Deep PPG for Better Heart Rate Estimation

D. Longitudinal predictions on ICU data

https://youtu.be/d98OUhllz4s

Andrei Burghelea, David Gutierrez, Denizhan Kara, and Sung Yoo Kim

***Abstract*—Photoplethysmography (PPG) is a low-cost, non-invasive, and optical technique using an infrared light to measure the volumetric variations of blood circulation in microvascular tissue from the skin surface. Improvements to PPG have brought heart rate measurements to wearable devices such as smartwatches and fitness trackers. Inferring cardiac information (e.g. heart rate) from PPG traces in the context of activity levels beyond the sedentary is extremely challenging, because of interferences caused by motion artifacts (MA). The work undertaken in this project is one of Predictive Modeling in which we attempt to predict the heart rate using the time-frequency spectra of synchronized PPG and accelerometer signals as input. The goal of this project is to use the novel large-scale dataset PPG-DaLiA introduced in [1], which includes a wide range of activities performed under close to real-life conditions, to attempt to reproduce the model used in the original reference paper. We also improved the reference model by adding the predicted activity as a feature developed using a separate activity prediction model. The predicted activity is added to the final connected layers in our convolutional neural network (CNN) model and improves the results by reducing the mean average error (MAE) and its variability, especially for the activities where the heart rates are elevated compared to the baseline (e.g. sport activities). The model performance is a subject-based MAE of 7.1±1.3 BPM, compared to 7.65±4.2 BPM for the reference model [1].**

I. Introduction

Photoplethysmography (PPG) is a low-cost, non-invasive, and optical technique using an infrared light to measure the volumetric variations of blood circulation in microvascular tissue from the skin surface. Electrocardiography (ECG) is currently the gold standard for heart rate measurement in clinical settings. More recently, improvements to photoplethysmography (PPG) have brought heart rate measurements to wearable devices, most notably in smartwatches such as the Apple Watch and fitness trackers like FitBit and Garmin devices. The technology has relatively accurate heart rate information to become commonly available in both clinical and everyday environments. Wearable devices are a large and growing segment of consumer devices with demand for increasingly advanced health signals fuelling the growth of more advanced sensor technology. PPG has enabled heart rate calculation in both rested and active states for the average consumer at a significantly reduced cost over ECGs and with the convenience of full mobility.

PPG has been traditionally used in a variety of devices with application domains ranging from medical to fitness. However, inferring cardiac information (e.g. heart rate) from PPG traces in the context of activity levels beyond the sedentary is extremely challenging, because of interferences caused by motion artifacts (MA). MAs are generally caused by the movement of the sensor module relative to the skin and affect the signal quality and the extraction of the parameters of interest (e.g. heart rate). Recently there have been a number of techniques developed for detecting, removing, or attenuating MA and estimating heart rate, including some based on artificial neural networks ([1]–[6]). Reference [7] provides a comprehensive review of the state-of-the-art research on heart rate estimation from wrist-worn PPG signals.

II. Related work

Certain algorithms explored in the reference [1] including SpaMa, SpaMaPlus, and Schaeck2017 are leveraged to try to eliminate noise in the PPG spectrum data. SpaMa can help eliminate sudden changes due to motion, SpaMaPus can help not carry over fluctuations or error into the next tracking periods, and Schaeck2017 can help decrease the noise with the possible use of multiple channels. These are some algorithms explored in the paper for the data processing that we will explore with potential others in our final report. Reference [3] details the importance of human activity recognition,

which is currently achievable using accompanying accelerometer data with high degrees of accuracy and precision. Recent developments of techniques for detecting, removing, or attenuating MA and estimating heart rate include the use of hybrid deep neural networks combining convolutional neural network (CNN) layers with long short-term memory layers (LSTM) in frameworks such as CorNET [4] and PP-Net [5]. This joint framework of CNN and LSTM combined with feature extraction provides a more accurate performance for heart rate estimation. Yet another recent approach is the DeepHeart framework [6], which uses a deep CNN ensemble for denoising PPG signals and removal of MA artifacts, combined with online analysis of the PPG spectrum. The performance of the approaches in [4]–[6] suggests that there is potential for improving the performance of the model in [1] by modifying the network architecture and the input processing.

III. Problem formulation

The work undertaken in this project is one of Predictive Modeling in which we attempted to predict the heart rate using the time-frequency spectra of synchronized PPG and accelerometer signals as input. Our aim was to replicate some of the work presented in the paper titled Deep PPG: Large Scale Heart Rate Estimation with Convolutional Neural Networks [1]. The goal is to use the novel large-scale dataset introduced in [1], which includes a wide range of activities performed under close to real-life conditions, to attempt to reproduce at least one model used in the original reference paper, as well as potentially introduce some improvements by adding the predicted activity as a feature generated using a separate predictive model. The dataset has a total of 11 attributes with over 8 million instances, categorized by activity such as sitting still, cycling, and ascending/descending of stairs and by subject – eight female and seven male with an average age of 30.69 years and some supplementary demographic information such as height, weight and fitness level which are all incorporated as features in addition to the biometric signals. This signal data includes raw sensor data with two devices RespiBan (chest-worn) and Empatica E4 (wrist-worn). The RespiBan provides the ECG, breathing, and motion signals sampled at 700 Hz. The Empatica provides additional data including the PPG signal (BVP), skin conductivity (EDA), and body temperature and motion, which are used as the main features in our models.

IV. Methodology

Reference [1] provides the details about the data collection protocol. We have used the same steps as identified in the reference paper. This means we have segmented the time-series data with a sliding window of length 8 seconds and a window shift of 2 seconds. As core features, we made use of only the first PPG channel and accelerometer channels in the X, Y and Z axis. We may consider additional PPG channels as input for our deep learning model if model development and timing allow. As a second step, we applied Fast Fourier Transform (FFT) on each time-series segment. The result of this step is Nch = 4 channel time-frequency spectra, one per signal channel. In the next step, we cut these spectra, keeping only the 0–4 Hz interval (4 Hz corresponds to 240 bpm). The resulting number of FFT points per segment and channel is NFFT=1025. Finally, z-normalization (zero mean and unit variance) is performed on each channel's spectrum. The final Nch time-frequency spectra serve as input for the deep learning model.

For the data preparation stage, we segmented the data with 8 seconds of sliding windows with a time rolling mechanism with 2 seconds shifts. This enabled us to extract features that we need from the data. Since we need both the time and frequency domain behavior of sensor channels, we extracted both time and frequency domain characteristics of each window. Time-domain characteristics include features indicating regularities and statistical measures on windows like mean, max, standard deviation, and unique/recurring value count, while frequency-domain features represent more complex, yet periodic information within each window. These features are FFT coefficients, number of CWT (continuous wavelet transform) peaks, and autoregression of each window.

Since we were unsure about which features or characteristics will be most contrastive towards heart-rate estimation, we ended up extracting a large number of features from each window that we think might provide information to our model. However,

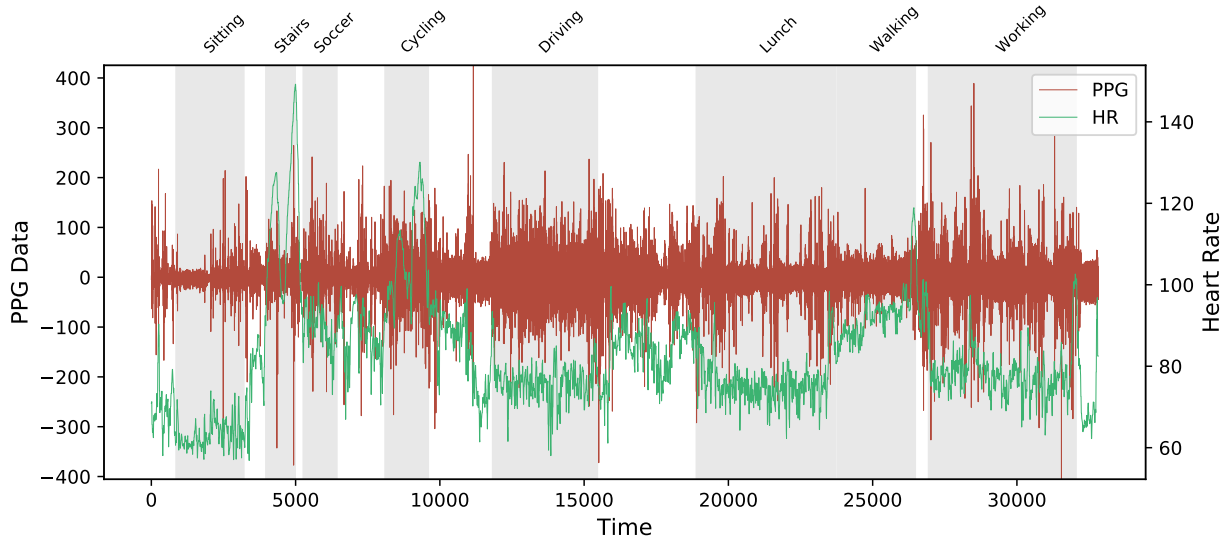


Fig. 1. A visualization of the raw data from the PPG-DaLia: Subject 2's BVP/PPG Data vs Heart Rate Ground Truth. Note the significant amount of noise in the BVP/PPG signal

this approach did lead to some correlation between features as well as some features that represent no information, which can be considered as noise. For this reason, we also applied univariate and multivariate feature selection techniques, which eliminated high correlation and noise features.

After the selection procedures, we tried a variety of modeling procedures. We built a deep learning model similar to Reference [1], as well as simpler options like tree-based models, which allowed us to perform benchmarking on our additional features with easier training procedures. Moreover, such models let us understand the information value of the considered features without being much affected by scaling and correlation issues between input features. Here, our goal is to determine which features are beneficial for our task as well as which architecture is likely to give us the best results.

For the initial training, we have extracted both time and frequency domain features from the pre-processed 8 second time windows and benchmarked them on a subject basis. The features include simple extractions like mean, maximum, and standard deviation of acceleration values in each window, which are intended to capture the time domain characteristics of the features, while more complex features like FFT coefficients, autocorrelation, and linear trend capture frequency domain characteristics of the signals, which are especially important to predict heart rate in the presence of high-frequency acceleration dynamics, which are very common during movements and sports activities for each subject.

In parallel to the improved features, we implemented a model similar to the original CNN model from Reference [1]. This allowed us to both implement the paper model itself, which is the minimum requirement for the project task, but also to build upon and improve the model itself with both architectural changes as well as new features, which we have experimented on in the initial training.

The performance metric is the same as the one used in [1], the mean absolute error (MAE) defined as:

$$MAE = \frac{1}{W}\sum_{w=1}^{W}|BPM_{est}(w) - BPM_{ref}(w)| \quad (1)$$

V. Experiments

Our initial target set at the start of the project was to achieve a lower or equal MAE than the reference paper [1] on 6/9 Activities. We also compared the MAE results for our model with the session-wise evaluation results for the reference model.

Our initial results using the boosted-tree model on a subject-wise training yielded an MAE between 1.6 and 2.8, which is a significant improvement over the MAE recorded in the reference paper [1]. This

improvement was the result of a simple gradient boosting model (XGBoost) without hyperparameter tuning. The evaluation was done by splitting each subject's data on a standard 80:10:10 train-test-validate basis with 80% of the data used for training. The initial results are presented in Table I.

Here, the superior training results are achieved with the use of more complex time series features. Based on the extracted data windows, we have extracted 15 different features that represent the time and frequency domain characteristics of the heart-rate signals. Our features include simple series statistics like mean, variation coefficient (scaled variance), minimum and maximum as well as more complex frequency domain indicators like FFT and autoregressive characteristics. Here, the simple statistics capture the "mean heart rate" information from features while frequency domain indicators act like an implicit "activity recognition" model. These frequency domain features capture the activity, as well as the corresponding signal frequencies for each predicted activity for the current window. As such, our features performed superior to the paper features as our features are covering paper features and and more. With these strong features, we have utilized a gradient boosting algorithm for benchmark purposes. The main reasons we chose XGBoost are as follows. First, it works well with large quantities of time series features, which inevitably have large correlation content. Since our feature pool is quite large, and generated features are from the same source, they carry some duplicate information content in the form of linear correlation. And since XGBoost is intrinsically a decision-tree based model, it does not get affected by high correlations and lets us perform the benchmarking. Secondly, since it is a boosting model, it performs well out of the box without requiring much tuning. For such models, the homogeneity of data is more important than the size, thus the model is allowing us to perform benchmarking of our features for the final model.

Subsequently, we built a convolutional neural network (CNN) similar to the model architecture proposed in Reference [1], in order to incorporate deep learning on activity-based training. The inputs to the deep learning model are four channels of time-frequency spectra corresponding to the x, y, and z accelerometer signals and the PPG signal (BVP), all collected from the wrist sensor.

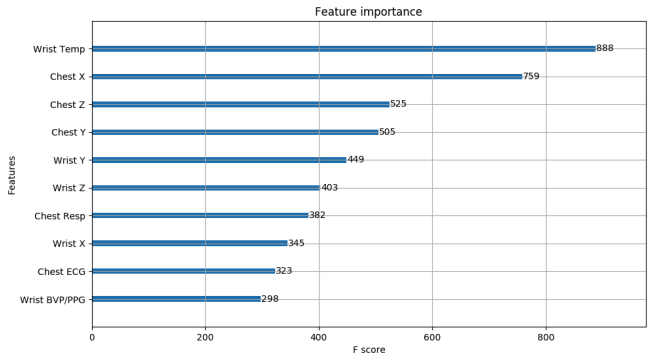


Fig. 2. A visualization of the untransformed feature importance ranking from the PPG-DaLia dataset relative to Ground Truth Heart Rate Prediction using Xgboost

The reference model [1] has a deep CNN architecture with two initial convolution layers and up to 8 additional convolution-maxpool layers. The first two layers are designed to fuse first the input channels (accelerometer measurements and PPG), and the segments used in heart rate tracking, respectively. The successive convolution-maxpool layers are designed to increase the learning effectiveness of the model. One last convolutional layer is included in the model in order to reduce the input dimension of the fully connected layer. The last two layers are fully connected, with the first flattening the input and the second one providing a single value estimated heart rate. Note that the reference model also included a dropout layer which was also included in our model. Based on our evaluations the dropout layer can be omitted without significant effects on the model performance. The model uses an exponential linear unit (ELU) as activation function for all the layers. The loss function is defined as the Mean Squared Error (MSE) between the estimated value for a given segment and the corresponding ground truth. For optimization the model uses the Adam optimizer [8]. We included the same filter size as the reference model to decrease the spatial size of the output. This decreased the number of features volume but increased the depth of the volume. The reference paper mentioned that this technique has been successfully implemented in other architectures like VGGNet, ResNet, and DenseNet [1]. Similar to the reference paper, we

implemented multiple scenarios to try and improve the network architecture. We tried ReLU activation function, different optimizers, and loss functions to try to improve model performance. In addition, we implemented different CNN layers with smaller filters to try and pick up finer detail. However, it seems though the initial CNN architecture similar to the paper [1] is already well set and any of the changes did not improve or otherwise drastically worsen the MAE score. We determined in the end to implement a deep learning model architecture similar to the reference paper to determine if the preprocessing and additional activity prediction would help our overall MAE score. Our initial model replicated the reference network architecture in [1] as described above. An illustration of the architecture is provided in Figure 3 - note that the figure also shows the predicted activity used as input in the final model (see below) but not in the initial model. The model was trained using a 80:20 train-test data split. The results from the initial model were rather disappointing as the performance of our model in terms of MAE was significantly higher than the reference model. Our initial model had higher MAE for almost all subjects and an average MAE of 13.37 $\pm$ 3.1 BPM, compared to the reference model average MAE of 7.65 $\pm$ 4.2 BPM.

To improve the model performance we initially envisioned the use of additional features we benchmarked in the initial training, which have performed very well. This included the development of an ensemble architecture that includes a simple activity recognition model. The activity is very important in the HR estimation and it is usually not explicitly available as an input to the model, thus it was identified early as a candidate to incorporate for model improvement, which we noted was proposed in some other works as well [3]. Moreover, we are aware that the interpretability aspect of the model is crucial for both improving and approval of it. We have performed a literature search on the interpretability aspect and found the SHAP algorithm [9], which can help us with the interpretation aspect of the model.

Several model changes were implemented in order to improve the model performance. The most significant improvement was the addition of the predicted activity as a feature using an additional prediction model. Here, we have developed an ensemble approach to prediction by training a "hypermodel" that is specifically trained for recognizing the user's current activity. The model is a lightweight Random Forest model developed for the multi-class classification task of activity recognition. It is feasible to be used in real-time in user's worn devices, and only uses the sensor data that is available in the wrist device, which are acceleration and BVP. Using this data in real-time, we have achieved between 87%-95% accuracy for activity prediction among subjects. Essentially, it has become a surrogate model for the main HR predictor, allowing the model to contrast better between activities and make more accurate predictions. The surrogate model decreased our mean MAE results, but more importantly, it has greatly decreased the standard deviation of errors, allowing the model to make much more stable predictions over the relatively more homogeneous partitioned data, as expected.

In this final configuration, the model performance was greatly improved compared to the initial model. Improvements were also evident in comparison to the reference model in [1], in particular for the activities where the heart rates are elevated compared to the baseline (e.g. sport activities). Table I shows the comparison of the results obtained. Our ensemble model with activity prediction had similar or better MAE for most subjects and an average MAE of 7.33 $\pm$ 1.15 BPM, compared to the reference model average MAE of 7.65 $\pm$ 4.2 BPM. Improvements were noted in particular for the activities where the heart rates are elevated compared to the baseline (e.g. sport activities). We also note that our ensemble model performance is more consistent between subjects and has significantly lower variability compared to the reference model.

Data processing was performed on our own individual computers with a processing time of around 15 minutes for all the subjects for general processing and around 45 minutes for all the subjects for FFT transform and normalization on a Core i7 Intel processor with 32 GB of RAM running Linux kernel 4.15.0. Training of the deep learning models was mostly performed on lambdalabs.com GPU cloud instances varying from 2xA6000 48 GB RAM 28 vCPU to 8x Tesla V100 16 GB RAM 92 vCPU.

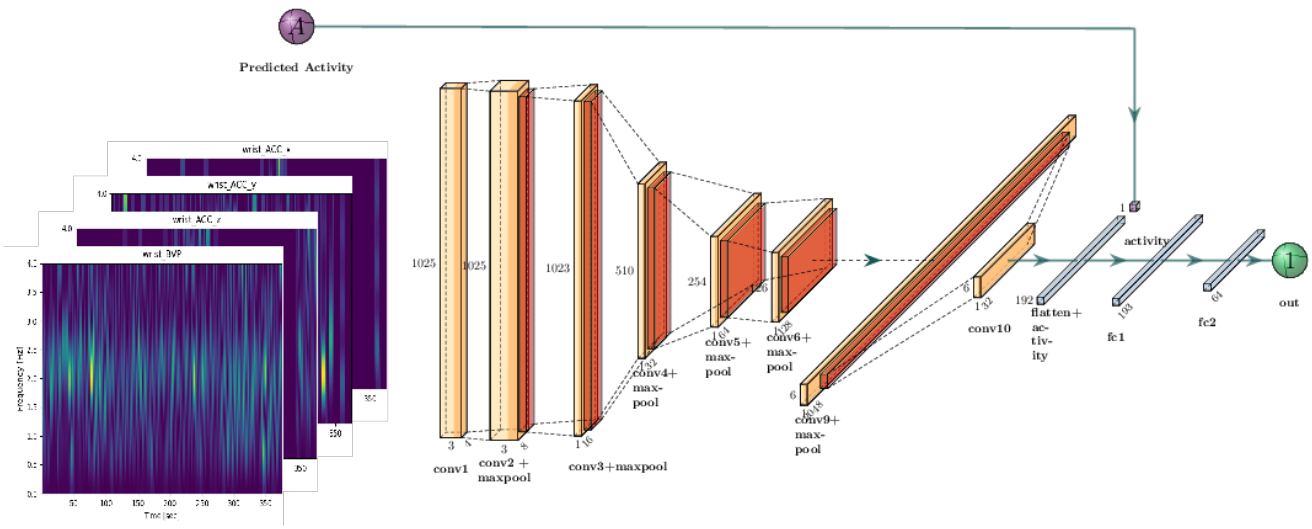
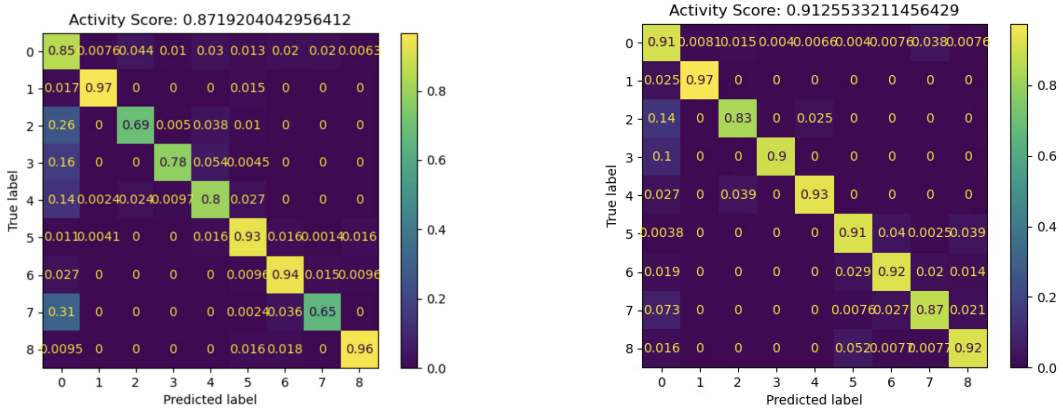


Fig. 3. Network architecture for the proposed model. The initial model did not include the predicted activity shown at the top. The predicted activity was included in the final model and is generated externally using an additional predictive model.



(a) Subject 2

(b) Subject 12

Fig. 4. Activity prediction model performance for Subject 2 (a) and Subject 12 (b). Most of the confusion is between transition actions (Label 0) and stairs climbing (Label 2) and table tennis (Label 3) as expected.

Training time ranged from 25 minutes to 65 minutes depending on the model and the training instance selected.

We cannot conclude without noting the significant better performance of the boosted tree model (XGBoost) compared to all the other models evaluated. Not only did the boosted tree model outperform all the other models in all the categories, but it did so with much lower MAE (average of 2.30 $\pm$ 0.34 BPM) and without any tuning. It is generally difficult to explain and justify how predictions are made with a neural network model. In contrast, a decision tree is easily explained, and the decision "workflow" through the decision tree can be readily implemented as a logical algorithm. For these reasons, we find that a tree model may be better suited and easier to implement as a predictive model for PPG-based heart rate predictions in wearable devices.

VI. Conclusion

In this paper, we sought to improve a state-of-the-art heart rate deep-learning prediction algorithm. One of the primary innovations to this end was

TABLE I
COMPARISON OF RESULTS FOR OUR MODELS WITH THE REFERENCE

Subject	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	S11	S12	S13	S14	S15	All
Reference CNN Average	8.45	7.92	5.96	7.86	18.97	13.55	5.16	11.49	10.65	6.07	9.87	9.95	5.25	5.85	5.25	8.82 $\pm$ 3.8
Reference CNN Ensemble	7.73	6.74	4.03	5.9	18.51	12.88	3.91	10.87	8.79	4.03	9.22	9.35	4.29	4.37	4.17	7.65 $\pm$ 4.2
Gradient Boosting (Xgboost)	2.32	2.1	2.3	2.22	2.13	2.43	2.58	2.29	2.11	1.94	2.85	1.64	2.7	2.83	2.03	2.30 $\pm$ 0.34
Initial CNN Model	12.13	9.91	11.99	9.55	15.38	17.80	13.28	8.62	15.30	11.38	18.65	11.14	17.56	13.76	14.16	13.37 $\pm$ 3.10
Ensemble CNN with Activity Prediction	7.24	7.09	7.81	6.30	7.25	7.71	7.36	5.46	7.49	7.51	9.38	5.61	9.48	8.03	6.24	7.33 $\pm$ 1.15

the preprocessing of various non-linear statistical features of the time-segmented windows of signals in order to get an equal or lower MAE score during evaluation of our model and the implementation of an ensemble model with an activity prediction. The preprocessed features took significant time to implement due to the size of the raw data and the filtering of the discrepancies that occur. The preprocessed data included FFT coefficients, autocorrelation, and linear trend capture frequency domain characteristics. Using the preprocessed features in a given window size, we were able to prove that the features were beneficial using a preliminary testing with XGBoost. We replicated the CNN model in Reference [1], and tried other deep learning architectures to lower the MAE score. Our ensemble CNN model matched or exceeded the performance of the reference CNN model from Reference [1]. Our boosted-tree model performed significantly better than any model we compared to, and we recommend it for use because it is generally easier to explain and to implement in wearable devices.

RESOURCES

Github repository for our code: https://github.com/denizhankara/PPG-DaLiA

Project contributions (in alphabetical order):

- Andrei Burghelea - Input Data Preparation, Model Architecture, Model Training, Documentation
- David Gutierrez - Data Visualization, Documentation, Research Structure
- Denizhan Kara - Input Data Preparation, Model Training, Documentation
- Sung Yoo Kim - Model Architecture, Model Training, Documentation

REFERENCES

[1] A. Reiss, I. Indlekofer, P. Schmidt, and K. Van Laerhoven, "Deep ppg: Large-scale heart rate estimation with convolutional neural networks," *Sensors*, vol. 19, no. 14, p. 3079, 2019.

[2] S. Salehizadeh, D. Dao, J. Bolkhovsky, C. Cho, Y. Mendelson, and K. H. Chon, "A novel time-varying spectral filtering algorithm for reconstruction of motion artifact corrupted heart rate signals during intense physical activities using a wearable photoplethysmogram sensor," *Sensors*, vol. 16, no. 1, p. 10, 2016.

[3] E. Brophy, W. Muehlhausen, A. F. Smeaton, and T. E. Ward, "Optimised convolutional neural networks for heart rate estimation and human activity recognition in wrist worn sensing applications," *arXiv preprint arXiv:2004.00505*, 2020.

[4] D. Biswas, L. Everson, M. Liu, M. Panwar, B.-E. Verhoef, S. Patki, C. H. Kim, A. Acharyya, C. Van Hoof, M. Konijnenburg *et al.*, "Cornet: Deep learning framework for ppg-based heart rate estimation and biometric identification in ambulant environment," *IEEE transactions on biomedical circuits and systems*, vol. 13, no. 2, pp. 282–291, 2019.

[5] M. Panwar, A. Gautam, D. Biswas, and A. Acharyya, "Pp-net: A deep learning framework for ppg-based blood pressure and heart rate estimation," *IEEE Sensors Journal*, vol. 20, no. 17, pp. 10 000–10 011, 2020.

[6] X. Chang, G. Li, G. Xing, K. Zhu, and L. Tu, "Deepheart: A deep learning approach for accurate heart rate estimation from ppg signals," *ACM Transactions on Sensor Networks (TOSN)*, vol. 17, no. 2, pp. 1–18, 2021.

[7] D. Biswas, N. Simões-Capela, C. Van Hoof, and N. Van Helleputte, "Heart rate estimation from wrist-worn photoplethysmography: A review," *IEEE Sensors Journal*, vol. 19, no. 16, pp. 6560–6570, 2019.

[8] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[9] S. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," *arXiv preprint arXiv:1705.07874*, 2017.