



Facial camera-based heart rate estimation using r-PPG convolutional neural networks

Title	Facial camera-based heart rate estimation using r-PPG convolutional neural networks
Author(s)	Moustafa, Mohamed; Lemley, Joseph; Corcoran, Peter
Publication Date	2023-08-26
Publisher	Irish Machine Vision and Image Processing Conference
Repository DOI	https://doi.org/10.5281/zenodo.8309793

Facial Camera-Based Heart Rate Estimation Using r-PPG Convolutional Neural Networks

Mohamed Moustafa^{1,2}, Joseph Lemley², and Peter Corcoran^{1,2}

¹*School of Engineering, University of Galway, Galway, Ireland*

²*Xperi Corporation, Galway, Ireland*

Abstract

The human heart plays an essential role in maintaining an individual's well-being. Therefore, monitoring heart behaviour and condition is important as it provides insights into various physical and psychological conditions. As it is not always convenient to attach sensors to an individual, remote heart signal estimation has become a widely popular field of study over the past two decades. This is commonly achieved by monitoring and extracting the remote photoplethysmography signal from the subject's face, followed by signal filtering and heart rate calculation. Recently, interest in heart rate estimation using supervised deep networks has risen as they have demonstrated better results compared to unsupervised computer vision techniques. This paper aims to explore the limitations of the conventional method of using a loss function to assess the accuracy of models in predicting heart rate from face videos. We present the findings of our study, where we trained and tested three state-of-the-art deep neural networks using publicly available datasets. Our results reveal a significant divergence between model loss and heart rate accuracy.

Keywords: Imaging, Computer Vision, Deep Learning, Convolutional Neural Network, Heart Rate.

1 Introduction

The human body is composed of various systems that enable individuals to function optimally in various aspects of their lives. While some medical conditions necessitate specialized and invasive procedures with costly equipment for examination, research has shown that several vital health metrics can be measured outside traditional healthcare settings, offering a comprehensive overview of an individual's health and well-being [Chen and McDuff, 2018, Cardone et al., 2020]. One such critical metric is heart rate, which provides valuable insights into an individual's physiological and psychological state.

In light of this, the implementation of remote heart rate sensing has become increasingly significant wherever contact-free sensors are available, such as vehicle in-cabin systems. This involves continuous monitoring of a specific region of interest (ROI), typically the face, to extract a heart signal, which is then processed to estimate the heart rate. While initial attempts of remote heart rate estimation relied on unsupervised computer vision image processing and analysis, recent advancements in supervised deep learning methods, particularly Convolution Neural Networks (CNNs), have demonstrated significantly improved results in this field [Liu et al., 2022]. These models are trained to predict the heart signal using face video or frames as input features by continuously adjusting the model's internal parameters (weights) to minimize the distance between predicted and ground truth values of the heart signal.

Typically, evaluating model performance relies on using the loss, which is a measure of the distance between the model output and the ground truth. The two mostly used methods of selecting which weights to save or deploy is by either using the last set of weights after training, or by finding the weights which give the lowest loss on the validation set (which is a portion of the training set used to monitor for issues such as over fitting). However, for models trained to estimate heart signal, the loss is not an accurate method to find out the best set of weights for heart rate estimation.

In this paper we explain the rationale behind this claim. As our main contribution, we present experimental data from three supervised deep models trained on the Pulse Rate Detection (PURE) [Stricker et al., 2014] and tested on the University Bourgogne Franche-Comte (UBFC) [Bobbia et al., 2019] public datasets to support the proposed claim. We compare our results to those presented in [Liu et al., 2022], using similar workflow for fair comparison, and show the reduced correlation between model loss and heart rate error, as compared with the correlation between validation and test set losses.

The structure of this paper is as follows: in section 2, we present background literature review of deep learning and remote heart rate estimation, with special focus on the deep learning models we use in this study. Section 3 will discuss the methodology we followed starting from the data used until model training. Section 4 will display the results obtained, explain how the models were evaluated, then discuss what conclusions are to be drawn from the results reached. Finally, section 5 will summarise the contributions of this paper and how they were reached.

2 Background

Deep Learning: Heart rate estimation has seen advancements through the application of deep learning techniques. Deep learning, a subset of machine learning, has shown promise in training mathematical models to make predictions and perform tasks using data [Zhang, 2020]. Deep neural networks (DNNs), a key component of deep learning, have gained attention for their ability to extract features at different levels of abstraction, eliminating the need for manual feature engineering [LeCun et al., 2015]. This characteristic makes DNNs well-suited for heart rate estimation tasks, as they can directly process inputs without the requirement of hand-crafted features. Training supervised DNNs involves the use of a loss function to calculate the distance between predictions (i.e., outputs) and ground truth labels, the gradients of the calculated loss are then used to optimize the internal model parameters; this process is referred to as back propagation [LeCun et al., 2015].

Simple DNNs, however, are not well-suited to dealing with certain types of inputs such as images, represented in a computer as a 3D tensor of the shape: (height x width x channel number), channel number being the number of values each pixel in an image has. Classic techniques involved flattening images into a single row of values then inputting those to the DNN. A special class of DNNs, CNNs, were developed to process image inputs by replacing the weighted sum with element-wise multiplication with a kernel followed by summation then some form of sub-sampling. This approach allowed patches of images to be processed together and drastically lowered the number of weights needed, as they were no longer dependant on the input size, but had a fixed number instead i.e., the kernel of shared parameters. Additionally, it allows the model to leverage spatial locality as patches of the image are processed together [LeCun et al., 2015].

Unsupervised heart signal estimation: Initial proof of concept for remote heart rate estimation has demonstrated that the photoplethysmography (PPG) signals, which measures blood oxygenation by monitoring how the light reflective properties of blood vary over time due to fluctuation in blood oxygen content, can be remotely measured on the human face using simple digital consumer cameras [Verkruysse et al., 2008]. This was done by extracting the heart signal from the video's green channel after spatially averaging the RGB video. This concept was further developed in [Poh et al., 2010a], where the facial region was detected from the RGB video then the three channels were separated. Following that, each channel is spatially averaged, and the resulting signal is de-trended and normalised. Independent component analysis (ICA) was applied on the three signals, each obtained from one channel, to separate three independent sources. The PPG signal was most visible in the second source signal. In [Monkaresi et al., 2013], the authors use a similar method to that in [Poh et al., 2010a], but, after ICA, a power spectrum analysis is carried out followed by one of two possible machine learning techniques: k-nearest neighbour and linear regression. The kNN approach proved to be much better at drastically reducing the error.

Supervised deep learning heart signal estimation: Deep learning methods have seen a sharp rise in utilization for the task of heart rate estimation. One of the two main approaches used is to train the model to estimate the PPG signal from facial images then calculate heart rate during the output post-processing phase of the pipeline.

DeepPhys, an end-to-end model, is used in [Chen and McDuff, 2018] for measurement of heart and breathing rate from video using a CNN. The model architecture presented consists of two branches, motion and appearance branches, each made up of several convolutional layers. The motion branch takes in normalized frame differences. Its purpose, in a manner similar to Imaging Ballistocardiography [Balakrishnan et al., 2013], is to extract motion information as minute body motions, resulting from the mechanical flow of blood, provide complementary cardiac information to the PPG signal. The appearance branch takes in frame averages and its main purpose is learning spatial masks which are then sent to the motion branch. This is done to reduce the effect of illumination and other external factors, acting as a form of attention mechanism.

Another end-to-end model of interest is the PhysNet architecture [Yu et al., 2019]. The authors present two possible implementations of this network, the first one consisting of 3D convolutional layers, where several frames are processed simultaneously, and the second uses traditional 2D convolutions followed by a recurrent neural network (RNN), where the output obtained from the previous frame is used in calculating the output based on the next frame. Overall the 3D convolution implementation displayed significantly better results than the 2D RNN implementation. This model is of interest as the authors not only use the model output for heart rate estimation, but they also evaluate the prediction accuracy for other metrics such as heart rate variability and atrial fibrillation detection.

The work proposed in [Liu et al., 2020] builds further upon the concepts presented in [Chen and McDuff, 2018] by introducing several convolutional attention networks (CANs) of a similar structure to the DeepPhys model. 3D-CAN uses 3-dimensional convolutions in both branches, hybrid-CAN uses them only in the motion branch, and the temporal-shift-CAN (TS-CAN) uses regular convolutions but applies a temporal-shift (TS) before each convolutional layer in the motion branch. The TS function allows for temporal information to be exchanged between frames [Lin et al., 2019] thus allowing classical convolutions access to temporal information. Additionally, this approach allows for inference time to drop to 25% of that of 3D-CAN while only slightly reducing the performance.

3 Methodology

During the training process, it is common practice to save the weights that result in the lowest loss on a validation set, which is a subset of the training data. However, in the case of estimating heart signals, the loss metric used may not accurately reflect the loss in heart rate estimation, as these are distinct tasks. Our objective is to question the conventional method of selecting the best weights for deployment. To achieve this, we train multiple models on publicly available data, saving the weights after each training epoch. We then calculate the heart signal loss on the validation and test sets as well as the heart rate root loss for each set of weights. Next, we compare the performance of models selected based on heart signal loss with those selected based on heart rate loss. In the subsequent section, we present the findings of our study, which indicate that a low overall signal loss does not necessarily correspond to low heart rate estimation error. Additionally, we demonstrate that there exists a weaker correlation between the signal losses and the heart rate loss, further emphasizing the need to consider alternative evaluation metrics when selecting the best-performing models for heart rate estimation.

Our experiments utilise the rppg-toolbox repository developed by [Liu et al., 2022]. For a fair comparison we follow the same training steps while saving the model weights after each epoch, then we use our own approach to find the best set of weights. With regards to specific hyperparameters and feature shapes, we follow the default configurations provided by [Liu et al., 2022] in their implementation.

3.1 Dataset

As we wanted to compare our results with the state-of-the-art, following the work presented in [Liu et al., 2022], we use the Pulse Rate Detection (PURE) [Stricker et al., 2014] and the University Bourgogne Franche-Comte (UBFC) [Bobbia et al., 2019] public heart rate datasets as separate train and test datasets.

PURE: The PURE dataset contains acquisitions of 10 subjects (8 male and 2 female) performing different, controlled head motions. Six one-minute acquisitions were carried out per subject, the videos were captured at a frame rate of 30 Hz with a cropped resolution of 640x480 pixels. The heart signal ground truth data was simultaneously collected using a finger clip pulse oximeter at a sampling rate of 60 Hz. The setups for each acquisition were: Steady, talking, slow translation, fast translation, small rotation, medium rotation.

UBFC: The UBFC dataset contains data of 42 subjects sitting in front of the camera playing a time sensitive mathematical game (aimed at augmenting heart rate) while simultaneously emulating a normal human-computer interaction scenario. The video acquisition is carried out at 30fps with a resolution of 640x480. A pulse oximeter was used to obtain the ground truth data.

3.2 Data Pre-processing

As the PURE dataset frame rate and sensor sampling rate are mismatched, we downsample the heart signal to 30Hz using interpolation. Following that, a Haar Cascade face detector is used on all the frames to locate the subject’s face and a square crop is extracted. The crop is resized to 72x72 for DeepPhys and TSCAN and 128x128 for PhysNet. Following that, the input feature (i.e. frames) representation is generated by calculating the normalized frame differences (differences between consecutive frames) and concatenating that to the standardized raw frames along the channel dimension to create a NumPy array of the shape $[N, W, H, C]$ where N is the length of the sequence, W is the width of the frames, H is the height of the frames, and C is the channels (with a value of 6). For faster data loading, each videos in the datasets is typically broken up into several “chunks” of non-overlapping 180 frame sequences (for PhysNet the chunk size is 128 instead as the model only supports factors of 512; this will be further explained below). The PPG waveforms (labels) are stored as numpy arrays in a $[1, N]$ format [Liu et al., 2022].

3.3 Deep Neural Networks

DeepPhys: a 2D CNN that consists of two branches, motion and appearance branches, each made up of several convolutional layers. The motion branch takes in normalized frame differences. while the appearance branch takes in frame averages. However, the implementation provided by [Liu et al., 2022] utilizes raw frames instead of frame averages. The appearance branch generates an attention mask which is then fed to the motion branch to emphasize regions of the face with high physiological information.

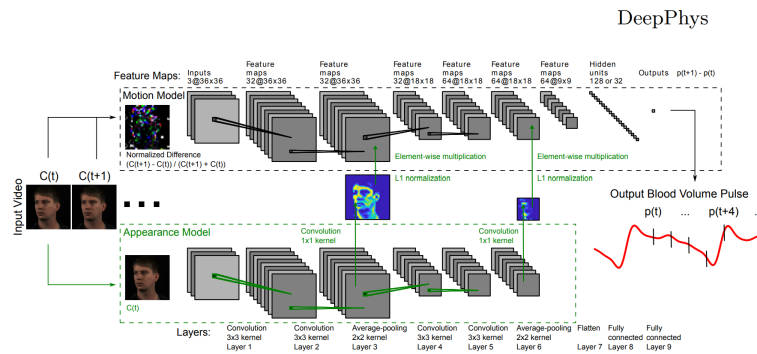


Figure 1: DeepPhys Architecture

TSCAN: An advanced implementation of DeepPhys where each of the main convolutional layers in the motion branch is preceded by a temporal shift function which performs a pixel shift across a certain number of frames (referred to as frame depth). As each frame represents a slice of time, this operation allows regular 2D convolutional to access temporal information without needing to use computationally-heavy 3D convolutions thus allowing for real-time applications.

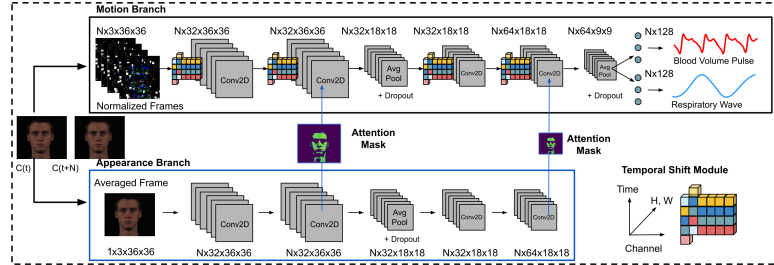


Figure 2: TSCAN Architecture

PhysNet: This model leverages 3D convolutions as part of a traditional feed-forward CNN. Unlike with DeepPhys, where the architecture complexity would mean that using 3D convolutions would prevent real-time applications, PhysNet’s 3D implementation is deemed by the authors as real-time compatible. And while they present a 2D convolutional RNN alternative implementation, the significant performance improvement demonstrated by the first implementation is why it is the implementation used in the rppg-toolbox as well as the presented work.

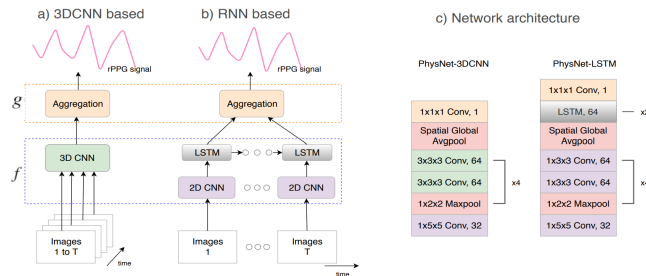


Figure 3: PhysNet Architecture

3.4 Model Training

While the authors in [Liu et al., 2022] present the results when training on both PURE and UBFC, the results shown in the second scenario were all beyond the accepted error range for remote heart rate estimation (< 5 bpm RMSE [Poh et al., 2010b]). Therefore, we choose to focus on the first case, training on PURE and testing on UBFC.

We trained all three architectures on 80% of the PURE dataset and used 20% as a validation set. We used a batch size of 4 for all models, however, in this implementation, each chunk is considered a single "item", meaning that batch of size 4 contains 720 frames (4 x 180); or in the case of PhysNet 512 frames (4 x 128). For the DeepPhys and TSCAN models we use a frame depth of 10.

The models are all trained for 30 epochs using the mean square error loss function as well as the AdamW optimizer [Loshchilov and Hutter, 2017]. The one cycle learning rate policy [Smith and Topin, 2019], which anneals the learning rate from an initial learning rate to some maximum learning rate and then from that maximum learning rate to some minimum learning rate much lower than the initial learning rate, was used. The maximum learning rate chosen was the initial learning rate itself, $9e-3$, which means that the scheduler essentially acts as learning rate decay policy. After each epoch, the model weights are saved, losses and accuracy are then calculated after training is complete.

4 Experimental Results

After training is finished, we calculate the validation set signal loss, test set signal loss, and test set heart rate loss. Following that, we calculate the test set heart rate loss for the model weights that gave the lowest validation set and test set signal losses and compare the heart rate accuracy for all three set of weight as well as with the results reported in [Liu et al., 2022] for all three architectures.

4.1 Heart Rate Error

To calculate heart rate error, the model outputs were concatenated then de-trended. The output signal is then filtered using a Butterworth bandpass filter with cutoff frequencies of [0.75, 2.5] Hz. The average ground truth and predicted beat per minute (bpm) heart rates are calculated for each subject using the Fast-Fourier Transform (FFT) power spectrum analysis. These bpm's are then used to calculate the overall model RMSE.

4.2 Results

We compare the results obtained using the best set of weights as defined by our approach, the validation loss, and the test loss, as well as the results reported by the author [Liu et al., 2022] in table 1.

For both DeepPhys and TSCAN models, we are only able to obtain results similar to those reported in [Liu et al., 2022] using our approach. In fact, following the two evaluation methods mentioned in the provided implementation (validation loss and last epoch), the results obtained are quite worse than the reported results.

PhysNet presented the most interesting results, using losses to find the "best" weights, the results we got were fairly close to the reported results (albeit beyond the acceptable range). However, using our approach, we found a set of weights that reduced the RMSE by a factor of 4, bringing it to near-medical accuracy.

Figure 4 compares the normalized model signal losses and heart rate loss per epoch for each of the models. At first glance, it is visible that validation and test signal losses have a strong correlation. This, however, is not applicable to the heart rate test error. For DeepPhys, the heart rate loss continuously fluctuates regardless of the downward trends exhibited by both signal losses. For the TSCAN model, accuracy heavily fluctuates within the first 5 epochs then remains fairly stable except at epoch 12. While it might be assumed this is correlated to the increase in both losses, we don't observe this behaviour at the other spikes in loss such as epochs 14 and 10. For PhysNet, the accuracy fluctuates quite rapidly, less so between epochs 15 to 20, while showing more sensitivity to changes in test signal loss.

This is further supported by the correlation coefficients presented in table 2 below. For both CAN models, the correlation coefficient of the losses is 20-70% higher than the correlation between the signal losses and heart rate RMSE. For PhysNet, the signal loss correlation is a bit lower, but the signal loss and heart rate RMSE correlation show proportional drop.

	Our Approach	Using Validation Loss	Using Test Loss	Last Epoch	Reported
DeepPhys	2.49	2.98	5.72	2.98	2.53
TSCAN	2.40	3.04	2.96	2.97	2.41
PhysNet	1.04	5.52	5.10	6.99	4.49

Table 1: Beat per minute RMSE comparison of our results and results reported in the rppg-toolbox paper

	Valid Loss and Test Loss	Valid Loss and HR RMSE	Test Loss and HR RMSE
DeepPhys	0.90	0.58	0.53
TSCAN	0.94	0.62	0.79
PhysNet	0.69	0.38	0.39

Table 2: Pearson product-moment correlation coefficients of losses and HR error for the three models

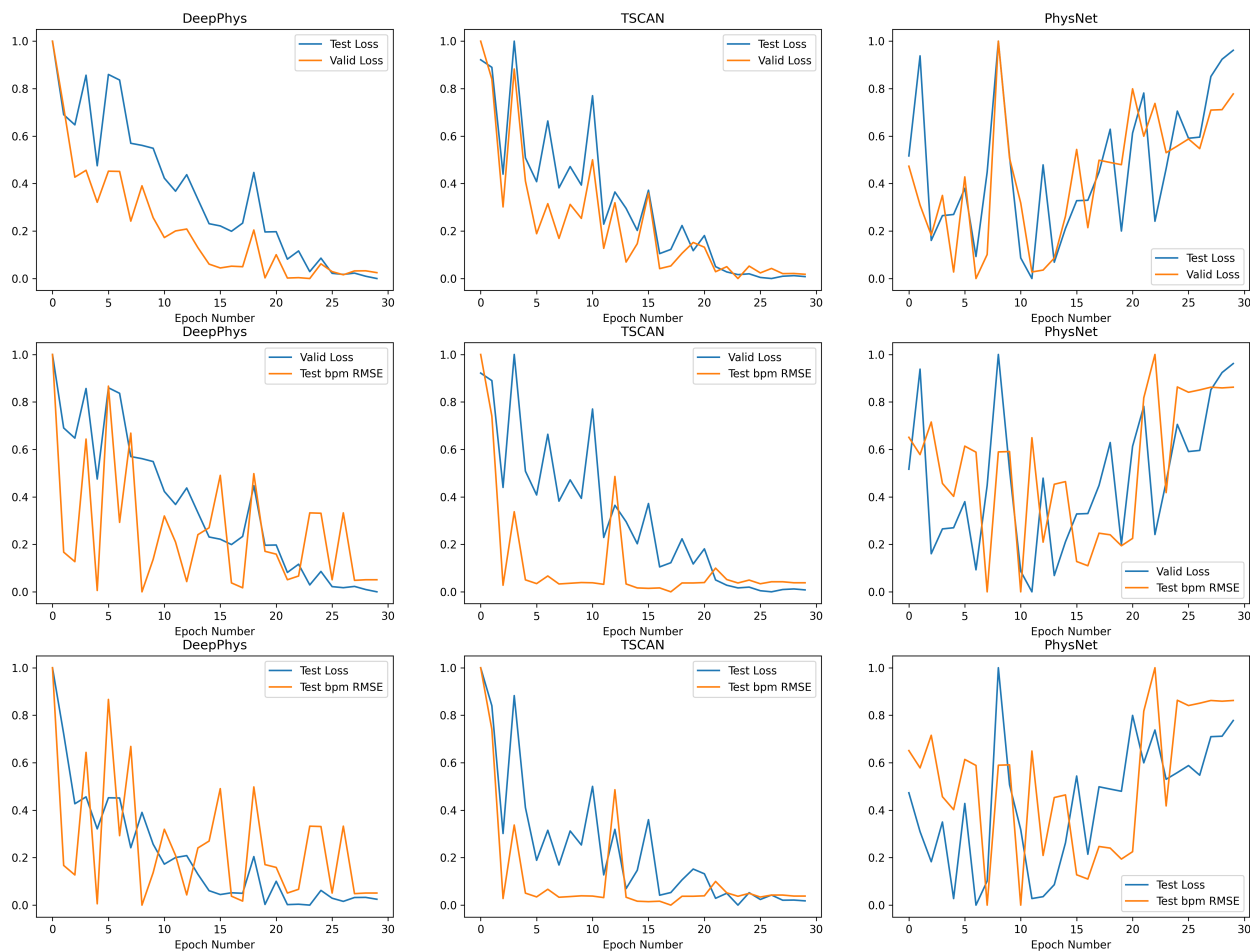


Figure 4: Comparison of losses and accuracy for all models

5 Conclusion

In this paper we present the hypothesis that the loss calculated directly from an video-based r-PPG supervised deep model, typically used for model evaluation, does not necessarily measure how well that model estimates the heart rate, given that those two are distinct tasks. Instead, the beat per minute root mean square error (or some other error metric) should be calculated for each set of weights saved during training to find the best model. We demonstrated how using this approach can allow researchers to find the best set of weights for three different state-of-the-art supervised deep learning networks trained and tested on publicly available data. Additionally, we showed that, across all training epochs, heart signal and heart rate losses do not have a strong correlation.

The results shown highlight the need to use error metrics crafted for the task the deployed model is meant to solve when selecting the model weights to utilize. It, however, does not discourage the use of a different error metric for training, as all the models presented in this study were trained using signal loss yet manage to demonstrate impressive performance once the right set of weights is found.

6 Acknowledgment

The research conducted in this publication was funded by the Irish Research Council under project ID EBPPG/2021/92 as a part of the Employment-Based Programme Postgraduate Scholarship in partnership with the Xperi Corporation.

References

- [Balakrishnan et al., 2013] Balakrishnan, G., Durand, F., and Guttag, J. (2013). Detecting pulse from head motions in video. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3430–3437.
- [Bobbia et al., 2019] Bobbia, S., Macwan, R., Benezeth, Y., Mansouri, A., and Dubois, J. (2019). Unsupervised skin tissue segmentation for remote photoplethysmography. *Pattern Recognition Letters*, 124:82–90.
- [Cardone et al., 2020] Cardone, D., Perpetuini, D., Filippini, C., Spadolini, E., Mancini, L., Chiarelli, A. M., and Merla, A. (2020). Driver stress state evaluation by means of thermal imaging: A supervised machine learning approach based on ecg signal. *Applied Sciences*, 10(16):5673.
- [Chen and McDuff, 2018] Chen, W. and McDuff, D. (2018). Deepphys: Video-based physiological measurement using convolutional attention networks. In *Proceedings of the european conference on computer vision (ECCV)*, pages 349–365.
- [LeCun et al., 2015] LeCun, Y., Bengio, Y., Hinton, G., et al. (2015). Deep learning. *nature*, 521 (7553), 436–444. *Google Scholar Google Scholar Cross Ref Cross Ref*, page 25.
- [Lin et al., 2019] Lin, J., Gan, C., and Han, S. (2019). Tsm: Temporal shift module for efficient video understanding. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 7083–7093.
- [Liu et al., 2020] Liu, X., Fromm, J., Patel, S., and McDuff, D. (2020). Multi-task temporal shift attention networks for on-device contactless vitals measurement. *Advances in Neural Information Processing Systems*, 33:19400–19411.
- [Liu et al., 2022] Liu, X., Zhang, X., Narayanswamy, G., Zhang, Y., Wang, Y., Patel, S., and McDuff, D. (2022). Deep physiological sensing toolbox. *arXiv preprint arXiv:2210.00716*.
- [Loshchilov and Hutter, 2017] Loshchilov, I. and Hutter, F. (2017). Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.
- [Monkaresi et al., 2013] Monkaresi, H., Calvo, R. A., and Yan, H. (2013). A machine learning approach to improve contactless heart rate monitoring using a webcam. *IEEE journal of biomedical and health informatics*, 18(4):1153–1160.
- [Poh et al., 2010a] Poh, M.-Z., McDuff, D. J., and Picard, R. W. (2010a). Advancements in noncontact, multiparameter physiological measurements using a webcam. *IEEE transactions on biomedical engineering*, 58(1):7–11.
- [Poh et al., 2010b] Poh, M.-Z., McDuff, D. J., and Picard, R. W. (2010b). Non-contact, automated cardiac pulse measurements using video imaging and blind source separation. *Optics express*, 18(10):10762–10774.
- [Smith and Topin, 2019] Smith, L. N. and Topin, N. (2019). Super-convergence: Very fast training of neural networks using large learning rates. In *Artificial intelligence and machine learning for multi-domain operations applications*, volume 11006, pages 369–386. SPIE.
- [Stricker et al., 2014] Stricker, R., Müller, S., and Gross, H.-M. (2014). Non-contact video-based pulse rate measurement on a mobile service robot. In *The 23rd IEEE International Symposium on Robot and Human Interactive Communication*, pages 1056–1062. IEEE.
- [Verkruysse et al., 2008] Verkruysse, W., Svaasand, L. O., and Nelson, J. S. (2008). Remote plethysmographic imaging using ambient light. *Optics express*, 16(26):21434–21445.

[Yu et al., 2019] Yu, Z., Li, X., and Zhao, G. (2019). Remote photoplethysmograph signal measurement from facial videos using spatio-temporal networks. *arXiv preprint arXiv:1905.02419*.

[Zhang, 2020] Zhang, X.-D. (2020). A matrix algebra approach to artificial intelligence.