Real-Time Estimation of Heart Rate in Situations Characterized by Dynamic Illumination using Remote Photoplethysmography

Patrik Hansen*, Marianela García Lozano*†, Farzad Kamrani*, Joel Brynielsson*†
*FOI Swedish Defence Research Agency, SE-164 90 Stockholm, Sweden
†KTH Royal Institute of Technology, SE-100 44 Stockholm, Sweden
{patrik.hansen, marianela.garcia.lozano, farzad.kamrani, joel.brynielsson}@foi.se

Abstract

Remote photoplethysmography (rPPG) is a technique that aims to remotely estimate the heart rate of an individual using an RGB camera. Although several studies use the rPPG methodology, it is usually applied in a laboratory in a controlled environment, where both the camera and the subject are static, and the illumination is ideal for the task. However, applying rPPG in a real-life scenario is much more demanding, since dynamic illumination issues arise. The work presented in this paper introduces a framework to estimate the heart rate of an individual in real-time using an RGB camera in a situation characterized by dynamic illumination. Such situations occur, for example, when either the camera or the subject is moving, and/or the face visibility is limited. The framework uses a face detection program to extract regions of interest on an individual's face. These regions are combined and constitute the input to a convolutional neural network, which is trained to estimate the heart rate in real-time. The method is evaluated on three publicly available datasets, and an in-house dataset specifically collected for the purpose of this study, that includes motions and dynamic illumination. The method shows good performance on all four datasets, outperforming other methods.

1. Introduction

Photoplethysmography (PPG) [2] is an optical measurement method for heart rate (HR) monitoring. A light source and a photodetector are used at the skin surface to measure the volumetric variations of blood circulation. The concept of remote photoplethysmography (rPPG) [28] introduces the ability to remotely estimate the HR of an individual using, e.g., an RGB camera.

Remote PPG uses the pulse-induced variation in light absorption of human tissue caused by changes in blood volume [13] to allow for cardiovascular measurements. Information about the HR can be acquired from the obtained rPPG signal [23]. However, signals obtained by the rPPG method are subject to large amounts of noise, which confounds measurements related to the targeted parameters [30]. Classical approaches that can be applied to connect signals obtained by rPPG to the vital parameters are different signal filtering techniques [11].

When working with rPPG signals, the main problem is its sensitivity to external disturbances such as motion and dynamic illumination. These sources of signal noise often dwarf the sought-after light intensity variations created by altering blood volume in the tissue. They also induce significant changes in the intensity perceived by the RGB sensor. The work presented in this paper aims to handle some of these challenges, which often arise if rPPG is used in a real-life scenario and in situations characterized by dynamic illumination; for instance, when either the camera or the subject whose HR is being measured is moving and parts of their face is occluded.

The rest of this paper is organized as follows. In Sec. 2, related work is discussed. Sec. 3 presents the proposed workflow, the applied method, and used datasets. The experimental setup is presented in Sec. 4, and results are presented in Sec. 5. In Sec. 6, an analysis and discussion of the results are presented. Finally, Sec. 7 comprises a few concluding remarks along with planned directions for future work.

2. Related Work

Since the first experimentation in 2008 with rPPG and consumer-level digital cameras for HR estimation [28], several approaches have been developed and refined. In [12], the authors develop their own rPPG signal filtering method, which is chrominance-based (referred to as *CHROM*). CHROM was shown to be more tolerant to motion-induced distortions than principal component analysis (PCA) and independent component analysis (ICA) based methods. An alternative method to rPPG-based HR estimation from video is to detect the minuscule head movements that heart

beats generate [3]. In [29], the authors propose a plane orthogonal-to-skin (POS) algorithm, which projects the PPG signal onto a plane orthogonal to the skin tone to extract the pulse signal. The POS algorithm was demonstrated to surpass the CHROM, PCA and ICA based methods. In a 2016 paper, the authors estimate a person's HR from an RGB video taken with a laptop webcam in an indoor environment with constant ambient light [22]. However, these classic methods often require prior knowledge for region of interest (ROI) selection, and do not generalize well to new data.

Two-step convolutional neural network (CNN) based methods, where one CNN is used to detect the face and extract the rPPG signal, and a second CNN estimates the HR from the signal, were suggested in 2017 [25]. An interesting take on the signal processing pipeline was suggested, where an attention network was used to calculate the frame difference and, thus, the motion representation [7]. Some proposals combine the first CNN with a long short-term memory (LSTM) network to process the temporal information contained in video sequences [17]. It is also possible to process the rPPG signals and estimate vital parameters using deep learning, which has been demonstrated in [8] with promising results in ideal settings. Nonetheless, these proposed neural network-based methods still rely on handcrafted features or aligned face images. Also, it is worth noting that there are difficulties in the statistical comparison of machine learning-based methods [11, 15, 27].

In 2020 Boccignone et al. [5] proposed an open framework where they implement eight of the classical rPPG methods, and compare their performance. Botina-Monsalve et al. [6] propose a real-time PPG signal estimation method and compare it with PhysNet [31], an end-to-end framework with spatio-temporal networks. Another real-time framework combines two LSTM networks with a signal quality attention mechanism to estimate the HR [14]. By using contrastive learning, a self-supervised method, in 2023, Birla et al. [4] improve the HR estimation over other methods.

3. Method

This section outlines the methodology and the experimental approach used to address non-contact HR estimation in dynamic illumination situations.

3.1. Workflow

To estimate the HR from a video sequence, an end-to-end framework is introduced. An illustration of this framework is shown in Fig. 1. The workflow consists of 5 different steps summarized as follows (in-depth descriptions of each step is given in the forthcoming sections):

1. *Input Video Sequence*. In the first step, the video sequence from which HR is to be estimated serves as the input to the computational process.
2. *ROI Extraction*. The five ROIs are extracted from each frame in the video sequence.
3. *ROI Video Sequence*. Using the extracted ROIs, a new image is created.
4. *Deep Learning Architecture*. To estimate the HR from the ROI video sequence, deep learning is used.
5. *Heart Rate Estimation*. In the last step of the framework, the HR is estimated.

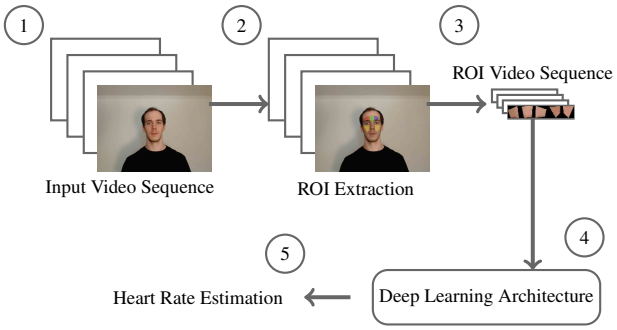


Fig. 1. Illustration of the end-to-end framework for estimation of HR from a video sequence.

Finally, to evaluate and validate the proposed method, different experiments are performed, see Sec. 4.

3.2. Data

The four datasets used for training and evaluation are presented in the following subsections.

3.2.1 Public Datasets

Three public datasets were used to train and evaluate the models. The first dataset is *Multimodal Spontaneous Emotion Database* (BP4D+), which is a 10 TB dataset created for the purpose of testing algorithms for analyzing human behavior [32]. The database includes 140 test subjects with varying gender, age, and ethnicity. The recorded physiological signal consists of blood pressure, respiration rate, HR, and electrodermal activity; all with a sampling frequency of 1000 Hz. The resolution of the 2D video camera is 1040×1392 pixels, with a frame rate of 25 FPS. In the database, the participants perform 10 different tasks to invoke emotions. Only videos with a blood pressure pulse that can be used to compute the HR as described in Sec. 3.5 are used.

The second dataset is *Pulse Rate Detection Dataset* (PURE),[1] which consists of video data of 10 participants

[1] https://www.tu-ilmenau.de/neurob/data-sets/pulse

and their corresponding pulse measurements [26]. Captured video data has a resolution of 640×480 pixels, with a frame rate of 30 FPS. The physiological measurements consist of their blood-volume pulse (BVP) and SpO_2 reading. The camera is placed at an average distance of 1.1 meters from the participants in a daylight illumination condition. Each participant is recorded in 6 different setups: steady, talking, slow transition, fast transition, small rotation, and medium rotation.

The third dataset is COHFACE, where physiological signals have been collected in more realistic settings [16]. The research presented in this paper made use of the COHFACE dataset made available by the Idiap Research Institute in Martigny, Switzerland. It consists of a total of 160 videos of 40 participants of different gender and age, along with their BVP and respiratory rates. The captured videos, taken under different lighting conditions, have a resolution of 640×480 pixels, with a frame rate of 20 FPS.

3.2.2 In-house Dataset – IHD

To mimic realistic scenarios with dynamic illumination settings, an in-house dataset (from now on referred to as the IHD dataset) was collected. The dataset consists of video data and HR collected from one individual. For the videos, a Logitech HD PRO Webcam C920 with fixed exposure and white balance was used to record the subject's face at a distance of about 1 meter from the camera. Each video was recorded at 20 FPS with a resolution of 640×480 pixels where the duration of each video is 60 seconds. To gather data on the HR of the subject during the recording of the video, the OXY-200 Desktop Pulse Oximeter was used [21].

Motions considered in this paper are head rotation and camera movements. A static scenario is also considered. The light originates from a single light bulb facing the subject. It is placed at a distance of about 1 meter from the subject at three different angles: 0, 45, and 90 degrees relative to the front of the subject (see Fig. 2).

For the head rotation and the static scenarios, there are three different light angles and two different settings for the light source, directed and non-directed. When considering the movement of the camera, only one angle of the light source is considered. Combined, these settings make up the three different main scenarios presented: static, head rotations, and moving camera. The recorded HR of all the data ranges from 46 to 163.

3.3. Selecting Region of Interest

The second step of the framework is about selecting the ROIs. The quality of the rPPG signal varies depending on where on the face the measurement is performed [18]. For example, the thickness of the skin is not the same all over the face, which results in a nonapparent blood volume tissue variation of the rPPG signal in different regions [10]. Hence, only certain regions of interest on the face should be considered.

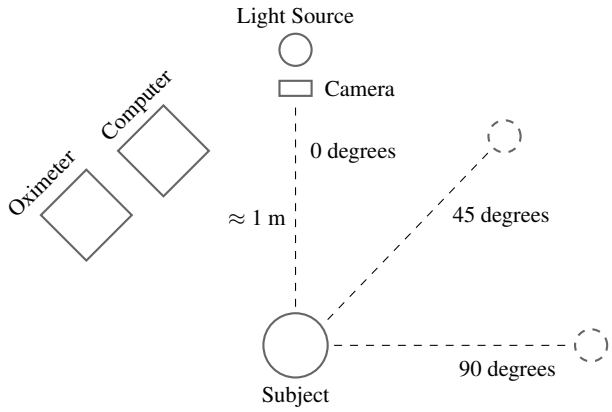


Fig. 2. Illustration of the setup for the gathering of data for the IHD dataset in the static scenario.

To track a face in the video and select ROIs, the open source Python library Mediapipe Face Mesh [19] is used, where both face mesh and face detection functionalities are available. The face detection tool possesses two modules; one for face detection and another for identifying landmarks. The face landmark module localizes 468 different facial key points where each key point is composed of Cartesian coordinates. In Fig. 3, the red circles mark the face landmarks used for outlining five different ROIs. The forehead is divided into three regions, the left, center, and right forehead, to allow for the potential exclusion of regions due to lack of visibility when a region is outside the line of sight from the camera. The left and right cheeks are also selected as ROIs due to their performance as rPPG measurement targets [18].

To extract the pixels of marked regions, one vertex is assigned to each face landmark outlining a respective region. A polygon is then created by interpolating lines between adjacent vertices. To decrease the computational complexity, the pixels inside of the polygon are extracted by first selecting a rectangle in the image defined by the maximum and minimum of the x and y values of the edges of the polygon, as illustrated in Fig. 4.

When the surface of an ROI is angled relative to the camera, the quality of the signal is decreased. Because of this, ROIs that are captured at an angle exceeding a too angled state are removed from down-stream processing. This is done by comparing the area of the ROI to the total area of all ROIs in the same region. The ROIs are divided into two different regions, the forehead and the cheeks. To compute the area of an ROI, the trapezoid formula for a simple polygon is used, where the area is described by

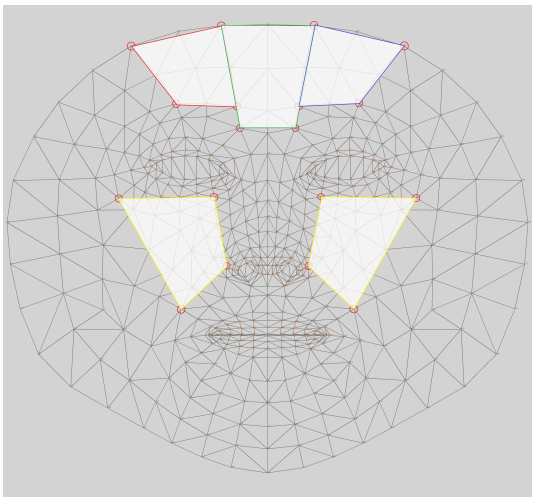


Fig. 3. Face landmarks that are created by the Mediapipe Face Mesh program, and extracted ROIs. The areas with red, green, and blue outlines represent the right, center and left forehead, and the areas with yellow outlines represent the right and left cheek.

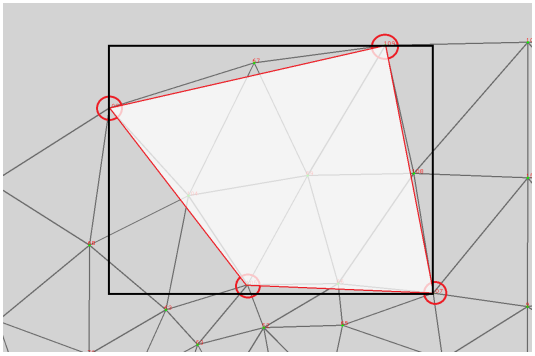


Fig. 4. Extraction of pixels by first selecting a rectangle defined by the maximum and minimum of the x and y values of the edges of the polygon, and then computing the pixels of the rectangle that belong to the polygon.

$A = \frac{1}{2}\left|\sum_{i=1}^{n-1}(y_i+y_{i+1})(x_i-x_{i+1})+(y_n+y_1)(x_n-x_1)\right|$, where n is the total number of vertices defining the polygon, x_i and y_i are the x and y coordinates of vertex i, and A is the total area of the ROI. To compute if an ROI is too angled in relation to the camera, the area of the ROI is considered. For the forehead and cheek regions, if the area of an ROI contributes to less than 20 and 30 percent of the total region area, respectively, the ROI is excluded from the computation of the rPPG signal.

To ensure that the quality of the data from each dataset is the same, all the data from the different datasets need to roughly have the same resolution. Also, a criterion in this work is that the method should be applicable in real-time. Due to this, the resolution of all datasets was lowered to 320×240 (except for the BP4D+ dataset that has a different aspect ratio, and therefore had its resolution changed to 260×348).

3.4. Extracting Regions of Interest

The relevant ROIs (see Sec. 3.3) are combined into a new image. First, for each ROI a temporary image is created with width and height equal to w and h, respectively. Here, $w = x_{max} - x_{min}$ and $h = y_{max} - y_{min}$, where x_{min}, y_{min}, x_{max}, and y_{max} are minimum and maximum values for x and y positions of pixels for each ROI. The temporary images are then resized into 20×20 images that are placed in a row by adding a black strip with width of 5 pixels between each two images, resulting in an image of size 20×120 as the input to the deep neural network.

The HR is time-dependent as it represents the number of heartbeats per minute. It is not possible to learn and predict the HR based on a single frame; instead, a sequence of frames with a corresponding HR is considered as a data point to be used for training. This sequence should be long enough to capture at least a couple of heartbeat cycles. Henceforth, the number of frames chosen is 40, which corresponds to 2 seconds.

To make a fair comparison of results between datasets, the frame rate had to be altered such that it is the same for all datasets. This was done by interpolating every pixel in the sequence of images of the extracted ROIs. A new sequence of images with a frame rate of 20 was obtained, by evaluating the interpolated pixels at every 0.05 second step.

3.5. Converting PPG Signal to Heart Rate

Since there is no exact universal method for determining HR, the HR in the datasets may have been computed differently. Hence, two similar rPPG signals can potentially be connected to two different HRs, even though they should be identical. To combat this, the HR is computed directly from the presented PPG signal in the different datasets to ensure it is computed in the same way. This was done by implementing a peak detection algorithm to compute the HR. The result of this process is shown in Fig. 5, where the HR is computed by taking the average time over nine peaks and converting it to one minute. The HR per minute is calculated by:

$$bpm_i = \begin{cases} bpm_{i-1}, & \text{if } t_i \text{ is not a peak,} \\ \sum_{k=j-4}^{j+4} \frac{60}{8(t'_{k+1}-t'_k)}, & \text{if } t_i \text{ is a peak,} \end{cases} \quad (1)$$

where bpm_i is the heart rate at time t_i, t_i is the time of the PPG signal at point i, t' is the series of times when peaks occur, and t'_j represents the peak corresponding to time t_i. The sum can be computed when there are at least four peaks to the right and left of the current time t_i. However, for every peak that does not exist, it can simply be skipped in the calculation and the number 8 in the denominator of Eq. (1) is decreased by one.

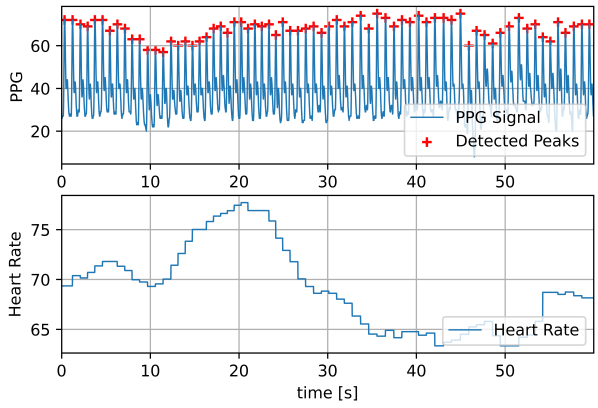


Fig. 5. The upper image shows an obtained PPG signal from an oximeter and the detected peaks of said signal. The lower image displays the HR calculated by Eq. (1) from the PPG signal.

When computing the HR of subjects in the BP4D+ dataset, introduced in Sec. 3.2.1, the blood pressure measurement over time is used, since no PPG signal is provided.

3.6. Deep Learning Architecture

The input data to the neural network consists of images with a width of 120 pixels (5 ROIs with a padding of 5 pixels between each ROI), a height of 20 pixels, and 3 color channels (red, green, and blue). The number of frames in each input sample is T, which represents the time duration, resulting in an input sample of size $T \times 20 \times 120 \times 3$. The value of T, that is, the number of frames, was set to 40, corresponding to 2 seconds. The deep neural network architecture proposed in this study, HR-rtCNN, is based on the 3 dimensional convolutional neural network (3DCNN). No padding is applied to any of the 3DCNN layers. In Tab. 1, the HR-rtCNN network is presented. Since the prediction of the HR is a regression task, a fully connected layer (Dense) with one node is added as the last layer of the deep neural network.

4. Experiments

To create a baseline, the elected framework was tested on the different public datasets. Then, for the experiments with dynamic illumination settings, data from the IHD dataset was used. To measure the real-time applicability of the HR-rtCNN, the mean computational time for processing a frame in IHD was measured in the pre-processing stage, as well as the inference time of the network. For the training of each network, the Adam optimizer was used with a learning rate of 0.0001. The rest of the hyperparameters were kept according to their default values in *Keras* [9]. Each network was trained for a maximum of 1000 epochs using early stopping with a patience of 50 epochs. The final network was saved as a frozen graph using *Tensorflow* [1]. The hardware used in this project is an AMD EPYC 7742 64-Core Processor, an NVIDIA A100 SXM4 40GB GPU, and one terabyte of RAM.

Tab. 1. 3DCNN-based network named (HR-rtCNN). The input to the network is of size $T \times 20 \times 120 \times 3$, where T is the number of frames. The total number of parameters is 1164897.

Layer	# of Nodes	Kernel Size
3DCNN	8	$1 \times 3 \times 3$
Batch Normalization		
3DCNN	32	$3 \times 1 \times 1$
Batch Normalization		
Dropout 0.5	–	–
Avg Pool	–	$2 \times 2 \times 2$
3DCNN	64	$3 \times 3 \times 3$
Batch Normalization		
Dropout 0.5	–	–
Avg Pool	–	$2 \times 2 \times 2$
3DCNN	128	$3 \times 3 \times 3$
Batch Normalization		
Dropout 0.5	–	–
Avg Pool	–	$2 \times 2 \times 2$
3DCNN	256	$3 \times 3 \times 3$
Batch Normalization		
Dropout 0.5	–	–
Global Avg Pool	–	–
Dense	1	–

The elected method was compared to the state-of-the-art method PhysNet [31] as well as the chrominance-based method CHROM [12]. The face detection was implemented using Mediapipe Face Mesh [19]. Since data sequences of 2 seconds are used and the PhysNet as well as the CHROM methods compute a filtered rPPG signal, the resulting HR was computed by implementing a peak detector and measuring the time between peaks. In the case where only one peak is present, the width of the wave corresponding to the peak was measured to compute the HR.

To measure the accuracy of the model, the *mean average error* (MAE), *root mean squared error* (RMSE), *mean absolute percentage error* (MAPE), *coefficient of determination* (R^2), and *Pearson correlation coefficient* (r) are used. The elected method is also evaluated on real-time data, where the mean time it takes for the neural network to process each frame in a video is measured.

4.1. Experiment Setup with the Public Datasets

The developed model was tested on the four different datasets according to Sec. 3.2. The deep neural network was trained on each of the public datasets separately. Every data sample was chosen such that there is no overlap

between data samples. For example, if the original video is 60 seconds and the time duration of each sample is set to 2 seconds, 30 data samples can be made from the original video. For each training session, 65% of the data samples in the dataset are used for training, 20% for validation, and 15% for testing. The number of data samples in the training, validation, and test sets for each of the different public datasets is given in Tab. 2. In the datasets, for most subjects, the recorded HR has a small deviation from the mean HR. To prevent this bias in the training, more weight should be given to data that is further from the mean HR in the loss function. Therefore, the MSE function is used as the loss function during the training.

Tab. 2. Number of data samples for the training, validation, and test sets for the datasets.

Dataset	Training Size	Validation Size	Testing Size
IHD	16342	5030	3772
COHFACE	3233	995	747
PURE	1310	404	303
BP4D+	18143	5583	4187

4.2. Experiment Setup with Dynamic Scenario Data

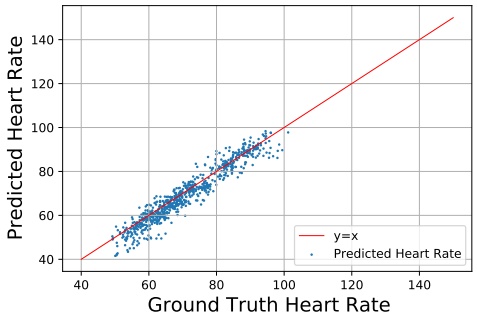
When performing the evaluation for the dynamic scenarios, the network was trained on the IHD dataset. The data split of the IHD dataset used for the training and evaluation of the deep neural network was 65% for training, 20% for validation, and 15% for testing, for each scenario. The data samples in the training, validation, and test sets were randomized from within each scenario. All data with uninterpretable PPG signal or blood pressure was excluded from the study. The total number of data samples in the training, validation, and test sets for the IHD dataset is given in Tab. 2. For the training of the network, the training and validation sets from all scenarios were combined, while the test sets were kept separate.

4.3. Measurement of Computational Time

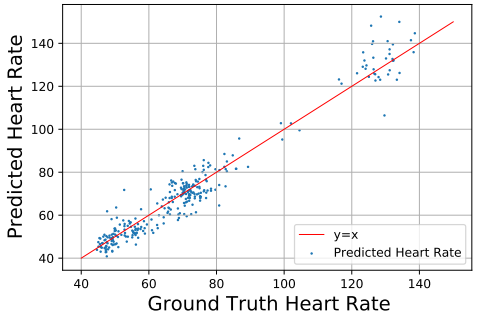
The measurement of the computational time for the elected method was separated in two parts. First, the computational time of the pre-processing stage was estimated by computing the time it takes for a frame to be pre-processed. The mean time, with corresponding standard deviation, was computed for each video in the IHD dataset. From the obtained values, the maximum mean and standard deviation were used. When measuring the inference time of the obtained network, the mean time with corresponding standard deviation of each data sample was computed.

5. Results

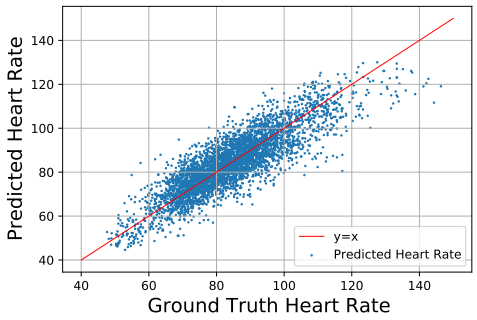
The experimental results of the trained neural network on test sets derived from the COHFACE, PURE, and BP4D+ datasets are presented in Tab. 3. Evaluation metrics used are MAE, RMSE, MAPE, R^2, and r, as mentioned in Sec. 4. The predicted HRs acquired from the HR-rtCNN model are plotted against the ground truth HRs for the COHFACE, PURE, and BP4D+ datasets in Fig. 6. This shows how well the network performs for the different ranges of HR for the public datasets.



(a) COHFACE



(b) PURE



(c) BP4D+

Fig. 6. The predicted HR of HR-rtCNN plotted against the ground truth with a time duration of 2 seconds for the (a) COHFACE, (b) PURE, and (c) BP4D+ datasets.

The results of the HR-rtCNN model on the in-house

Tab. 3. Evaluation results of the neural network architectures as well as the CHROM method on the test sets derived from the public datasets. The time duration of each video sequence is 2 seconds, corresponding to 40 frames.

Dataset	HR-rtCNN					PhysNet					CHROM				
	MAE	RMSE	MAPE	R^2	r	MAE	RMSE	MAPE	R^2	r	MAE	RMSE	MAPE	R^2	r
COHFACE	2.92	3.83	4.26	0.89	0.96	6.81	10.11	9.85	0.26	0.69	36.81	49.83	53.96	-17.38	0.08
PURE	4.30	5.73	6.10	0.94	0.97	3.87	8.82	6.54	0.86	0.93	21.45	38.59	35.74	-1.56	0.34
BP4D+	5.69	7.33	6.80	0.73	0.86	4.31	6.71	5.27	0.78	0.89	15.17	26.97	19.01	-2.71	0.31

dataset are displayed in Tab. 4. The predicted HR is plotted against the ground truth HR for each scenario in Fig. 7. The results of the computational time measurements, as described in Sec. 4.3, is displayed in Tab. 5.

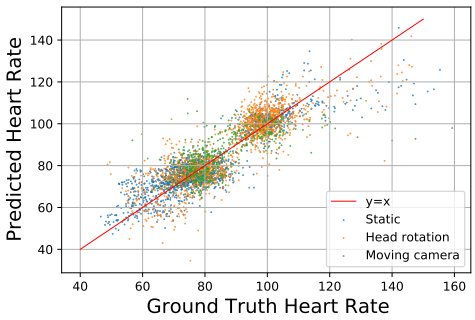


Fig. 7. The predicted HR of HR-rtCNN for the test set for the different scenarios in the IHD dataset. The time duration is 2 seconds.

6. Discussion

Overall, the prediction results for all the datasets achieved good results for all metrics when using HR-rtCNN, as seen in Tab. 3. The high R^2 value indicates that the neural network has found patterns in the data. The objective function used for the training was the MSE function, meaning that the optimization aims to minimize the RMSE for every dataset. It might be possible to achieve lower values of MAE and MAPE if the objective function incorporates these metrics. However, the MAE and MAPE metrics might present deceptive results for the context of HRs in the datasets. For MAE, there will be a bias towards more common HRs in the training data, which worsen the prediction of HRs that deviate from the norm. When scaling the error based on the actual value of the HR, as in MAPE, the absolute value of the error will have a bias towards lower HRs, as a specific value of the error would have a much higher impact on the MAPE value for lower HRs than for higher ones.

The CHROM method performs poorly, as seen in Tab. 3 and Tab. 4. This may be due to that the short 2-second video sequences make it extra susceptible to interference from noise. Since CHROM follows a preset formula, no special care is taken to adapt to the specific settings in the datasets, which further decreases its performance. This may explain why it performs worse than the deep learning based methods.

When comparing HR-rtCNN with PhysNet, HR-rtCNN generally achieves better results for all metrics for each of the different datasets (see Tab. 4). The reason for this could be that the HR-rtCNN is an end-to-end network that directly connects a sequence of pre-processed images to the HR while PhysNet computes a filtered rPPG signal. Because the time duration of the sequence is 2 seconds, an accurate HR could be difficult to compute by measuring the distance between peaks. This would significantly decrease the accuracy in the presence of low-quality data. The COHFACE dataset has about 0.04 bits per pixel, while BP4D+ and PURE have about 4 and 10 bits per pixel, respectively. This indicates that the COHFACE dataset has been compressed to a high degree, lowering the quality of the rPPG signal present in the video [20]. This might be the cause of why a higher R^2 value was achieved for PURE, even though there is dynamic illumination present whilst COHFACE is static. In the BP4D+ dataset, there is a great number of participants, creating more variation in the data. There is also dynamic illumination induced by spontaneous motions of participants in the video data, causing a decrease in accuracy for both HR-rtCNN and PhysNet. The PURE and BP4D+ datasets have a greater range of HRs than the COHFACE dataset, as observed in Fig. 6. A consequence of this might be an increase in variations of the predicted HR, resulting in higher values for the metrics MAE, RMSE, and MAPE.

In the IHD dataset, more extensive dynamic illumination is present, affecting the results negatively for both HR-rtCNN and PhysNet, possibly explaining why it performs worse compared to the public datasets. In Fig. 7, a notable bias can be observed, where there appears to be 2 different clusters of data points. The HR being recorded at a normal and an elevated state caused this. The higher heart state was achieved by performing cardiovascular exercises before the recording. Not enough time might have elapsed for the HR to reach normal levels before ending the recording, explain-

Tab. 4. Evaluation results of the neural network architectures as well as the CHROM method on the test set derived from the IHD dataset for the different scenarios. The metrics displayed are MAE, RMSE, MAPE, R^2, and r. The time duration of each video sequence is 2 seconds, corresponding to 40 frames.

Scenario	HR-rtCNN					PhysNet					CHROM				
	MAE	RMSE	MAPE	R^2	r	MAE	RMSE	MAPE	R^2	r	MAE	RMSE	MAPE	R^2	r
Static	5.60	7.43	7.15	0.79	0.89	6.23	9.78	7.99	0.62	0.81	33.62	43.78	43.30	-6.62	0.01
Head Rotation	6.29	8.33	7.43	0.71	0.86	13.47	20.27	15.94	-0.67	0.40	32.51	42.16	38.15	-6.52	-0.03
Moving Camera	6.23	8.92	7.26	0.45	0.69	22.74	29.60	28.29	-4.49	0.02	32.65	42.48	40.21	-10.52	0.01

Tab. 5. Mean computational time with corresponding standard deviation of the frame pre-processing time and inference time.

Method	Pre-processing	Inference Time
HR-rtCNN	14.9 ± 6.7 ms	3.0 ± 0.4 ms
PhysNet	101.8 ± 51.0 ms	7.4 ± 2.2 ms

ing the presence of 2 separate clusters in Fig. 7. The network performs noticeably worse for HRs above 120, which might be due to the lack of training data for those ranges of the HR.

In Tab. 4, the results for the test set of the 3 different scenarios in the IHD dataset are displayed. For the static scenario, both methods attain better performance, which was expected. The introduction of head rotations resulted in a noticeably lowered predictive capability. For PhysNet, a negative R^2 value is obtained, meaning that it performs worse than constantly predicting the HR as the mean HR of the test set. When introducing movement of the camera, the predictive capability of PhysNet is next to none. For the HR-rtCNN method, there is a significant decrease in the R^2 and r metrics. This could be due to the increased difficulty for the Mediapipe Face Mesh software to accurately detect the face. There is also an underrepresentation of data collected with a moving camera, which could have contributed to the increased error in the evaluation.

The resulting computational time for the two deep learning based methods is observable in Tab. 5. Due to the video sequences having 20 frames per second, there is 50 ms between each frame. To achieve real-time performance, the computations need to be faster than this. For HR-rtCNN, the combined time for the pre-processing and the inference is about 18 ms, implying real-time applicability. The low standard deviation of both the pre-processing time and the inference time also support this. For PhysNet, the pre-processing takes about 101 ms, which does not allow for real-time usage. The reason for the difference in computational time of the pre-processing is caused by PhysNet using the original resolution of the videos, as well as extracting a 128×128 normalized region of the face.

7. Conclusions and Future Work

The main contribution of the work presented in this paper is the proposal of an end-to-end neural network based framework designed to estimate subjects' HRs in real-time from short two-second video sequences accompanied by dynamic illumination. Five face ROIs are extracted, enabling a higher resilience to the subject's movements and removing the background, thereby resulting in a better signal quality. The framework also predicts the HR directly instead of predicting a PPG signal, as many other methods do. Furthermore, the evaluation of the deep neural network on the COHFACE, PURE, BP4D+, and IHD datasets, shows that it generally outperforms other methods with an RMSE value of 8.920 and R^2 value of 0.452 for the most challenging situations. Also, with an average computing time of 18 ms per frame (compared to 50 ms between frames), faster than real-time performance is ensured. Hence, the main conclusion is that it is possible to estimate the HR of a subject from a distance with good performance in realistic situations characterized by movements and dynamic illumination. However, considering the limitations of the dataset used for training, which inevitably result in a relatively poor domain generalization of the current model, a recommendation for the future is to train a model using a larger dataset collected through a well-designed and specific dynamic illumination to obtain a better generalization ability.

One of the most limiting factors when working with rPPG based on deep learning, is the need for representative data, displaying demographic diversity, situations with dynamic illumination, and pixel distance sizes allowing for greater HR detection. Therefore, gathering new data should be prioritized to make the method more practically applicable. Generation of rPPG signals with HR is an optional approach to circumvent this limitation. However, even though GAN-based methods for PPG signal generation have been proposed [24], more in-depth evaluations of HR estimation networks are needed.

Acknowledgements. This work has received funding from the European Union Horizon 2020 program (grant agreement no. 101021957 – NIGHTINGALE).

References

[1] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org. 5

[2] John Allen. Photoplethysmography and its application in clinical physiological measurement. *Physiological Measurement*, 28(3), 2007. 1

[3] Guha Balakrishnan, Fredo Durand, and John Guttag. Detecting pulse from head motions in video. In *Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3430–3437. IEEE, 2013. 2

[4] Lokendra Birla, Sneha Shukla, Anup Kumar Gupta, and Puneet Gupta. ALPINE: Improving remote heart rate estimation using contrastive learning. In *Proceedings of the IEEE Winter Conference on Applications of Computer Vision*, pages 5029–5038. IEEE, 2023. 2

[5] Giuseppe Boccignone, Donatello Conte, Vittorio Cuculo, Alessandro D'Amelio, Giuliano Grossi, and Raffaella Lanzarotti. An open framework for remote-PPG methods and their assessment. *IEEE Access*, 8:216083–216103, 2020. 2

[6] Deivid Botina-Monsalve, Yannick Benezeth, and Johel Miteran. RTrPPG: An ultra light 3DCNN for real-time remote photoplethysmography. In *Proceedings of the 2022 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2146–2154. IEEE, 2022. 2

[7] Weixuan Chen and Daniel McDuff. DeepPhys: Video-based physiological measurement using convolutional attention networks. In *Proceedings of the 2018 European Conference on Computer Vision (ECCV)*, pages 349–365. Springer, 2018. 2

[8] Chun-Hong Cheng, Kwan-Long Wong, Jing-Wei Chin, Tsz-Tai Chan, and Richard H. Y. So. Deep learning methods for remote heart rate measurement: A review and future research agenda. *Sensors*, 21(18), 2021. 2

[9] François Chollet. Keras. https://github.com/fchollet/keras, 2015. 5

[10] Karan Chopra, Daniel Calva, Michael Sosin, Kashyap Komarraju Tadisina, Abhishake Banda, Carla De La Cruz, Muhammad R. Chaudhry, Teklu Legesse, Cinithia B. Drachenberg, Paul N. Manson, and Michael R. Christy. A comprehensive examination of topographic thickness of skin in the human face. *Aesthetic Surgery Journal*, 35(8):1007–1013, 2015. 3

[11] Ananyananda Dasari, Sakthi Kumar Arul Prakash, László A. Jeni, and Conrad S. Tucker. Evaluation of biases in remote photoplethysmography methods. *npj Digital Medicine*, 4(1), 2021. 1, 2

[12] Gerard de Haan and Vincent Jeanne. Robust pulse rate from chrominance-based rPPG. *IEEE Transactions on Biomedical Engineering*, 60(10):2878–2886, 2013. 1, 5

[13] Mohamed Elgendi. *PPG signal analysis: An introduction using MATLAB*, chapter 2, pages 27–52. CRC Press, 1 edition, 2020. 1

[14] Haoyuan Gao, Xiaopei Wu, Jidong Geng, and Yang Lv. Remote heart rate estimation by signal quality attention network. In *Proceedings of the 2022 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2122–2129, 2022. 2

[15] Magdalena Graczyk, Tadeusz Lasota, Zbigniew Telec, and Bogdan Trawiński. Nonparametric statistical analysis of machine learning algorithms for regression problems. In *Proceedings of the International Conference on Knowledge-Based and Intelligent Information and Engineering Systems*, pages 111–120. Springer, 2010. 2

[16] Guillaume Heusch, André Anjos, and Sébastien Marcel. A reproducible study on remote heart rate measurement. *arXiv*, 2017. 3

[17] Min Hu, Dong Guo, Xiaohua Wang, Peng Ge, and Qian Chu. A novel spatial-temporal convolutional neural network for remote photoplethysmography. In *Proceedings of the 2019 12th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*. IEEE, 2019. 2

[18] Dae-Yeol Kim, Kwangkee Lee, and Chae-Bong Sohn. Assessment of ROI selection for facial video-based rPPG. *Sensors*, 21(23), 2021. 3

[19] Camillo Lugaresi, Jiuqiang Tang, Hadon Nash, Chris McClanahan, Esha Uboweja, Michael Hays, Fan Zhang, Chuo-Ling Chang, Ming Yong, Juhyun Lee, Wan-Teh Chang, Wei Hua, Manfred Georg, and Matthias Grundmann. Mediapipe: A framework for perceiving and processing reality. In *Proceedings of the 2019 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2019. 3, 5

[20] Daniel J. McDuff, Ethan B. Blackford, and Justin R. Estepp. The impact of video compression on remote cardiac pulse measurement using imaging photoplethysmography. In *Proceedings of the 12th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2017)*, pages 63–70. IEEE, 2017. 7

[21] OXY-200. Desktop pulse oximeter. https://www.gimaitaly.com/prodotti.asp?sku=35101&dept_selected=622&dept_id=6220. Accessed: 2022-10-17. 3

[22] Hamidur Rahman, Mobyen Uddin Ahmed, Shahina Begum, and Peter Funk. Real time heart rate monitoring from facial RGB color video using webcam. In *Proceedings of the 29th Annual Workshop of the Swedish Artificial Intelligence Society (SAIS)*. Linköping University Electronic Press, 2016. 2

[23] F. Bereksi Reguig. Photoplethysmogram signal analysis for detecting vital physiological parameters: An evaluating study. In *Proceedings of the 2016 International Symposium on Signal, Image, Video and Communications (ISIVC)*, pages 167–173. IEEE, 2016. 1

[24] Rencheng Song, Huan Chen, Juan Cheng, Chang Li, Yu Liu, and Xun Chen. PulseGAN: Learning to generate realistic pulse waveforms in remote photoplethysmography. *IEEE Journal of Biomedical and Health Informatics*, 25(5):1373–1384, 2021. 8

[25] Radim Spetlik, Vojtech Franc, Jan Cech, and Jiri Matas. Visual heart rate estimation with convolutional neural network. In *Proceedings of the British Machine Vision Conference (BMVC)*, 2018. 2

[26] Ronny Stricker, Steffen Muller, and Horst-Michael Gross. Non-contact video-based pulse rate measurement on a mobile service robot. In *Proceedings of the 23rd IEEE International Symposium on Robot and Human Interactive Communication (Ro-Man 2014)*, pages 1056–1062. IEEE, 2014. 3

[27] Antonio Torralba and Alexei A. Efros. Unbiased look at dataset bias. In *Proceedings of the 2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1521–1528. IEEE, 2011. 2

[28] Wim Verkruysse, Lars O. Svaasand, and J. Stuart Nelson. Remote plethysmographic imaging using ambient light. *Optics Express*, 16(26):21434–21445, 2008. 1

[29] Wenjin Wang, Albertus C. den Brinker, Sander Stuijk, and Gerard de Haan. Robust heart rate from fitness videos. *Physiological Measurement*, 38(6):1023, 2017. 2

[30] Wenjin Wang, Sander Stuijk, and Gerard de Haan. Exploiting spatial redundancy of image sensor for motion robust rPPG. *IEEE Transactions on Biomedical Engineering*, 62(2):415–425, 2015. 1

[31] Zitong Yu, Xiaobai Li, and Guoying Zhao. Remote photoplethysmograph signal measurement from facial videos using spatio-temporal networks. In *Proceedings of the British Machine Vision Conference (BMVC)*, 2019. 2, 5

[32] Zheng Zhang, Jeffrey M. Girard, Yue Wu, Xing Zhang, Peng Liu, Umur Ciftci, Shaun Canavan, Michael Reale, Andrew Horowitz, Huiyuan Yang, Jeffrey F. Cohn, Qiang Ji, and Lijun Yin. Multimodal spontaneous emotion corpus for human behavior analysis. In *Proceedings of the 2016 IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3438–3446. IEEE, 2016. 2