Prediction of blood volume pulse waveform features using remote PPG

Ruben Sangers

TU Delft

Prediction of blood volume pulse waveform features using remote PPG

by

Ruben Sangers

to obtain the degree of Master of Science
at the Delft University of Technology,
to be defended publicly on Friday July 1, 2022 at 1:00 PM.

Student number: 4591771
Project duration: October 27, 2021 – July 1, 2022
Thesis committee: Dr. J. C. van Gemert, TU Delft, supervisor
Dr. W. P. Brinkman, TU Delft
An electronic version of this thesis is available at http://repository.tudelft.nl/.

TU Delft

.

Preface

This report documents my research on remote photoplethysmography for my Master Thesis to obtain the Computer Science Master degree at TU Delft. My work was conducted at the Computer Vision Lab, under supervision of Dr. Jan van Gemert along with daily supervision of Marian Bittner. This document is structured into two parts: the first part is a scientific paper, presenting novel insights into the influence of input representations on the prediction of blood volume pulse features using remote photoplethysmography, the second part of my thesis contains supplementary material, consisting of background information, additional experiments and a discussion.

Six years ago, I started my studies as a nanobiologist, but I soon became fascinated about the diverse possibilities of Artificial Intelligence. It was this fascination that convinced me to switch to Computer Science after obtaining my Bachelor degree, a decision that I have not regretted. The lectures by Jan van Gemert on deep learning only enlarged my enthusiasm, and especially applications of AI in the visual domain are my biggest interest. It is therefore not a surprise that I chose to do this Master Thesis at the Computer Vision lab.

First of all, I would like to thank Marian Bittner for his daily supervision and help in writing the research paper. Our discussions have always been very insightful for me and your experience has helped me a lot in developing my work. Secondly, I would like to thank Jan van Gemert for our meetings, your perspective always resulted in new ideas and your guidelines have helped me a lot to become a better researcher. Additionally, I would like to extend my gratitude to Dr. Willem-Paul Brinkman for his interest in my work and his role as a member of my Thesis committee.

Lastly, I would like to thank my family, friends and girlfriend for their support during my thesis. I could always rely on you to keep me motivated and your perspectives on my Thesis have greatly helped me to place my research in the bigger picture. I am proud to end my hard work of the last six years with this document and I hope that it will help people in their research.

Ruben Sangers
Delft, June 2022

Contents

1

Research paper

Prediction of blood volume pulse waveform features using remote PPG

Ruben Sangers, Marian Bittner and Jan van Gemert

Delft University of Technology, The Netherlands

Abstract. Contactless measurement of changes in blood volume by exploiting the color fluctuations in the face is a technique commonly referred to as remote photoplethysmography (rPPG). Recent developments show promising results for heart rate estimation from low-cost cameras, making applications in remote healthcare possible. Remote PPG applications in at home diagnostics focus predominantly on heart rate estimation, while other features of the blood volume pulse can provide valuable information as well, but have scarcely been studied using rPPG. In this work, we aim to lay a foundation for rPPG feature prediction. We study pulse wave prediction using a variety of input representations, model architectures and datasets to thereby investigate which combined approaches are the most promising. Our results show which input representation is most suitable based on the feature of interest and demonstrate the ability to predict pulse waveform features using rPPG. These results take the first steps towards including the prediction of a wide range of waveform properties to make more remote health monitoring possible.

Keywords: remote photoplethysmography, blood volume pulse, waveform features, deep learning, input representations

1 Introduction

Monitoring of heart rate and heart rate related measures, in an at-home setting, holds great and exciting opportunities for long-term care for infants or elderly people as it allows for remote diagnosis, which can be more time effective for doctors and reduce the risk of spreading contagious diseases. Traditionally, contact based methods have been used for heart rate monitoring, such as photoplethysmography (PPG), which aims to measure the blood volume pulse from light absorption of the skin using a contact sensor at a finger or ear lobe, allowing cardiovascular measurement of the patient. However, this way of measuring blood volume changes has several disadvantages, as it requires specialized hardware and direct skin contact to the patient, which might be uncomfortable or even harmful [1][30]. In contrast to traditional PPG, remote PPG (rPPG) does not rely on contact sensors but aims to measure the photoplethysmogram, or blood volume changes, using a camera directed at the face. This has numerous

advantages, as it does not rely on specialized sensors, enables at-home diagnostics and prevents harmful skin-contact for infants and patients with a sensitive skin.

Over the last decades, remote PPG has seen a huge increase in interest across academic fields, however its focus has been primarily on the prediction of heart rate, as it has many applications in healthcare. However , a wide range of other features that characterise the photoplethysmography waveform exist, which have barely been studied with remote PPG. These are features of the pulse waveform, such as its amplitude and rise time (Figure 2), which can give insights into important cardiovascular properties that in turn are related to physiological conditions and the well-being of the patient such as blood pressure [33], stroke volume [23], arteriosclerosis [46] and anxiety [45]. Knowing these properties could aid physicians in giving more accurate diagnoses via video chat or monitor longitudinal difference in a home setting. However, as many of the features are less pronounced than the peaks and valleys of individual heartbeats in the blood volume pulse, an accurate non-contact way of measuring the pulse is necessary.

While the first rPPG methods were based on signal processing techniques, in recent years deep learning techniques, which have already been successfully applied to multiple problems in computer vision, have achieved state-of-the-art performance on remote heart rate estimation. Recent advancements in remote PPG research have made it possible to measure heart rate within error rates of two pulses per minute [21]. Commonly one or multiple frames of the raw video are pre-processed before being used as input for deep learning models. As the influence of input representation and model architecture have only been investigated in the context of heart rate, we will study how well other features of the PPG waveform can be predicted to thereby get a more complete view of the possibilities of remote PPG.

In this paper, we study the prediction of four different pulse waveform features: the heart rate, rise time, pulse wave amplitude and the pulse area. Moreover, we hypothesize that feature estimation can be made easier when choosing a suitable representation. We therefore study the effect of three input representations on the prediction, which differ in the region of interest that is used (the complete frame or pre-selected parts of a face) and their domain (frequency- or time-domain). We thereby aim to give a broad insight into the opportunities of measuring a variety of pulse waveform features using deep learning techniques and their relation with input representations (Figure 1).

Summarized, our contributions are as follows:

- We show that direct prediction of PPG pulse waveform features using deep neural networks is possible, to the best of our knowledge we are the first to do so.
- Our study gives insights into which waveform features have the most potential to be measured accurately.

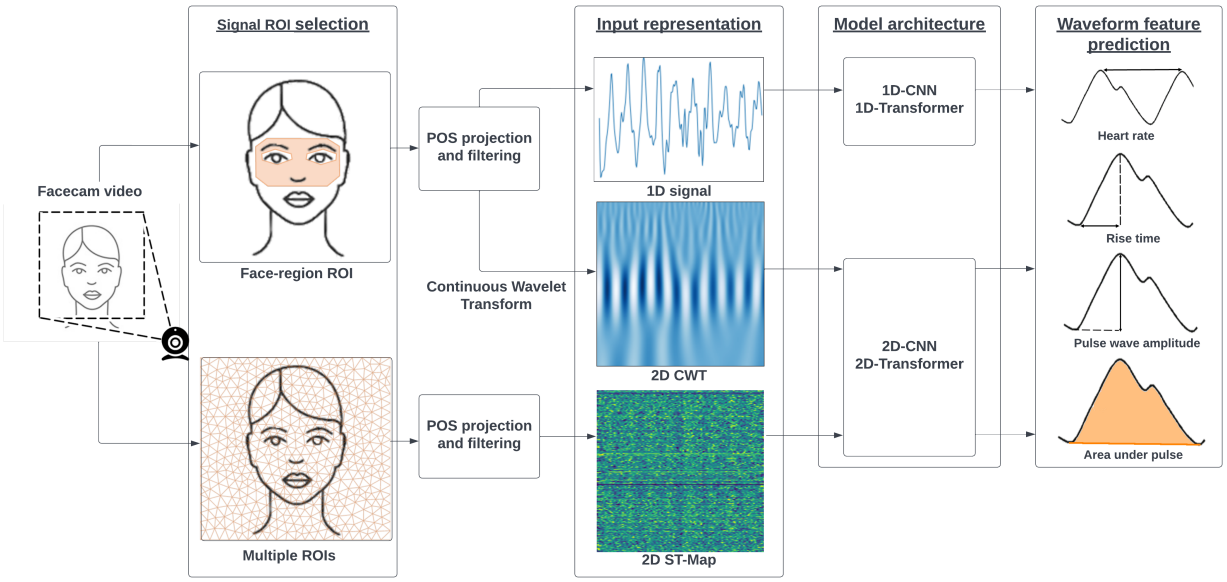

Fig. 1. Overview of our methodology. We investigate PPG waveform feature prediction using three different ways of representing the input video files to our models. Two different model-types are evaluated on direct prediction of a range of pulse waveform features from these input representations to get insights into the possibilities of PPG feature prediction.

– We show the influence of various input representations on the waveform feature prediction performance.

2 Related work

2.1 PPG waveform features

Next to its use to track heart rate across multiple waves in the blood volume pulse, the shape of a single wave holds important additional information. The PPG waveform resulting from the blood volume pulse in general consists of two peaks: the systolic and diastolic peak respectively. A study by Elgendi [15] was one of the first to highlight the different PPG features to increase the understanding of the embedded information in the PPG curve. Current literature has already shown the possibility of estimating blood pressure based on the full morphology of the PPG waveform [33], but also individual features hold various correlations with underlying physiological conditions of the patient. The rise time, the time between the pulse wave begin and the systolic peak, has for example been shown to correlate with arterial stiffness [2] and cardiovascular diseases such as hypertension [13] and arteriosclerosis [46], while the amplitude of the systolic peak, also known as the pulse wave amplitude, is related with stroke volume [23] and the local vascular distensibility [14], which can give important insight into cardiovascular health. Overall the most important waveform features can be classified into temporal features, amplitude related features and area related features. An overview of the most common pulse wave features and

some of their correlations with physiological conditions can be found in Table 1. These features can give valuable insight into the physiology of a patient, however, the main focus on current contact less vital sign research in recent years has been on remote heart rate estimation.

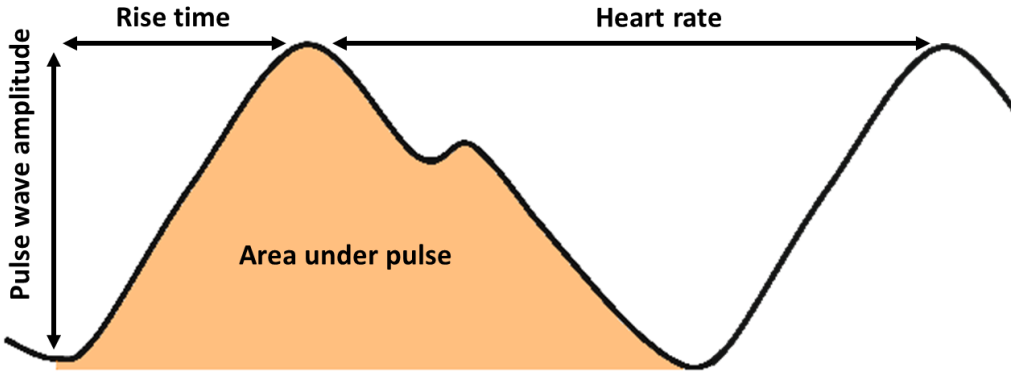

Fig. 2. Schematic overview of the blood volume pulse waveform features we study. In the figure, the change in blood volume over time is shown.

Table 1. Overview of pulse waveform features and their correlations with physiological conditions. + and − indicate a positive or negative correlation respectively.

Class	Feature	Correlations
Temporal	Pulse propagation time	− Age [22]
		+ Artery stiffness [28]
	Diastolic time	− Blood pressure [37]
	Pulse width half-height	+ Systemic vascular resistance [4]
		− Blood pressure [37][3]
	Heart rate variability	+ Stress-levels and anxiety [10]
	Rise time	+ Arteriosclerosis [2][13]
		+ Hypertension [2][13]
	Dicrotic notch time	− Heavy exercise [42]
Amplitude	Pulse wave amplitude	+ Stroke volume [23]
		+ Local vascular distensibility [14]
		+ Blood pressure [12]
	Augmentation index	+ Arterial stiffness [9]
	Stiffness index	+ Age [22]
Area	Pulse area	− Motor reaction to skin incision [31]
	Inflection point area	+ Cardiovascular diseases [43]

2.2 Remote photoplethysmography

Verkruysse et al. [40] were one of the first to show the possibility of measuring the PPG pulse using an RGB camera. Using low-cost equipment and natural lighting,

they showed that it is possible to characterize vital signs such as heart rate by measuring the color changes in the face over time. The underlying principle causing these light reflection fluctuations of the skin follows the Beer-Lambert law [35], which states that the light absorption of blood is proportional to the hemoglobin concentration in the blood and the penetration of light into the skin. The cardiac cycle results in varying hemoglobin concentrations of the blood, leading to a varying light absorption by the skin. The remote PPG (rPPG) setup commonly consists of three elements: a light source, a person, and a camera. Based on Shafer's dichromatic reflectance model [32], the reflection of a skin pixel over time can be modelled by the specular reflection and the diffuse reflection of the skin [44]. The specular reflection is a mirror-like light reflection by the skin, while the diffuse reflection is associated with the absorption of light by the skin tissue. As the specular reflection does not contain information of the blood volume pulse, the goal of remote PPG methods is in general to focus on the diffuse reflection using signal-processing or data-driven methods to thereby measure the blood volume pulse.

2.3 Signal processing rPPG methods

The findings by Verkruysse et al. [40] sparked the development of specialized signal processing pipelines and algorithms to increase the quality of the blood volume pulse signal by removing noise. Existing algorithms for signal decomposition were used as blind-source signal separation techniques such as independent component analysis (ICA) [29] and principal component analysis (PCA) [5], which work especially well for removing noise small in amplitude and periodic. Another commonly used type of methods relies on transformation of the RGB signal to alternative color spaces to, for example, eliminate the influence of specular reflections on the signal. Examples of this are the hue channel of the HSV color space, which does not depend on the intensity of the reflected light from the surface [39], or the POS method, which defines a plane orthogonal to the skin tone on which the pulsatile signal is projected, to remove influence of different skin tones [44]. Additional commonly used operations for remote PPG are spatial averaging to cancel out noise by camera quantisation, (band-pass) filtering to remove signals that are not in the range of natural heart rate frequencies [36], or transforming the rPPG signal to the frequency-domain, for example by using power spectral density estimation [29] or the Continuous Wavelet Transform (CWT) [8]. These methods can be used on their own, or in combination with data-driven methods to pre-process the rPPG signal.

2.4 Data-driven rPPG methods

In recent years, the shift has been made from the use of purely signal processing methods to data-driven methods, usually involving deep learning. The strength of these methods is that, by learning features from data instead of hand-coding them, they can detect more complex features and are often better able to handle a large variety in data. They are, however, often far more computationally

expensive, and require a large amount of training data to be able to perform well.

Deep-learning based rPPG methods for heart rate estimation can in general be divided into two categories: direct heart rate estimation, or PPG wave estimation followed by post-processing to extract the heart rate. DeepPhys [11] has been a very successful model for PPG wave estimation. It consists of two branches, an appearance branch and a motion branch. The appearance branch is trained on producing a spatial attention mask for the video, which is then combined with the normalized frame differences to estimate the rPPG signal by the motion branch. MTTS-CAN [20] builds onto DeepPhys, but improves the time efficiency of the rPPG computation by introducing a temporal shift module. This allows the sharing of temporal information without the need for computationally expensive 3D convolutions, making real-time heart rate estimation possible. The authors of the PhysFormer [47] architecture were one of the few that have applied a Transformer model for PPG waveform estimation. They claim that by using a Transformer they are able to learn the long-range spatial-temporal interactions of the blood volume pulse more successfully.

One of the first attempts to attempt direct heart rate prediction is the HR-CNN by Spetlik et al. [34]. It uses two convolutional neural networks (CNNs): one functioning as an extractor to estimate the rPPG signal, which is then used as the input to a second CNN which predicts the heart rate from this signal. Rhythmnet [26] uses a different approach, as they use a spatial-temporal map to represent the input, from which they then train a CNN to estimate the heart rate from 10 second spatial-temporal maps. In a similar way, we will attempt to directly predict PPG waveform features using remote PPG.

2.5 Region of interest selection

Most algorithms, whether signal processing based or data-driven, rely on the face for extraction of the signal. A common first step is therefore often to define a region of interest in the face from which the pulsatile signal is extracted. There are several ways this is done. Early research into remote PPG relied on the subject not moving such that a region of interest, usually around the cheeks and forehead, could be selected to extract the pixel signal from. As this can not be assumed in practice, later methods apply face detection methods such as the Viola-Jones algorithm [41] to extract the pixel color from a fixed region of the face. The main disadvantage of these methods however, is that by pre-defining the region of interest you limit the amount of information you feed into the algorithm: although the region of interest is chosen based on the visibility of the blood volume pulse signal, other regions with only a weak signal or even none can be relevant as well. Nowara et al. [27] for example developed a method that learns an inverse attention mask, to thereby find the regions containing no pulsatile signal which can then be used to estimate noise and illumination changes. Niu et al. [26] uses face detection, but instead of pre-defining the region of interest they divide the face into n regions of equal size, and then have their model learn which of the ROIs to use to what extend in its prediction. IBIS [7] is

a similar technique, but instead of dividing the video into n equal sized squares, it divides pixels into regions that are closest in the spatial as well as in the color domain. This thereby keeps the contours of the image intact, resulting in much less distorted colour signals for each region [44]. However, as all of these input representations focus on heart rate estimation, it is unclear if they work equally well for the estimation of waveform properties.

3 Method

In this work, we explore direct-estimation methods for predicting PPG waveform features using deep neural networks. We do this using different rPPG signal representations, including pre-processing steps to reduce noise in the original signal, and test multiple common deep learning architectures.

3.1 Data pre-processing

For simplicity, we cut each video into fragments of 10 seconds to serve as input for our algorithms. For each fragment, we calculate the average value for each of our waveform features within this 10 second time frame. We use contact-PPG signals measured at the finger to train our algorithms. From these signals, we obtain the ground truth pulse wave features using a derivative based peak- and valley-detection algorithm [6].

The features characterising the pulse waveform can in general be divided into three classes: temporal, amplitude and area features. To study the ability to detect a wide range of waveform features, we chose to study at least one feature from each of these categories. The features we will look at are the following: the heart rate (HR), rise time (RT), pulse wave amplitude (PWA) and the area under the pulse (AUP) (Figure 2).

3.2 Input representations

Different input representations present the input signal in an alternative way, thereby affecting the way waveform features are presented. The effect of the representation might differ based on the feature of interest, and therefore we compare three different input representations (Figure 3) and their effect on the capability of predicting waveform feature values.

Our first input representation is the 1D pre-processed signal. To generate this representation, we start of with face finding using the Viola-Jones algorithm [41] and facial landmarks detection using an active appearance model [19]. These facial landmarks are then used to select the upper region of the face (excluding the eyes) over which we perform spatial averaging to capture a 1D RGB pixel signal over time [16]. This step is then followed by projection to a plane orthogonal to the skin tone using the POS method [44] and lastly a Butterworth band-pass filter with cutoff frequencies 0.5 and 6 Hz.

As we aim for consistency in processing all input representations, we use the same pre-processing pipeline for the Continuous Wavelet Transform (CWT) representation. The only difference is that after generating the one-dimensional signal, we apply a CWT using the Morlet mother wavelet to get a two-dimensional representation of the frequencies present in the signal.

Our third representation is a spatial-temporal map (ST-map). For this, we use the IBIS [7] method, for spatially segmenting the video into n superpixels based on the location and colour of pixels. Here, we chose $n = 240$ and $c = 20$, which defines the weighting between spatial and colour information on determining superpixel groupings. This gives us n temporal signals, each containing the average RGB pixel values of each of the superpixels over time. Similar to the other two representations, we project these signals using the POS method followed by the same band-pass filter to remove unwanted noise.

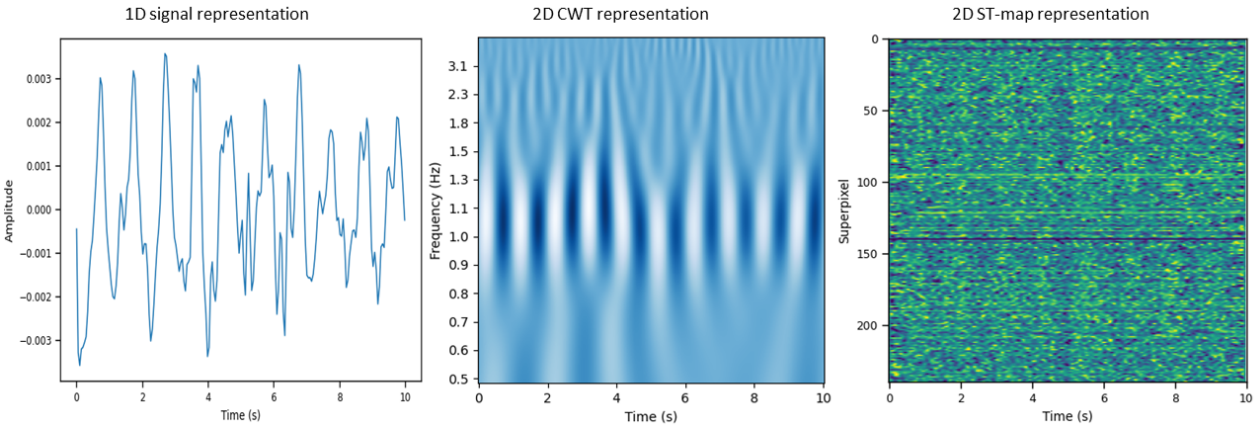

Fig. 3. Visualisation of the different input representations we evaluate for waveform feature prediction. The representations differ based on their domain (time- or frequency-domain) and region of interest that is used (the complete frame or pre-selected parts of the face).

3.3 Models and training procedure

We use two different types of neural network architectures: a CNN, which uses convolutional operations to extract features, and a Transformer, which uses self-attention to learn the relations within the data. As these two models differ in their fundamental operation to learn features, we expect that they might benefit from different input representations. At the same time, by using two different model types we can evaluate the suitability of the different representations in a wider range of situations.

For the CNN, we use the widely used Resnet18 [17] and a one-dimensional adaptation of this architecture for the 1D data [18] using the same hyperparameters but with 10 residual blocks. The Transformer model we use for the two-dimensional data is the base model of the Data-efficient Image Transformer

(DeIT) [38] while for the one-dimensional data, we use an architecture similar to [24], where we use four convolutional layers to learn an embedding, four attention heads, eight self-attention layers and a model dimension $d_m = 256$.

Based on training curves, we use 200, 20 and 20 epochs for training on the the VIPL-HR dataset for the 1D-, CWT- and ST-map- representations respectively, while for the VicarPPGBeyond dataset, we use 200, 60 and 60 training epochs respectively. After every epoch, we evaluate the performance on the model on the validation set, and we chose the model with the best validation set performance as our final model, which we evaluate on the test set. We train the models using an L1-loss and they are evaluated using the Mean Absolute Error (MAE), Root Mean Square Error (RMSE) and Pearson correlation coefficient (ρ).

3.4 Baseline

To be able to test if our models are capable of learning relevant features from the input, we design a naive baseline algorithm that achieves in general the lowest possible error without using any input features. It does this by calculating the average of all target output values it has previously seen, and uses this value as its prediction when presented a new input.

4 Experiments

4.1 Datasets

We test our method on two publicly available datasets VIPL-HR [25] and VicarPPGBeyond (not published yet). VIPL-HR is a dataset commonly used for heart rate estimation and was especially collected for the training of data-driven rPPG methods. It was collected by the Institute of Computing Technology Chinese Academy of Sciences, and is a dataset that presents scenarios that are much less constrained than in previous databases. The dataset includes variations in illumination, head movement and a diversity of camera devices to thereby mimic a natural environment for remote PPG measurement. The data consists of 107 participants and 3130 videos in which the participants follow 9 different scenario's with respect to e.g. head movement and recording method. To reduce the dataset size, videos are compressed using the MJPG codec. The videos have a frame rate of 25 or 30 fps (depending on the camera device) and are on average 30 seconds in length.

The VicarPPGBeyond dataset has recently been acquired with the goal of measuring a variety of vital signs such as respiration rate, blood oxygen saturation and heart rate. The data consists of 15 subjects in natural light conditions, recorded with a consumer-grade camera (Logitech C925e). Videos have been acquired with a duration of 2-3 minutes, in which the participants perform various tasks such as holding their breath for 30 seconds and performing small motions. The recordings are acquired with a frame rate of 30 fps and are uncompressed. This dataset will be made publicly available in a future publication. Distributions of the heart rate (HR), rise time (RT), pulse wave amplitude (PWA) and pulse area (AUP) for both datasets can be found in Figure 4.

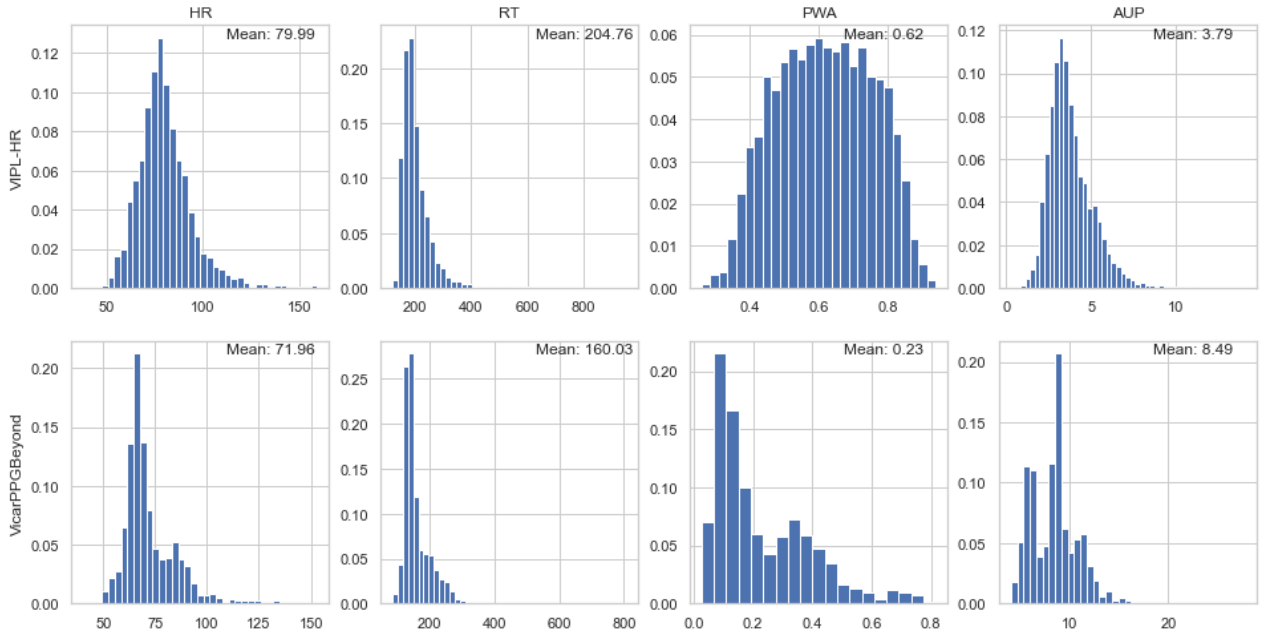

Fig. 4. Distributions of the heart rate (HR), rise time (RT), pulse wave amplitude (PWA) and area under pulse (AUP) for the 10-second clips of the VIPL-HR and VicarPPGBeyond dataset. We see that the VIPL-HR dataset in general has distributions that have a more Gaussian shape than the VicarPPGBeyond distributions, possibly due to the higher number of participants included in the VIPL-HR dataset.

4.2 Evaluation methods on PPG signal

Experimental setup: The remote PPG signal can be regarded as an extremely noisy version of a PPG signal, with a varying distance between sensor and skin, inconsistent composition of light-source and camera quantization as potential sources of noise amongst others. The PPG ground truth provided with the analyzed dataset can be thus regarded as the 'cleanest' version of the remote PPG signal. We used these clean signals to design a toy-experiment to test whether our methods are capable of predicting individual waveform features under ideal circumstances. With the reasoning that if a method is not capable to predict waveform features on the ground-truth signal, there is little hope that it would be able to do this on the real signals which are far noisier.

The different model types are trained on predicting one of the waveform features from the ground-truth signal for VicarPPGBeyond and VIPL-HR. Two different input representations are tested: the 1D-signal representation and the 2D-CWT. The input representations are generated according to our described methodology (section 3.2), excluding the filtering and POS-projection step. Models are trained using a subject-exclusive 70/15/15 train/validation/test split, meaning that participants only appear in one of the three sets.

Results: Our results can be found in Figure 5. The true-vs-estimated graphs can give us a clear insight into how well the different model-input combinations are able to predict the value of the waveform features. We see that the 1D-CNN is quite successful in predicting all four features: the predicted values follow the

true-values line relatively well, except for the pulse wave amplitude prediction on VicarPPGBeyond. The CWT-CNN performs comparable, although it seems a bit worse on both PWA prediction tasks. The Transformer (TF) trained on the CWT representations is less successful: on half of the features, it seems to do reasonable well, while on others, it converges to predicting the same value for every input. Lastly, the 1D-TF is not able to learn any of the waveform features, predicting the same value regardless of the input.

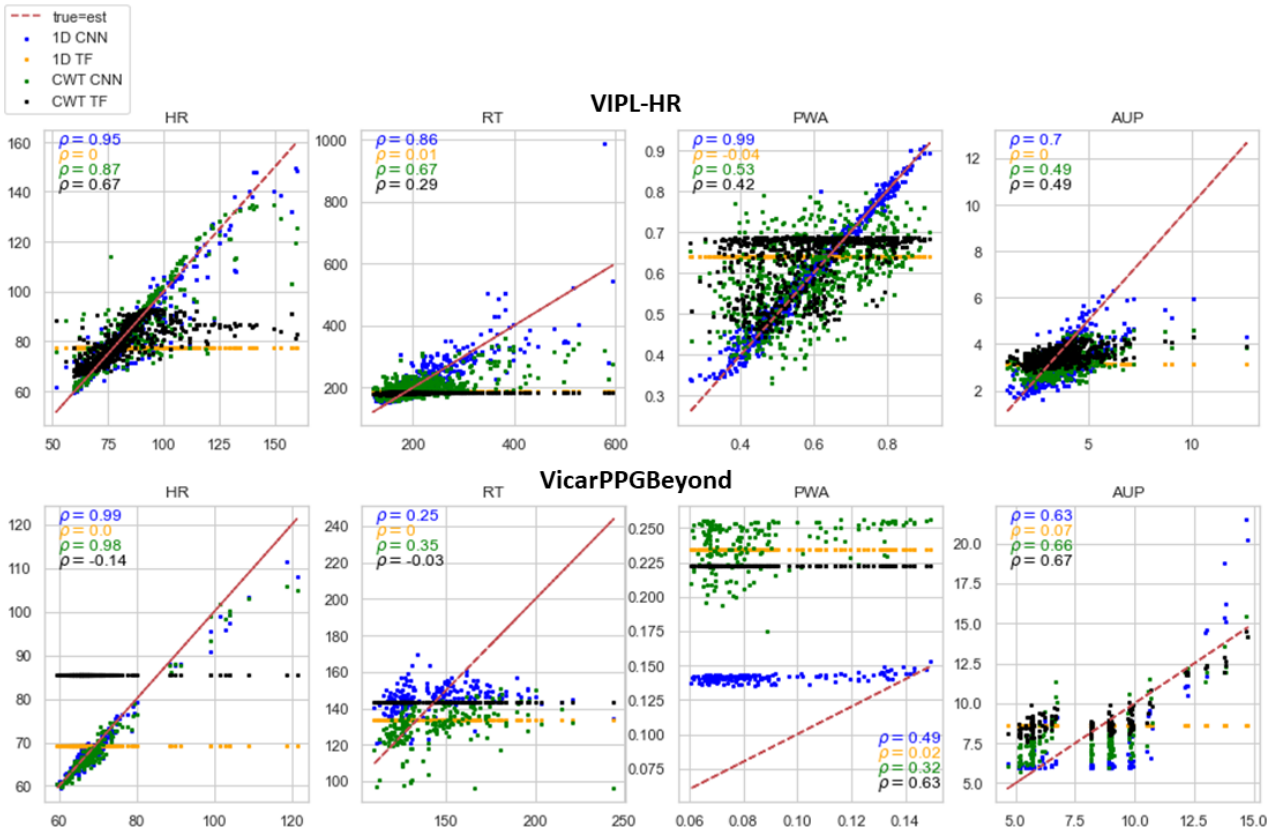

Fig. 5. True-vs-estimated values for the different methods trained on predicting feature values from PPG signals of VicarPPGBeyond and VIPL-HR. Pearson correlations are shown as annotations. As can be seen, most methods perform well for all features on the VIPL-HR dataset, while they struggle for predicting rise time (RT) and pulse wave amplitude (PWA) on VicarPPGBeyond. Also notice that both Transformer (TF) models are outperformed by the CNNs, especially the TF trained on 1D data.

4.3 Influence of input representation on waveform feature prediction

Experimental setup: We train the different model types on predicting one of the four waveform features using one of the three different input representations, to investigate whether it is possible to estimate individual waveform features from the far more noisy rPPG signals and at the same time determine which pipeline is best suited for which representation. We do this on both the VicarPPGBeyond and the VIPL-HR dataset and use 7-fold subject-exclusive

cross-validation. We take the average over all seven folds and report this as the final performance.

Evaluation metrics: To compare the performance of the method-input combinations for each of the waveform features, we use the Mean Absolute Error (MAE), Root Mean Square Error (RMSE) and Pearson correlation coefficient between the predicted feature value and the ground-truth value for each 10-second signal clip.

Results: Our results are depicted in Table 2, 3 and 4, and have been visualized in Figure 6. Based on MAE and RMSE (Table 2 and 3 respectively), we can see that the best model-input combination differs based on the waveform feature of interest. For predicting heart rate, the CNN trained on the one-dimensional input representation seems to be the best choice as it outperforms the baseline reasonably well. For rise time prediction, the best model-input combination is less clear: overall, the 1D-Transformer seems best, though it is only slightly better than the 1D-CNN and CWT-Transformer performance. For pulse wave amplitude prediction, the Transformer trained on ST-maps performs the best on VicarPPGBeyond, although the difference with the baseline is only marginal and the baseline outperforms all methods on VIPL-HR. Lastly for pulse area prediction, the 1D-CNN seems again the best choice.

If we consider the Pearson correlation coefficients (Table 4, Figure 6), however, we see that some models have not been able to learn any useful features, achieving a correlation close to zero. Especially both spatial-temporal map methods perform bad, essentially predicting the same value for every input. The same holds for the Transformer trained on 1D-input signals: although it achieved an error well below baseline on predicting rise time for VicarPPGBeyond, this can probably be attributed to mere chance as the average ρ-values indicate that it does not learn any correlation.

If we combine the information from the error-values and Pearson correlation, the results suggest that the 1D-CNN, CWT-CNN and CWT-TF are the most successful in predicting waveform features. Especially the heart rate and pulse area can be predicted fairly well, while the rise time and pulse wave amplitude can not be predicted using the current experimental setup.

4.4 Influence of filtering on waveform feature prediction

Experimental setup: In the pre-processing of all three input representations, we use the same band-pass filter to exclude high- and low-frequency signals from the input. To evaluate the role of this filtering step in the final performance of our methods and to identify if we do not filter out important information, we have tested the performance of our methods with- and without band-pass filter on the VIPL-HR dataset. For this, we use a fixed 70/15/15 participant-exclusive train/validation/test split and denote the average performance over 5 different training iterations to test for significance.

Table 2. Mean absolute error for the different models and input representations on prediction of waveform features on VicarPPGBeyond and VIPL-HR. Scores outperforming the baseline are underlined and best performing model per feature is shown in bold. Error rates were obtained as the average using 7-fold cross-validation. Based on the MAE, we see that there is at least one model for each waveform feature that outperforms the baseline, except for PWA prediction on VIPL-HR.

MAE		**VicarPPGBeyond**				**VIPL-HR**			
		HR	**RT**	**PWA**	**AUP**	**HR**	**RT**	**PWA**	**AUP**
1D signal	**CNN**	**4.38**	17.7	0.156	2.00	**7.94**	**32.3**	0.121	**0.854**
	TF	10.05	**16.3**	0.175	2.30	10.36	32.9	**0.120**	0.982
2D CWT	**CNN**	5.43	22.5	0.185	**1.92**	8.08	33.3	0.121	0.859
	TF	14.69	16.8	0.162	2.02	10.78	33.5	0.125	0.887
2D ST-map	**CNN**	13.13	30.6	0.175	2.92	12.21	32.9	0.137	1.505
	TF	12.76	19.8	**0.151**	2.67	10.63	32.7	0.125	1.397
Baseline		11.60	25.6	0.157	2.13	10.44	35.8	0.119	0.996

Table 3. Root mean square error for the different models and input representations on prediction of waveform features on VicarPPGBeyond and VIPL-HR. Scores outperforming the baseline are underlined and best performing model per feature is shown in bold. Error rates were obtained as the average using 7-fold cross-validation. We see that there is at least one model for each waveform feature that outperforms the baseline, except for PWA prediction on VIPL-HR, which is in accordance with the MAE results.

RMSE		**VicarPPGBeyond**				**VIPL-HR**			
		HR	**RT**	**PWA**	**AUP**	**HR**	**RT**	**PWA**	**AUP**
1D signal	**CNN**	**6.70**	21.1	0.175	**2.60**	**11.1**	53.3	0.145	**1.17**
	TF	12.46	19.8	0.191	2.97	13.3	**51.2**	**0.142**	1.32
2D CWT	**CNN**	7.74	26.1	0.203	2.67	**11.1**	53.8	0.145	**1.17**
	TF	16.69	**19.7**	0.179	2.73	13.9	53.2	0.148	1.21
2D ST-map	**CNN**	15.66	34.2	0.193	3.64	39.7	54.2	0.439	4.26
	TF	15.23	22.9	**0.167**	3.35	13.6	52.9	0.152	1.95
Baseline		13.67	27.9	0.172	2.80	13.4	51.5	0.140	1.31

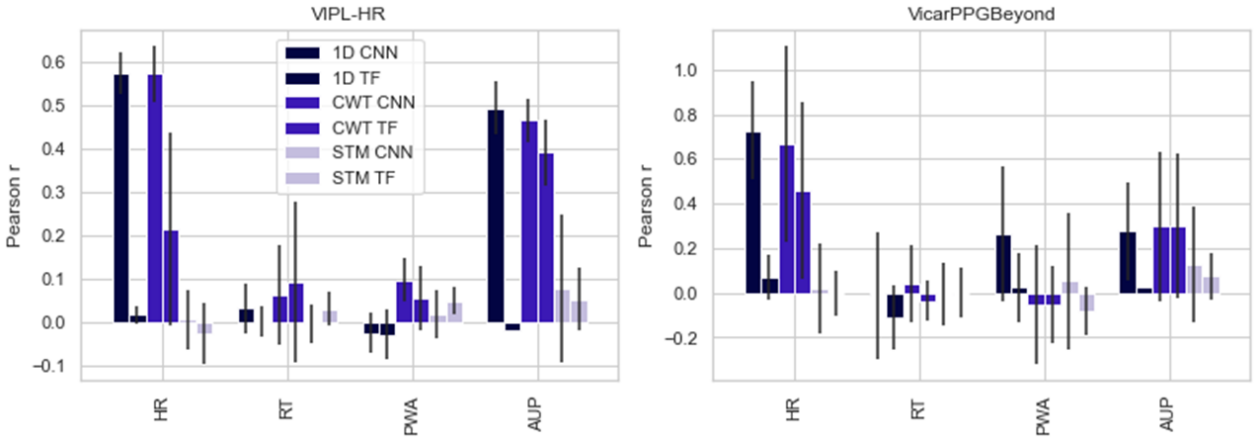

Fig. 6. Pearson correlation between the predicted and true values for the different models and input representations on prediction of waveform features on VIPL-HR (left) and VicarPPGBeyond (right). Pearson correlation coefficients were obtained as the average using 7-fold cross-validation and standard deviations are shown as error bars. We see that only for heart rate (HR) and area under pulse (AUP) prediction, there are models that achieve a reasonable correlation. Moreover, the standard deviation between folds is relatively high, indicating that the choice of fold greatly influences model performance.

Table 4. Pearson correlation coefficient between the true and estimated values for the different models and input representations on prediction of waveform features on VicarPPGBeyond and VIPL-HR. Coefficients were obtained as the average using 7-fold cross-validation and highest correlation per feature is shown in bold. Although some models achieve a good MAE and RMSE, based on the Pearson correlation we see that only the 1D-CNN and both models trained on CWT representations have learned a reasonable correlation. Moreover, the rise time and pulse wave amplitude can not be predicted using the current experimental setup.

Pearson ρ		**VicarPPGBeyond**				**VIPL-HR**			
		HR	**RT**	**PWA**	**AUP**	**HR**	**RT**	**PWA**	**AUP**
1D signal	**CNN**	**0.73**	-0.01	**0.27**	0.28	**0.58**	0.03	-0.02	**0.49**
	TF	0.07	-0.11	0.02	0.03	0.02	0.00	-0.03	-0.02
2D CWT	**CNN**	0.67	**0.04**	-0.05	**0.30**	0.57	0.06	**0.10**	0.47
	TF	0.46	-0.03	-0.06	**0.30**	0.21	**0.09**	0.06	0.39
2D ST-map	**CNN**	0.02	-0.01	0.05	0.12	0.01	0.00	0.02	0.08
	TF	0.00	0.00	-0.08	0.08	-0.02	0.03	0.05	0.05

Evaluation metrics: We evaluate the performance using the Mean Absolute Error and we use an independent equal-variance two-sample t-test to test for significant differences.

Results: From Table 5, we can see that the obtained MAE values do not differ much upon removing the band-pass filter. In all cases, we see that the unfiltered methods do not perform significantly better than their filtered variant, indicating that our band-pass filter does not remove any information that would be useful in prediction. For rise-time prediction we even see that the band-pass filter results in a small but significant drop in MAE.

Table 5. Mean absolute error for the different models and input representations on prediction of waveform features either with or without band-pass-filtering. Best performing method scores per filtered-unfiltered category are shown in bold ($* = p < 0.05$ significance) and error rates show the average over five iterations. The results show that using a band-pass filter gives at least as good performance as without a filter.

	HR	**RT**	**PWA**	**AUP**
1D CNN filtered	**9.21**	**43.59***	**0.12**	**0.34**
1D CNN unfiltered	9.46	44.10	**0.12**	0.35
1D TF filtered	11.37	**44.15**	**0.11**	**0.36**
1D TF unfiltered	**11.34**	44.34	**0.11**	**0.36**
CWT CNN filtered	**9.45**	**43.04***	**0.11**	**0.35**
CWT CNN unfiltered	9.62	43.42	**0.11**	**0.35**
CWT TF filtered	12.08	**43.85***	**0.11**	**0.39**
CWT TF unfiltered	**12.02**	43.89	**0.11**	0.40

5 Discussion and Limitations

We have studied the ability to predict PPG waveform features for multiple datasets, model-types and input representations, to thereby give a broad insight into the possibility of pulse feature measurement. The MAE, RMSE and Pearson correlation values that our methods obtain show the ability to predict heart rate and pulse area, while the rise time and pulse wave amplitude seem to be more challenging features, which our tested models were not able to predict.

As was expected compared to the toy problem, where we estimated waveform features directly from the PPG waveform, the additional noise in the rPPG signal makes the task of estimating waveform features a lot more difficult. In the toy example the features that were predicted with the highest accuracy were the heart rate and the area under the pulse, while the rise time and pulse wave amplitude could only be predicted relatively well on the VIPL-HR dataset (Figure 5). We show that while heart rate and area under the pulse still show good correlation when trained on pre-processed rPPG signals, the same does not hold true for the pulse wave amplitude and rise time prediction.

Moreover, our results demonstrate the influence of input representation on the prediction performance. In our method design, we have specifically not used any specialized model architectures to thereby give a general overview of the relation between input representations and waveform features. In general, the one-dimensional input representation in combination with a CNN is the most suitable based on our experimental set-up, although the models trained on the CWT representation achieve a comparable Pearson correlation for pulse area prediction.

One possible explanation of the bad performance of all tested pipelines on the rise time prediction is the low sampling frequency of the rPPG signal. The prediction of rise time and pulse wave amplitude depends on being able to detect the start and peak of the systolic wave. In the datasets we study, the average rise-time over 10-second time windows is approximately 205 ms in VIPL-HR, and 160 ms in VicarPPGBeyond (Figure 4). As both datasets consist of videos with a frame-rate of maximally 30 fps, this means that the rise-time often has a duration of only 6 frames. If an rPPG method is only one frame off in its rise time prediction, this thereby already gives it an error of 33 ms. If we want to improve over the rise time prediction error below 33 ms that we obtain with our methods, it is therefore important to collect datasets with a higher frame-rate as otherwise the time-resolution will play a big role in the obtained performance.

Another notable detail is that the performance of our methods varies greatly between folds, especially in the VicarPPGBeyond dataset. This makes cross-validation important, as otherwise the obtained error is largely influenced by the chosen data split. This can be seen in Figure 6, where the standard deviation of the Pearson correlation is often higher than the average. An example is the pulse wave amplitude prediction using the 1D-CNN, where the Pearson correlation varies between -0.01 and 0.82. This indicates that the relatively low amount of participants and data volume of VicarPPGBeyond makes it sensitive to the chosen train/validation/test split.

If we compare the performance of our methods on both datasets, we would expect to achieve better results on the VicarPPGBeyond dataset as it contains videos that are uncompressed and contain only small motions with respect to the VIPL-HR videos. However, the performance we achieve is not much better, except for heart rate prediction. We think that this might be due to the lower volume of the VicarPPGBeyond dataset. Our experiment where we train on the PPG signals instead of rPPG signals (Figure 5) also supports this hypothesis, as we see that even on the clean PPG signals, our methods struggle to predict the rise-time and pulse wave amplitude for VicarPPGBeyond. We would therefore suggest studying waveform features on large datasets, as the data volume might highly impact the performance.

Although we aim to give a broad insight into methods for PPG waveform feature estimation, there are still many possible methods that are out of the scope of the current study. Based on successful results of direct heart-rate estimation methods, we focus specifically on direct feature estimation methods in

this study. However, there exist many rPPG methods that predict heart-rate indirectly, by optimizing for predicting the PPG waveform and then extracting the heart-rate by post-processing. Investigating how well these methods are able to predict various waveform features might be interesting. Other possibilities include the use of purely signal-processing based methods, or the use of purely deep-learning based methods, for example by using a PPG waveform estimation model in combination with our CNN model trained on predicting waveform features from 1D-signals.

The results presented in this paper give first insights into the possibilities of PPG waveform feature prediction using remote PPG. Although there is still much progress to make, the insights we obtain can serve as indication of promising directions to develop more advanced waveform feature prediction models. Ultimately, this can thereby allow further steps towards home-care robotics.

References

1. Aarts, L.A., Jeanne, V., Cleary, J.P., Lieber, C., Nelson, J.S., Oetomo, S.B., Verkruysse, W.: Non-contact heart rate monitoring utilizing camera photoplethysmography in the neonatal intensive care unit—a pilot study. Early human development **89**(12), 943–948 (2013)
2. Alty, S.R., Angarita-Jaimes, N., Millasseau, S.C., Chowienczyk, P.J.: Predicting arterial stiffness from the digital volume pulse waveform. IEEE Transactions on Biomedical Engineering **54**(12), 2268–2275 (2007)
3. Awad, A.A., Ghobashy, M.A.M., Stout, R.G., Silverman, D.G., Shelley, K.H.: How does the plethysmogram derived from the pulse oximeter relate to arterial blood pressure in coronary artery bypass graft patients? Anesthesia & Analgesia **93**(6), 1466–1471 (2001)
4. Awad, A.A., Haddadin, A.S., Tantawy, H., Badr, T.M., Stout, R.G., Silverman, D.G., Shelley, K.H.: The relationship between the photoplethysmographic waveform and systemic vascular resistance. Journal of clinical monitoring and computing **21**(6), 365–372 (2007)
5. Balakrishnan, G., Durand, F., Guttag, J.: Detecting pulse from head motions in video. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 3430–3437 (2013)
6. Billauer, E.: peakdet: Peak detection using matlab. Detect Peaks in a Vector, Billauer, E., Haifa, Israel, accessed July **20**, 2012 (2012)
7. Bobbia, S., Macwan, R., Benezeth, Y., Nakamura, K., Gomez, R., Dubois, J.: Iterative boundaries implicit identification for superpixels segmentation: a real-time approach. IEEE Access **9**, 77250–77263 (2021)
8. Bousefsaf, F., Djeldjli, D., Ouzar, Y., Maaoui, C., Pruski, A.: ippg 2 cppg: Reconstructing contact from imaging photoplethysmographic signals using u-net architectures. Computers in Biology and Medicine **138**, 104860 (2021)
9. Brown, M.: Similarities and differences between augmentation index and pulse wave velocity in the assessment of arterial stiffness. Qjm **92**(10), 595–600 (1999)
10. Camm, A.J., Malik, M., Bigger, J.T., Breithardt, G., Cerutti, S., Cohen, R.J., Coumel, P., Fallen, E.L., Kennedy, H.L., Kleiger, R.E., et al.: Heart rate variability: standards of measurement, physiological interpretation and clinical use. task force of the european society of cardiology and the north american society of pacing and electrophysiology (1996)
11. Chen, W., McDuff, D.: Deepphys: Video-based physiological measurement using convolutional attention networks. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 349–365 (2018)
12. Chua, E.C.P., Redmond, S.J., McDarby, G., Heneghan, C.: Towards using photoplethysmogram amplitude to measure blood pressure during sleep. Annals of biomedical engineering **38**(3), 945–954 (2010)
13. Dillon, J.B., Hertzman, A.B.: The form of the volume pulse in the finger pad in health, arteriosclerosis, and hypertension. American Heart Journal **21**(2), 172–190 (1941)
14. Dorlas, J., Nijboer, J.: Photo-electric plethysmography as a monitoring device in anaesthesia: application and interpretation. British journal of anaesthesia **57**(5), 524–530 (1985)
15. Elgendi, M.: On the analysis of fingertip photoplethysmogram signals. Current cardiology reviews **8**(1), 14–25 (2012)

16. Gudi, A., Bittner, M., van Gemert, J.: Real-time webcam heart-rate and variability estimation with clean ground truth for evaluation. Applied Sciences **10**(23), 8630 (2020)
17. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)
18. Hong, S., Xu, Y., Khare, A., Priambada, S., Maher, K., Aljiffry, A., Sun, J., Tumanov, A.: Holmes: Health online model ensemble serving for deep learning models in intensive care units. In: Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. pp. 1614–1624 (2020)
19. Kuilenburg, H.v., Wiering, M., Uyl, M.d.: A model based method for automatic facial expression recognition. In: European conference on machine learning. pp. 194–205. Springer (2005)
20. Liu, X., Fromm, J., Patel, S., McDuff, D.: Multi-task temporal shift attention networks for on-device contactless vitals measurement. Advances in Neural Information Processing Systems **33**, 19400–19411 (2020)
21. Liu, X., Hill, B.L., Jiang, Z., Patel, S., McDuff, D.: Efficientphys: Enabling simple, fast and accurate camera-based vitals measurement. arXiv preprint arXiv:2110.04447 (2021)
22. Millasseau, S.C., Kelly, R., Ritter, J., Chowienczyk, P.: Determination of age-related increases in large artery stiffness by digital pulse contour analysis. Clinical science **103**(4), 371–377 (2002)
23. Murray, W.B., Foster, P.A.: The peripheral pulse wave: information overlooked. Journal of clinical monitoring **12**(5), 365–377 (1996)
24. Natarajan, A., Chang, Y., Mariani, S., Rahman, A., Boverman, G., Vij, S., Rubin, J.: A wide and deep transformer neural network for 12-lead ecg classification. In: 2020 Computing in Cardiology. pp. 1–4. IEEE (2020)
25. Niu, X., Han, H., Shan, S., Chen, X.: Vipl-hr: A multi-modal database for pulse estimation from less-constrained face video. In: Asian Conference on Computer Vision. pp. 562–576. Springer (2018)
26. Niu, X., Shan, S., Han, H., Chen, X.: Rhythmnet: End-to-end heart rate estimation from face via spatial-temporal representation. IEEE Transactions on Image Processing **29**, 2409–2423 (2019)
27. Nowara, E., McDuff, D., Veeraraghavan, A.: The benefit of distraction: Denoising remote vitals measurements using inverse attention. arXiv preprint arXiv:2010.07770 (2020)
28. Pereira, T., Tran, N., Gadhoumi, K., Pelter, M.M., Do, D.H., Lee, R.J., Colorado, R., Meisel, K., Hu, X.: Photoplethysmography based atrial fibrillation detection: a review. NPJ digital medicine **3**(1), 1–12 (2020)
29. Poh, M.Z., McDuff, D.J., Picard, R.W.: Advancements in noncontact, multiparameter physiological measurements using a webcam. IEEE transactions on biomedical engineering **58**(1), 7–11 (2010)
30. Rouast, P.V., Adam, M.T., Chiong, R., Cornforth, D., Lux, E.: Remote heart rate measurement using low-cost rgb face video: a technical literature review. Frontiers of Computer Science **12**(5), 858–872 (2018)
31. Seitsonen, E., Korhonen, I., Van Gils, M., Huiku, M., Lötjönen, J., Korttila, K., Yli-Hankala, A.: Eeg spectral entropy, heart rate, photoplethysmography and motor responses to skin incision during sevoflurane anaesthesia. Acta Anaesthesiologica Scandinavica **49**(3), 284–292 (2005)
32. Shafer, S.A.: Using color to separate reflection components. Color Research & Application **10**(4), 210–218 (1985)

33. Shin, H., Min, S.D.: Feasibility study for the non-invasive blood pressure estimation based on ppg morphology: Normotensive subject study. Biomedical engineering online **16**(1), 1–14 (2017)
34. Špetlík, R., Franc, V., Matas, J.: Visual heart rate estimation with convolutional neural network. In: Proceedings of the british machine vision conference, Newcastle, UK. pp. 3–6 (2018)
35. Swinehart, D.F.: The beer-lambert law. Journal of chemical education **39**(7), 333 (1962)
36. Takano, C., Ohta, Y.: Heart rate measurement based on a time-lapse image. Medical engineering & physics **29**(8), 853–857 (2007)
37. Teng, X., Zhang, Y.: Continuous and noninvasive estimation of arterial blood pressure using a photoplethysmographic approach. In: Proceedings of the 25th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (IEEE Cat. No. 03CH37439). vol. 4, pp. 3153–3156. IEEE (2003)
38. Touvron, H., Cord, M., Douze, M., Massa, F., Sablayrolles, A., Jégou, H.: Training data-efficient image transformers & distillation through attention. In: International Conference on Machine Learning. pp. 10347–10357. PMLR (2021)
39. Tsouri, G.R., Li, Z.: On the benefits of alternative color spaces for noncontact heart rate measurements using standard red-green-blue cameras. Journal of biomedical optics **20**(4), 048002 (2015)
40. Verkruysse, W., Svaasand, L.O., Nelson, J.S.: Remote plethysmographic imaging using ambient light. Optics express **16**(26), 21434–21445 (2008)
41. Viola, P., Jones, M.: Rapid object detection using a boosted cascade of simple features. In: Proceedings of the 2001 IEEE computer society conference on computer vision and pattern recognition. CVPR 2001. vol. 1, pp. I–I. Ieee (2001)
42. Wang, A., Yang, L., Wen, W., Zhang, S., Hao, D., Khalid, S.G., Zheng, D.: Quantification of radial arterial pulse characteristics change during exercise and recovery. The Journal of Physiological Sciences **68**(2), 113–120 (2018)
43. Wang, L., Pickwell-MacPherson, E., Liang, Y., Zhang, Y.T.: Noninvasive cardiac output estimation using a novel photoplethysmogram index. In: 2009 annual international conference of the IEEE engineering in medicine and biology society. pp. 1746–1749. IEEE (2009)
44. Wang, W., Den Brinker, A.C., Stuijk, S., De Haan, G.: Algorithmic principles of remote ppg. IEEE Transactions on Biomedical Engineering **64**(7), 1479–1491 (2016)
45. von Wowern, E., Östling, G., Nilsson, P.M., Olofsson, P.: Digital photoplethysmography for assessment of arterial stiffness: repeatability and comparison with applanation tonometry. PloS one **10**(8), e0135659 (2015)
46. Wu, H.T., Liu, C.C., Lin, P.H., Chung, H.M., Liu, M.C., Yip, H.K., Liu, A.B., Sun, C.K.: Novel application of parameters in waveform contour analysis for assessing arterial stiffness in aged and atherosclerotic subjects. Atherosclerosis **213**(1), 173–177 (2010)
47. Yu, Z., Shen, Y., Shi, J., Zhao, H., Torr, P., Zhao, G.: Physformer: Facial video-based physiological measurement with temporal difference transformer. arXiv preprint arXiv:2111.12082 (2021)

2

Introduction

Photoplethysmography (PPG) is a technique to measure the blood volume pulse of a patient using a contact sensor commonly placed around the finger or ear lobe [6]. Fluctuations in the hemoglobin concentration of the blood due to the blood pulse result in changes in light absorption by the skin tissue [17], which can be measured and used to derive the underlying pulse. Remote PPG (rPPG) attempts to do the same, but using a camera directed at the face instead of making contact. Although the light absorption fluctuations are small, they can be detected from several meters distance by a camera and by using specialized techniques, the underlying blood volume pulse can be derived.

Remote photoplethysmography can have many positive applications. First of all, it makes it possible to measure the blood volume pulse in patients that are harmed by the skin-contact of traditional PPG devices, such as infants [3] and people with a sensitive skin [22]. Moreover, it allows for remote health diagnosis, as rPPG can be done using a webcam or phone camera instead of specialized healthcare devices, thereby largely decreasing the threshold for health checkups, especially in rural areas.

Currently, remote PPG methods have been developed for estimating the heart rate of a patient with great success [15]. However, many other features can be derived from the blood volume pulse to describe its morphology, for example the pulse wave amplitude and the area under the pulse [9]. These features have correlations with many underlying physiological conditions of the patient, e.g. its blood pressure or the presence of cardiovascular diseases. Being able to measure these features using remote PPG would therefore allow for a more detailed health screening of a patient.

In this work, we will investigate the possibility of measuring various blood volume pulse features using remote PPG. As we are one of the first works to do this, we will attempt to do this using various input pre-processing pipelines, which we call input representations, in combination with current state-of-art deep learning architectures to predict pulse wave features from a facecam video. We thereby aim to give insight into the possibilities of pulse feature measurement using remote PPG and the influence of input representations on this prediction. We present our results in the form of a paper (Chapter 1), background information (Chapter 3), additional experiments (Chapter 4) and conclude with an ethical discussion and future recommendations (Chapter 5).

3

Background

In this section, I will give background information of the Deep Learning and signal processing techniques that I have used in my thesis. This chapter expects that the reader is familiar with the basics of neural networks, such as activation functions, fully connected networks, convolutions and backpropagation.

3.1. Model architectures

3.1.1. Convolutional neural networks

Convolutional neural networks (CNNs) have been one of the most used types of deep learning models in the last decade. First introduced in AlexNet [14], these models consist of several convolutional layers to learn to extract features from an input. Although they were introduced for 2D data, specifically images, they have also achieved success on 1D time series data [28], 3D data such as videos [12], and could theoretically also be used on higher dimensionality input, although they exponentially increase in size with increasing dimensions. These models have shown state-of-the-art performance in various computer vision problems [16][29][21], and their application also extends to other domains such as healthcare [18]. Although the CNN is in theory able to learn any complex non-linear relation, there have been studies that suggest it generally relies on low-level features [4] such as textures [10] in its decision making.

3.1.2. Residual connections

As deep learning architectures become deeper, they generally become harder to converge because of exploding or vanishing gradients. Because of this issue, He et al. [11] introduced the residual- or skip connection, most commonly known from their widely used Resnet architecture. Traditionally in neural networks, data flows through each layer sequentially, where the output of one layer becomes the input of the layer that follows. Residual connections, however, skip the subsequent layer, and are multiplied instead with the identity matrix to keep the original information intact (Figure 3.1). There are various theories why this is a good idea, one of which is that these residual connections make the network behave as an ensemble of methods, thereby avoiding gradient problems by having shallower paths in these ensembles [26].

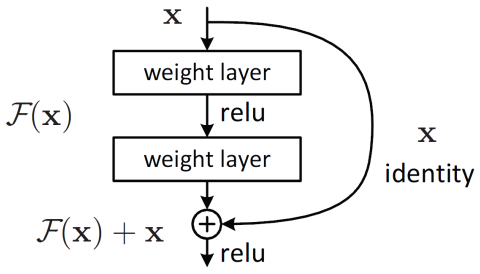

Figure 3.1: Schematic view of a residual connection [11]

3.1.3. The Transformer

More recently, the Transformer architecture has been introduced [25]. It has originally been designed for the natural language processing field, especially for text translation. Traditionally, for these problems, recurrent neural network models have been the most appropriate model, as their ability to learn temporal-dependencies from data allows them to learn relations between subsequent words in a sentence. These models however where in practise not able to learn long-distance dependencies spanning more than several sentences, and this is the main issue that the Transformer model improves on. The Transformer relies on self-attention to learn relationships between elements in a sequence. It does this using query, key and value matrices based on the following equation: $Attention(Q,K,V) = softmax(\frac{QK^T}{\sqrt{d_k}})V$.

Without going much into detail, the query-, key- and value-matrices (Q, K and V respectively) are obtained using matrix-vector multiplication between a word embedding x and a learned weight matrix W_q, W_k or W_v. By multiplying the key- and query matrices as QK^T, it returns the similarity of each word with each of the other words in the sequence. After scaling this with the word embedding dimensionality d_k, multiplication with value matrix V then results in an output that represents a weighted retrieval of the value of the words in the sequence. Often, multiple 'attention-heads' are used to be able to capture multiple relationships between words, where each head has different W_q, W_k and W_v matrices.

In the Transformer model, the self-attention operation described above is combined with residual connections and fully connected layers into an encoder layer (Figure 3.2). Multiple encoder layers can be stacked, where the output of each layer will be the input of the following layer. The output of the encoder can then be used in combination with a decoder architecture to then output a sequence (as in the original Transformer implementation) or a scalar value in our case.

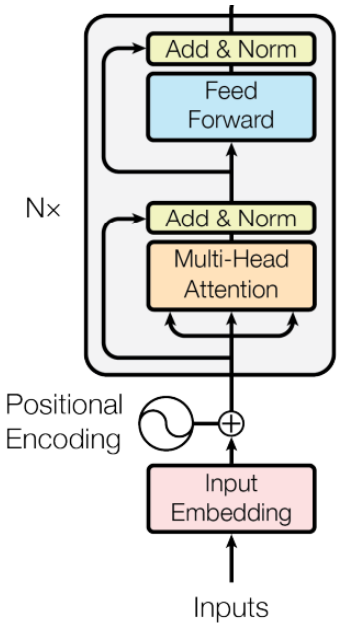

Figure 3.2: Schematic overview of the encoder architecture of the Transformer by [25].

3.1.4. Vision Transformers

In theory, the same recipe for the Transformer could also be applied on 2D data such as images: give every pixel a positional encoding and treat the $n \times n$ image as a sequence with length n^2. In reality however, this is infeasible as the self-attention operations have $\mathcal{O}(N^2 d)$ complexity, with N the sequence length. The solution to this which Dosovitskiy et al. [8] propose in their Vision Transformer (ViT) is to compress 16×16 pixel patches into one scalar, thereby greatly decreasing the computational complexity. Despite this compression, the ViT performs really well in practise and has proven to outperform the CNN in many computer vision tasks [24][8].

3.2. Model training

3.2.1. Adam optimizer

Supervised learning of a neural network in general involves three steps: present a sample from the training set, calculate the loss based on output and target values, and back-propagate the loss to update the network weights. Using stochastic gradient descent, weights θ are updated based on the learning rate ϵ and a gradient approximated by mini-batches m':

$$\theta' = \theta - \epsilon \frac{1}{m'} \sum_{i=1}^{m'} \nabla_\theta L(y_i, f(x_i), \theta), \tag{3.1}$$

with $f(x_i)$ the predicted value based on data i in the mini-batch and y_i its target value.

However, because stochastic gradient descent uses an approximated gradient based on mini-batches and only uses current estimated gradients to update weights, it can have problems with convergence. In Figure 3.3, an example is shown of a possible weight update trajectory using stochastic gradient descent on two weight parameters θ_1 and θ_2. As can be seen, this trajectory is far from optimal, as instead of going in a straight direction towards the optimum, it often deviates from this direction due to the stochastic nature of the mini-batches. Optimizers have therefore been designed that use a more complex update step to overcome this issue. In my thesis, I have used the widely adopted Adam optimizer [13], which combines two different methods that both rely on the exponential weighted moving average (EWMA) to use previous update steps in its current update: Momentum and RMSprop.

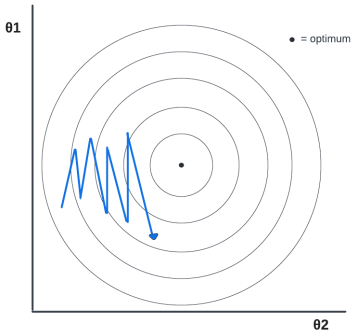

Figure 3.3: Example of a stochastic gradient descent weight update trajectory in a two-dimensional weight landscape.

Stochastic gradient descent with momentum uses the EWMA of previous gradients combined with the current gradient to update the weights: $\theta' = \theta - \epsilon v_i$, with $v_i = \rho v_{i-1} + (1-\rho)\nabla_\theta$, where ρ is a tuneable hyperparameter and ∇_θ is the estimated gradient for the current weights. This thereby smooths the average over noisy gradient approximations. RMSProp uses the EWMA to estimate the squared gradient: $r_i = \rho r_{i-1} + (1-\rho)\nabla_\theta^2$. It then uses this to divide the current mini-batch gradient by, to thereby smooth the zero-centered variance of noisy update steps: $\theta' = \theta - \epsilon \frac{\nabla_\theta}{\sqrt{r_i}}$. By combining momentum and RMSprop, Adam is thereby able to accelerate model convergence and limit the stochastics caused by mini-batches.

3.2.2. K-fold cross-validation

In a basic model training procedure, the dataset is split into three sets: a training-set, validation-set and a test-set. The training-set is used, as the name says, when training the model, followed by the validation phase were each trained model is tested on a set of data it has not previously seen to tune hyper-parameters and chose the model that generalizes best. After this, the best performing model is chosen, which is then finally evaluated on the data in the test-set.

If the dataset is low in volume however, it is often better to use a more rigorous training procedure called cross-validation because of two reasons: to increase the data that can be used for training, and to decrease the dependency on the chosen test- and validation-set. The procedure goes as follows: 1) shuffle the data, 2) divide the data into k parts, also called folds, 3) train k different models, were fold k will be divided into a test- and validation-set and all other folds will be the training data for that model, 4) take the average of the performance of all k models as the final performance (Figure 3.4).

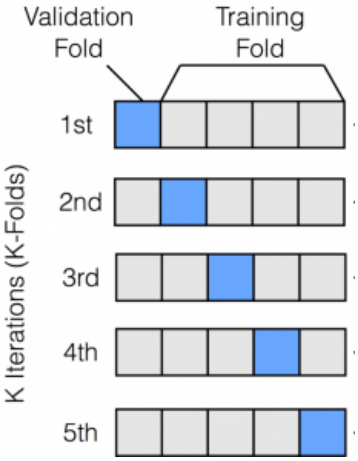

Figure 3.4: K-fold cross-validation visualized [1].

3.3. Remote PPG and signal processing

3.3.1. Skin reflection model

To get a good understanding of the remote photoplethysmography problem, it is important to understand the basic principles of light reflection by the skin. Mathematically, we can define the reflection of a skin pixel as an RGB-signal over time [27] using Shafer's dichromatic reflection model [23]:

$$C_k(t) = I(t) \cdot (v_s(t) + v_d(t)) + v_n(t), \tag{3.2}$$

where $C_k(t)$ denotes the RGB values of pixel k at timepoint t. This value is dependent on $I(t)$, which is the illumination level affected by variations in the light intensity of the source as well as by changes in the distance between skin, light-source and sensor. $I(t)$ is modulated by the specular reflection $v_s(t)$ and the diffuse reflection $v_d(t)$. Lastly, $v_n(t)$ denotes the quantisation noise by the camera.

The specular reflection is light reflection by the skin comparable to a mirror (Figure 3.5): it is affected by the geometric structure of the skin and does not contain any pulsatile information. Its value over

time can be denoted as

$$v_s(t) = u_s \cdot (s_0 + s(t)), \tag{3.3}$$

where u_s is the unit color vector of the light spectrum from the source, and s_0 and $s(t)$ respectively reflect the stationary and time dependent parts of specular light reflection.

The diffuse reflection contains the relevant signal that we aim to capture using remote photoplethysmography: it reflects the light absorption by the skin tissue and is affected by the blood volume pulse. The diffuse reflection $v_d(t)$ can be denoted as:

$$v_d(t) = u_d \cdot d_0 + u_p \cdot p(t), \tag{3.4}$$

with u_d and d_0 the color vector of the skin tissue and stationary strength of reflection respectively, u_p the relative strength in pulse of the RGB-channels and $p(t)$ the blood pulse.

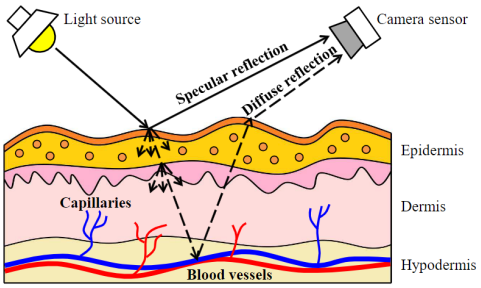

Figure 3.5: Visualisation of the specular and diffuse light reflection of the skin [27].

Substituting the above equations into the general formula and some rewriting gives:

$$C_k(t) = I_0 \cdot (1 + i(t)) \cdot (u_s \cdot (s_0 + s(t)) + u_d \cdot d_0 + u_p \cdot p(t)) + v_n(t). \tag{3.5}$$

Thus, the color value of pixels over time can be expressed as the combination of a stationary part I_0 and a time-dependent part $I_0 \cdot i(t)$. Our goal is the measure the pulsatile signal $p(t)$, but because the specular reflection in general is a signal that is much higher in amplitude than the diffuse reflection, this is oftentimes a challenging problem.

3.3.2. POS method

The Plane-Orthogonal-to-Skin (POS) method by Wang et al. [27] builds onto the skin reflection model defined in the previous section. As we concluded there, the pixel color at some timepoint depends on stationary components as well as time-dependent components of the specular and diffusion reflection. As we are only interested in measuring the diffusion reflection which is affected by the blood volume pulse, an optimal method would allow us to negate all the other effects. Wang et al. aim to do this by estimating a plane orthogonal to the skin and projecting the pixel color signal onto this plane. As the diffusion reflection is largely independent of the skin tone [7], projecting the signal onto this plane allows the removal of a large part of the intensity variations and specular reflection while keeping the signal of interest intact.

3.3.3. Butterworth filter

Filtering is an operation that has often been used for remote photoplethysmography, especially in combination with other signal processing techniques. The reason behind this is that there is only a range of desirable frequencies that we want to include in photoplethysmography measurement: the blood volume pulse of somebody at rest naturally has a frequency between 60 and 100 bpm. To account for extreme cases, e.g. heavy exercise and hearth disorders, in general a broader passband is used such that no useful information is filtered out, for example between 25 and 250 bpm.

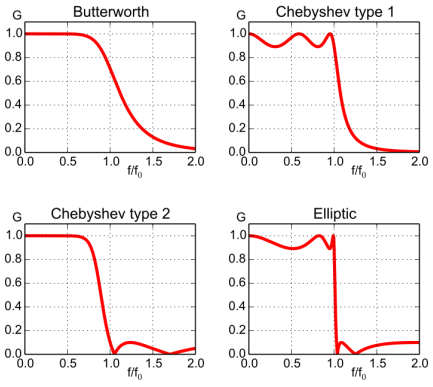

Figure 3.6: Comparison of the gain around the cut-off frequency for different filter types[20].

The Butterworth-filter is one of the most popular methods to perform band-pass filtering. It is especially designed to have a frequency response that is as flat as possible in the passband, meaning that it has a maximally uniform sensitivity to all frequencies within the passband [5]. The disadvantage of this filter is that it has a relatively broad roll-off around the cut-off frequencies, which results in some frequencies out of the passband being not completely filtered out (Figure 3.6). As long as we chose a broad range of frequencies for the passband, the flat response of the Butterworth filter makes sure that we do not filter out any

useful signals, and our neural networks can learn to negate the effects of the frequencies that are not filtered out due to the broad roll-off of the filter. This makes the Butterworth filter a suitable choice for our application.

3.3.4. Continuous wavelet transform

The continuous wavelet transform (CWT) is by definition the convolution of a one-dimensional input signal with a set of functions derived from a chosen 'mother wavelet'. For this mother wavelet various functions can be used, one of which is the Morlet wavelet 3.7. This mother wavelet is used as a source function to derive a set of daughter wavelets by translation and scaling. The mathematical definition for the continuous wavelet transform is thereby as follows:

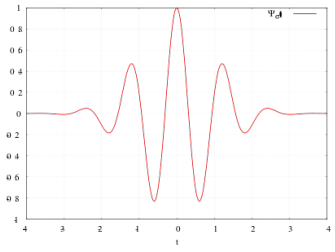

Figure 3.7: The Morlet wavelet [2].

$$X_w(a,b) = \frac{1}{|a|^{1/2}} \int_{-\infty}^{\infty} x(t)\bar{\psi}\frac{t-b}{a}dt, \tag{3.6}$$

with ψ the mother wavelet, which is scaled by a and translated by b, $x(t)$ is the original signal and X_w its continuous wavelet transform. Using a broad range of daughter wavelets, a representation of the frequencies present in the input signal can thereby be generated consisting of a real and imaginary part as shown in Figure 3.8. The original signal can also be derived from the CWT representation using the inverse continuous wavelet transform.

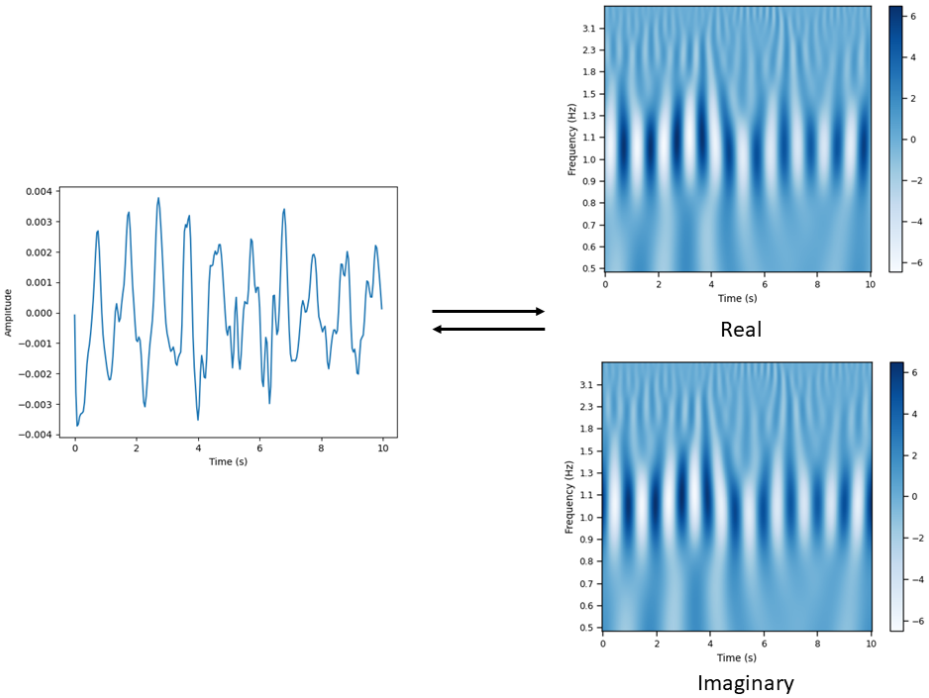

Figure 3.8: An example of a one-dimensional signal (left) and its continuous wavelet transform representation (right).

4

Additional experiments

In this chapter, we describe some of the additional experiments that did not make it into the paper to give more insight in the underlying methods and possible alternatives.

4.1. Ground-truth evaluation

In my thesis, I aim to predict four different waveform features: the heart rate, rise time, pulse wave amplitude and the pulse area. To be able to train a neural network on this task, we need ground-truth values for these features. Fortunately, all four of these signals can be derived using annotations of the start and the peak of the systolic phase as follows (Figure 4.1):

- Heart rate: time between consecutive systolic peaks, converted to the amount of peaks per minute.
- Rise time: time between the start of the systolic phase and the systolic peak.
- Pulse wave amplitude: amplitude difference between the start of the systolic phase and the systolic peak.
- Area under pulse: area under the signal between two consecutive starts of the systolic phase.

Unfortunately, the start of the systolic phase has not been annotated for both datasets, and the peak-annotations are only available for VIPL-HR. Therefore, a peak- and valley-detection algorithm has been used to detect the peaks and valleys in the ground truth signal. This algorithm is based on the first-derivative of the signal, and uses a threshold to account for noise and smaller peaks and valleys in the signal. The pseudo-code of this algorithm can be found in Algorithm 1.

To evaluate the performance of this algorithm, I have annotated peaks and valleys for a part of the ground-truth signals for the VIPL-HR and VicarPPGBeyond dataset (Figure 4.2). To be precise, the following procedure was used:

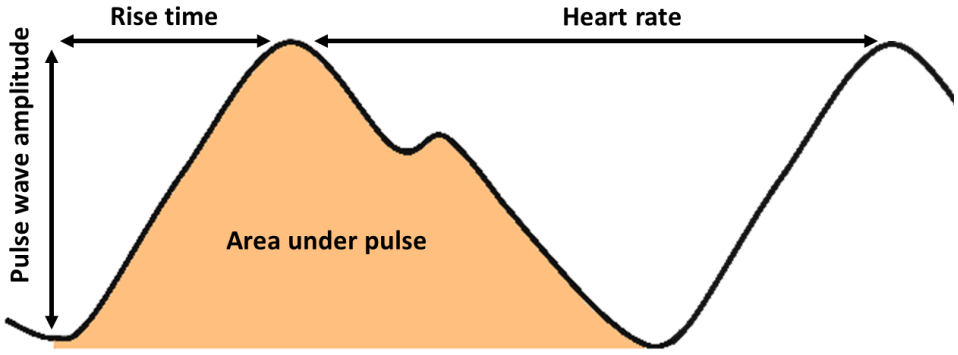

Figure 4.1: Overview of the four pulse waveform features we study.

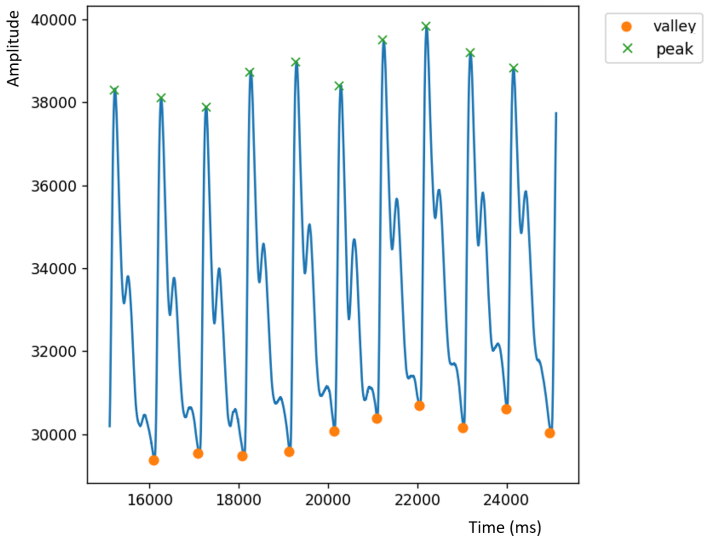

Figure 4.2: Example annotation of peaks and valleys in a randomly chosen 10-second window of a Vicar ground-truth PPG signal.

Algorithm 1 Pseudocode of peak- and valley-detection

```
procedure peaksAndValleys(signal, peak_delta)
    norm_sig ← signal - Mean(signal)          Normalize signal and peak delta
    delta ← peak_delta · Max(norm_sig)

    peaks ← {}, valleys ← {}
    mxpos ← 0, mnpos ← 0
    look_for_max ← True
    mx ← −∞, mn ← ∞

    for idx, x ∈ Enumerate(norm_sig) do    Go over consecutive datapoints
        if x > mx then                      If a new max is found, save it
            mx ← x
            mxpos ← idx
        if x < mn then                      If a new min is found, save it
            mn ← x
            mnpos ← idx
        if look_for_max then
            if x < mx - delta then          If the current datapoint is lower than the threshold, save the peak
                peaks ← peaks ∪ x
                mn ← x, mnpos ← idx
                look_for_max ← False
        else
            if x > mn + delta then          If the current datapoint is higher than the threshold, save the valley
                valleys ← valleys ∪ x
                mx ← x, mxpos ← idx
                look_for_max ← True
    return peaks, valleys
```

- For VIPL-HR, only the valleys were not yet annotated. Therefore, for every participant (107 in total), randomly one video out of the nine scenarios was chosen. From this video, we chose a random 10-second window to be annotated. As the recordings are on average 30 seconds in length, this accounts for approximately 1/27th of the total dataset randomly sampled.

	VicarPPGBeyond		VIPL-HR	
	Peaks	**Valleys**	**Peaks**	**Valleys**
Precision	0.960	0.992	0.975	0.918
Recall	0.989	0.957	0.988	0.891

Table 4.1: Precision and recall values for evaluation of the ground-truth annotation algorithm.

- For VicarPPGBeyond, we annotate both peaks and valleys using the following procedure: for each of the 105 recordings (7 scenarios for all 15 participants), a random 10-second window is chosen to be annotated. The VicarPPGBeyond recordings are on average approximate 130 seconds in length, making the annotation 1/13th of the complete dataset.

The peaks and valleys that were found using the peak- and valley-detection algorithm were then compared with the ground truth values. For this, we use the following criterium: if the peak or valley found by the algorithm is within 100 ms of the ground-truth peak/valley, it is correct, otherwise it is considered incorrect. This 100 ms tolerance was chosen as within this range, it would not significantly affect the calculation of the feature values for a 10-second time-window and to account for small errors in the annotation.

The precision and recall were then used to evaluate the performance of the algorithm as follows:

$$Precision = \frac{n_{correct}}{n_{detected}}, \tag{4.1}$$

$$Recall = \frac{n_{correct}}{n_{true}}, \tag{4.2}$$

with $n_{correct}$ the amount of correctly detected peaks/valleys, $n_{detected}$ the total amount of peaks/valleys detected by the algorithm and n_{true} the true total amount of peaks/valleys. The values that we obtain are summarized in Table 4.1. We consider these values good enough for our application, and therefore we have chosen to continue with using this algorithm.

4.2. Hyperparameter tuning

In this study, we use two model architecture types: the convolutional neural network (CNN) and the Transformer. For the 2D input representations, we use a Resnet18 [26] and DeIT-base model [24] (similar to the Vision Transformer described in Chapter 3.1.4) respectively, which have a predefined size and amount of layers shown to perform well on a wide range of tasks. For the 1D input representation, we use similar model architectures but their optimal size and number of layers are more variable and are a tuneable hyperparameter. We therefore perform a minor grid-search to tune their optimal hyperparameters for our PPG waveform estimation task.

For the one-dimensional Resnet, there is one hyperparameter that we tune: the number of blocks $nBlocks$. The Resnet architecture is comprised of the following layers (disregarding any possible activation functions or normalization layers): one convolution layer, followed by n residual blocks, and ending with a final fully-connected layer. Each residual block consists of a residual connection and two convolution layers, which have a number of filters that doubles every two blocks. By changing the number of blocks, we can thus increase the depth of the model and simultaneously increase the number of filters in the final convolution layers.

We tune three hyperparameters for the one-dimensional Transformer, which are the number of heads $nHeads$, the number of self-attention blocks $nBlocks$ and the dimension of the feedforward layers in the self-attention blocks $dHid$ (see Figure 3.2 for the Transformer building blocks). Increasing the value of these hyperparameters will increase the complexity of the model.

We perform a small grid-search over hyperparameter values for both architectures by evaluating the models on the MAE they obtain. We do this using a fixed subject-exclusive 70/15/15 train/validate/test split on VIPL-HR. The results are shown in Table 4.2 and 4.3.

4.3. Training curves

Training curves are a well-known tool to keep track of model training and to check when your model starts to overfit. A common train- and test-curve of the model training on the waveform feature prediction

nBlocks	2	4	6	8	10	12
MAE	9.79	16.89	9.18	7.49	**7.01**	8.80

Table 4.2: Hyperparameter tuning of the number of residual blocks in the 1D Resnet architecture based on the MAE on VIPL-HR. Lowest MAE value is shown in bold.

nHeads		2			4			8		
nBlocks		2	4	8	2	4	8	2	4	8
dHid	512	20.05	18.49	18.44	19.20	19.04	16.36	19.76	18.10	18.28
	1024	21.00	16.58	18.50	19.67	19.10	18.67	19.35	19.71	17.80
	2048	19.59	19.10	18.10	18.50	18.91	11.10	17.84	17.85	17.64

Table 4.3: Hyperparameter tuning of the number of self-attention blocks, the number of attention heads and the hidden dimension of the feedforward layers in the 1D Transformer architecture based on the MAE on VIPL-HR. Lowest MAE value is shown in bold.

task is shown in Figure 4.3. What can be seen from this, is that the performance on the training set is relatively smooth: after the first few epochs, the difference in train-loss between consecutive epochs is only marginal and the loss steadily decreases over time. The test-set performance is far from smooth however, as we see large upward and downward spikes in test-loss between consecutive epochs, often showing a more than 10% increase or decrease in performance. Moreover, the test-loss does not in general decrease over time: in the first half, we can see that the loss on the test-set decreases but hereafter, this is not the case anymore. We therefore can not simply use the training-set performance as an indicator of the best model to use, as this often does not generalize. Therefore, an additional validation-set has been used so that the validation-set performance can be utilized to chose the best model, which we then evaluate on the test-set.

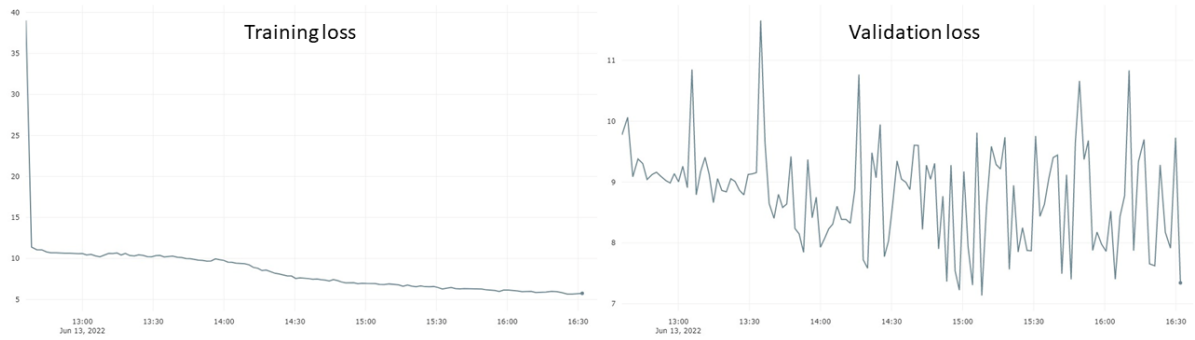

Figure 4.3: Example of a train- and test-curve for HR prediction on VIPL. As can be seen, a small training loss does not necessarily correspond to a small test-loss.

4.4. Spatial-temporal map alternative

The spatial-temporal map representation that we use in this study has been inspired by the RhythmNet model [19]. In their study, they use a 5x5 grid of square regions as ROIs, from which they extract the average RGB value over time (comparable to our approach, but using different ROIs). They then project the obtained RGB values to the YUV-space, which divides the signal into a luminance component (Y), and two chrominance components (U, V).

In my thesis, I use a projection to a plane orthogonal to the skin in combination with filtering instead of YUV-space projection. This method has shown to be successful for estimating heart rate using 1D rPPG-signals [27] and by keeping all pre-processing steps consistent for all three input representations, we can compare them without having to account for the effects of different pre-processing.

To test if this affects the performance of the models trained on the spatial-temporal map representations, I have tested both pre-processing methods (YUV-space projection or POS-projection + filtering) on the VicarPPGBeyond dataset. For this, I have used a subject-exclusive 70/15/15 train/validate/test split and trained both the CNN and the Transformer (TF) on predicting the four different waveform fea-

tures from the spatial-temporal maps. We take the average over 5 different iterations to account for randomness in training and use an independent equal-variance two-sample t-test to test for significant differences.

Our results are summarized in Table 4.4. As can be seen, we do see a difference in performance per method when they use the YUV-space instead of the POS-method. For four out of eight pairwise-comparisons, these differences are significant as well. This means that the POS-projection method outperforms the YUV-space for PWA and AUP prediction using a Transformer, while it is outperformed for HR prediction in combination with a Transformer, and PWA prediction using a CNN. There is thus no single-best choice that is best in all cases, but the results indicate that YUV-space projection might be an interesting alternative option.

		HR	RT	PWA	AUP
CNN	**POS**	**6.39**	25.8	0.237	**0.119**
	YUV	8.24	**19.67**	**0.144***	0.124
TF	**POS**	5.88	20.9	**0.120***	**0.111***
	YUV	**5.48***	**19.3**	0.324	0.132

Table 4.4: Mean absolute error for the different models on prediction of waveform features either using POS and filtering or using YUV-colorspace projection of the spatial-temporal map representations for VicarPPGBeyond. Best performing method scores per POS-YUV pairwise-comparison are shown in bold ($* = p < 0.05$ significance). Error rates show the average over five iterations.

5 Discussion

5.1. Ethical considerations

Aside from the technical possibilities of remote PPG, it is important to consider the ethical implications the development of this technology entails. First of all, there are many positive effects that this technology can bring. The possibility of remote measurement of the blood volume pulse can highly decrease the threshold for checking the physiological health of someone, not only by making this measurement possible without the need of specialized contact sensors, but also by making online diagnosis possible. Especially in more rural areas, this can greatly decrease the effort of health diagnosis. The development of remote photoplethysmography thereby also has the effect of allowing everyone, as long as they have a camera, to daily check-up on their health, which will in particular benefit poorer people for which this would normally be too costly.

However, there are other effects that remote PPG can have which can be detrimental if they are disregarded. First of all, it is possible that this technology will be maliciously used. The medical data that remote PPG can collect from someone should always be collected with consent of this individual and should be handled properly. Companies or other authorities might have motives to collect your medical data, for example to measure your emotions, to test if you speak the truth and to check if you are in good health. Further development of remote PPG might make it possible to reveal more of your medical health from a simple facecam video, especially if rPPG measurement will encompass more than only your heart rate. Suitable measures should therefore be taken to make sure that this will not be possible, although there is not much active research yet into disabling remote PPG measurement. Technical solutions could for example filter out the blood volume pulse signal, as the pulse signal is only minor and removing it would therefore not significantly affect the quality of the resulting video. It is important to start research into this subject, as it might only be a matter of time until remote PPG will be maliciously used.

Another, less obvious, adverse effect of remote PPG technology are the issues that the adoption of this technology in healthcare can have. First, an online diagnosis cannot entirely replace physical appointments with a general practitioner. In person, the doctor can often use subtle clues about the behaviour of a patient to determine their diagnosis and in healthcare, people often also have the desire to have face-contact, especially for people who are less familiar with technology. Relying on remote PPG also has the danger of overestimating the qualities of such a system. This will especially be the case if the user of the rPPG method, e.g. a doctor, is not familiar with the technology itself and the situations in which it might fail. Moreover, there is the danger of health anxiety for the patient undergoing a remote PPG check. A detailed remote PPG screening might for example indicate that there is a tiny chance of a cardiovascular disease because one's blood values might have marginally changed over the past few weeks. Although the chance of a disease is very low, users will constantly be confronted with their health information, causing distress.

It is important that not only the advantages but also the negative effects of remote PPG technology

are carefully considered before this technique will be used in practise. If used correctly, remote PPG can provide a low-cost and easy method to give everyone the ability to have their health screened. It is our shared responsibility as researchers on this topic that this technology will be used as intended.

5.2. Challenges and future work

In this work, we explore the possibilities of estimating PPG pulse waveform features using remote PPG. In the experiment where we train on PPG signals we show that, with a clean enough signal, we are able to estimate pulse waveform features with a relatively high correlation, while training on the rPPG signals shows that some features are harder to predict than others in practise.

There are still many challenges before this technology can be used in practise. For example, the high standard deviations between different folds when training on the VicarPPGBeyond dataset shows the importance of a widely distributed training dataset, as the results otherwise do not generalize to other persons. In our paper, we also address the importance of datasets with a high frame-rate, especially for measuring pulse waveform features, as the temporal resolution can play a big role for features with a low time duration. Lastly, it is important to look into different method types to see how they compare with the results we obtain and investigate their suitability for measuring waveform features. Methods trained on PPG waveform prediction, e.g. MTTS-CAN [15], have shown to be very successful for estimating heart rate. Although they do not directly optimize for predicting waveform features, it can be interesting how well they are able to estimate those in combination with post-processing methods to extract the feature values from the predicted wave.

The range of different possibilities to estimate PPG waveform features using remote PPG is endless, and there is no doubt that there will be methods that improve over the results we obtain in this study. However, with the work we present in this paper, we hope to give insights into the possibilities of PPG feature prediction and suitable methods to do so.

Bibliography

[1] AndroidKT pytorch k-fold cross-validation using dataloader and sklearn. https://androidkt.com/pytorch-k-fold-cross-validation-using-dataloader-and-sklearn/, accessed: 2022-06-10

[2] Wavelet. https://www.wikiwand.com/simple/Wavelet, accessed: 2022-06-13

[3] Aarts, L.A., Jeanne, V., Cleary, J.P., Lieber, C., Nelson, J.S., Oetomo, S.B., Verkruysse, W.: Non-contact heart rate monitoring utilizing camera photoplethysmography in the neonatal intensive care unit—a pilot study. Early human development **89**(12), 943–948 (2013)

[4] Brendel, W., Bethge, M.: Approximating cnns with bag-of-local-features models works surprisingly well on imagenet. arXiv preprint arXiv:1904.00760 (2019)

[5] Butterworth, S., et al.: On the theory of filter amplifiers. Wireless Engineer **7**(6), 536–541 (1930)

[6] Castaneda, D., Esparza, A., Ghamari, M., Soltanpur, C., Nazeran, H.: A review on wearable photoplethysmography sensors and their potential future applications in health care. International journal of biosensors & bioelectronics **4**(4), 195 (2018)

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

[8] Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929 (2020)

[9] Elgendi, M.: On the analysis of fingertip photoplethysmogram signals. Current cardiology reviews **8**(1), 14–25 (2012)

[10] Geirhos, R., Rubisch, P., Michaelis, C., Bethge, M., Wichmann, F.A., Brendel, W.: Imagenet-trained cnns are biased towards texture; increasing shape bias improves accuracy and robustness. arXiv preprint arXiv:1811.12231 (2018)

[11] He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)

[12] Kamnitsas, K., Ledig, C., Newcombe, V.F., Simpson, J.P., Kane, A.D., Menon, D.K., Rueckert, D., Glocker, B.: Efficient multi-scale 3d cnn with fully connected crf for accurate brain lesion segmentation. Medical image analysis **36**, 61–78 (2017)

[13] Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)

[14] Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. Advances in neural information processing systems **25** (2012)

[15] Liu, X., Fromm, J., Patel, S., McDuff, D.: Multi-task temporal shift attention networks for on-device contactless vitals measurement. Advances in Neural Information Processing Systems **33**, 19400–19411 (2020)

[16] Luvizon, D.C., Picard, D., Tabia, H.: 2d/3d pose estimation and action recognition using multitask deep learning. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 5137–5146 (2018)

[17] McDuff, D.: Camera measurement of physiological vital signs. arXiv preprint arXiv:2111.11547 (2021)

[18] Miotto, R., Wang, F., Wang, S., Jiang, X., Dudley, J.T.: Deep learning for healthcare: review, opportunities and challenges. Briefings in bioinformatics **19**(6), 1236–1246 (2018)

[19] Niu, X., Shan, S., Han, H., Chen, X.: Rhythmnet: End-to-end heart rate estimation from face via spatial-temporal representation. IEEE Transactions on Image Processing **29**, 2409–2423 (2019)

[20] Podder, P., Hasan, M., Islam, M., Sayeed, M., et al.: Design and implementation of butterworth, chebyshev-i and elliptic filter for speech signal analysis. arXiv preprint arXiv:2002.03130 (2020)

[21] Reed, S., Akata, Z., Yan, X., Logeswaran, L., Schiele, B., Lee, H.: Generative adversarial text to image synthesis. In: International conference on machine learning. pp. 1060–1069. PMLR (2016)

[22] Rouast, P.V., Adam, M.T., Chiong, R., Cornforth, D., Lux, E.: Remote heart rate measurement using low-cost rgb face video: a technical literature review. Frontiers of Computer Science **12**(5), 858–872 (2018)

[23] Shafer, S.A.: Using color to separate reflection components. Color Research & Application **10**(4), 210–218 (1985)

[24] Touvron, H., Cord, M., Douze, M., Massa, F., Sablayrolles, A., Jégou, H.: Training data-efficient image transformers & distillation through attention. In: International Conference on Machine Learning. pp. 10347–10357. PMLR (2021)

[25] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. Advances in neural information processing systems **30** (2017)

[26] Veit, A., Wilber, M.J., Belongie, S.: Residual networks behave like ensembles of relatively shallow networks. Advances in neural information processing systems **29** (2016)

[27] Wang, W., Den Brinker, A.C., Stuijk, S., De Haan, G.: Algorithmic principles of remote ppg. IEEE Transactions on Biomedical Engineering **64**(7), 1479–1491 (2016)

[28] Zhao, B., Lu, H., Chen, S., Liu, J., Wu, D.: Convolutional neural networks for time series classification. Journal of Systems Engineering and Electronics **28**(1), 162–169 (2017)

[29] Zhao, Z.Q., Zheng, P., Xu, S.t., Wu, X.: Object detection with deep learning: A review. IEEE transactions on neural networks and learning systems **30**(11), 3212–3232 (2019)