Version of Record: https://www.sciencedirect.com/science/article/pii/S0167865517303860
Manuscript_041e0114e2e36a6785e31e90a8bb0436

Pattern Recognition Letters
journal homepage: www.elsevier.com

Unsupervised skin tissue segmentation for remote photoplethysmography

Serge Bobbia[a,**], Richard Macwan[a], Yannick Benezeth[a], Alamin Mansouri[a], Julien Dubois[a]

[a] *Univ. Bourgogne Franche-Comté LE2I FRE2005, CNRS, ENSAM F-21000 Dijon, France*
serge.bobbia@ubfc.fr, richard.macwan@ubfc.fr, yannick.benezeth@ubfc.fr, alamin.mansouri@ubfc.fr, julien.dubois@ubfc.fr

ABSTRACT

Segmentation is a critical step for many algorithms, especially for remote photoplethysmography (rPPG) applications as only the skin surface provides information. Moreover, it has been shown that the rPPG signal is not distributed homogeneously across the skin. Most of the time, algorithms get input information from face detection provided by a supervised learning of physical appearance and skin pixel selection. However, both methods show several limitations. In this paper, we propose a simple approach to implicitly select skin tissues based on their distinct pulsatility feature. The input video frames are decomposed into several temporal superpixels from which the pulse signals are extracted. A pulsatility measure from each temporal superpixel is then used to merge the pulse traces and estimate the photoplethysmogram signal. Since the most pulsatile signals provide high quality information, areas where the information is predominant are favored. We evaluated our contribution using a new publicly available dataset dedicated to rPPG algorithms comparison. The results of our experiments show that our method outperforms state of the art algorithms, without any critical face or skin detection.

1. Introduction

Photoplethysmography (PPG) is a non-invasive technique for detecting microvascular blood volume changes in tissues. Nowadays, PPG is applied ubiquitously in many settings where a contact PPG sensor (also known as pulse oximeter) is typically attached to a finger or patched to the skin. Basically, contact PPG sensors are used to determine the heart rate and oxygen saturation in blood. The principle of this technology is actually very simple as it only requires a light source and a photodetector. The light source illuminates the tissue and the photodetector measures the small variations in transmitted or reflected light associated with changes in perfusion in the tissue [1].

However, conventional contact PPG sensors are not suitable in situations of skin damage or when unconstrained movement is required. Moreover, it has been shown that pressure of the conventional clip sensors tends to affect the waveform of PPG signal because of the contact force between the finger and the sensor [2].

With the emergence of camera-based health care monitoring, remote photoplethysmography (rPPG) has recently been developed as it allows remote physiological measurements without expensive and specific hardware. It has been shown recently [3] that it is possible to recover the cardiovascular pulse wave measuring variations of back-scattered light remotely, using only ambient light and low-cost vision systems. Since this seminal work, there has been rapid growth in the literature pertaining to remote PPG techniques.

Most methods share a common pipeline-based framework [4, 5, 6, 7]: regions of interest (ROI) are first detected and tracked over frames, RGB channels are then combined to estimate the pulse signal, which is then filtered and analyzed to extract physiological parameters such as heart rate or respiration rate. An interesting and comprehensive state of the art paper on PPG and rPPG has been recently proposed by Sun and Thakor [1]. Microvascular blood volume changes in tissues induce subtle skin color variations over time. This pulsatile information is mixed in the light reflected by the tissue with other signals such as incoming light changes or shadow casting variations due to movements. This mixed signal is then captured by the camera. This suggests that Blind Source Separation (BSS) techniques can separate the different sources and

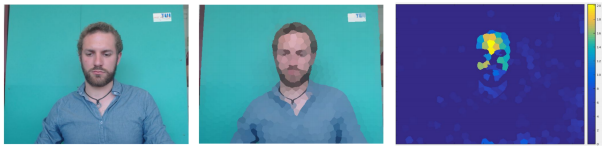
**Corresponding author:
e-mail: serge.bobbia@ubfc.fr (Serge Bobbia)

isolate pulse signal. To this end, a linear combination of RGB time traces can be estimated maximizing independence of estimated sources. Independent Component Analysis (ICA) is a very common algorithm used in several works [5] or [8]. In another work, Lewandowska et al. [9] used Principal Component Analysis (PCA) and proper channel selection to extract the rPPG pulse signal. Unlike the BSS-based methods, some other methods use prior knowledge of the color vectors of the contributing components to control the de-mixing process. We can cite CHROM [6], PBV [10] or POS [11] algorithms that determine optimal RGB combinations to retrieve the pulsatile signal.

RPPG estimation methods use the spatially averaged RGB values of pixels in a Region Of Interest (ROI) to generate a temporal RGB signal. The selection of ROI is a critical first step to obtain reliable pulse signals and must contain as many skin pixels as possible. Several approaches have been proposed for ROI selection in the video stream. In earlier studies, manual selection of the ROI have been used [3, 12]. ROI can also be defined based on the results of classical face detection [8] and tracking algorithms [13] and possibly refined with a skin pixel classification [14]. Instead of selecting and tracking face or skin pixels, some methods focus on smart ROI selection paradigms. For example, in [15], we proposed to use temporal superpixels to extract candidate pulse signals which were then merged into an rPPG signal using pulsatility criteria.Wang *et al.* [16] have used the pulsatility criteria to make a robust living skin classification. In a related work [14], they extracted the rPPG signal by constructing pixel based rPPG sensors. Lately, Tulyakov *et al.* [17] have developed a matrix completion approach in which several traces from several ROIs are combined using an optimization procedure. Indeed, the spatial distribution of the rPPG information allows cross validation and error estimation between several temporal traces. Most methods cited above use a supervised segmentation of the ROI as face detection and skin detection. The proposed method [15] and related work by [18] present a different approach based on unsupervised pixel clustering. Theses clusters are used to detect living skin areas that contribute significantly to rPPG information.

Pixels in the ROI are then usually spatially averaged and the process is repeated for each video frame. The result of this process is a time series which is later used to obtain rPPG signal. It has been shown in several studies that the quality of the ROI has a direct impact on the quality of the rPPG signal [19]. First, because a smaller number of skin pixels leads to larger quantized RGB errors, it can be observed that the quality of rPPG signal deteriorates while down-sampling the ROI. This may be understood as the reduction of the sensor noise amplitude by a factor equal to the square root of the number of pixels used in the averaging process [20]. Second, the quality is also affected by the percentage of non-skin pixels in the ROI [7]. All rPPG algorithms suffer from performance degradation when the ROI is not properly selected. These two remarks are fairly intuitive but it is actually quite difficult in practice to get a well-defined ROI, that is stable over time, without performing complex calculations.

Moreover, as shown by [11], the rPPG signal is not distributed homogeneously across the skin. Some skin regions



(a) (b) (c)

Fig. 1. Pulsatility measures estimated from various temporal superpixels. (a) input frame, (b) temporal superpixel segmentation and (c) pulsatility measures (blue means low pulsatility measures and yellow/orange is high).

contain more PPG signal than others. For example, we observed that the signal-to-noise ratio (SNR) of photoplethysmogram signals extracted from forehead or cheekbones are clearly higher than those obtained from the chin. Figure 1 presents the SNR of rPPG signals calculated from several skin regions.

To overcome these limitations, we propose a new method that implicitly selects ROI that represents living skin tissue and that favors regions of interest where the pulse trace is more predominant. We use the term *implicit* to differentiate our method with those that require critical pre-processing steps for ROI selection and tracking. ROI selection is based on the fact that only the skin tissue of a living subject generates pulsatility, as opposed to conventional approaches based on face detection, tracking and skin segmentation. The input video stream is decomposed into several temporal superpixels from which pulse signals are extracted. A pulsatility measure for each temporal superpixel is then used to merge the pulse traces and estimate the photoplethysmogram signal. This approach can be used with any rPPG algorithm. In this paper, we experimentally validated the proposed automatic living skin tissue segmentation for ROI selection using 5 different methods: Green [3], Green-Red [21], PCA [9], chrominance-based (also known as CHROM) [6] because this method is definitely one of the most reliable rPPG methods, and Plane-Orthogonal-to-Skin (called POS) [11].

The closest contribution to our work, to the best of our knowledge, was done by Wang *et al.* [18], called Voxel-Pulse-Spectral (VPS). In order to detect a living subject in a video using physiological features, VPS extracts voxel-based rPPG signals, then a similarity matrix is built and matrix decomposition with hierarchical fusion is used to identify and combine the voxels. Like our approach, VPS does not rely on ROI selection as a preliminary step. However, our objective is different. We select and combine the ROIs that allow the estimation of an rPPG signal using the weigthed fusion framework without any tedious ROI selection. Moreover, our method uses temporal superpixels tailored to video data rather than supervoxels such as in VPS that are designed for 3D volumetric data. In contrast to supervoxel, object parts in different frames are tracked by the same temporal superpixel. Guazzi *et al.* [20] also use the fusion of several pulse traces but the video is simply divided into contiguous square blocks. The temporal superpixel segmentation is more suited to rPPG algorithms to handle motion scenarios.

As explained previously in the Introduction, the implicit living skin tissue selection method has already been presented in previous work [15]. In this paper, we propose to extend noticeably the experimental study comparing 5 recent state of the art rPPG algorithms. Moreover, a new dataset dedicated to rPPG

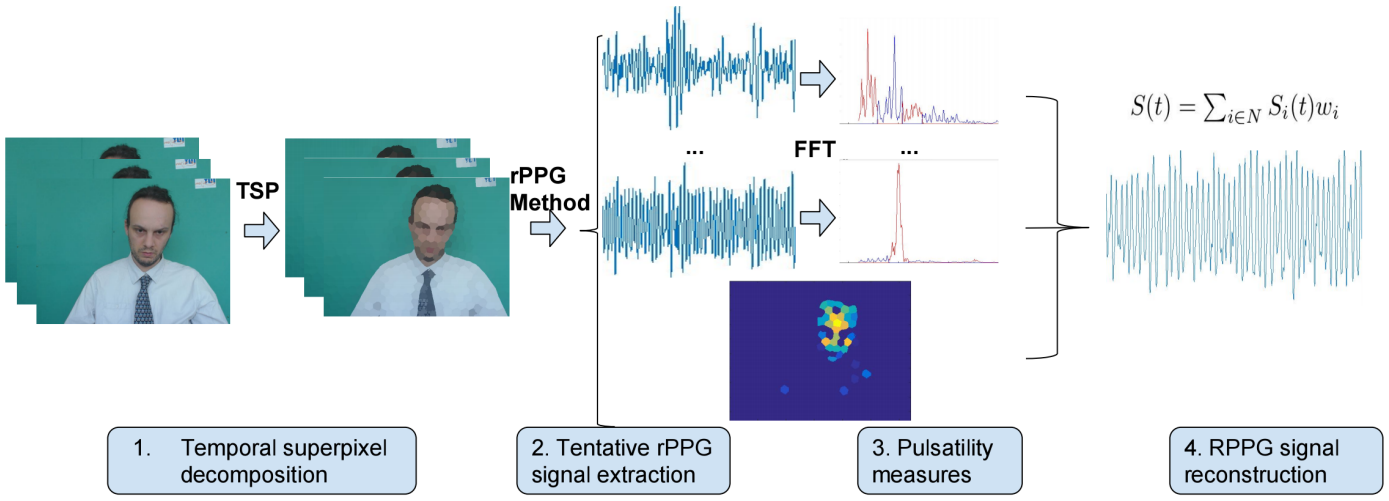


Fig. 2. Overview of the proposed method. (1) Input video stream is decomposed into temporally consistent superpixels. (2) *Tentative* rPPG signal is extracted from each TSP. (3) A pulsatility measure is estimated for each ROI. Blue signal is the convolution of the periodogram by h_{signal} and red signal is the convolution by h_{noise}. (4) A weighted average of all the *tentative* rPPG signals is finally computed.

algorithm evaluation has been acquired, and is presented in the paper and is made publicly available for further comparison with the community.

The rest of the paper is organised as follows. The method is described in section 2 with the temporal superpixel segmentation, the pulsatility measure and the signals fusion procedure. Section 3 presents a new publicly available dataset along with the implemented rPPG methods and the metrics used to compare our unsupervised superpixel-based ROI selection with conventional ROI selection. Results and discussions are presented in section 4 while the conclusion is presented in section 5.

2. Method

The overview of the proposed method is shown in Figure 2. The algorithm can be decomposed into four main steps: (1) the video stream is first decomposed into temporally consistent superpixels (later called TSP for Temporal SuperPixels). (2) Then, a tentative rPPG signal is extracted from each TSP. (3) A pulsatility measure is estimated for each TSP to find contributive signals and (4) a weighted average of all the rPPG signals is computed where the weights are given by the pulsatility measure.

2.1. Temporal superpixel construction

The first step of our method is the segmentation of the video stream into temporally consistent superpixels. If a superpixel is a set of pixels that are local, coherent, and which preserve most of the structure necessary for segmentation [22], temporal superpixels can be defined as a set of video pixels that are local in space and track the same part of an object across time [23]. In this work, we use the TSP method proposed by Chang *et al.* [23] which we found to be a good compromise between precision and speed. This method is based on the Simple Linear Iterative Clustering (called SLIC) [24] decomposition. It has been shown that SLIC is very efficient and is among the fastest superpixel methods [25].

The construction of the TSP is based on an iterative process that propagate the superpixel construction in a coherent way from a frame to the next one. The superpixel construction process can be summed up as the assignation of a 5-dimensional feature vector for each pixel, with the x- and y-location coordinate, and the three components of the *lab* colorspace as introduced by SLIC [24]. Then, the algorithm is performed in two steps. First, a k-means clustering in order to aggregate pixels into clusters with a high compacity. Second, enhance coherence in the clustering by removing isolated pixels and enforce that every superpixel is a single 4-connected cluster. The temporal propagation of the clustering is performed by adding optical-flow information [26] to the process which provides a dense tracking of the pixel acceleration. This dense tracking is then used to build a specific kernel called *bilateral kernel* to model both smoothness and discontinuities that are consistent with flows. The cluster deformation is thus highly related to the motion between frames. From our perspective, this ability to keep spatial coherence and to track the same skin region over time is a significant advantage. Indeed, it reduces the input signal noise due to specular information variations from a region to another one.

2.2. Pulse signal extraction

To segment contributive clusters, (*e.g.* skin areas) from non-contributive ones we construct *tentative* rPPG signals from every TSP. For each video frame, the pixel values in each superpixel are spatially averaged. The result of this process is a set of N RGB time series $x_i^c(t)$, where $c \in \{R, G, B\}$ is the color channel, t is the frame index and $i = 1, 2, \ldots, K$ where K is the number of TSP:

$$x_i^c(t) = \frac{\sum_{k=1}^{M_i(t)} I_{k,i}^c(t)}{M_i(t)} \tag{1}$$

where $M_i(t)$ is the number of pixels in the i^{th} TSP at time t and $I^c_{k,i}(t)$ the k^{th} pixel value at time t and color channel c.

The RGB temporal traces are then pre-processed by normalization, detrended using a smoothness priors approach [27] and band-pass filtered with a Butterworth filter. The rPPG signal can be then extracted using any existing method. In this work, we decided to use the chrominance-based method (also known as *CHROM*) [6] because this method is one of the most simple and reliable rPPG methods. Other rPPG methods have been implemented and evaluated, the results of which are presented in section 3. *CHROM* applies simple linear combinations of RGB channels and obtains a very good performance with low computational complexity. Let $y^c_i(t)$ be the RGB time series obtained after pre-processing. The *CHROM* method projects these RGB values onto two orthogonal chrominance vectors X_i and Y_i:

$$\begin{aligned} X_i(t) &= 3y^R_i(t) - 2y^G_i(t), \\ Y_i(t) &= 1.5y^R_i(t) + y^G_i(t) - 1.5y^B_i(t). \end{aligned} \tag{2}$$

The pulse signal S_i of the i^{th} TSP is finally calculated with $S_i(t) = X_i(t) - \alpha Y_i(t)$ where $\alpha_i = \sigma(X_i)/\sigma(Y_i)$. Because X_i and Y_i are two orthogonal chrominance signals, PPG-induced variations will likely be different in X_i and Y_i, while motion affects both chrominance signals identically.

2.3. Pulsatility measure

Only the skin tissue of a living subject exhibits pulsatility, therefore pulse signals calculated from some superpixels only contain noise (on non-skin areas). Figure 3 (a) presents the periodogram of a pulse signal estimated from skin area while Figure 3 (b) presents the periodogram of a pulse signal estimated from the background. In the frequency domain, the pulsatile, cardiac-synchronous signal, exhibits an important peak centered on the fundamental frequency of heart rate, possibly its second harmonic and limited information at other frequencies. To measure the quality of rPPG signals, we estimate signal-to-noise ratio (SNR) defined as the ratio of the power of the main pulsatile component and the power of background noise, computed in dB due to the wide dynamic range of the signals.

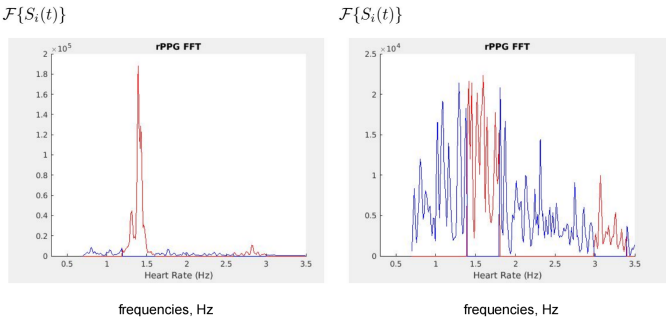


Fig. 3. Periodogram examples of 2 *tentative* rPPG signals estimated from (a) skin area and (b) background.

The pulsatility measure of the i^{th} TSP is estimated by:

$$SNR_i = 10\log_{10}\left(\frac{\int_{f_1}^{f_2} h^i_{signal}(f)|\mathcal{F}\{S_i(t)\}|^2 \mathrm{d}f}{\int_{f_1}^{f_2} h^i_{noise}(f)|\mathcal{F}\{S_i(t)\}|^2 \mathrm{d}f}\right) \tag{3}$$

where $\mathcal{F}\{S_i(t)\}$ is the Fourier transform of the rPPG signal of the i^{th} TSP, f_1 and f_2 the lower and upper limit of the integral defined by the possible physiological range of the heart rate (40 to 240 bpm in our case), and a double-step function h, for the first and second harmonics, defined by the convolution:

$$\begin{aligned} h^i_{signal}(f) &= [\delta(f - f^i_0) + \delta(f - 2f^i_0)] * \prod(\pm f_r) \\ h^i_{noise}(f) &= 1 - h^i_{signal}(f) \end{aligned} \tag{4}$$

with δ the Dirac delta function, f^i_0 the fundamental frequency (*i.e.* peak of the periodogram), convoluted with the *rect* function, noted as $\prod$ of half-width f_r. SNR_i will be high for skin TSP and low for background ones.

2.4. rPPG signal fusion

The final rPPG signal $S(t)$ is then obtained by a weighted average of all *tentative* pulse signals $S_i(t)$, *i.e.* $S(t) = \sum_{i\in K} S_i(t)w_i$ where weightings w_i are a function of the main pulsatile component SNR:

$$w_i = \frac{10^{SNR_i}}{\sum_{i\in K} 10^{SNR_i}} \tag{5}$$

The weights are normalized and in order to conserve the relative contribution of each rPPG signal, they are defined with the $\log^{-1}(x)$ function (*i.e.* 10^x). The weighting favors TSP that have a high main pulsatile component SNR as these are more likely to represent skin areas. For example, in Figure 4, the final rPPG signal is made up of mainly four *tentative* rPPG signals.

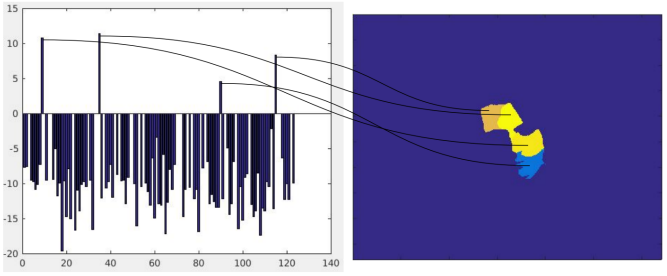


Fig. 4. Examples of SNR values (dB) and its corresponding superpixel.

3. Experiments

This section presents the experimental setup for evaluating the proposed method. First, we describe a new publicly available dataset. Then we present the rPPG methods that have been implemented to evaluate the unsupervised ROI segmentation method. Then, we present the evaluation metrics and finally we compare our implicit ROI selection with regular face detection/tracking and skin detection approach.

3.1. UBFC-RPPG Video dataset

We introduce here a new dataset, called UBFC-RPPG, composed of 43 videos, where each video is synchronized with a pulse oximeter finger clip sensor (Contec Medical CMS50E)

for the ground truth. Each video is about 2 minutes long and recorded with a low cost webcam (Logitech C920 HD pro) at 30 frames per second with a resolution of 640 × 480 in uncompressed 8-bits RGB format. The dataset is available on our project page[1] on demand.

Fig. 5. Dataset sample images.

The subjects sits in front of the camera about one-meter away as shown in Figure 5. Subjects were required to play a time sensitive mathematical game that supposedly raises the heart rate and also emulates the scenario of a normal activity in front of a computer. Experiments are conducted on full video sequences.

3.2. Benchmark algorithms

The TSP algorithm used to construct the time series RGB input signals was implemented based on the publicly available MATLAB code [23]. To evaluate the proposed method, we implemented five algorithms widely used in the literature, with significant performance variations.

3.2.1. Compared rPPG methods

In the *Green* method [3], the component G is directly used as the pulse signal S_i of the i^{th} TSP. The plethysmographic signal is not homogeneously distributed through the RGB channels and the contribution of the green component has been shown to be far more important than the two others.

The *Green−Red* method [21] set S_i as the difference between the normalized G and R components of the time series. This combination provides surprisingly good results considering its simplicity. It states that the motion and illumination variations over the skin surface generate noise that are almost equally distributed in green and red channels.

The *PCA* method [9] computes the principal component analysis of the triplet (R, G, B) of the time series $x_i^c(t)$. The three different channels contribute to the same observation. It constructs a signal with a maximized variance that can be used as the pulsatile sources.

Similarly to *CHROM* presented in section 2, the Plane-Orthogonal-to-Skin method (called POS) [11] applies linear combination on two different orthogonal vector X_i and Y_i:

$$\begin{aligned} X_i(t) &= y_i^G(t) - y_i^B(t), \\ Y_i(t) &= -2y_i^R(t) + y_i^G(t) + 1.5y_i^B(t). \end{aligned} \tag{6}$$

Finally $S_i(t) = X_i(t) + \alpha Y_i(t)$ where $\alpha_i = \sigma(X_i)/\sigma(Y_i)$. X_i and Y_i define a plane orthogonal to skin, that minimizes the specular information due to motion and illumination variations on the skin surface.

[1] http://ilt.u-bourgogne.fr/benezeth/projects/UBFCrPPG

Each algorithm presented above is used to extract the plethysmographic information from the time traces $x_i^c(t)$ and is applied to a sliding window of 20 seconds with a step of 0.5 seconds. The consecutive resultant signals are then overlapped.

Once the pulse signal of each TSP is extracted from the complete video sequences, the signals are detrended and filtered using a frequency bandwidth of $[0.7; 3.5]$ Hz. The SNR calculation is performed using a Dirac window that has been experimentally fixed to $2 \times f_r = 0.35$ Hz. The SNR is estimated within the range $[1; 3.5]$ Hz using the complete rPPG signal. It is important to note here that the TSPs with significant discontinuities due to tracking failure are just discarded for the final rPPG signal construction.

3.2.2. ROI selection methods

Several approaches have been introduced for ROI selection and tracking. In this paper, we compare our implicit ROI segmentation with three regular methods, namely *face*, *cropped* and *skin* as they are respectively used in [3], [4] and [14]. In *face*, face detection and tracking was performed using the Viola-Jones [28] and the Kanade-Lucas-Tomasi [29] implementations provided by the computer vision toolbox of MATLAB. In *cropped*, the center 60% width and full height of the box is selected as the ROI. Finally, in *skin*, skin detection as formulated by Conaire *et al.* [30] was performed to select the candidate pixels in the face ROI of each frame. Figure 6 show examples of these threes ROI.

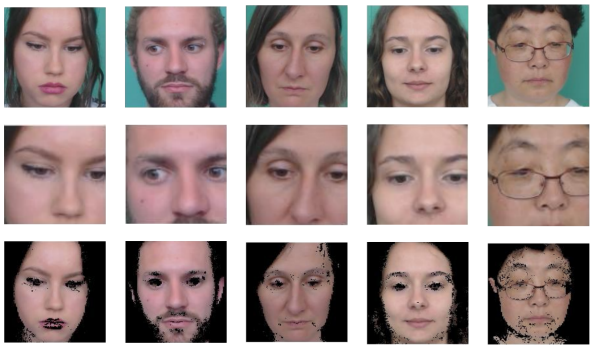


Fig. 6. Segmentation result examples for the reference method with row 1: face detection, row 2: face detection + crop and row 3: face detection + skin segmentation.

We use the same pre-processing and filtering, described in 2, for all the methods. For each video, we estimate heart rate in a sliding window framework. Heart rate is given by the position of the peaks on the frequency axis. The same heart rate estimation procedure was used on the PPG signal recorded with the contact sensor, on the rPPG signal given by the reference methods and the rPPG signal given by our method. All the computation steps from the segmentation to the metrics were performed on MATLAB on a single core on an Intel i7-4790 CPU @ 3.60 GHz platform.

3.3. Benchmark metrics

The following metrics are applied for all the methods introduced and used for comparison:

Table 1. rPPG methods comparison. *CROP* **for the face detection then crop regular method,** *SKIN* **for the face detection then skin detection regular method and** $K = 150$ **for our method with a number of superpixels specified at** 150**.**

Evaluation metrics	Method	Green	Green-Red	PCA	CHROM	POS
Estimation at 2.5 BPM	***FACE***	0.522	0.343	**0.442**	0.75	0.729
	CROP	**0.56**	0.422	0.392	0.796	0.759
	SKIN	0.491	0.535	0.435	0.822	**0.795**
	Our method	0.516	**0.595**	0.462	**0.826**	0.782
Estimation at 5 BPM	***FACE***	0.614	0.409	0.516	0.766	0.73
	CROP	0.683	0.65	0.589	0.862	0.863
	SKIN	**0.739**	0.782	**0.701**	0.861	0.862
	Our method	0.628	**0.821**	0.509	**0.89**	**0.885**
Pearson correlation	***FACE***	0.321	0.135	0.11	0.581	0.571
	CROP	0.78	0.748	0.689	0.94	0.952
	SKIN	**0.821**	0.855	**0.794**	0.943	0.941
	Our method	0.669	**0.904**	0.557	**0.961**	**0.958**
RMSE	***FACE***	18.405	24.198	23.364	9.519	10.033
	CROP	**10.036**	12.813	16.053	3.783	3.838
	SKIN	10.131	8.702	**11.081**	3.157	**3.695**
	Our method	16.84	**6.773**	21.224	**2.388**	6.773
Mean SNR	***FACE***	-2.075	-4.04	-2.976	1.684	0.098
	CROP	-0.225	0.266	-0.83	4.083	3.415
	SKIN	**0.598**	3.245	**0.544**	4.315	3.245
	Our method	-0.234	**3.95**	-2.304	**4.967**	**5.175**

- **Pearson correlation factor** r is the correlation between heart rate estimated from the rPPG signal and the heart rate estimated from the PPG reference signal.
- **Root mean square error** (RMSE) is the quadratic error calculated between the measured value and the ground truth.
- **Precision at 2.5 or 5 bpm**. This metric represents the percentage of estimations where the absolute error is under a threshold (2.5 or 5 bpm)
- **Mean SNR**. The average SNR of the rPPG signal estimations. The bandwidth is set to $[0.7; 3.5]$ Hz and the Dirac width is set to 0.1 Hz

The estimations at 2.5 and 5 bpm are expressed within the range $[0; 1]$ as well as the Pearson correlation factor metric. The $RMSE$ is unit-less as it is the quadratic error measurement and the $Mean\ SNR$ is expressed in dB.

4. Results and discussions

In this section we present the results and further investigation conducted to evaluate the contribution and performance of our segmentation method. The overall results are summarized in Table 1. The best results per algorithm (*Green*, $Green - Red$, etc.) are marked in bold and the best results per metric ($RMSE$, $Mean\ SNR$, etc.) are marked in red in the table. For this study, the superpixels number was fixed to 150 (K = 150).

4.1. Evaluation with several rPPG algorithms

In the first experiment, we evaluate the contribution of the proposed ROI segmentation method for all rPPG algorithms. The first observation is that our method outperforms other ROI segmentation with CHROM, POS and Green-Red. For example, the *Mean SNR* metric for *POS* increased by 59% (about 2dB) using our implicit ROI segmentation compared to *Crop* or *Skin*. This improvement is clearly significant.

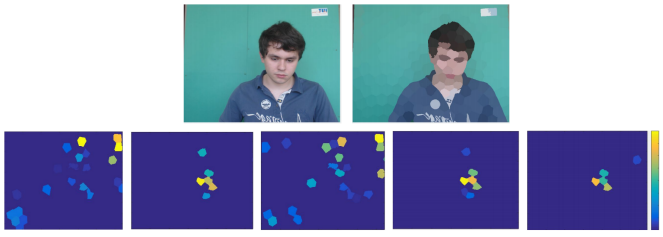


Fig. 7. SNR **value estimated for each TSP, for** SNR **in range** $[0; 8]$ dB**. First row, input frame and superpixels segmentation. Second row, results for** *Green*, $Green - Red$, *PCA*, *CHROM* **and** *POS* **method.**

Although, it is interesting to note that our method does not improve results of *Green* and *PCA*, *Skin* ROI is usually the best segmentation algorithm for these two rPPG algorithms. This can be attributed to the low intrinsic performances of these two methods. Indeed, *Green* and *PCA* obtain on average the worst performances. As a consequence, the weight difference between skin and background areas will be smaller if SNR is low for skin area.

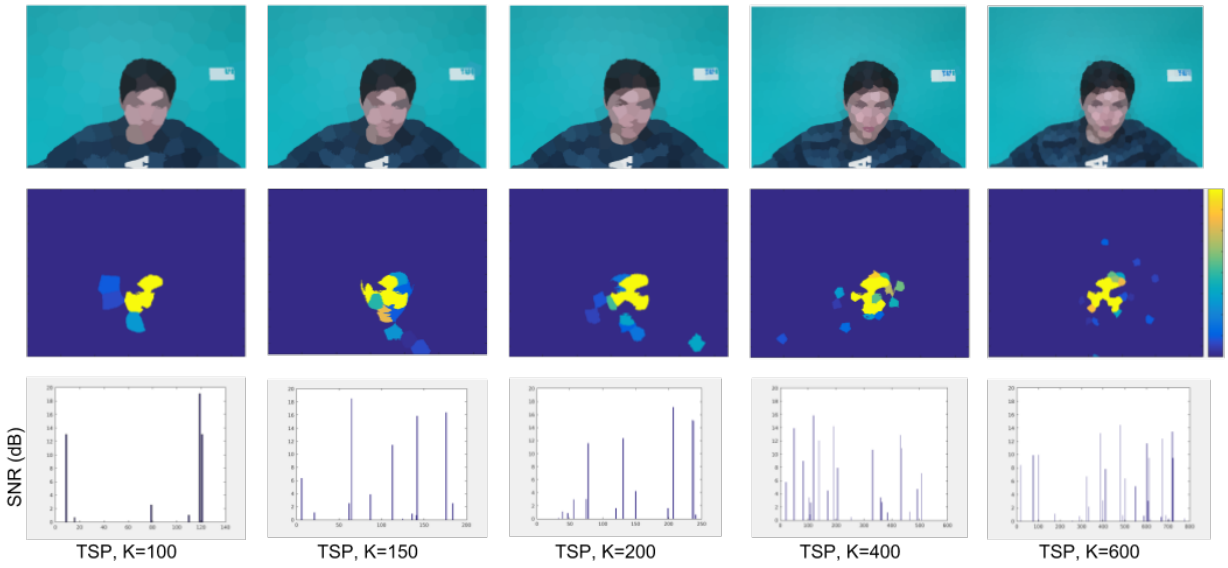


Fig. 8. SNR **metric assigned to corresponding** TSP **for varying resolution :** 100 **then** 150 **then** 200 **then** 400 **then** 600**. Row 1, averaged pixels value per** TSP**. Row 2,** SNR **value in range** $[0;8]$ dB**. Row 3,** SNR **value in** dB **for every** TSP**.**

To illustrate this observation, Figure 7 shows SNR values for each TSP. We clearly see that cheeks and forehead are correctly segmented with $Green - red$, $CHROM$ and POS while contributive TSP are wrongly spread over the frame with $Green$ and PCA. Because the pulsatility criteria is measured on a very small frequency range $[1;3.5]$ Hz with a large rectangle function, *i.e.* $0.35Hz$, the SNR estimation considers that 27% of the information is signal (and the rest is noise). This could lead to wrong segmentation. In Figure 7, even some background areas have a quite high SNR.

In Table 2, we present the running times for the preprocessing (namely normalization, detrending and filtering) and the processing of 20 seconds of RGB traces. These running times are provided for information only. As they were measured with non-optimized code on MATLAB, they are indicative only of the relative processing times. It is interesting to note that the differences between the methods are very low, about 1 ms (except for PCA) and may be negligible compared to other processings (mainly TSP). However, it is worth noting that all processing was performed per superpixel in our method. Consequently, the relative difference between the processing times would be higher with a larger number of superpixels. Finally, it is possible to implement an optimized and parallelized version of our method, since the processing of each TSP is independent.

Table 2. Running time comparison to process 20 seconds of RGB traces.

	Green	Green-Red	PCA	CHROM	POS
Time (ms)	12.13	12.27	15.5	13.21	13.33

Finally, even if it is possible to implement an optimized and parallelized version of TSP, it is important to note that our proposed ROI segmentation method is significantly slower than other simpler methods.

4.2. Segmentation impact

In the second experiment, we evaluate the influence of the resolution of the spatial segmentation defined with the number of superpixels per frame. According to results presented in Table 1, $CHROM$ rPPG algorithm is used for all subsequent experiments. Results are presented in Figure 9 and 10 varying the number of superpixels per frame from 100 to 600. For comparison, we also added the performance obtained with CROP and SKIN in the Figures.

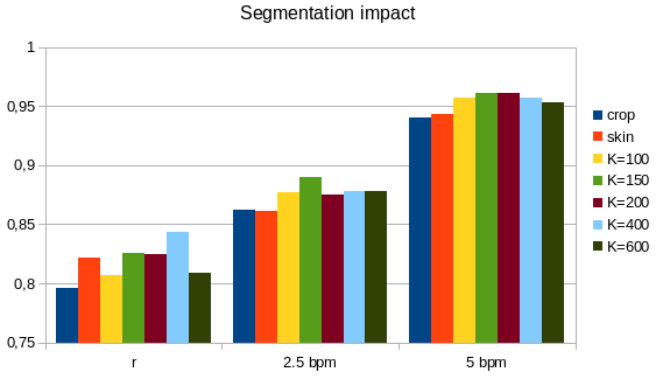


Fig. 9. Metrics results for varying amount of TSP. First Pearson correlation factor, second precision at 2.5 bpm and third precision at 5 bpm.

Every metric shows at least similar results as the $CROP$ and $SKIN$ regular methods for all the segmentation levels. The Pearson correlation factor remains quite stable over the study with results varying from 80% to almost 85%. The precision metrics are consistenly high with at least 95% of correct estimation at every segmentation level for the 5 bpm estimation. Our segmentation method outperforms the face detection approach in both $CROP$ and $SKIN$ scenario as well as the precision at 2.5 bpm with at least 87% of correct estimation. These

three metrics, within the range [0; 1], show that our method performs better in terms of heart-rate estimation for all resolutions. Moreover, the $K = 150$ TSP resolution performs the best for all metrics indicating an optimal resolution for our method. Next, it is important to emphasize that, whatever the resolution used, RMSE is always lower than 3.5. Finally, for a resolution of K = 150 superpixels, we obtain a very good SNR of almost 5 dB.

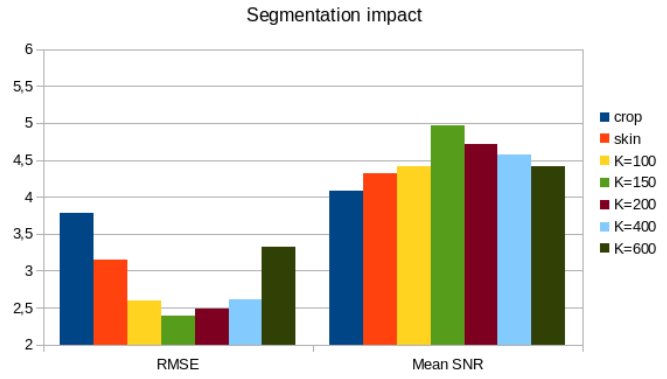


Fig. 10. Metrics results for varying amount of TSP. First RMSE and second mean SNR value.

Results obtained with all the metrics finally show the same tendency. Several points can be highlighted here. First of all, although the performances remain rather robust to the changes in resolution, as shown in Figure 8, there is a growth and decline in performance around the best resolution K=150. This can be explained by the combination of at least two main phenomena. On one hand, the quality of the measure clearly deteriorates as we consider less skin pixels because of quantization noise. On the other hand, the rPPG signal is not distributed homogeneously across the skin and consequently, a large ROI tends to average everything. With very few superpixels, it is possible to have several areas, with different levels of pulsatility, grouped into the same superpixels. In that case, highly informative skin regions will not contribute more than other ones.

4.3. Temporal superpixels segmentation limitations

In this work, we use the TSP method proposed by Chang *et al.* [23] which we found to be a good compromise between precision and speed. However, we experimentally observed that this segmentation does have some limitations. As the TSPs do not have long-time continuity, tracking could be lost in case of rotation or occlusion of the subjects in front of the camera even for a short period of time. The optical flow also interrupts the tracking if the pixel acceleration for a couple of frames is measured too high. Indeed, as we are working with a frame-rate of 30 fps, the optical-flow computation step in TSP is highly sensitive to motion. In Figure 11, we show that for some videos there is a very large number of tracking failures (yellow superpixels in the figure).

Also, superpixel boundaries may vary in time. This variation generates high frequency noise and in some cases can interfere with the estimated rPPG signal. It was observed with our dataset that the TSP algorithm performs well enough to avoid this case and allows a good estimation of the rPPG signal in most cases. Further tests would be need to clearly identify the limitations on varying scenarios and different frame rates.

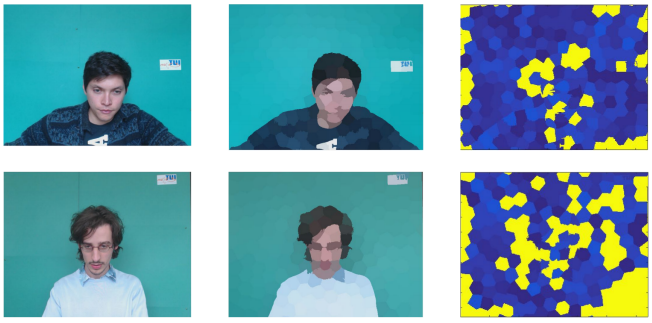


Fig. 11. TSP tracking failure examples. The second column shows the TSP segmentation while the third column highlights in yellow the tracking failures.

5. Conclusion

In the present study, we have described, implemented, and evaluated a new rPPG method that implicitly selects living skin tissue via their distinct pulsatility feature. Photoplethysmogram signals are estimated with the weighted fusion of several *tentative* rPPG signals computed on a set of temporal superpixels. Based on a new publicly available dataset of 43 subjects, the results of this study have demonstrated that the rPPG signals could be remotely estimated without any tedious ROI selection. Furthermore, our method always outperforms the supervised reference methods, namely $FACE$, $CROP$ and $SKIN$. Our method improves signal quality from 15.1% to 59.4%, based on the signal quality SNR metric. This makes the heart rate estimation more accurate in almost every TSP resolution implemented in this study.

Because the TSP algorithm is computationally very intensive, further developments include using other spatio-temporal representation that would consider the important information that the rPPG signal is not distributed homogeneously across the skin but more computationally efficient. We also plan to test our method on a more complex dataset and we are also planning to continue this work to handle multiple individuals in the scene.

Acknowledgments

This research was supported by the Conseil Régional de Bourgogne Franche-Comté, France and the Fond Européen de Développement Régional (FEDER).

References

[1] Y. Sun, N. Thakor, Photoplethysmography revisited: From contact to noncontact, from point to imaging, IEEE Trans. on Biomedical Engineering 63 (2016) 463–477.

[2] X. Teng, Y. Zhang, The effect of contacting force on photoplethysmographic signals, Physiological Measurement 25 (2004) 1323–1335.

[3] W. Verkruysse, L. O. Svaasand, J. S. Nelson, Remote plethysmographic imaging using ambient light, Optics express 16 (2008) 21434–21445.

[4] M. Poh, D. McDuff, R. Picard, Non-contact automated cardiac pulse measurements using video imaging and blind source separation, Optics express 18 (2010) 10762–10774.

[5] B. Kim, S. Yoo, Motion artifact reduction in photoplethysmography using independent component analysis, IEEE Trans. on Biomedical Engineering 53 (2006) 566–568.

[6] G. de Haan, V. Jeanne, Robust pulse rate from chrominance-based rppg, IEEE Trans. on Biomedical Engineering 60 (2013) 2878–2886.

[7] W. Wang, S. Stuijk, G. de Haan, Novel algorithm for remote photoplethysmography: Spatial subspace rotation, IEEE Trans. on Biomedical Engineering 63 (2016) 1974 – 1984.

[8] M. Z. Poh, D. J. McDuff, R. W. Picard, Advancements in non- contact, multiparameter physiological measurements using a webcam, IEEE Trans. on Biomedical Engineering 58 (2011) 7–11.

[9] M. Lewandowska, J. Rumiński, T. Kocejko, J. Nowak, Measuring pulse rate with a webcam, a non-contact method for evaluating cardiac activity, in: Computer Science and Information Systems (FedCSIS), 2011 Federated Conference on, IEEE, 2011, pp. 405–410.

[10] G. De Haan, A. Van Leest, Improved motion robustness of remote-ppg by using the blood volume pulse signature, Physiological measurement 35 (9) (2014) 1913.

[11] A. Kamshilin1, E. Nippolainen, I. Sidorov, P. Vasilev, N. Erofeev, N. Podolian, R. Romashko, A new look at the essence of the imaging photoplethysmography, Scientific Reports (2015) 5.

[12] Y. Sun, S. Hu, V. Azorin-Peris, S. Greenwald, J. Chambers, Y. Zhu, Motion-compensated noncontact imaging photoplethysmography to monitor cardiorespiratory status during exercise, Journal of Biomedical Optics (2011) 16.

[13] H. Tasli, A. Gudi, M. Uyl, Remote ppg based vital sign measurement using adaptive facial regions, IEEE International Conference on Image Processing (2014) 1410–1414.

[14] W. Wang, S. Stuijk, G. de Haan, Exploiting spatial-redundancy of image sensor for motion robust rppg, IEEE Trans. On Biomedical Engineering 62 (2015) 415–425.

[15] S. Bobbia, Y. Benezeth, J. Dubois, Remote photoplethysmography based on implicit living skin tissue segmentation, in: 23rd International Conference on Pattern Recognition (ICPR 2016), 2016.

[16] W. Wang, S. Stuijk, G. de Haan, Living-skin classification via remote-ppg, IEEE Transactions on Biomedical Engineering PP (99) (2017) 1–1.

[17] S. Tulyakov, X. Alameda-Pineda, E. Ricci, L. Yin, J. F. Cohn, N. Sebe, Self-adaptive matrix completion for heart rate estimation from face videos under realistic conditions, in: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.

[18] W. Wang, S. Stuijk, G. de Haan, Unsupervised subject detection via remote ppg, IEEE Trans. On Biomedical Engineering 62 (2015) 2629–2637.

[19] F. Bousefsaf, C. Maaoui, A. Pruski, Continuous wavelet filtering on webcam photoplethysmographic signals to remotely assess the instantaneous heart rate, Biomedical Signal Processing and Control 8 (2013) 568–574.

[20] A. Guazzi, M. Villarroel, J. Jorge, J. Daly, M. Frise, P. Robbins, L. Tarassenko, Non-contact measurement of oxygen saturation with an rgb camera, Biomedical Optics Express 6 (2015) 3320–3338.

[21] M. Hülsbusch, An image-based functional method for opto-electronic detection of skin perfusion, Ph.D. dissertation, Dept. Elect. Eng. RWTH Aachen Univ, Aachen Germany, 2008.

[22] X. Ren, J. Malik, Learning a classification model for segmentation, Proc. IEEE Conference on Computer Vision and Pattern Recognition 1 (2003) 10–17.

[23] J. Chang, D. Wei, J. Fisher, A video representation using temporal superpixels, Proc. IEEE Conference on Computer Vision and Pattern Recognition (2013) 2051–2058.

[24] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, , S. Ssstrunk, Slic superpixels compared to state-of-the-art superpixel methods, IEEE Trans. on Pattern Analysis and Machine Intelligence 34 (2012) 2274–2282.

[25] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, S. Süsstrunk, Slic superpixels compared to state-of-the-art superpixel methods, IEEE transactions on pattern analysis and machine intelligence 34 (11) (2012) 2274–2282.

[26] B. Horn, K. Berthold, G. S. Brian, Determining optical flow, Artificial intelligence 17 (1-3) (1981) 185–203.

[27] M. Tarvainen, P. Ranta-Aho, P. Karjalainen, An advanced detrending method with application to hrv analysis, IEEE Trans. on Biomedical Engineering 49 (2002) 172–175.

[28] P. Viola, M. Jones, Rapid object detection using a boosted cascade of simple features, IEEE Conference on Computer Vision and Pattern Recognition 1 (2001) 511–518.

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

[30] C. O. Conaire, N. E. O'Connor, A. F. Smeaton, Detector adaptation by maximising agreement between independent data sources, in: Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on, IEEE, 2007, pp. 1–6.