. 6, pp. 568–574, 2013.

[20] M. Rapczynski, P. Werner, and A. Al-Hamadi, "Continuous low latency heart rate estimation from painful faces in real time," in *Proc. 23rd Int. Conf. Pattern Recognit.*, 2016, pp. 1165–1170.

[21] S. Bobbia, R. Macwan, Y. Benezeth, A. Mansouri, and J. Dubois, "Unsupervised skin tissue segmentation for remote photoplethysmography," *Pattern Recognit. Lett.*, vol. 124, pp. 82–90, Jun. 2019.

[22] L. M. Po, L. Feng, Y. Li, X. Xu, T. C. H. Cheung, and K. W. Cheung, "Block-based adaptive ROI for remote photoplethysmography," *Multimedia Tools Appl.*, vol. 77, no. 6, pp. 6503–6529, 2018.

[23] S. Chaichulee *et al.*, "Multi-task convolutional neural network for patient detection and skin segmentation in continuous non-contact vital sign monitoring," in *Proc. 12th IEEE Int. Conf. Automat. Face Gesture Recognit., FG 1st Int. Workshop Adaptive Shot Learn. Gesture Understanding Prod.*, 2017, pp. 266–272.

[24] R. Špetlík, V. Franc, J. Čech, and J. Matas, "Visual heart rate estimation with convolutional neural network," in *Proc. Brit. Mach. Vis. Conf.*, 2018, pp. 1–12.

[25] A. Trumpp *et al.*, "Skin detection and tracking for camera-based photoplethysmography using a bayesian classifier and level set segmentation," in *Bildverarbeitung für die Medizin* 2017, ser. Informatikaktuell, K. H. Maier-Hein, geb. Fritzsche, T. M. Deserno, geb. Lehmann, H. Handels, and T. Tolxdorff, Eds. Berlin, Heidelberg: Springer, 2017, pp. 43–48.

[26] C. D. Soffientini, E. De Bernardi, G. Baselli, and I. El Naqa, "GMM guided automated level set algorithm for PET image segmentation," *IFMBE Proc.*, vol. 51, no. 1, pp. 368–371, 2015.

[27] OpenCV team, "OpenCV 4.1.1 documentation," 2019. [Online]. Available: https://docs.opencv.org/4.1.1/index.html

[28] J. A. Sethian, *Level Set Methods and Fast Marching Methods*. Cambridge, U.K.: Cambridge Univ. Press, 1999.

[29] K. D. Toennies, *Guide to Medical Image Analysis*, (Ser. Advances in Computer Vision and Pattern Recognition). London: Berlin, Germany: Springer, 2017.

[30] T. F. Chan and L. A. Vese, "Active contours without edges," *IEEE Trans. Image Process.*, vol. 10, no. 2, pp. 266–277, Feb. 2001.

[31] T. F. Chan and L. A. Vese, "An active contour model without edges," in *Proc. Scale-Space*, 1999, pp. 141–151.

[32] M. J. Jones and J. M. Rehg, "Statistical color models with application to skin detection," *Int. J. Comput. Vis.*, vol. 46, no. 1, pp. 81–96, 2002.

[33] S. Zaunseder, A. Trumpp, H. Ernst, M. Frster, and H. Malberg, "Spatio-temporal analysis of blood perfusion by imaging photoplethysmography," in *Optical Diagnostics and Sensing XVIII: Toward Point-of-Care Diagnostics*, G. L. Coté, Ed., vol. 10501, Int. Soc. Opt. and Photon,. SPIE, 2018, pp. 178–191.

[34] R. Stricker, S. Muller, and H.-M. Gross, "Non-contact video-based pulse rate measurement on a mobile service robot," in *Proc. 23rd IEEE Int. Symp. Robot Hum. Interactive Commun.*, Aug. 2014, pp. 1056–1062.

[35] J. Shi and Tomasi, "Good features to track," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 1994, pp. 593–600.

[36] X. Li, J. Chen, G. Zhao, and M. Pietikäinen, "Remote heart rate measurement from face videos under realistic situations," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 4264–4271.

[37] "Dlib 19.21 Documentation," 2020. [Online]. Available: http://dlib.net/

[38] W. Wang, L. Vosters, and A. C. den Brinker, "Continuous-spectrum infrared illuminator for camera-PPG in darkness," *Sensors*, vol. 20, no. 11, May 2020, Art. no. 3044.

[39] R. Macwan, Y. Benezeth, and A. Mansouri, "Remote photoplethysmography with constrained ICA using periodicity and chrominance constraints," *BioMed. Eng. OnLine*, vol. 17, no. 1, pp. 1–22, 2018.

[40] P. Li, Y. Benezeth, K. Nakamura, R. Gomez, and F. Yang, "Model-based region of interest segmentation for remote photoplethysmography," in *Proc. 14th Int. Joint Conf. Comput. Vis., Imag. Comput. Graph. Theory Appl.*, 2019, pp. 383–388.

[41] G. Heusch, A. Anjos, and S. Marcel, "A reproducible study on remote heart rate measurement," 2017, *arXiv:1709.00962*.

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

[43] D. J. McDuff, E. B. Blackford, and J. R. Estepp, "The impact of video compression on remote cardiac pulse measurement using imaging photoplethysmography," in *Proc. 12th IEEE Int. Conf. Automat. Face Gesture Recognit., FG 2017-1st Int. Workshop Adaptive Shot Learn. Gesture Understanding Prod.*, 2017, pp. 63–70.

[44] M. Rapczynski, P. Werner, and A. Al-Hamadi, "Effects of video encoding on camera-based heart rate estimation," *IEEE Trans. Biomed. Eng.*, vol. 66, no. 12, pp. 3360–3370, Dec. 2019.

[45] ISO/IEC 14496-2:2004, "Information technology—Coding of audio-visual objects—Part 2: Visual," Int. Org. Standardization, Geneva, CH, Standard, Jun. 2004.

[46] T. Blöcher, S. Krause, K. Zhou, J. Zeilfelder, and W. Stork, "VitalCamSet - a dataset for photoplethysmography imaging," in *Proc. IEEE Sensors Appl. Symp.*, Mar. 2019, pp. 1–6.