Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

CENTER FOR
PROJECT-BASED
LEARNING

Department of Information Technology and Electrical Engineering

Fall Semester 2024

PPG Signal Enhancement via Sensor Fusion: IMU-Driven Artifact Removal Using Transformers Architecture

Semester Thesis

Andri Furrer
andfurrer@student.ethz.ch

January 2025

Supervisors: Dr. Tommaso Polonelli, tommaso.polonelli@pbl.ee.ethz.ch
(external from Politecnico di Milano) Alice Scandelli, alice.scandelli@polimi.it
Aurelio Teliti, aurelio.teliti@polimi.it

Professor: Prof. Dr. Luca Benini

Acknowledgements

I want to express my gratitude to my supervisors Dr. Tommaso Polonelli, Alice Scandelli and Aurelio Teliti for their much-appreciated guidance, advice and helpful discussions. I appreciated learning from them and I am thankful for their feedback on my work.

Special thanks to Prof. Dr. Luca Benini for granting me the opportunity to work on this project and to the Center for Project-Based Learning (PBL) for providing the appreciated infrastructure.

I thank Andrea Bruder for his valuable feedback and constructive criticism of my work and report. Finally, I thank my family and friends for their much-valued opinions and everlasting support.

Abstract

Many wearables today have photoplethysmography (PPG) sensors included as they provide information on vital signs through efficient mapping to the gold standard electrocardiogram (ECG) signal, as directly measuring the ECG is unfeasible. This is important as wearable devices have become omnipresent devices playing an important role in providing meaningful vital sign data. However, as the sensors are heavily influenced by movements of the human body, the PPG signals are distorted by motion artifacts (MA). Researchers have extensively focused on traditional methods for MA removal, including filtering and algorithmic methods, while recently deep learning architecture gained more attention. This project aims to implement a transformer architecture using PPG signals as inputs to reliably predict the corresponding ECG signal, using a novel approach of also adding inertial measurement unit (IMU) data, consisting of accelerometer and gyroscope data, to the PPG signal to gain more robust and reliable ECG predictions. The architecture of choice is an encoder-only transformer, consisting of an embedding layer, a positional encoding layer, several encoder layers and an output layer. First, this transformer architecture is trained with only PPG inputs on a dataset with relatively calm physical activities before it is trained on a second dataset with more energetic activities. With a personalized setting using a single subject and a single activity, the model achieved a mean squared error (MSE) value of 0.013. The results imply that the model is working decently on calmer activities as well as on single subjects performing a single activity, while generalization to more energetic activities and multiple subjects appears to be mediocre. In the next step IMU data consisting of accelerometer and gyroscope data is used together with the PPG signal as inputs to train and evaluate the model again on both datasets. The lowest MSE value of 0.025 using a single subject and a single activity is a bit higher than before without IMU data, but overall the values of the used metrics increased for the better. The extended input helped the model to understand some features of the underlying model, but it could not elevate the model's performance significantly. Nonetheless, using IMU together with PPG is beneficial for understanding the underlying motion and could play an important role in future MA reduction methods.

Declaration of Originality

I hereby confirm that I am the sole author of the written work here enclosed and that I have compiled it in my own words. Parts excepted are corrections of form and content by the supervisor. For a detailed version of the declaration of originality, please refer to Appendix B

Andri Furrer,
Zurich, January 2025

Contents

List of Figures

List of Tables

Chapter 1

Introduction

PPG is a broadly applicable optical technique to measure blood volume changes. A PPG sensor uses an LED as a light source and a photodiode as a sensing part to measure the light. In figure 1.1 are the two modalities shown, the transmission mode that is used in the common finger clip solutions, where LEDs and photodiodes are placed on opposite sides of the body, and the reflective modality where LEDs and photodiodes are placed on the same side and the light is reflected by the tissues which is commonly used in smartwatches. Typically three different wavelengths are used, green, red and infrared (IR) and from these measurements, vital signs such as heart rate, heart rate variability, respiration rate, and oxygen saturation can be retrieved. This non-invasive and practical technique is widely used in wearable smart devices, notably smartwatches, earables, fitness trackers, and smart glasses.

The main challenge consists of human body movements during daily activities that cause noise and disturbance of PPG signals which can be observed in figure 1.2, known as MA, as clean PPG signals are essential to retrieve reliable vital signs. Hence there exist different methods to detect and remove these MA where most of these methods focus on traditional signal processing algorithms, that achieve a high signal-to-noise ratio (SNR) such as the Butterworth-independent component analysis (ICA)-Adaptive Filter MA removing technique proposed by Luke et al. (2018), or machine learning methods for MA detection such as the gated residual network (GRN)-transformer proposed by Le (2023), achieving 98% accuracy in artifact detection [1], [2]. This project focuses on an approach with a modified version of the transformer architecture, as introduced by Vaswani et al. (2017), and uses datasets that have PPG signals with the corresponding ground-truth ECG signals, together with motion information provided by an IMU sensor, that is accelerometer and gyroscope data [3]. An example of recorded gyroscope data during the activities of sitting and running is provided in figure 1.3. The project aims to

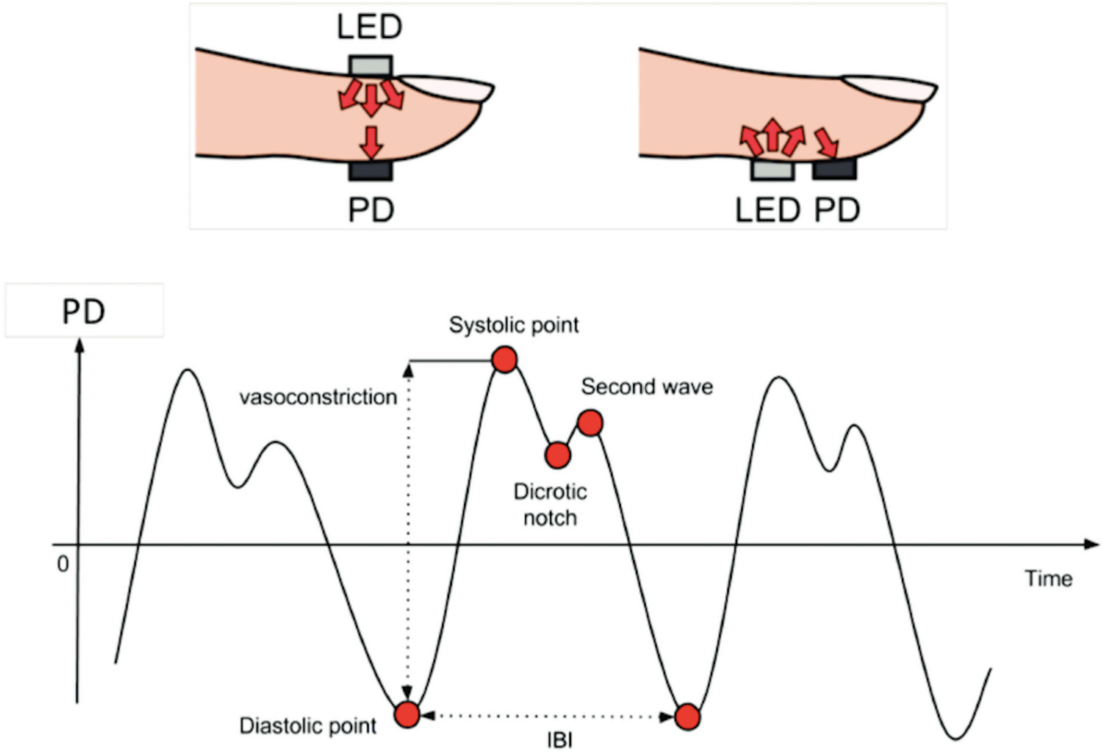


Figure 1.1.: The LED and photodiode setup on the left finger corresponds to the transmission mode and the setup to the right corresponds to the reflective mode. The lower part shows the shape of a typical PPG signal form and its characteristic points [4].

merge the PPG and IMU information to predict a robust ECG signal, that can be later used to extract vital signs.

1.1. Motivation

For the aforementioned vital measurements, such as heart rate, heart rate variability, respiration rate and oxygen saturation the ECG signal is still considered to be the gold standard by many [5], [6]. Nevertheless, ECG requires electrodes attached to the human body, which is unsuitable for long-term data acquisition, and therefore PPG sensors, which can be placed in various devices, are a viable and convenient alternative [7]. However, body movements during daily activities also affect the PPG sensor and introduce MA that need to be addressed when retrieving the ECG signal.

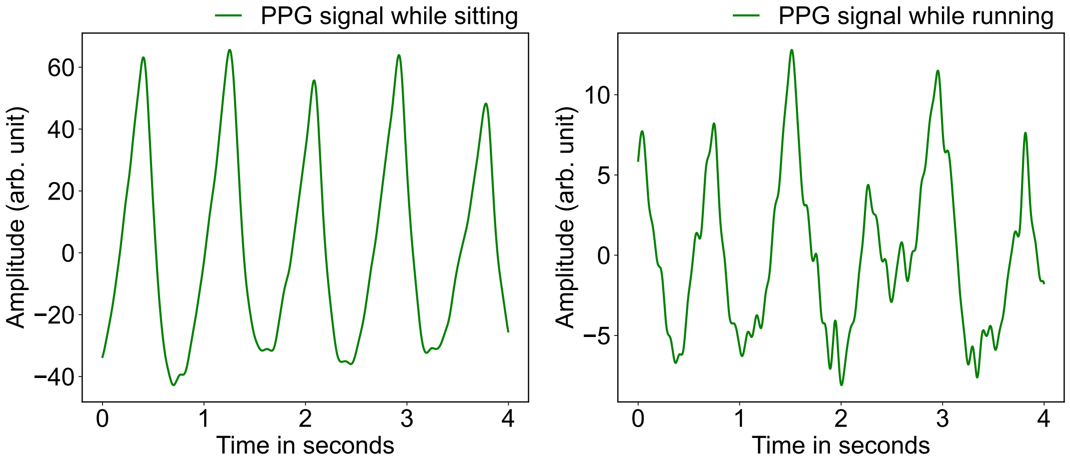


Figure 1.2.: The plot to the left shows a PPG signal recorded during sitting and to the right is a PPG signal during running where the induced MA can be observed.

The transformer architecture originally designed for natural language processing (NLP) has proven to be very effective for many other tasks since its introduction. Due to the transformer's ability to track long-term dependencies, the architecture has the potential for time-series forecasting. Qingsong Wen et al. (2023) show the properties and applications of different variations of the transformer architecture for time series [8].

This project aims to develop a transformer-based architecture for robust MA removal and denoising of PPG signals, by predicting the corresponding ECG signal. In the first step, the architecture will utilize a dataset of PPG signals from the finger, affected by MA, to train a transformer model, matching the accuracy of current state-of-the-art transformer models [9], [10]. Next, a dataset of PPG signals from the wrist with more vigorous activities is used, to compare the model's performance on this dataset concerning the first one with relatively still activities [11]. Once this is achieved, IMU data is used to extend the model's input and to compare the newly trained model's performance to the vanilla transformer architecture in the first two steps. Once this is achieved, the model could be optimized for microcontroller deployment, with particular attention to power consumption, memory footprint, and latency.

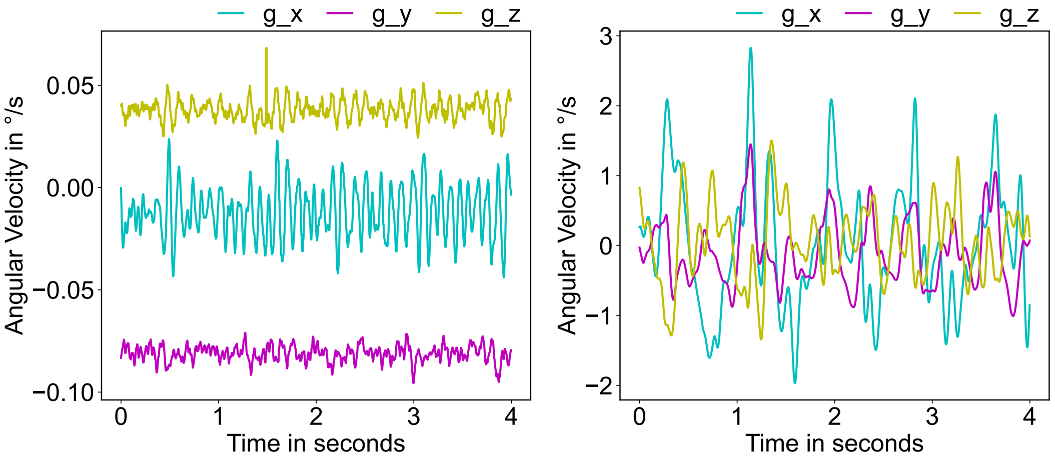


Figure 1.3.: The plot to the left shows the three gyroscope values recorded during sitting and the plot to the right shows the three gyroscope values during running, where the higher angular velocity can be observed.

1.2. Objective

1.2.1. Research Questions

1. Implement a working transformers-based model working on time-series signals and apply the model to a dataset containing PPG signals recorded from the finger to predict the ECG signal.
2. Train and apply the same model to a dataset of PPG signals recorded from the wrist with more vigorous activities, to compare the model's performance on this dataset concerning the first one from the finger with relatively still activities.
3. Add IMU data and use it together with the PPG as input, train the same model architecture again on both datasets and compare it to the performance of the model of the first two steps.

1.2.2. Outline

The transformer architecture and the used datasets are first introduced in Chapter 2 before the related work is covered Chapter in 3. Chapter 4 dives into the preprocessing, training and evaluation steps and the implementation details, followed by the obtained results in Chapter 5. The findings are contextualized in the discussion Chapter 6 before the conclusion and future outlooks are provided in Chapter 7.

Chapter 2

Theory / Background

2.1. Transformers

The transformer architecture introduced by Vaswani et al. (2017) relies on the attention mechanism, where this attention mechanism calculates the relative importance of the components of the input sequences to the transformer [3]. This enables the transformer network to capture long-term dependencies. The vanilla transformer that can be seen in figure 2.1 is based on an encoder block that extracts the features of the given input sequence, and a decoder part that generates the output based on the decoder input and the received encoder embedding, processed with the attention mechanism. This vanilla model architecture can be used as an encoder-only, decoder-only or full transformer network model and adapted according to the underlying task by adding layers or changing components such as the attention mechanism.

2.2. Finger dataset

The pulse transit time PPG dataset was created by Mehrgardt et al. (2022) and is hereafter referred to as the finger dataset [9], [10]. This dataset contains time-synchronized data from multiple sensors placed at the body including PPG, IMU and ECG from twentytwo subjects performing three different physical activities. The subjects performed the activities *sitting*, *walking* and *running* for some minutes with the setup illustrated in figure 2.2, wearing a finger clip containing the PPG and IMU sensors and three electrodes recorded the ECG in parallel. The PPG sensors were placed at the finger in two distinct locations, the distal and proximal phalanx, yielding measurements for three different wavelengths, red, green and infrared and inertial data from the accelerometer and gyroscope. The IMU data contains the acceleration in x-, y- and z-direction measured

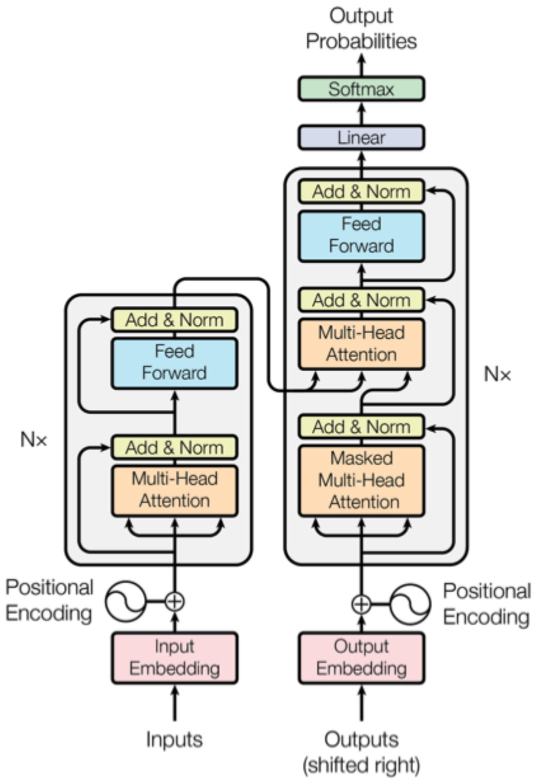


Figure 2.1.: The transformer architecture was proposed by Vaswani et al. (2017), with the encoder block to the left and the decoder block to the right of the figure [3].

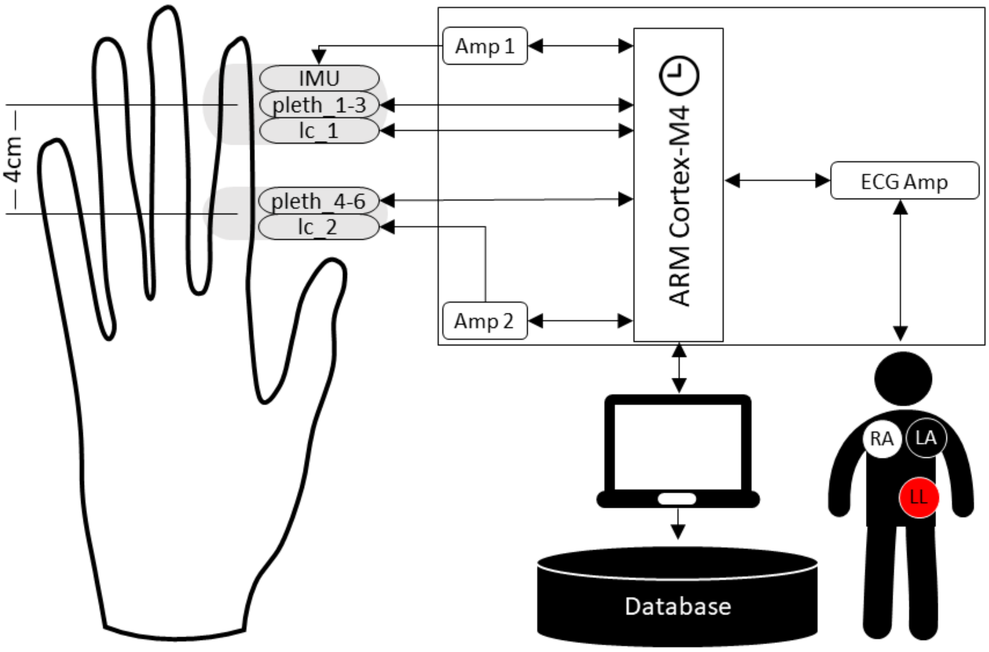


Figure 2.2.: The device setup for the data acquisition of the finger dataset illustrates the different sensor locations [9].

in g as well as the gyroscope data for the angular velocity around the x-, y-, and z-axis measured in $°\backslash s$. All sensors were connected to a microcontroller, reading the sensors within a two ms window corresponding to 500 Hz, apart from that all the sensor data is unfiltered. The dataset contains 16'217'132 samples, is well balanced across the activities with 203 running, 202 sitting and 202 walking minutes recorded and is stored in comma-separated-value (CSV) files together with a time column.

2.3. Wrist dataset

The wrist PPG during exercise dataset was created by Jarchi et al. (2017) and is hereafter referred to as the wrist dataset [11]. This dataset contains PPG and IMU data from sensors placed on the wrist recorded from eight subjects during the physical activities *walking*, *running* and two different bike riding configurations, *low resistance bike* and *high resistance bike*. For reference reasons, an ECG from the chest recorded with two electrodes is included. The IMU data contains the acceleration in x-, y- and z-direction measured in g with two configurations low and wide, where the low configuration has a lower noise floor to measure smaller motion components and the wide configuration can measure larger motion components when the low accelerometer saturates. A gyroscope is also included to track the angular velocity around the x-, y-, and z-axis measured in $^{\circ}\backslash s$. All sensors were sampled with 256 Hz and the authors applied PPG and ECG filtering before converting it to waveform database (WFDB) format, while also adding a time column. Specifically, they mentioned that the cycling PPG traces were low pass filtered using a second order infinite impulse response (IIR) Butterworth digital filter with a 15 Hz cut-off and zero group delay with the Matlab filtfilt command. All ECG records have a 50 Hz notch filter applied as part of the Actiwave control software to remove mains interference [12]. The dataset contains 1'735'142 samples and is slightly unbalanced as not all subjects participated in all activities, hence there are roughly 14 minutes of high resistance bike, 20 minutes of low resistance bike, 18 minutes of running and 25 minutes of walking activities recorded.

Chapter 3

Related Work

Researchers have extensively focused on traditional methods for MA detection and removal, including filtering and algorithmic methods. For instance, Wijshoff et al. (2017) proposed a generic algorithm to remove MA, allowing to recover artifact-reduced PPG signals for beat-to-beat analysis while walking on a treadmill [13]. The results showed a more stable interbeat interval (IBI) and oxygen saturation level (SpO_2) when the step rate and pulse rate were distinct, that is the external physical motion and the pulse rate of the heart, indicating effective MA reduction. Luke et al. (2018) proposed three different MA removal algorithms: ICA-Adaptive Filter, Butterworth-ICA-Adaptive Filter and Butterworth-Wavelet Transform algorithm, where the Butterworth-Wavelet Transform Algorithm provides a high SNR of 41.5235 dB during the fingertip movement of applying pressure without compromising any signal characteristics, hence matching the heart rate and oxygen saturation (SpO_2) levels closely [1]. Nabavi & Bhadra (2020), mentioned the lack of independence between PPG signals for ICA and the computational complexity for adaptive filtering as a drawback and proposed a new sensor fusion technique, achieving an accuracy higher than 95 % in estimating cardiac parameters [14]. This technique acquires data from a PPG sensor and accelerometer to reduce MA by filtering out the frequencies of motion through stopband filters, which does not require any preprocessing or conditioning and is not dependent on the quality of the reference signal. Le et al. (2024) proposed an extended transformer architecture for MA detection, the GRN-transformer, and found that adding a GRN layer as a separate layer is more beneficial than integrating it within the transformer's Attention mechanism. Further, the GRN-transformer achieves 98 % accuracy, 91 % precision, 96 % recall and an F1 score of 94 % if the sigmoid activation function is used for the gated linear unit (GLU), a crucial element of the GRN-transformer [2]. Pinto et. al (2024) proposed a modified U-Net architecture, which is essentially a modified convolutional network, to infer the ECG signal from the PPG signals [15]. They used four different datasets, from which the PPG-daily life activities (DaLiA) is closest to real-world activities, and achieved for the two different model settings personalized and generalized MSE values of 0.015 and 0.026 respectively.

As PPG sensors are often deployed on small wearables, the research community faced the demand for lightweight and energy-efficient methods of detecting and removing MA, for example not using an accelerometer uses less energy, computation power and hence overall less hardware. Zargari et al. (2021) proposed a low-power non-accelerometer-based PPG MA removal by using Cycle generative adversarial networks (GAN). The method includes a two-stage approach, where first noisy signals get detected by a convolutional neural network (CNN) and then through CycleGAN, the clean signal is reconstructed. The noise detection module has an accuracy of 99 % and CycleGAN outperforms other methods while being 9.5 times more energy-efficient [16]. Zheng et al. (2023) proposed a lightweight deep neural network called Tiny-PPG for internet of things (IoT) edge devices for MA detection [17]. With additional model pruning, Tiny-PPG performed state-of-the-art detection accuracy of 87.4 % while only having 19'726 parameters. This is an exemplary case for the trade-off between the size of the model, computation time needed and accuracy, where using a tiny model is more feasible to load on a microcontroller but the accuracy is less compared to a larger model.

Chapter 4

Model Implementation

4.1. Repository

The entire implementation of the project can be found on GitLab: transformer repository and Github: transformer repository.

4.2. Transformer Architecture Choice

Figure 2.1 from Chapter 2 shows the two main blocks of the transformer architecture, the encoder and decoder block. The decoder block is good for generating data that is similar to its input, whereas the encoder is great for capturing features of its input data. The combination of both these blocks combines their respective strength and was the first model choice to create a working transformer architecture to predict the ECG signal from the input PPG signals. The PPG signals were used as input for the encoder block and the ECG was used for the decoder input, shifted by a single step to apply teacher forcing, a method to feed the real target value to the model. However, this approach has some severe limitations. First, this approach calls for auto-regressive decoding during inference on the validation and test set. During auto-regressive the model makes its first prediction with the given input and a special start token. In the next round, the model uses again the input from before, including the next given input and the model prediction from the last step. Auto-regressive decoding depends on the model's prediction, which requires sequential computation instead of parallel, which is normally one of the great strengths of transformers. However, in this case, predicting the ECG on a sequential basis takes longer than predicting the sequence in parallel. A second limitation lies in the drift off of the predicted ECG values, where the model relies confidently on its own previously made ECG predictions, due to the extensive training with teacher forcing.

Lastly, MA reduction in the field should remain mainly on PPG signals, as the ECG signal is often not present. One could argue, that this does not matter during training when actual ground-truth ECG signals are present, but it is nonetheless beneficial if the model does not rely on the ECG signals such that it reflects the conditions in the field. That is why the model of choice for the ECG prediction is an encoder-only transformer, that is predestined for sequence-to-sequence regression and relies only on the PPG and eventual IMU signals as input.

4.3. Encoder-Only Transformer

The encoder-only transformer is built of several crucial components which are explained here together with the general workflow of the architecture and the complete model can be seen in the pipeline scheme in figure 4.1. The embedding layer corresponds to a mapping layer that takes the input sequence d_{input} and scales it to the embedding dimension d_{model}, which is the model's internal dimensionality. Next, positional encoding is applied such as in equation 4.1 and 4.2, to keep track of the position of the tokens in a sequence, as the transformer architecture lacks an inherent sense of order for the tokens.

$$PE_{(pos,2i)} = sin(\frac{pos}{10000^{\frac{2i}{d_{model}}}}) \tag{4.1}$$

$$PE_{(pos,2i+1)} = cos(\frac{pos}{10000^{\frac{2i}{d_{model}}}}) \tag{4.2}$$

The main part of the model consists of the number of encoder layers n_{layers}, where each layer consists of multi-head self-attention that captures dependencies across the entire input sequence, a feedforward network that applies non-linear transformations to the input and residual connections, that skip components, normalisation and dropout parts to ensure stability during training and regularization. These encoder layers process the sequence to extract a feature-aware representation, that is lastly applied to a fully connected output layer, that reduces the dimensionality from d_{model} to the desired output dimension d_{output}.

4.4. Preprocessing

This section covers the general aspects of the required preprocessing, whereas the dataset-specific preprocessing steps are described in the subsections below. Apart from dataset-specific filtering, normalization is needed to ensure stability during training and proper gradient updates. To address subject and activity-wise differences in the acquired PPG, ECG and IMU signals, the signals were normalised for the individual subjects and activities that are present in the respective datasets to the range of $(-1, 1)$, using a MinMaxScaler (sklearn.preprocessing.MinMaxScaler) [18]. This helps mitigate vanishing or

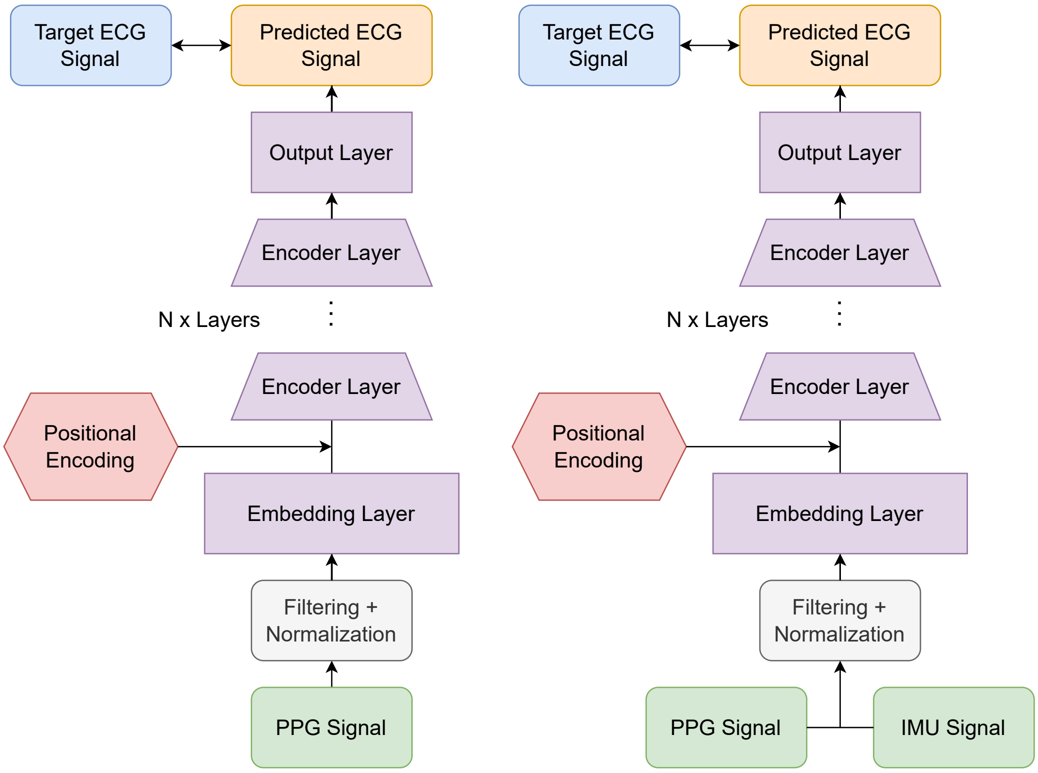


Figure 4.1.: The figure gives an overview of the underlying model architecture where the left block shows the model architecture used in the first two tasks with only using PPG signals as input, and the right block shows the model as it is used in the third task, where IMU data, consisting of accelerometer and gyroscope data, is used together with the PPG signal as input.

exploding gradients while training. The model was implemented with the PyTorch library with tensors (torch.Tensor) as a data structure, to also allow proper device handling [19]. As the transformer handles sequences, the samples get aligned to sequences of a fixed length, that can be set before training. To capture an entire beat interval the sequence length is set to a 1000 in the standard configuration, to account for the 500 Hz sampling rate in the finger dataset. The sliding window was set to 100, which is 10 % of the sequence length.

To ensure a validating and testing step on unseen data, roughly 70 % of the dataset was used for training, 15 % for validation and 15 % for testing. When multiple subjects were trained the subjects were split across the three different sets randomly, that means for training and test runs with multiple subjects the model was tested on data from new subjects that it had not seen during training. When the model was trained only on a single subject, then the splits were taken from this same subject and the model was validated and tested with data from the same subject. To ensure proper data loading, data loaders (torch.utils.data.DataLoader) were used for the train, validation and test set, and the shuffling option was set to true, to ensure the necessary variances across batches. The splitting into the different train, validation and test sets including the shuffling process required a random generator with a fixed random seed to replicate the results. Together with the batch size and the sequence length the input to the encoder model consisted of the tensor $[batch\ size, sequence\ length,\ d_{input}]$ and the desired output $[batch\ size, sequence\ length,\ d_{output}]$. This means for every input PPG sequence the model predicts an equally long ECG sequence from the same time interval.

4.4.1. Finger dataset

The finger dataset as introduced in Chapter 2 was preprocessed by the application of a Butterworth bandpass filter with a sampling frequency of 500 Hz, as well as low- and high-cut frequencies of 0.4 Hz and 10 Hz respectively, on all three different wavelengths of the PPG signals as well as the ECG signal. The figures 4.2, 4.3 and 4.4 show a random sequence for the PPG signal, and figure 4.5 shows the ECG signal for the same sequence. To capture all the information in the IMU, the gyroscope signals shown in figure 4.6 and the accelerometer signals shown in figure 4.7 were not filtered. The input dimension d_{input} for the first task for the transformer model was set to three when the finger dataset was used, as the three PPG signals red, green and IR are present in this dataset. For the IMU part the input dimension d_{input} was set to nine, to adjust for the additional six parameters, consisting of the acceleration in x-, y- and z-direction as well as the gyroscope data for the angular velocity around the x-, y-, and z-axis.

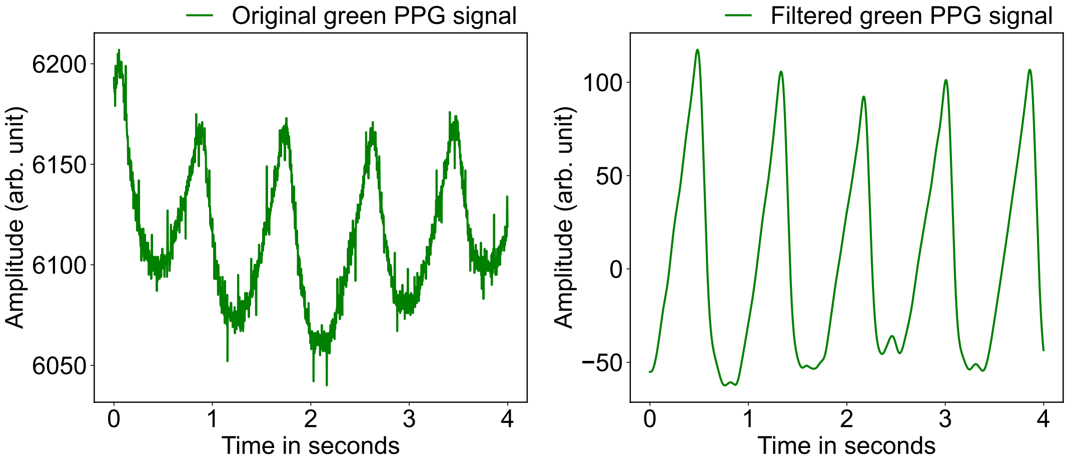


Figure 4.2.: The plot shows an exemplary sequence from a subject during the sitting activity of a PPG signal with the green wavelength in the original and filtered version from the finger dataset.

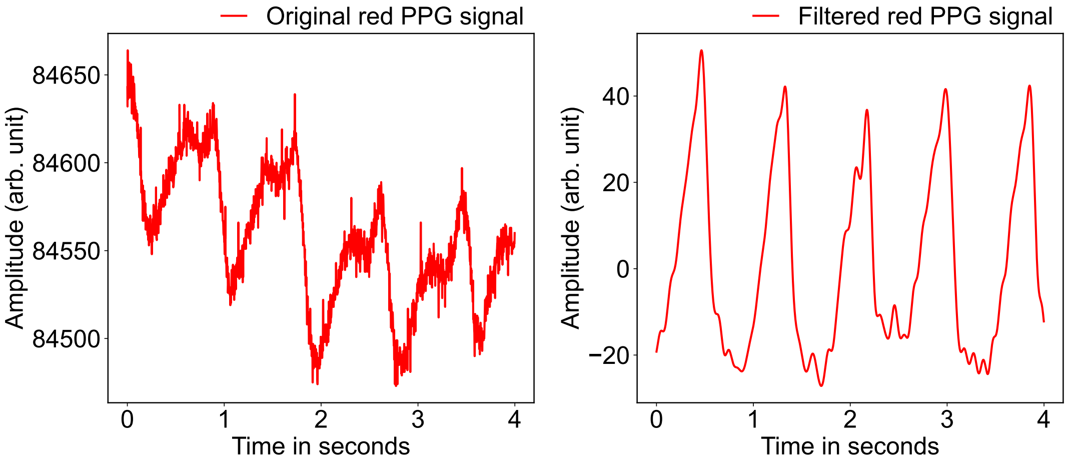


Figure 4.3.: The plot shows an exemplary sequence from a subject during the sitting activity of a PPG signal with the red wavelength in the original and filtered version from the finger dataset.

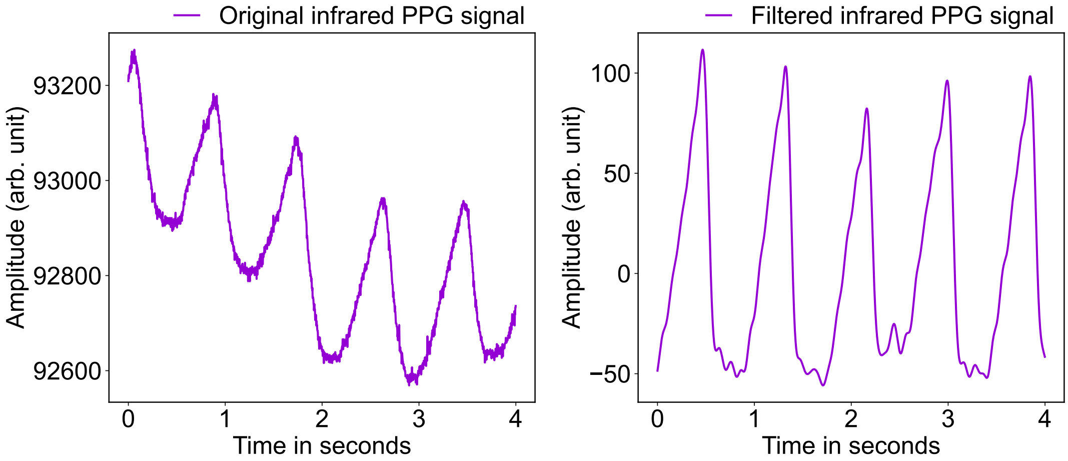


Figure 4.4.: The plot shows an exemplary sequence from a subject during the sitting activity of a PPG signal with the infrared wavelength in the original and filtered version from the finger dataset.

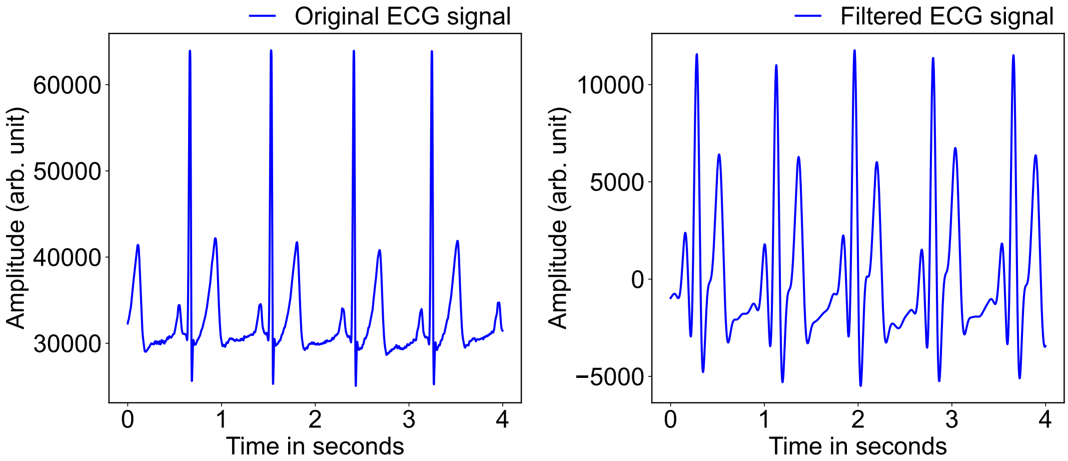


Figure 4.5.: The plot shows an exemplary sequence from a subject during the sitting activity of the ECG signal in the original and filtered version from the finger dataset.

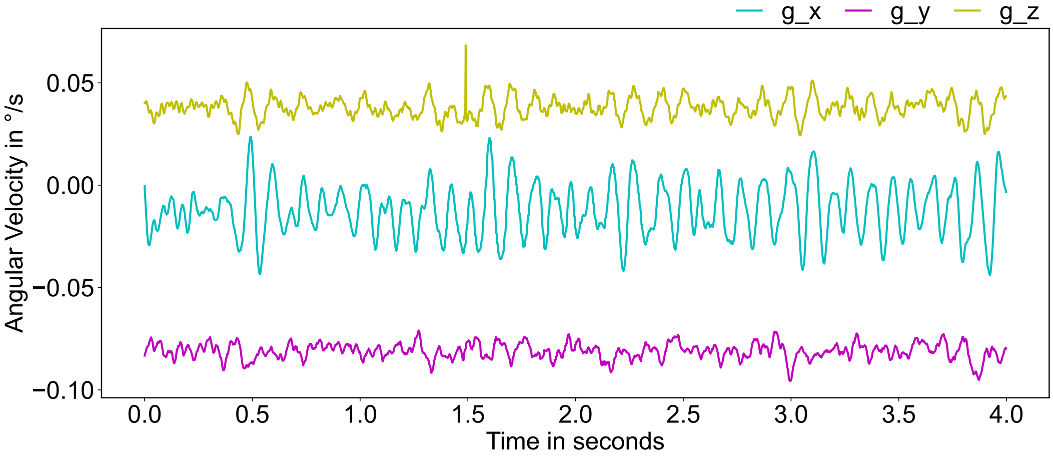


Figure 4.6.: The plot shows an exemplary sequence from a subject during the sitting activity of the three gyroscope signals from the finger dataset.

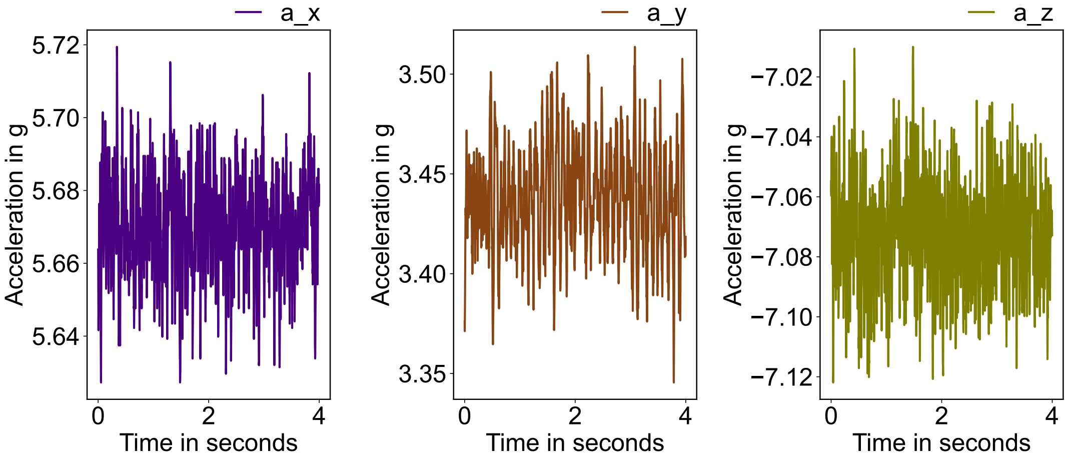


Figure 4.7.: The plot shows an exemplary sequence from a subject during the sitting activity of the three accelerometer signals from the finger dataset.

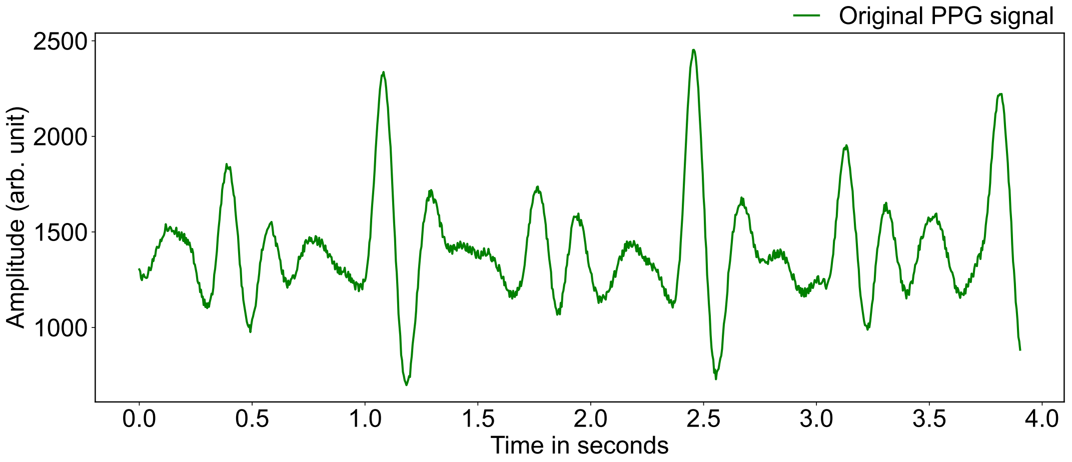


Figure 4.8.: The plot shows an exemplary sequence from a subject during the walking activity of the original PPG signal from the wrist dataset.

4.4.2. Wrist dataset

The PPG signals from the wrist dataset introduced in Chapter 2 were already filtered and a random sequence from the original signal is shown in figure 4.8. The ECG signal was preprocessed by the application of a Butterworth bandpass filter with a sampling frequency of 256 Hz, as well as low- and high-cut frequencies of 0.4 Hz and 10 Hz respectively, and an exemplary ECG sequence can be seen in 4.9. To capture all the information in the IMU, the accelerometer and gyroscope signals were not filtered. Figure 4.10 shows the three gyroscope signals, figure 4.11 shows the three accelerometer signals in low noise configuration as explained in Chapter 2, whereas the other three accelerometer signals in wide range configuration are shown in figure 4.12. The input dimension d_{input} for the first task for the transformer model was set to one when the wrist dataset was used, as only a single PPG channel is present in this dataset. For the IMU part the input dimension d_{input} was set to ten, to adjust for the additional nine parameters, consisting of the accelerometer data in x-, y- and z-direction for both configurations low and wide, as well as the gyroscope data for the angular velocity around the x-, y-, and z-axis.

4.5. Training

The Transformer has several parameters that can either be chosen as one sees fit or are given by the constraints of the design. While input and output dimensions are fixed according to the used data, the embedding dimension, the number of heads and the number of transformer layers can be set. In general, it can be said, that the higher these

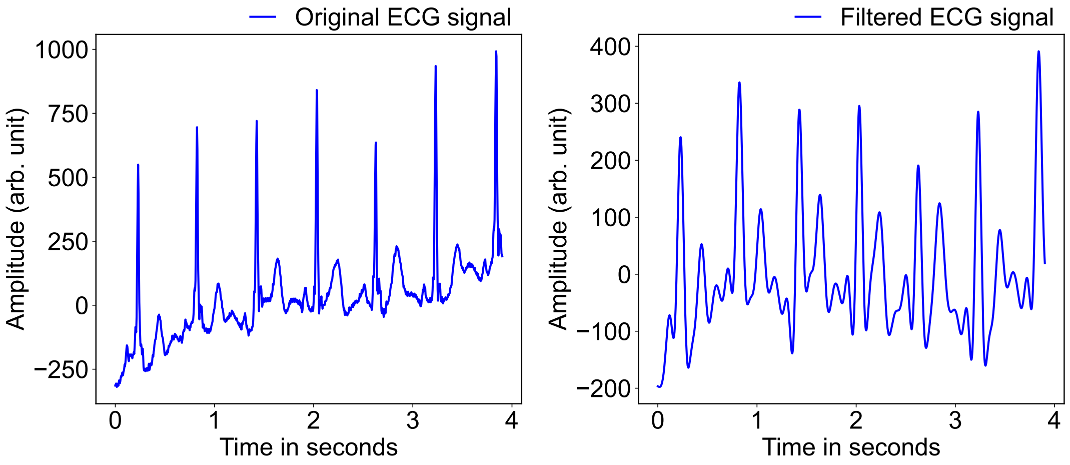


Figure 4.9.: The plot shows an exemplary sequence from a subject during the walking activity of the ECG signal in the original and filtered version from the wrist dataset.

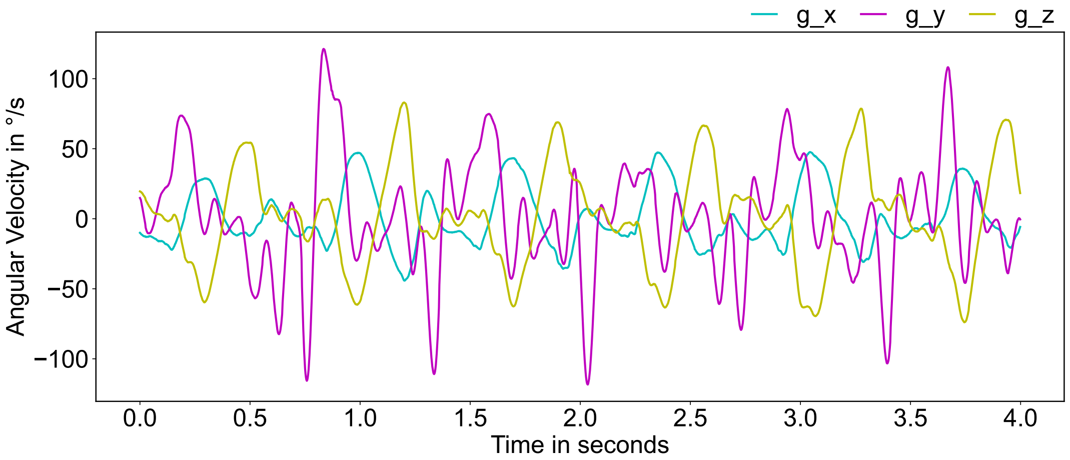


Figure 4.10.: The plot shows an exemplary sequence from a subject during the walking activity of the three gyroscope signals from the wrist dataset.

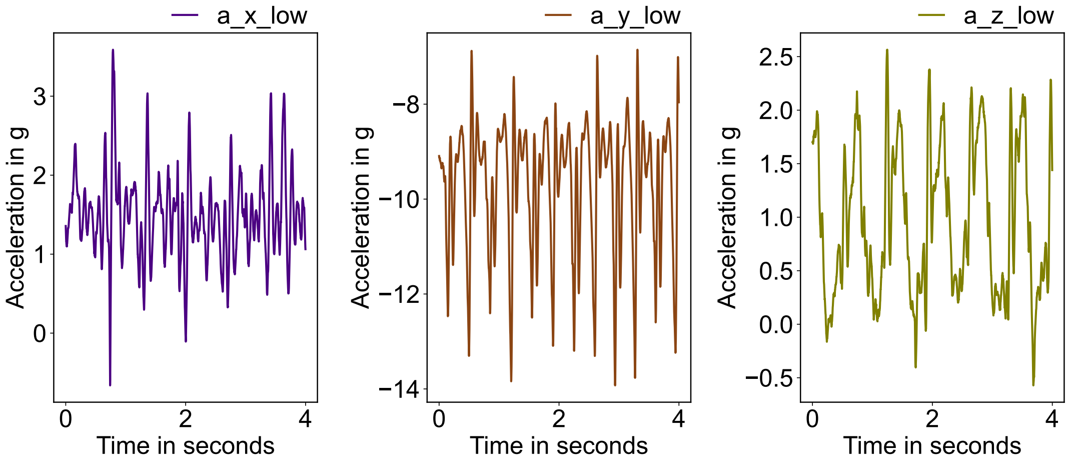


Figure 4.11.: The plot shows an exemplary sequence from a subject during the walking activity of the three accelerometer signals in low noise configuration mode from the wrist dataset.

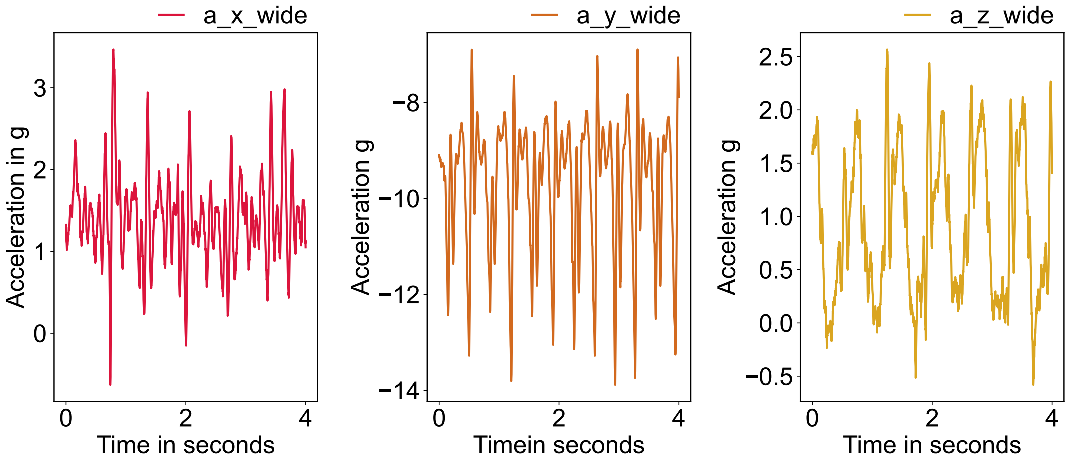


Figure 4.12.: The plot shows an exemplary sequence from a subject during the walking activity of the three accelerometer signals in wide-range configuration mode from the wrist dataset.

parameters are set the more parameters the model has to train and the more complex the model, which allows for better model performance but also increases the risk of overfitting. However, there exists a tradeoff between computing time and memory usage. To make use of parallelism, the model is trained in batches and the higher the batch size, the more memory is required but the computation is faster. The model was trained for 300 epochs to enable sufficient training and early stopping with a patience of 10 epochs was used as well as saving checkpoints of the model. The AdamW (torch.optim.AdamW) optimizer was used and the initial learning rate was set to 0.001 and adjusted using a learning rate scheduler (torch.optim.lr-scheduler.ReduceLROnPlateau). The standard MSE (torch.nn.MSELoss) loss function was used to calculate the loss. The model was trained with floating point 32 precision (torch.float32) and has a total trainable amount of 485'649 parameters which corresponds to a parameter size of 0.61 megabytes (MB).

4.6. Evaluation

As the input and target sequences are both time series data, six common metrics that also apply to time series were used to evaluate the performance of a model where the calculation mainly relies on the open-source library SciPy [20]. The metrics were calculated after the test set was applied to the model with the lowest validation loss to compare the performance of the model on the different datasets and input.

4.6.1. Euclidean Distance

The Euclidean distance (scipy.spatial.distance.euclidean) measures the distance between two points in a two-dimensional plane and can be therefore used to calculate the distance between time series. Equation 4.3 shows the calculation of the Euclidean distance:

$$Euclidian = \sqrt{\sum (x_i - y_i)^2} \tag{4.3}$$

whereas x_i and y_i are the points of the two input vectors.

4.6.2. DTW Distance

In time series analysis DTW marks an algorithm to compute the similarity between two sequences, where these two sequences can also be time-shifted, meaning they are wary in speed. This metric is particularly desired as it can align complex sequences with local shifts in time and noisy environments. However, the computation has a time complexity of $O(NM)$, where N and M mark the lengths of the two input sequences, which makes batching and downsampling a necessity when the DTW algorithm is applied, using an open source library [21].

4.6.3. Pearson Correlation

The Pearson correlation coefficient (scipy.stats.pearsonr) measures the linear correlation between two data sets, where the possible values range from -1 to 1, where 1 means perfect correlation, 0 means no correlation and -1 means perfect inverse correlation. The Pearson correlation coefficient is calculated as seen in equation 4.4:

$$p_{x,y} = \frac{cov(X,Y)}{\sigma_X \sigma_Y} = \frac{\sum_{i=1}^{n}(x_i - \overline{x})(y_i - \overline{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \overline{x_i})^2}\sqrt{\sum_{i=1}^{n}(y_i - \overline{y_i})^2}} \tag{4.4}$$

whereas σ_X and σ_Y are the standard deviations of X and Y respectively, n is the sample size, x_i, y_i are the indexed sample points and $\overline{x}$, $\overline{y}$ are the corresponding means.

4.6.4. Spearman Correlation

The Spearman's rank correlation coefficient (scipy.stats.spearmanr) measures the monotonic relationship between two data sets. The values range from -1 to 1 where 1 means perfect correlation, 0 means no correlation and -1 means perfect inverse correlation. The coefficient can be used to determine how well two variables are similar. The Spearman correlation can be calculated with equation 4.5:

$$r_s = \frac{cov(R[X], R[Y])}{\sigma_R[X]\sigma_R[Y]} = 1 - \frac{6\sum d_i^2}{n(n^2-1)} \tag{4.5}$$

whereas $R[X]$, $R[Y]$ and σ_X, σ_Y are the rank variables and standard deviations of X and Y respectively, $d_i = R[X] - R[Y]$ is the difference of the two ranks of each observation and n is the number of observations.

4.6.5. MSE

The MSE calculates the average error by squaring the difference between the actual and predicted values. The MSE is calculated with the following equation 4.6:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y_i})^2 \tag{4.6}$$

whereas n is the number of data points, y_i is the actual value for the i-th data point and $\hat{y}_i$ is the predicted value for the i-th data point.

4.6.6. MAE

The MAE calculates the average error of the absolute difference between the actual and predicted values. Equation 4.7 shows the MAE error:

$$MAE = \frac{1}{n}\sum_{i=1}^{n} y_i - \hat{y}_i \tag{4.7}$$

whereas n is the number of data points, y_i is the actual value for the i-th data point and $\hat{y}_i$ is the predicted value for the i-th data point.

Chapter 5

Results

5.1. Encoder-Only Transformer with PPG as Input

The first two tasks involved training and evaluating the encoder-only transformer model with only PPG as input, on the two datasets. Since different activities in the datasets affect the body in varying ways, the transformer model was independently trained several times on different subsets of the datasets. Specifically, the model was trained separately for every activity in the dataset, once with only one randomly selected subject and once using data from all subjects performing that activity. For the finger dataset, this resulted in seven training runs, with a single subject performing sitting, walking and running, all subjects performing sitting, walking and running as well as the complete dataset. For the wrist dataset, there were nine training runs, a single subject performing walking, running, low and high resistance bike, as well as all subjects performing walking, running, low and high resistance bike and the complete dataset. This approach allows for a comparative analysis of model performance across different activities and the two datasets.

The standard configuration for all training and evaluation runs consisted of a sequence length of 1000 samples, a sliding window of 100, an embedding dimension d_{model} of 144, the input dimension d_{input} was set to 3 for the finger dataset and set to 1 for the wrist dataset, the output dimension d_{output} was set to 1, 6 transformer heads n_{head}, 4 layers n_{layers}, a batch size of 16, a dropout level of 0.1, a learning rate of 0.001 and 300 epochs n_{epochs}. This configuration was used consistently across all training and evaluation runs.

5.1.1. Results Finger Dataset

Table 5.1 presents the evaluation metrics of the encoder-only Transformer model using PPG signals from the finger dataset for the seven above-mentioned training runs, demonstrating how the transformer performed on different subsets of the finger dataset.

When trained on a single subject performing a single activity, the model achieves its best result for the calm sitting activity, with Pearson and Spearman correlation coefficients of 0.946 and 0.924, respectively, indicating a strong relationship between the predicted and actual ECG values. The MSE has a value of 0.013, the lowest across all training runs, as well as the Euclidean distance with 55.209. In contrast, performance on a single activity is lowest for the most energetic activity running with Pearson and Spearman correlation coefficients dropping to 0.821 and 0.734, and the MSE and Euclidean distance with values of 0.028 and 83.079. Figure 5.1 illustrates accurate ECG predictions when the model was trained on a single subject performing the sitting activity.

When training was expanded to include all subjects performing the same activity, performance decreased significantly across all metrics. For example, in the sitting activity, the Pearson correlation coefficient dropped from 0.946 to 0.029, and the MSE increased from 0.013 to 0.080. This trend is consistent across the walking and running activities, highlighting the difficulty of generalizing across multiple subjects. Figure 5.2 shows this loss of predictive accuracy, for training on all subjects performing the walking activity.

When trained on all subjects and all activities, that is on the complete dataset, model performance was further affected. The Spearman correlation coefficient improved slightly to 0.074 and the Pearson correlation coefficient to 0.224, compared to training on all subjects within a single activity but remained significantly lower than single-subject training. The Euclidean distance increased substantially to 928.238, and the MSE rose to 0.119. Figure 5.3 depicts some correct predictions but with notable outliers for training on all subjects and activities.

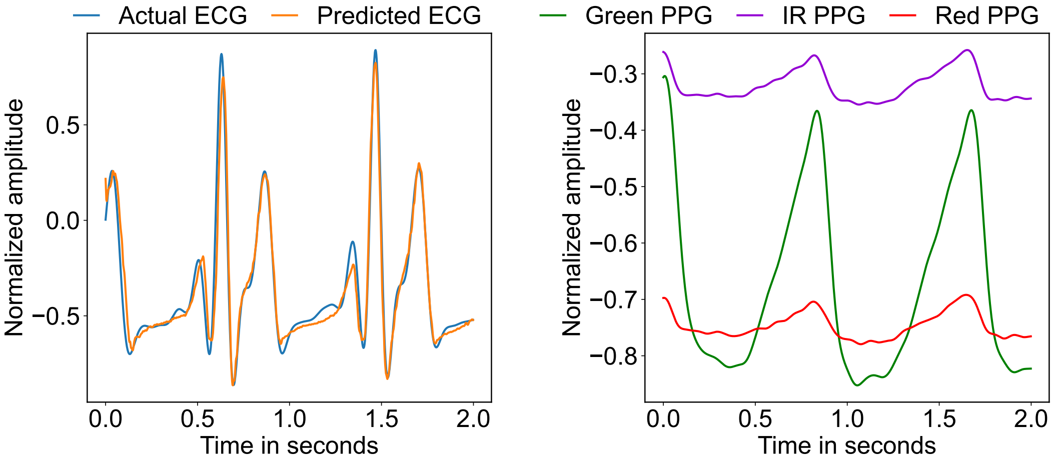


Figure 5.1.: This figure shows the predicted and actual ECG signals derived from the input PPG signals as they can be seen on the right, of a randomly selected sequence from the test set. It illustrates the model's inference performance after being trained on data from a single subject of the finger dataset performing the sitting activity.

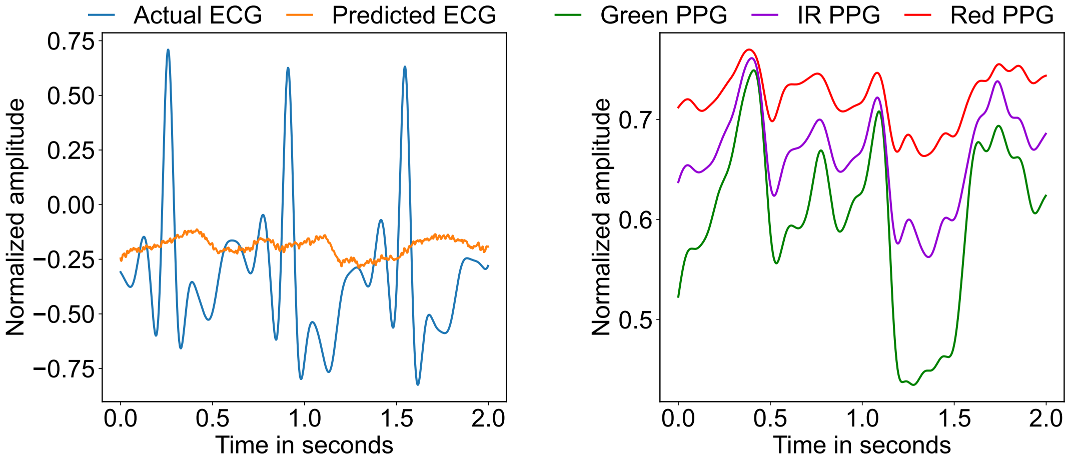


Figure 5.2.: This figure shows the predicted and actual ECG signals derived from the input PPG signals as they can be seen on the right, of a randomly selected sequence from the test set. It illustrates the model's inference performance after being trained on data from all subjects of the finger dataset performing the walking activity.

Model	Subject and Activity	Metrics					
		Euclidean	DTW	Pearson	Spearman	MSE	MAE
Encoder	single subject sit	55.209	0.063	0.946	0.924	**0.013**	0.063
	single subject walk	73.067	0.100	0.845	0.784	0.022	0.100
	single subject run	83.079	0.126	0.821	0.734	0.028	0.126
	all subjects sit	440.724	0.221	0.029	-0.311	0.080	0.222
	all subjects walk	513.410	0.263	0.062	0.043	0.110	0.263
	all subjects run	409.717	0.206	-0.210	-0.290	0.070	0.206
	all subjects and activities	928.238	0.270	0.224	0.074	0.119	0.269

Table 5.1.: The table shows the metrics for the encoder model architecture and the finger dataset using only PPG signals as input, with the shown subject and activity combinations corresponding to the seven training runs described above.

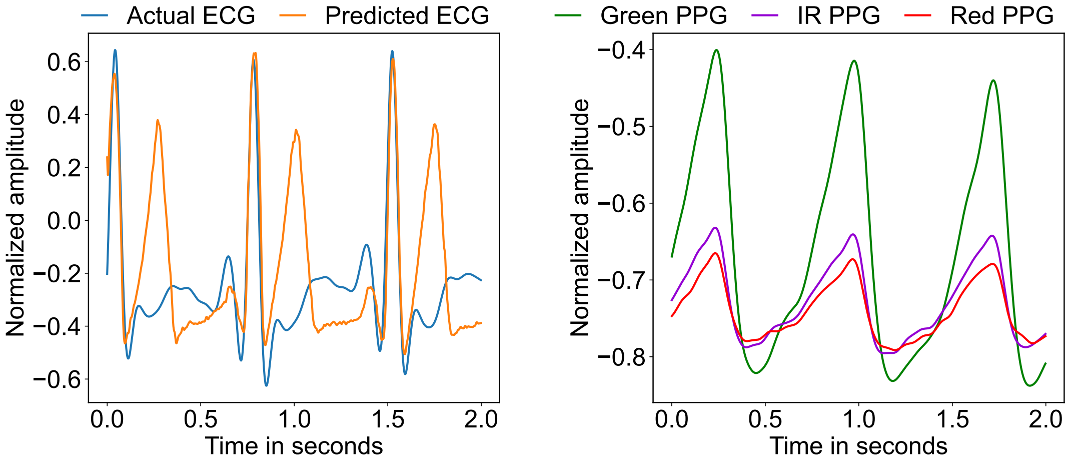


Figure 5.3.: This figure shows the predicted and actual ECG signals derived from the input PPG signals as they can be seen on the right, of a randomly selected sequence from the test set. It illustrates the model's inference performance after being trained on the complete finger dataset, meaning data from all subjects of the finger dataset performing all activities.

5.1.2. Results Wrist Dataset

Table 5.2 presents the evaluation metrics of the encoder-only transformer model using PPG signals from the wrist dataset for the nine above-mentioned training runs, demonstrating how the transformer performed on different subsets of the wrist dataset. The model performed poorly across all setups, as reflected by the high MAE and Pearson and Spearman correlation coefficient values close to zero, indicating a lack of meaningful correlation between the predicted and actual ECG values.

Among the training runs with a single subject, the high resistance bike activity achieved the lowest Euclidean distance of 34.209 and MSE value of 0.016 as well as the highest, though still weak, Pearson and Spearman correlation coefficient values of 0.071 and 0.073, respectively. However, figure 5.4 shows that even in this case, the alignment between the predicted and actual ECG sequences remains marginal.

When trained on all subjects performing a single activity, performance stayed mediocre. Notably, training on the activity low resistance bike resulted in the lowest MSE of 0.035 among these setups. The Pearson and Spearman correlation coefficient values remained close to zero across all activities, indicating that the model struggles to generalize across multiple subjects in the wrist dataset. Figure 5.5 presents the mediocre prediction for the low-resistance bike activity when trained on all subjects performing that activity.

Interestingly, when trained on all subjects and all activities, the model yielded a Pearson correlation coefficient of -0.752 and a Spearman correlation coefficient of -0.704, suggesting a strong but inverse relationship between the predicted and actual ECG values. However, as illustrated in figure 5.6, the predicted ECG sequence deviates significantly from the actual target sequence, further confirming the model's poor predictive ability on this dataset.

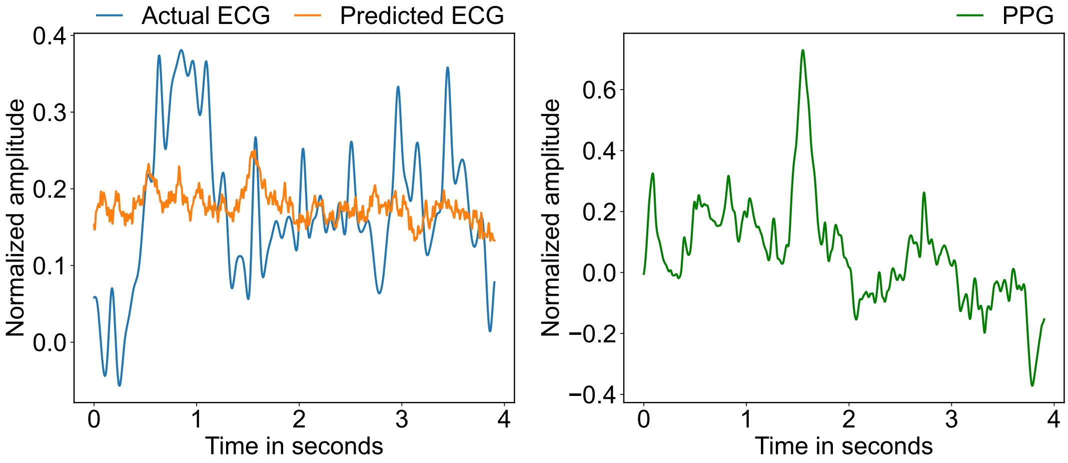


Figure 5.4.: This figure shows the predicted and actual ECG signals derived from the input PPG signal as it can be seen on the right, of a randomly selected sequence from the test set. It illustrates the model's inference performance after being trained on data from a single subject of the wrist dataset performing the high resistance bike activity.

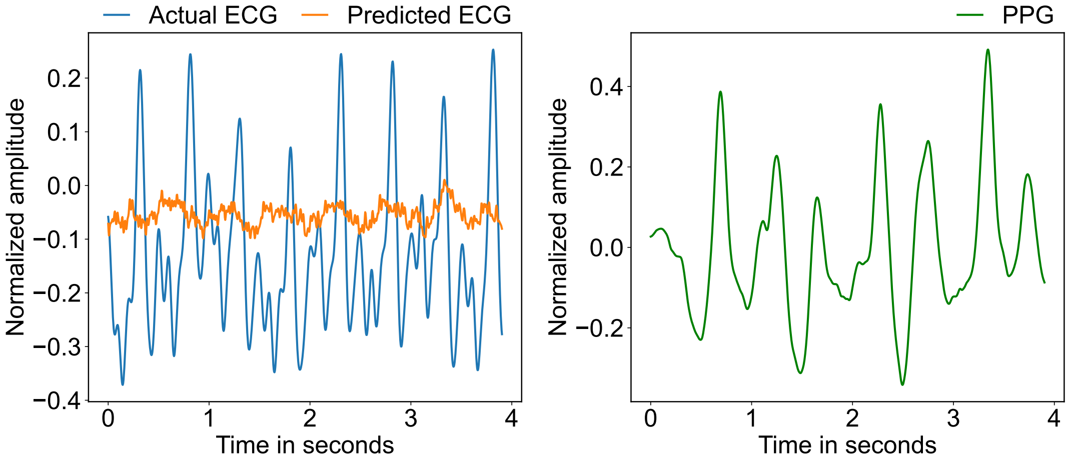


Figure 5.5.: This figure shows the predicted and actual ECG signals derived from the input PPG signal as it can be seen on the right, of a randomly selected sequence from the test set. It illustrates the model's inference performance after being trained on data from all subjects of the wrist dataset performing the low resistance bike activity.

Model	Subject and Activity	Metrics					
		Euclidean	DTW	Pearson	Spearman	MSE	MAE
Encoder	single subject walk	49.216	0.125	0.003	0.008	0.033	0.125
	single subject run	55.540	0.121	-0.007	-0.016	0.040	0.121
	single subject low resistance bike	58.628	0.147	0.018	0.022	0.046	0.147
	single subject high resistance bike	34.209	0.095	0.071	0.073	**0.016**	0.095
	all subjects walk	261.697	0.121	-0.007	-0.005	0.067	0.206
	all subjects run	323.101	0.374	0.004	0.019	0.144	0.375
	all subjects low resistance bike	156.463	0.155	-0.014	-0.012	0.035	0.155
	all subjects high resistance bike	187.516	0.195	-0.089	-0.100	0.050	0.195
	all subjects and activities	378.834	0.243	-0.752	-0.704	0.082	0.243

Table 5.2.: The table shows the metrics for the encoder model architecture and the wrist dataset using only PPG signals as input, with the shown subject and activity combinations, corresponding to the nine training runs described above.

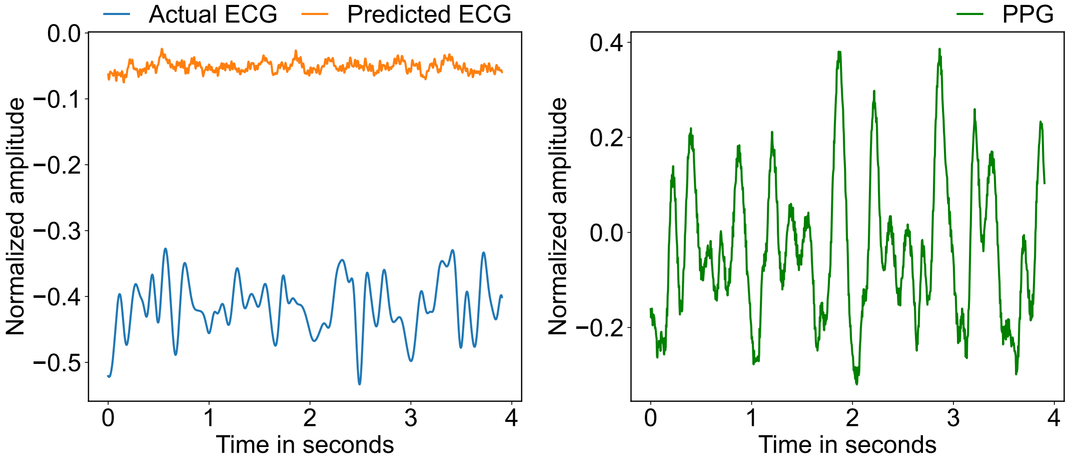


Figure 5.6.: This figure shows the predicted and actual ECG signals derived from the input PPG signal as it can be seen on the right, of a randomly selected sequence from the test set. It illustrates the model's inference performance after being trained on the complete wrist dataset, meaning data from all subjects of the wrist dataset performing all activities.

5.2. Encoder-Only Transformer with PPG and IMU as Inputs

In this step, the IMU data, which includes gyroscope and accelerometer signals, is incorporated alongside the PPG signals as input to the encoder-only transformer model and both datasets are trained and evaluated again. As before, the transformer model was independently trained several times on different subsets of the datasets. The model was trained separately for every activity in the dataset, once with only one randomly selected subject and once using data from all subjects performing that activity. This resulted again in seven training runs for the finger dataset and nine training runs for the wrist dataset.

The standard configuration for all training and evaluation runs consisted of a sequence length of 1000 samples, a sliding window of 100, an embedding dimension d_{model} of 144, the input dimension d_{input} was set to 9 for the finger dataset and set to 10 for the wrist dataset, the output dimension d_{output} was set to 1, 6 transformer heads n_{head}, 4 layers n_{layers}, a batch size of 16, a dropout level of 0.1, a learning rate of 0.001 and 300 epochs n_{epochs}. This configuration was used consistently across all training and evaluation runs.

5.2.1. Results Finger Dataset

Table 5.3 presents the evaluation metrics of the encoder-only transformer model using PPG and IMU signals from the finger dataset for the seven above-mentioned training runs, demonstrating how the transformer performed on different subsets of the finger dataset.

For single subject activities, the model performs well for the activities of sitting and walking, with high Pearson correlation coefficients of 0.870 and 0.825 and low MSE values of 0.030 and 0.025, respectively. Figure 5.7 shows the corresponding prediction of the walking activity. However, the running activity presents challenges, reflected by a much higher MSE value of 0.085 and lower Pearson and Spearman correlation coefficients of 0.120 and 0.139.

When including all subjects for a single activity, the sitting activity shows improved performance, with higher Pearson and Spearman correlation coefficients of 0.372 and 0.318, suggesting better tracking of the sequence, albeit a higher MSE value of 0.114. The walking activity still shows a low Pearson and Spearman correlation coefficient of 0.108 and 0.116, with only a slight improvement of the MSE value with 0.091. The predicted sequence is quite close to the actual one, as can be seen in figure 5.8.

For all subjects and activities, that is the complete dataset, the model's performance declines, as evidenced by the increased Euclidean distance of 1495.230 and lower correlation coefficients of 0.136 and 0.144 for Pearson and Spearman, although it is less influenced by outliers. These observations can be confirmed by looking at figure 5.9, showing the predicted ECG sequence.

Model	**Subject and Activity**	**Metrics**					
		Euclidean	**DTW**	**Pearson**	**Spearman**	**MSE**	**MAE**
Encoder	single subject sit	84.829	0.106	0.870	0.802	0.030	0.106
	single subject walk	77.388	0.101	0.825	0.790	0.025	0.101
	single subject run	143.878	0.214	0.120	0.139	0.085	0.214
	all subjects sit	912.297	0.261	0.372	0.318	0.114	0.261
	all subjects walk	813.034	0.230	0.108	0.116	0.091	0.230
	all subjects run	758.055	0.215	-0.017	-0.024	0.079	0.215
	all subjects and activities	1495.230	0.245	0.136	0.144	0.106	0.246

Table 5.3.: The table shows the metrics for the encoder model architecture and the finger dataset using PPG and IMU signals as input, with the shown subject and activity combinations corresponding to the seven training runs described above.

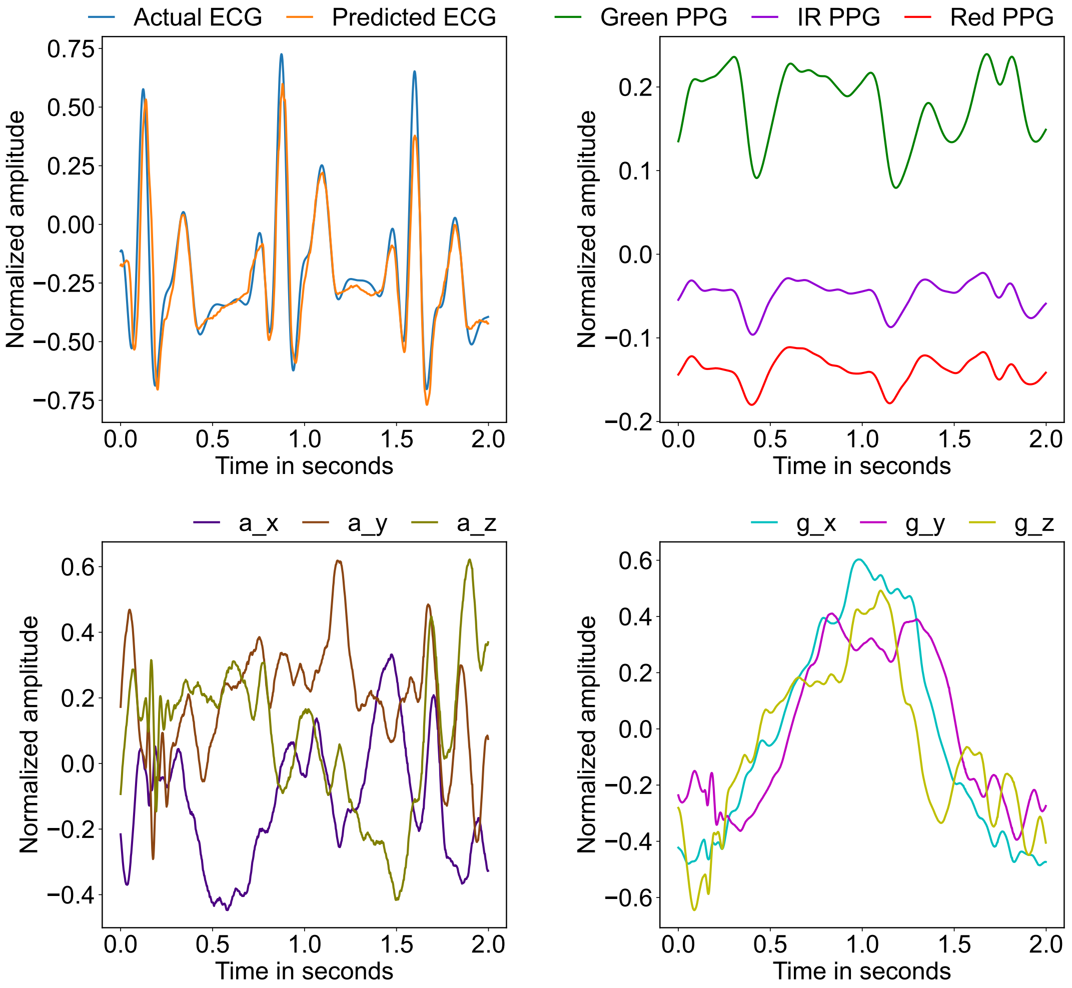


Figure 5.7.: This figure shows the predicted and actual ECG signals derived from the shown input PPG and IMU signals of a randomly selected sequence from the test set. It illustrates the model's inference performance after being trained on data from a single subject of the finger dataset performing the walking activity.

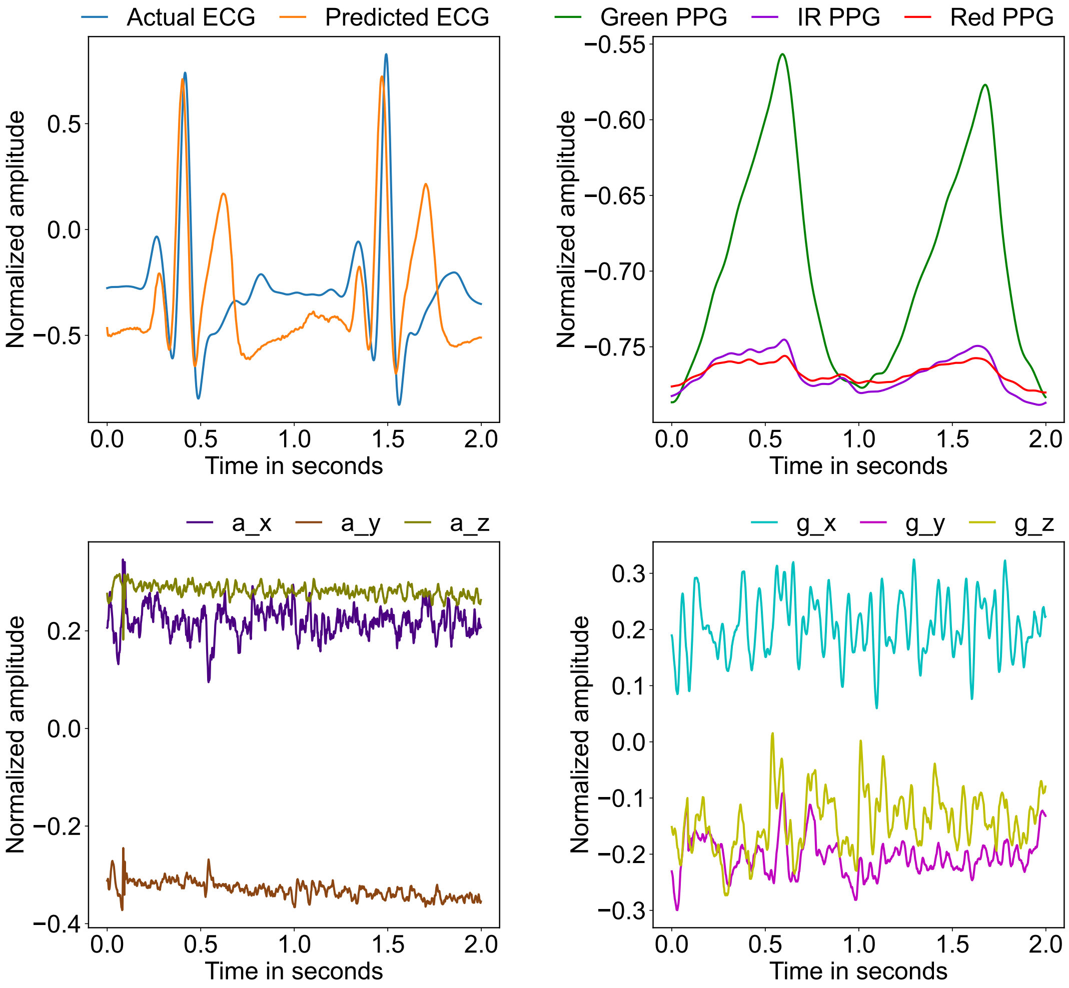


Figure 5.8.: This figure shows the predicted and actual ECG signals derived from the shown input PPG and IMU signals of a randomly selected sequence from the test set. It illustrates the model's inference performance after being trained on data from all subjects of the finger dataset performing the sitting activity.

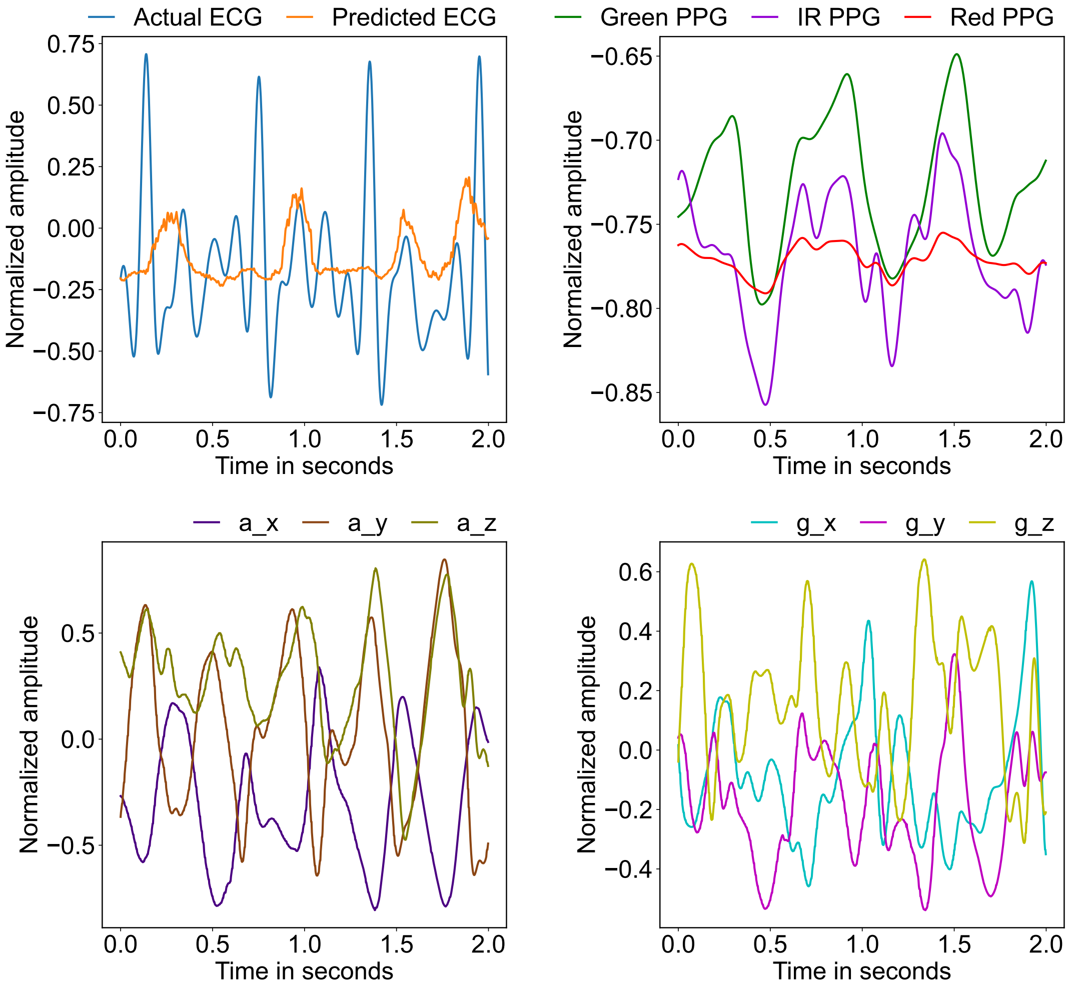


Figure 5.9.: This figure shows the predicted and actual ECG signals derived from the shown input PPG and IMU signals of a randomly selected sequence from the test set. It illustrates the model's inference performance after being trained on the complete finger dataset, meaning data from all subjects of the finger dataset performing all activities.

5.2.2. Results Wrist Dataset

Table 5.2 presents the evaluation metrics of the encoder-only transformer model using PPG and IMU signals from the finger dataset for the nine above-mentioned training runs, demonstrating how the transformer performed on different subsets of the wrist dataset. The model performed poorly across all setups, as reflected by the high MAE and Pearson and Spearman correlation coefficient values close to zero, indicating a lack of meaningful correlation between the predicted and actual ECG values.

For the single subject activities, the model struggles to track the actual ECG signal. For instance, the walking activity has a Pearson correlation coefficient of 0.114, a Spearman correlation coefficient of 0.051, and a substantial MSE of 0.033, as confirmed by the predicted sequence in figure 5.10. As without IMU signals as input the best performing single subject activity was the high resistance bike activity, with a Pearson correlation coefficient of 0.337, Spearman correlation coefficient of 0.319, and an MSE of 0.015.

The training runs with all subjects for a single activity show intermediate performance, with many results between those of the single subject runs and the entire dataset run. Figure 5.11 confirms this, showing a slight offset of the predicted sequence concerning the actual ECG sequence, and getting a Pearson correlation coefficient of -0.005, Spearman correlation coefficient of -0.018, and an MSE of 0.108.

When all subjects and activities are considered, the model performs better, with 0.764 and 0.698 as Pearson and Spearman correlation coefficients respectively, and an MSE of 0.057, but shows some large offset in the predicted sequence compared to the actual ECG sequence, as it can be seen in figure 5.12.

Model	**Subject and activity**	**Metrics**					
		Euclidean	**DTW**	**Pearson**	**Spearman**	**MSE**	**MAE**
	single subject walk	49.342	0.130	0.114	0.051	0.033	0.131
	single subject run	55.152	0.120	0.112	0.115	0.039	0.120
	single subject low resistance bike	58.401	0.147	0.104	0.082	0.045	0.147
	single subject high resistance bike	32.356	0.091	0.337	0.319	0.015	0.091
Encoder	all subjects walk	240.262	0.186	-0.012	-0.016	0.057	0.186
	all subjects run	298.565	0.341	-0.009	-0.000	0.123	0.341
	all subjects low resistance bike	146.458	0.139	-0.006	-0.002	0.031	0.139
	all subjects high resistance bike	277.045	0.306	-0.005	-0.018	0.108	0.306
	all subjects and activities	314.903	0.208	**0.764**	**0.698**	0.057	0.208

Table 5.4.: The table shows the metrics for the encoder model architecture and the wrist dataset using PPG and IMU signals as input, with the shown subject and activity combinations, corresponding to the nine training runs described above.

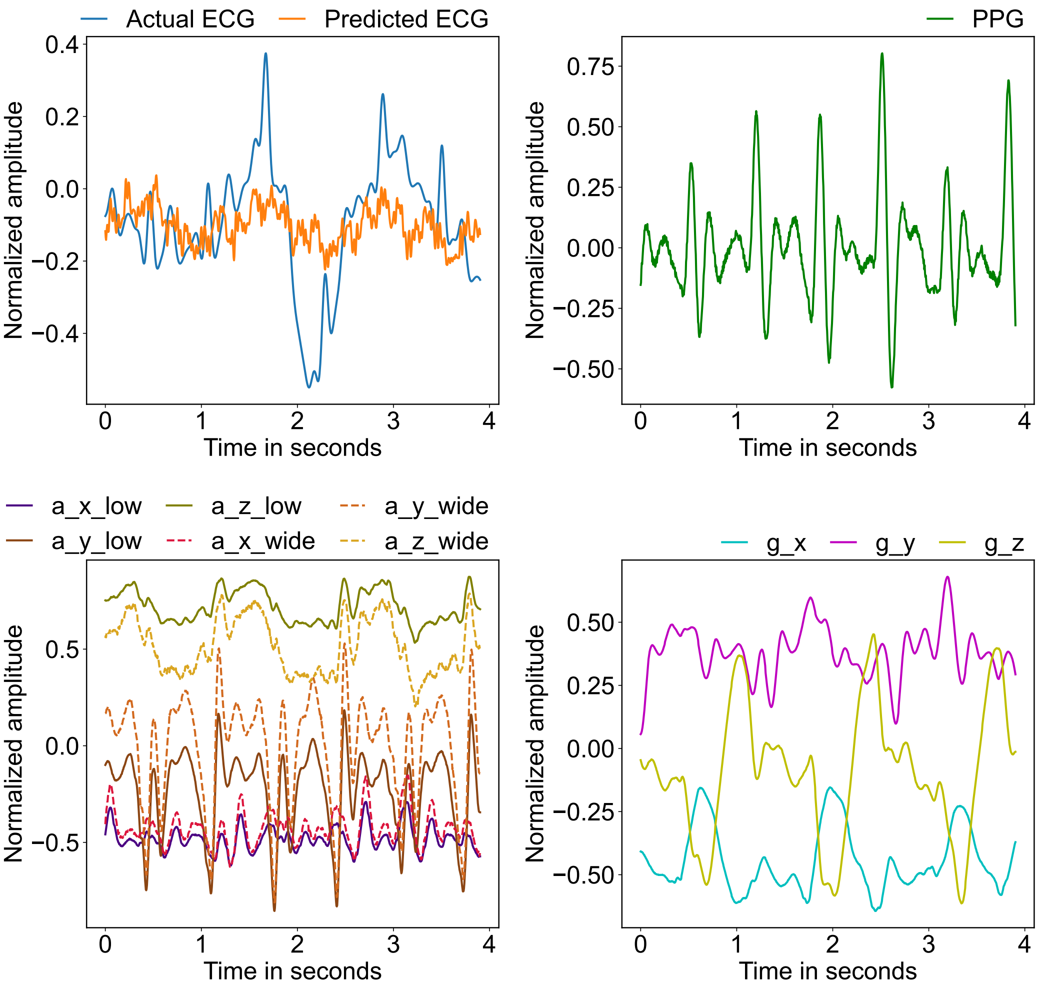


Figure 5.10.: This figure shows the predicted and actual ECG signals derived from the shown input PPG and IMU signals of a randomly selected sequence from the test set. It illustrates the model's inference performance after being trained on data from a single subject of the wrist dataset performing the walking activity.

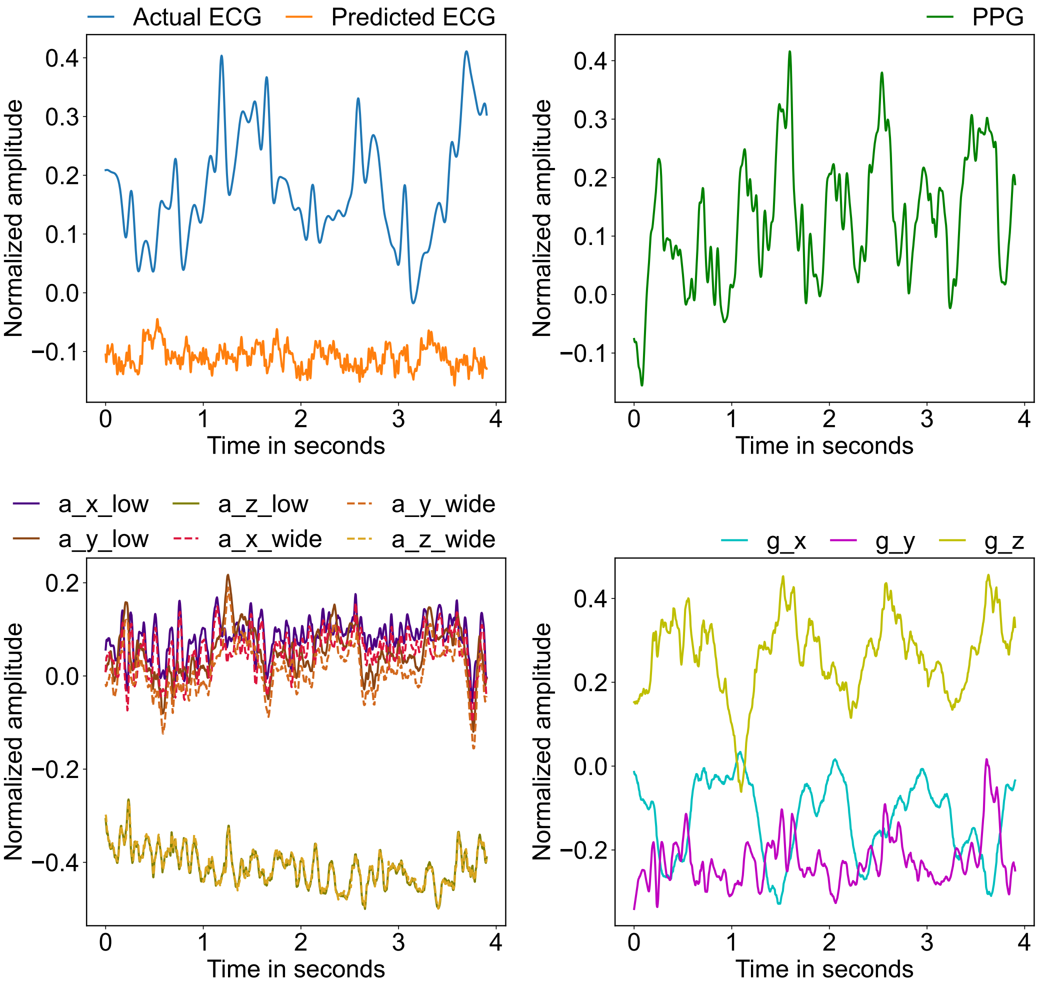


Figure 5.11.: This figure shows the predicted and actual ECG signals derived from the shown input PPG and IMU signals of a randomly selected sequence from the test set. It illustrates the model's inference performance after being trained on data from all subjects of the wrist dataset performing the high resistance bike activity.

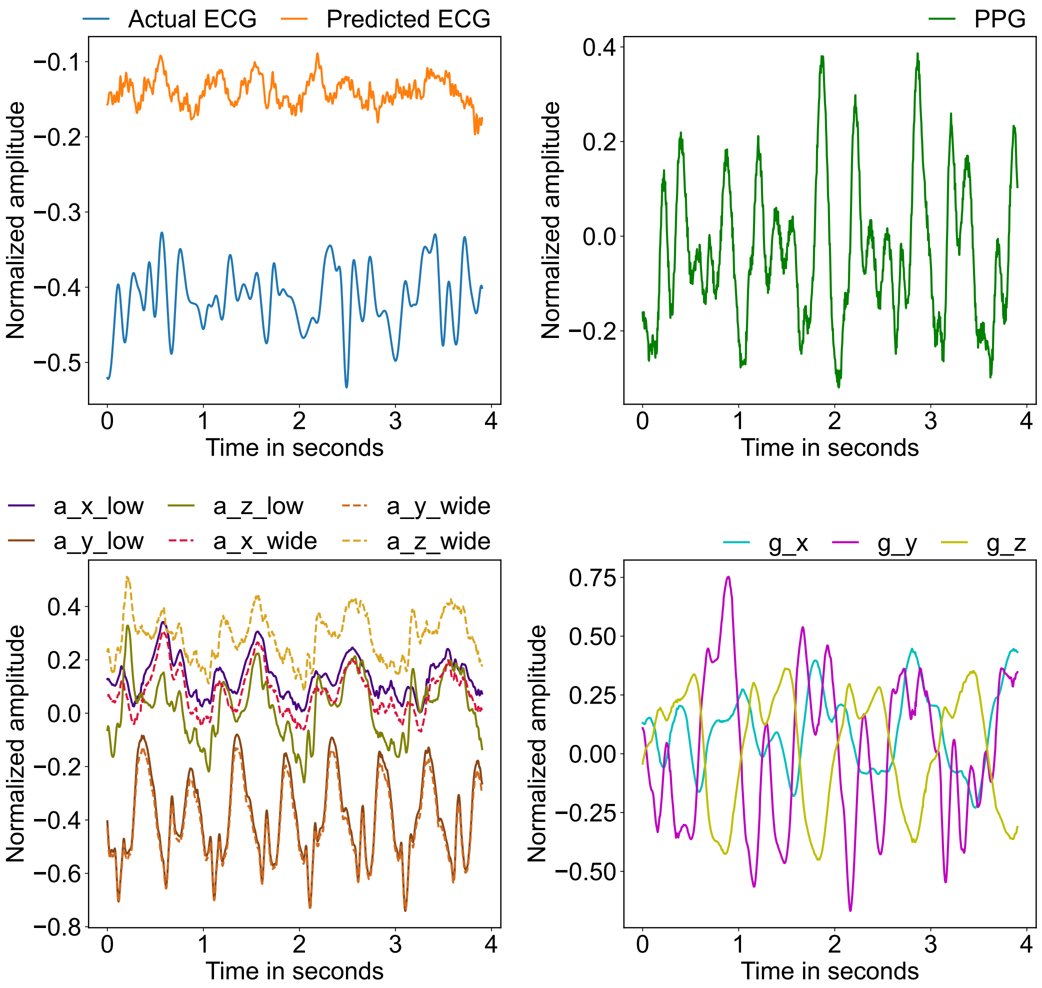


Figure 5.12.: This figure shows the predicted and actual ECG signals derived from the shown input PPG and IMU signals of a randomly selected sequence from the test set. It illustrates the model's inference performance after being trained on the complete wrist dataset, meaning data from all subjects of the wrist dataset performing all activities.

Chapter 6

Discussion

The training, validating and testing of the encoder-only transformer model is working so that all desired metrics described in Chapter 4 can be calculated and predictions can be made, based on the trained model architecture. One can select all hyperparameters that are used to train the model, including the desired input consisting either of only PPG signals or PPG and IMU signals, to train the model and get the specific predictions. The following sections describe the obtained results of the three tasks, compare them to related work, draw conclusions and point out further investigation possibilities.

6.1. Encoder-Only Transformer with PPG as Input

6.1.1. Finger Dataset

The model works decently on one single subject with a single activity, where we achieve MSE values of 0.013, 0.022 and 0.028 compared to a value of 0.023 for their subject-specific model and the PPG-DaLiA dataset achieved by Pinto et al. (2024) [15]. For the calm activity of sitting we achieve the best performance, the most energetic activity running has the worst performance, and the activity of walking is somewhere in between. This aligns with the increase in MA for more energetic activities. When multiple subjects are considered the performance of the model is worse, the MAE ranges from 0.206 to 0.263, which was also observed by Pinto et al. (2024), albeit less drastically with a MAE value of 0.127 for their group model [15]. Here is no obvious difference between the performance of the three activities visible, which leads to the conclusion that based on the degradation in performance with multiple subjects, the model is not capable of generalizing the transformation from PPG to ECG and is overfitting on single subjects. The performance is worst on the entire dataset, but the Pearson and Spearman coefficients are somewhere in between the two cases described above. This aligns with the two cases above, as there are now all subjects but also the calmer activities present, where the model in some cases can achieve somehow meaningful predictions. As the model has partial success careful change of the architecture, a thorough investigation of the hyperparameters or other techniques, such as data augmentation, could be used to enhance the performance of the model.

6.1.2. Wrist Dataset

The model has the lowest MSE values on a single subject with a single activity, where we achieve MSE values of 0.016, 0.033, 0.040 and 0.046. Except for the low resistance bike activity, these values are considerably higher than the value of 0.023 obtained by Pinto et al. (2024), for their subject-specific model and the PPG-DaLiA dataset [15]. Similar to the finger dataset the calmer activities achieve better performance, the most energetic activities running and high resistance bike have the worst performance. When multiple subjects are considered the performance of the model is worse, the MAE ranges from 0.155 to 0.375, which is overall higher than Pinto et al. (2024) achieved, with a MAE value of 0.127 for their group model [15]. The model fails to extract the context of its PPG input and is not able to capture the character of the target ECG signal. This is also true for the entire dataset, where the model again struggles and has strong negative Pearson and Spearman coefficients. The model is not able to replicate the performance of the finger dataset on the wrist dataset. This task included testing the model's ability to a more energetic dataset and it was expected that the model would probably perform worse, albeit less than it did. Reasons could be that the wrist dataset has roughly ten times fewer samples and only one PPG channel, leading in total to thirty times less input data. On the other hand, the wrist dataset contains four activities, which means less data to train on per activity, and overall more energetic activities, as the sitting activity that was present in the finger dataset is missing. Data augmentation is a tool that could be investigated further together with an entirely new dataset, that has comparable samples and three PPG signals, with similar activities.

6.2. Encoder-Only Transformer with PPG and IMU as Inputs

6.2.1. Finger Dataset

The model works decently on one single subject with a single activity, where we achieve MSE values of 0.025, 0.030 and 0.085 compared to MSE values of 0.013, 0.022 and 0.028 before. The activities of sitting and walking did not change too much, but the running activity is now considerably worse. A reason could be that the already with MA highly influenced PPG and then adding six more IMU signals confused the model, where the model is no longer confident in predicting the ECG target sequence. When multiple subjects are considered the performance of the model on the sitting activity stands out, significantly improving as we can see with a DTW value of 0.261, which is close to the single subject on running value of 0.214. The Pearson and Spearman correlation coefficient values of 0.029 and -0.311 are now dwarfed by values of 0.372 and 0.318. The walking and running activities show similar but less noticeable improvements. Interesting is the performance on the entire dataset, whereas the metrics are slightly better except for the Euclidean value which is now higher at 1495.230 compared to the previous 928.238, as well as the Pearson correlation coefficient, with 0.136 compared to the previously achieved 0.224. This leads to the conclusion, that the predictions are slightly closer to the actual target value, but fail to capture the full character of it. Overall, using the IMU data as input helps the model to understand the underlying motion better to some extent, however, it is not the final solution for reliable predictions. Further investigation could focus on using only gyroscope or accelerometer data, as well as different preprocessing steps to get the best context for the model to use.

6.2.2. Wrist Dataset

The model achieves now on a single subject with a single activity MSE values of 0.015, 0.033, 0.039 and 0.045, where we achieved MSE values of 0.016, 0.033, 0.040 and 0.046 before with only PPG signals as input. No activity profited specifically from adding the IMU data, as again the calmer activities achieved better performance and the more energetic activities running and high resistance bike had the worst performance. When multiple subjects are considered the performance of the model is once more worse, the MAE ranges from 0.139 to 0.341 to a previous range of 0.155 to 0.375. Even with the IMU data the model fails to extract the context of its PPG and IMU input and is not able to capture the character of the target ECG signal. However, this does not occur for the entire dataset, where the model has now highly positive Pearson and Spearman correlation coefficients instead of strong negative ones. The model seems to capture at least some information out of the input but fails to reproduce essentially the amplitude, as the MSE, MAE and Euclidean values are still high. This shows that using the IMU as input has some limitations, meaning when the model is not able to predict something approximately close to the actual target sequence, adding IMU data to the PPG signal can not fix the underlying problem. Further investigation could focus on getting more data or a more capable model, as already described before in the results section for the wrist dataset using only PPG as input. Measures such as using a dataset with factors of more samples, extensive data augmentation and then using the IMU data as an additional tool, could be promising further steps.

Chapter 7

Conclusion and Future Work

The findings of this project on ECG prediction using only PPG as input are in alignment with previous work. Models using a single subject on a single activity or multiple subjects on a single activity show similar performance degrading when more subjects or more complex activities with extended MA are used to train and evaluate the mode. However, only a single subject on a single activity from the finger dataset such as sitting, walking or running, can achieve a similar state-of-the-art performance as it was achieved by Pinto et. al (2024) [15]. In addition, datasets with more energetic or complex activities are more exposed to MA which makes it generally harder for the model to reduce them and retrieve the actual ECG signal. The model performed badly on the wrist dataset, with more energetic activities and way fewer samples. The project contributed by exploring the feasibility of adding IMU data to capture the underlying motion and showed that overall IMU contributes a limited, but still significant factor to increase the model performance in every subject and activity combination. However, this is not a method to leap a model to new levels way beyond its initial performance with only PPG signals as input, as it was seen on the wrist dataset, but rather a feature that should be explored if IMU data is present for the task to solve. There are many possible future contributions such as using a more sophisticated encoder model architecture, or even a different transformer architecture, to achieve reliable ECG predictions. As the wrist performance indicates, the amount of data samples seems to be of great importance, hence using larger datasets with similar activities and parameters as well as data augmentation methods seems a good way to start further investigations. Future work could also include optimising the model in terms of model size and performance, with an additional focus on the challenges when deploying it on a microcontroller.

Appendix A

Task Description

ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

CENTER FOR
PROJECT-BASED
LEARNING

Task Description for a Semester Project on

PPG Signal Enhancement via Sensor Fusion: IMU-Driven Artifact Removal Using Transformers Architecture

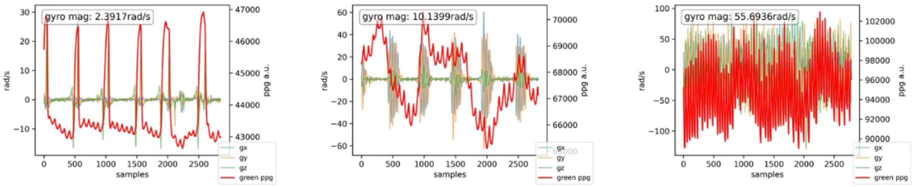
at the Department of Information Technology and
Electrical Engineering

for

Andri Furrer
andfurrer@student.ethz.ch

Advisors:	Dr. Tommaso Polonelli, tommaso.polonelli@pbl.ee.ethz.ch (external from Politecnico di Milano) Alice Scandelli alice.scandelli@polimi.it, Aurelio Teliti aurelio.teliti@polimi.it
Professor:	Prof. Dr. Luca Benini, lbenini@iis.ee.ethz.ch
Handout Date:	14.10.2024
Due Date:	20.01.2025

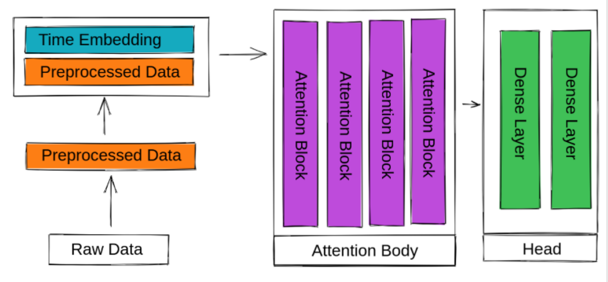
1 Project Goals



(a) Lip Puller.

(b) Nod.

(c) Running.



Photoplethysmography (PPG) is a widely used, non-invasive optical technique for monitoring vital signs by detecting blood volume changes in the body's peripheral circulation. While PPG signals are commonly used in wearable health devices like smartwatches, fitness trackers, and earables, they are highly susceptible to noise and motion artifacts (MA). Movements during daily activities, particularly those involving the wrist, arm, or head, introduce significant artifacts that can degrade signal quality, affecting the accuracy of heart rate, blood pressure, and other physiological measurements. Traditional filtering methods often struggle to remove these artifacts without compromising critical physiological data, underscoring the need for more advanced signal processing approaches to ensure reliable monitoring.

Recent research has focused on enhancing PPG signal quality through motion artifact removal and detection techniques. Nabavi & Bhadra (2020) proposed a fusion method combining PPG and accelerometer data to filter out motion-related frequencies, achieving over 95% accuracy in estimating cardiac parameters. Luke et al. (2018) developed three algorithms for motion artifact removal, with the Butterworth-Wavelet Transform Algorithm showing promising results. Le (2023) introduced the GRN-Transformer, integrating Gated Residual Networks into the Transformer architecture, outperforming conventional methods with 98% accuracy in artifact detection. Building on this, Le (2024) explored self-supervised learning techniques to improve Transformer models' performance on limited PPG datasets. Contrastive learning emerged as the most effective approach, with a novel contrastive loss function facilitating smoother training and better convergence. These advancements demonstrate significant progress in PPG signal enhancement, potentially improving clinical decision support systems and patient outcomes.

This project aims to develop a Transformer-based architecture for robust artifact removal and denoising of PPG signals, using IMU (Inertial Measurement Unit), matching the accuracy of current state-of-the-art transformer models. This project will utilize a dataset of PPG signals affected by motion artifacts, synchronized with IMU data, to train a model based on transformer architecture. Once this is achieved, the focus will shift to optimizing the model for deployment on microcontrollers, with particular attention to power consumption, memory footprint, and latency.

2 Tasks

The project will be split into three phases, as described below:

Phase 1 (Week 1-2)

1. Investigate the state-of-the-art of PPG artifacts removal and detection with AI/ML techniques.
2. Get in touch with the current tools for design and implement deep learning models
3. Study and get used to the concept of Transformers Architecture

Phase 2 (weeks 3-11)

1. Implement a working Transformers-based model working on a timeseries signal
2. Apply the developed model to the provided PPG+IMU datasets
3. Test, evaluation the main metrics of the developed model.
4. (optional) Quantize the model and compare with the non-quantize model
5. (optional) Deploy on the model on a microcontroller

Phase 3 (week 12-14)

1. Final presentation
2. Final report

Milestones

The following milestones need to be reached during the thesis:

- Implement a working Transformer-based Model based on PPG+IMU data for the reconstruction of the PPG signal corrupted by the motion artifacts
- Final report and presentation.

3 Project Organization

During the thesis, students will gain experience in the independent solution of a technical-scientific problem by applying the acquired specialist and social skills. The grade is based on the following: student effort; thoroughness and learning curve; achieving qualitative and quantitative results with a scientific approach; supporting practical findings with theoretical background and literature investigations; final presentation and report; documentation and reproducibility. All theses include an oral presentation, a written report and are graded. The report and presentation need to have publication grade quality to achieve a good grade. Students are graded based on the official ITET grading form[1]. For students of IIS (Prof. Benini) a special grading scheme exists, please contact your supervisor for details there. Before starting, the project must be registered in myStudies and all required documents need to be handed in for archiving by PBL.

3.1 Laboratory Rules

The students agree to follow the lab rules set by PBL staff, for detail please contact us. The most important points are:

- All ETH safety regulations need to be followed[2], in addition to ones given by PBL staff
- No device in the lab is used without introduction by your supervisor or PBL staff
- No device leaves the lab without being officially borrowed, this is done by PBL staff and needs your Legi.
- Any damage to devices or tools needs to be reported immediately to PBL staff.

[1] https://ethz.ch/content/dam/ethz/special-interest/itet/department/Studies/Forms/Grading%20Form.xlsx

[2] https://ethz.ch/staffnet/en/service/safety-security-health-environment/sicherheit-in-laboren-und-werkstaetten/laborsicherheit.html

- The Lab-desk is clean and free for others after you finished your task, or when you take longer breaks. All tools are correctly sorted into their drawers/cupboards when you leave.

3.2 Weekly Report

There will be a weekly report/meeting held between the student and the assistants. The exact time and location of these meetings will be determined within the first week of the project in order to fit the students and the assistants schedule. These meetings will be used to evaluate the status and document the progress of the project (required to be done by the student). Beside these regular meetings, additional meetings can be organized to address urgent issues as well. The weekly report, along with all other relevant documents (source code, datasheets, papers, etc), should be uploaded to a clouding service, such as Polybox and shared with the assistants.

3.3 Project Plan

Within the first month of the project, you will be asked to prepare a project plan. This plan should identify the tasks to be performed during the project and sets deadlines for those tasks. The prepared plan will be a topic of discussion of the first week's meeting between you and your advisers. Note that the project plan should be updated constantly depending on the project's status.

3.4 Final Report and paper

PDF copies of the final report written in English are to be turned in. Basic references will be provided by the supervisors by mail and at the meetings during the whole project, but the students are expected to add a considerable amount of their own literature research to the project ("state of the art").

3.5 Final Presentation

There will be a presentation (15 min presentation and 5 min Q&A for BT/ST and 20 min presentation and 10 min Q&A for MT) at the end of this project in order to present your results to a wider audience. The exact date will be determined towards the end of the work.

References:

Will be provided by the supervisors by mail and at the meetings during the whole project.

Place and Date Zurich, 11.10.24 Signature Student

Appendix B

Declaration of Originality

ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Declaration of originality

The signed declaration of originality is a component of every written paper or thesis authored during the course of studies. In consultation with the supervisor, one of the following three options must be selected:

○ I confirm that I authored the work in question independently and in my own words, i.e. that no one helped me to author it. Suggestions from the supervisor regarding language and content are excepted. I used no generative artificial intelligence technologies[1].

○ I confirm that I authored the work in question independently and in my own words, i.e. that no one helped me to author it. Suggestions from the supervisor regarding language and content are excepted. I used and cited generative artificial intelligence technologies[2].

● I confirm that I authored the work in question independently and in my own words, i.e. that no one helped me to author it. Suggestions from the supervisor regarding language and content are excepted. I used generative artificial intelligence technologies[3]. In consultation with the supervisor, I did not cite them.

Title of paper or thesis:

PPG Signal Enhancement via Sensor Fusion: IMU-Driven Artifact Removal Using Transformers Architecture

Authored by:
If the work was compiled in a group, the names of all authors are required.

Last name(s):	**First name(s):**
Furrer	Andri

With my signature I confirm the following:

- I have adhered to the rules set out in the Citation Guide.
- I have documented all methods, data and processes truthfully and fully.
- I have mentioned all persons who were significant facilitators of the work.

I am aware that the work may be screened electronically for originality.

Place, date	**Signature(s)**
Zurich, 19, January, 2025	A Furrer

If the work was compiled in a group, the names of all authors are required. Through their signatures they vouch jointly for the entire content of the written work.

[1] E.g. ChatGPT, DALL E 2, Google Bard
[2] E.g. ChatGPT, DALL E 2, Google Bard
[3] E.g. ChatGPT, DALL E 2, Google Bard

Bibliography

[1] A. Luke, S. Shaji, and K. A. Unnikrishna Menon, “Motion artifact removal and feature extraction from ppg signals using efficient signal processing algorithms,” in *2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, 2018, pp. 624–630.

[2] T.-D. Le, “Transformer meets gated residual networks to enhance photoplethysmogram artifact detection informed by mutual information neural estimation,” 2024, [Accessed 2-January-2025]. [Online]. Available: https://arxiv.org/abs/2405.16177

[3] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, “Attention is all you need,” 2023, [Accessed 2-January-2025]. [Online]. Available: https://arxiv.org/abs/1706.03762

[4] P. Celka, N. Granqvist, and H. Schwabl, “Traditional tibetan pulse reading in the digital era,” vol. 3, 02 2019.

[5] T. Stracina, M. Ronzhina, R. Redina, and M. Novakova, “Golden standard or obsolete method? review of ecg applications in clinical and experimental context,” *Frontiers in Physiology*, vol. 13, p. 867033, 2022, [Accessed 2-January-2025]. [Online]. Available: https://doi.org/10.3389/fphys.2022.867033

[6] FibriCheck Academy, “Chapter 4: The differences and similarities between ecg and ppg,” 2025, accessed 17-January-2025. [Online]. Available: https://academy.fibricheck.com/hc/en-be/articles/6383416554396-Chapter-4-The-differences-and-similarities-between-ECG-and-PPG#:~:text=âœŤ%EF%B8%8F%20According%20to%20the%20ESC,without%20a%20prior%20AF%20diagnosis

[7] C. Mirescu and S. W. Harden, “Photoplethysmography as a potential alternative to electrocardiography for recording heart rate intervals used in variability analysis,” *Journal of Medicine and Life*, vol. 5, no. Spec Issue, pp. 123–128, 2012.

[8] Q. Wen, T. Zhou, C. Zhang, W. Chen, Z. Ma, J. Yan, and L. Sun, "Transformers in time series: A survey," 2023, [Accessed 2-January-2025]. [Online]. Available: https://arxiv.org/abs/2202.07125

[9] P. Mehrgardt, M. Khushi, S. Poon, and A. Withana, "Pulse transit time ppg dataset (version 1.1.0)," https://doi.org/10.13026/jpan-6n92, 2022, [Accessed 15-January-2025].

[10] A. L. Goldberger, L. A. N. Amaral, L. Glass, J. M. Hausdorff, P. C. Ivanov, R. G. Mark, and H. E. Stanley, "Physiobank, physiotoolkit, and physionet: Components of a new research resource for complex physiologic signals," *Circulation [Online]*, vol. 101, no. 23, pp. e215–e220, 2000, [Accessed 15-January-2025]. [Online]. Available: https://doi.org/10.1161/01.CIR.101.23.e215

[11] D. Jarchi and A. J. Casson, "Description of a database containing wrist ppg signals recorded during physical exercise with both accelerometer and gyroscope measures of motion," *Data*, vol. 2, no. 1, p. 1, 2017, [Accessed 15-January-2025]. [Online]. Available: https://doi.org/10.3390/data2010001

[12] Camntech Actiwave, "Camntech actiwave," http://www.camntech.com/, accessed 17-January-2025.

[13] W. RW, MischiM, and A. RM, "Reduction of periodic motion artifacts in photoplethysmography," *IEEE Transactions on Biomedical Engineering*, vol. 64, no. 1, pp. 196–207, Jan 2017, epub 2016 Apr 12.

[14] S. Nabavi and S. Bhadra, "A robust fusion method for motion artifacts reduction in photoplethysmography signal," *IEEE Transactions on Instrumentation and Measurement*, vol. 69, no. 12, pp. 9599–9608, 2020, [Accessed 2-January-2025].

[15] R. A. Pinto, H. S. De Oliveira, E. Souto, R. Giusti, and R. Veras, "Inferring ecg waveforms from ppg signals with a modified u-net neural network," *Sensors*, vol. 24, no. 18, 2024, [Accessed 2-January-2025]. [Online]. Available: https://www.mdpi.com/1424-8220/24/18/6046

[16] A. H. A. Zargari, S. A. H. Aqajari, H. Khodabandeh, A. M. Rahmani, and F. Kurdahi, "An accurate non-accelerometer-based ppg motion artifact removal technique using cyclegan," 2021, [Accessed 2-January-2025]. [Online]. Available: https://arxiv.org/abs/2106.11512

[17] Y. Zheng, C. Wu, P. Cai, Z. Zhong, H. Huang, and Y. Jiang, "Tiny-ppg: A lightweight deep neural network for real-time detection of motion artifacts in photoplethysmogram signals on edge devices," 2023, [Accessed 2-January-2025]. [Online]. Available: https://arxiv.org/abs/2305.03308

[18] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[19] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Köpf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, "Pytorch: An imperative style, high-performance deep learning library," 2019, [Accessed 2-January-2025]. [Online]. Available: https://arxiv.org/abs/1912.01703

[20] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. J. Millman, N. Mayorov, A. R. J. Nelson, E. Jones, R. Kern, E. Larson, C. J. Carey, İ. Polat, Y. Feng, E. W. Moore, J. VanderPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E. A. Quintero, C. R. Harris, A. M. Archibald, A. H. Ribeiro, F. Pedregosa, P. van Mulbregt, and SciPy 1.0 Contributors, "SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python," *Nature Methods*, vol. 17, pp. 261–272, 2020.

[21] T. Giorgino, "Computing and visualizing dynamic time warping alignments in r: The dtw package," *Journal of Statistical Software*, vol. 31, no. 7, pp. 1–24, 2009, [Accessed 17-January-2025]. [Online]. Available: https://doi.org/10.18637/jss.v031.i07