Shallow-PPGNet: A Simple yet Effective Network for Hypertension Detection

Pin-Ying Wu Ye-Sel Lee Yin Lei
University of California, San Diego
La Jolla, CA

Abstract

The Human Photoplethysmography (PPG) signal is a simple and non-invasive optical technique used to detect volumetric changes in blood in the peripheral circulation and can reveal the condition of the cardiovascular system in real-time. On the other hand, Blood Pressure (BP) is the measurement of the force exerted on artery walls and can be divided into several categories by their values, including Normotension (NT), Prehypertension (PHT), Stage 1 and Stage 2 Hypertension (HT), and Hypertension crisis, leading to many diseases such as heart failure, stroke, and kidney failure. Measuring PPG signals and real-time BP estimation is possible by using wearable devices. Therefore, we propose a ***Shallow-PPGNet*** *to predict whether the patient has blood pressure diseases based on PPG signals, especially Prehypertension and Hypertension, to early diagnose and avoid potential diseases. In addition, we also implement the multi-class classification task (NT/PHT/HT), which provides more detailed information to both patients and doctors than binary classification. The task is helpful in medical applications because the doctor can give early treatment to the patients base on the high true positive rate. Our model (****Shallow-PPGNet****) improves over 10% accuracy than the state-of-the-art and achieves 71.13% and 77.22% accuracy in NT/PHT and NT/HT tasks on PPG-BP [14] Database, and 80.2%, 89.5% on MIMIC-II Dataset [13] respectively.*

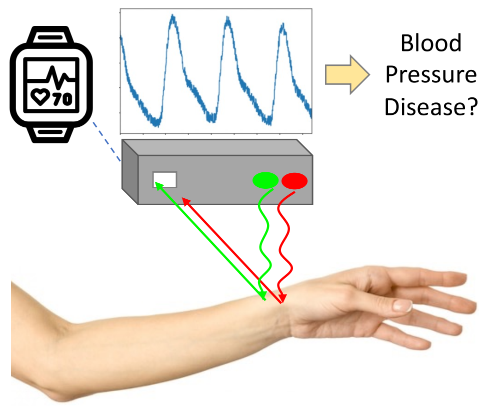

Figure 1. **Overview**. PPG signal is easily accessible without invasion and thus is widely adopted in smartwatch settings. In this project, we focus on detecting blood pressure diseases, i.e. hypertension, with PPG signals.

1. Introduction

Cardiovascular diseases (CVDs) have been a leading cause of mortality worldwide, accounting for approximately 31% of all deaths globally [11]. Hypertension [20], or high blood pressure, is a medical condition in which the force of blood against the walls of arteries is consistently too high, which is a significant risk factor for CVDs. Monitoring blood pressure persistently then becomes essential for early diagnosis and management of hypertension and the prevention of CVDs [21]. Blood pressure can be detected using a variety of methods, including traditional sphygmomanometers, automated blood pressure monitors, and photoplethysmography (PPG) signals. While professional devices in medical grade can provide more accurate measures of blood pressure, they are more bulky, expensive, and not easily portable [2]. Photoplethysmography (PPG) [2], an optical sensing-based technique that measures changes in blood volume in peripheral tissues, then becomes a popular choice for continual monitoring of cardiovascular function, due to its non-invasive, low-cost, and portable properties, which is widely integrated into wearable devices like smartwatches for monitoring health conditions in daily lives.

Owing to its importance for detecting CVDs, there are research works on hypertension detection with PPG signals [21], which usually formulate the problem as a classification task, where each signal segment is classified into Normotension (NT), Prehypertension (PHT), and Hypertension (HT). While machine learning approaches have been a powerful tool for solving complex tasks in the medical domain, from image analysis [18] to disease diagnosis [12] and drug discovery [19], it is also applied to hypertension detection and shown promising results. However, the performance is still undesired and generalized poorly to unseen data. We attribute the reasons to three challenges.

Firstly, the data quality may be influenced by the noise signals during the measure, which can make it more difficult to distinguish between normal and abnormal cases. Secondly, due to ethics and privacy concerns, data acquisition is arduous for medical datasets, which results in insufficient labeled data for training the machine learning models. Thirdly, as NT and PHT signals appear more frequently than HT signals in natural, the collected datasets [14] typically have significant data imbalance in the class distributions, leading to biased predictions toward more populated classes such as NT and PHT.

In this paper, we aim at developing a machine-learning algorithm for hypertension detection, with a focus on addressing the first two challenges. While not addressing the third, we analyze the data distribution and point out the imbalanced problem, which can inspire other future works. To enhance the data quality, we employ a data preprocessing procedure, consisting of a set of filters (e.g. median filter, roll filter, Chebyshev filter) to remove noises, without changing or distorting the original signal patterns. On the other hand, to address the problem of the small dataset, we propose a shallow network to analyze PPG signals and extract relevant features for predicting blood pressure diseases, namely **Shallow-PPGNet**, which leverages convolutional kernels to discover the sequential patterns in the PPG signals. Shallow-PPGNet consists of three convolutional layers with varying kernel sizes, followed by two fully connected layers. Specifically, the first convolutional layer performs feature extraction at the lowest level from the input signal, and the second and third convolutional layers extract higher-level features. The output of the last convolutional layer is flattened and fed into two fully connected layers for classification. We adopt this shallow design for two reasons. First, from the perspective of the curse of dimensionality, as the number of features grows, the number of samples required for training increases exponentially. As the dataset is not large-scale, we make the network as simple as possible for better estimation of the model parameters. Second, a lightweight model has more potential to be deployed for applications on edge devices like smartwatches, since the resource required for running is less.

We evaluate our model using two publicly available datasets of PPG signals [14, 13]. Without bells and whistles, **Shallow-PPGNet** achieves superior accuracy than state-of-the-art methods in predicting blood pressure diseases, with the accuracy of 77.22% and 89.51% on PPG-BP [14] and MIMIC-II [13] datasets respectively. Our results demonstrate the potential of PPG signals as a valuable tool for the early detection and monitoring of blood pressure diseases. Furthermore, by leveraging the temporal information present in PPG signals, our model can identify key features indicative of hypertension, providing clinicians with valuable insights into the underlying mechanisms of the disease. Therefore, this work has significant implications for improving the management of hypertension and reducing the burden of CVDs worldwide.

In conclusion, our study provides evidence that PPG signals can effectively predict blood pressure diseases. Furthermore, our shallow convolutional neural network model demonstrates high accuracy in identifying relevant features in PPG signals for predicting hypertension. This work highlights the potential of PPG signals as a valuable tool for the early detection and monitoring of blood pressure diseases. It can serve as a basis for future research in this area. Ultimately, applying machine learning algorithms to PPG signals has significant implications for improving the management of hypertension and reducing the burden of CVDs worldwide.

2. Related Work

It has been years since scientists and researchers started trying to detect CVDs through machine-learning approaches. In 2014, Golino et. al [10] proposed a classification model to detect hypertension and pre-hypertension cases based on a decision tree and managed to achieve a sensitivity of 58.38%, a specificity of 69.70%, and an AUC of 0.688. In 2018, Lopez et al.[5] developed a regression model and obtained a sensitivity of 77%, a specificity of 68%, and an AUC of 0.73. An example of a model that makes use of a more complex machine learning technique is one developed, again by Lopez et al. [6], based on ANN with a multilayer perceptron architecture. This new model managed to achieve a sensitivity of 40%, a specificity of 87%, a precision of 57.8%, and an AUC of 0.77.

While these models are based on different machine learning techniques, they all perform their classifications based on a variety of clinical and physiological data that may be difficult to obtain. The decision tree model by Golino et al. [10] requires features such as BMI, waist and hip circumference (WC and HC, respectively), and waist–hip ratio (WHR) as inputs. For Lopez's models, while they are able to achieve better performances, they are fitted on data published by National Health and Nutrition Examination Survey (or NHANES). The dataset contains thousands of columns of data, including but not limited to the subject's physical indices such as BMI, his/her dietary habits, smoking history, drug use, etc. Some of the features are extremely difficult to be measured effectively in a laboratory or a clinic setting, so the only way to have access to them is through a questionnaire. As a result, these data may be inaccurate due to reasons like personal biases. These factors prohibited the models from being used widely and effectively.

In recent years, with the development of digital wearable devices such as smartwatches, PPG signal has become much more accessible. As a result, many researchers decided to use it to classify CVDs. In 2019, Luo et al. [17] developed a CNN model for hypertension prediction. The model is trained over a well-known dataset called Multiparameter Intelligent Monitoring in Intensive Care(MIMIC) II [13]. Luo et al. also compared their proposed model with models using other machine learning techniques such as KNN, J48, Random Forest, SMO, Native Bayes, and Logistics, and they claimed that results from CNN gave the best performance. In 2020, Tjahjadi et al. [9] proposed a model that extracts information from PPG signals using a short-time Fourier transform (STFT) and then uses the resulting features to train a bidirectional long short-term memory (BLSTM) network. Tjahjadi et al. claimed that their process required less time and achieved better performance than a normal CNN approach.

In 2018, a short-recorded PPG dataset collected for the purpose of blood pressure monitoring was published in China [14]. This dataset contains 657 data segments from 219 subjects, and each data segment is 2.1 seconds. Four years later, Gupta et al. [7] proposed a new CNN-LSTM model based on this dataset. Their model uses the given PPG segments to classify three different cases: normotension (NT), prehypertension (PHT), and hypertension (HT). In their article, the authors claim that they achieved 61.07% and 67.76% accuracy in binary classifications of NT vs PHT and NT vs HT.

3. Preliminaries

In this section, we give a brief review of neural networks and convolutional neural networks.

3.1. Neural Networks

The Neural Network (NN) is a model that imitates the neurons in the biological brain. Each neuron is comprised of a set of parameters that can be learned through mathematical methods. In a neural network, there can be multiple neuron layers, each is composed of a set of neurons, e.g. input layer, hidden layers, and the output layer. Neurons between layers are interconnected, and the neurons at the later layers can be stimulated by the neurons from the former layers. The network's input layer can be fed with various kinds of signals, such as biological signals, image signals, and audio signals. Therefore, it can be applied to several domains, including biomedical technology, image processing, computer vision, natural language processing, acoustic, communications engineering, and so on.

The weights and the bias that reside in each neuron, which indicates the degree they contribute to the output, are the parameters that the NN should learn. These parameters define a mathematical model as a function of the mapping from the input to the output. Since the function is differentiable, the values of the model parameters can be estimated with algorithms like gradient descent to find the local minimum of the cost function, which can be estimated with a number of input-output pairs.

3.2. Convolutional Neural Networks

Convolutional Neural Networks (CNNs) have become increasingly popular in recent years due to their effectiveness in various applications, including image and speech recognition, natural language processing, and time-series analysis. One of the critical advantages of CNNs is their ability to capture temporal or spatial patterns in the input data by exploiting the local connectivity pattern, which means that each neuron in the network is only connected to a small region of the input signal, allowing the network to detect local patterns or features at different time scales and is particularly useful for analyzing time-series data, such as photoplethysmography (PPG) or electroencephalography (EEG) signals, which contain complex temporal patterns that can be difficult to capture using traditional neural networks.

Furthermore, the most significant difference between CNNs and NNs is that the weight sharing in CNNs allows the same set of learned weights to be applied to different parts of the input signal. As a result, it reduces the number of parameters that need to be learned, which is important for applications with limited training data.

In addition to local connectivity and weight sharing, CNNs also use pooling layers to reduce the dimensionality of the output from each convolutional layer. Pooling layers can extract invariant features from the input signal, which can be helpful when working with noisy or variable data, and also reduce overfitting by combining multiple feature maps into a single output. CNNs typically have multiple layers, with each layer learning increasingly complex features. The hierarchical feature extraction allows the network to learn high-level representations of the input signal, which can be used for tasks such as classification or regression.

4. Shallow-PPGNet

In this section, we formulate the problem of hypertension detection, and introduce the data preprocessing procedure and the proposed network design.

4.1. Problem Formulation

PPG signals are 1-d serial data. To perform hypertension detection, a typical way is to apply a sliding window truncating multiple frames with the same window length for classification, i.e. $\{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_N\}$, where $\mathbf{x}_i \in \mathbb{R}^L$, L and N are the window length and number of segments respectively. Following prior works [7], each frame is viewed as an independent sample, and the model aims at generating a class label for each input PPG segment. In this paper, we study the problem with three tasks:

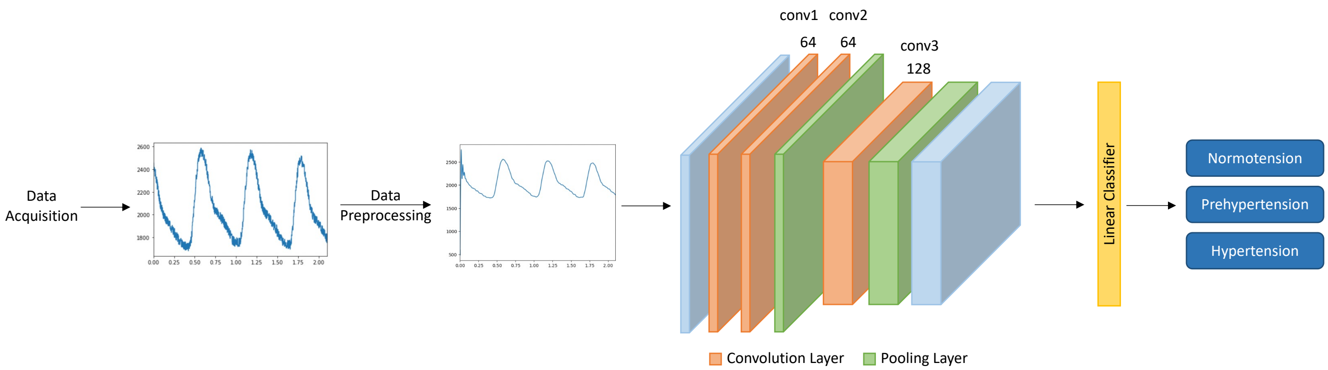

Figure 2. **Model Architecture**. We first preprocess the data with the steps described in subsection 4.2. The processed data is then fed into a convolutional neural network for feature extraction and classification. The network is composed of three convolutional layers as defined in Table 1.

- NT/PHT classification
- NT/HT classification
- NT/PHT/HT classification

where NT/PHT/HT is the most difficult task, as it required classifying the samples into three categories instead of two. NT/PHT is the second because it is a fine-grained classification task, where the difference between NT/PHT signals are minor than NT/HT signals.

4.2. Data Preprocessing

In 2016 IEEE International Conference on Consumer Electronics (ICCE), Dahee Ban and Sungoh Kwon pointed out that there are two major sources of noises in PPG signals, namely movement noise and high-frequency noise [3]. The common causes of high-frequency noise are thermal noise and electromagnetic interference in cables. On the other hand, voluntary or involuntary movements of a subject during the process of signal recording can lead to movement noises. As a result, it is necessary to apply data preprocessing first so that the signals will be more accurate and thus allow for better training results.

In our approach, we first apply a median filter with kernel size 23 to remove noises in the signal, followed by a roll filter with the same kernel size. This initial step allows us to replace the data points that are locally too high or too low in frequency with local medians and means. The underlying assumption behind this step is that local data values that are abnormally too high or low in frequency result from noises, and that the local noise is Gaussian-distributed with mean 0. Afterward, we apply a 4th-order Chebyshev filter with a cutoff frequency of 25 and a minimum attenuation stopband of 10 [15], which is designed to remove any unwanted high-frequency components from the signal. While Liang et al. [15] alleged that for the given PPG-BP dataset the best filtering approach is a backward-forward Chebyshev filter, we compare the performance of a normal Chebyshev filter and a backward-forward Chebyshev filter and find that the difference is almost negligible. Therefore, we decided to not apply a backward-forward filter to further complicate the problem. As we can see from the graphs Figure 3, after using the combination of these filters, we can effectively remove noise and improve the quality of the signal without changing or distorting the original signal patterns, which is important for many applications in signal processing and analysis.

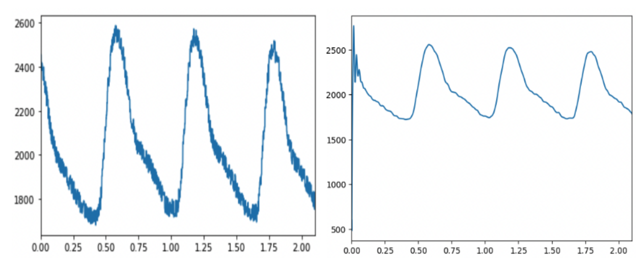

Figure 3. **Data Preprocessing**. The original signal (left) is passed through a set of preprocessing steps (e.g. median filter, roll filter, Chebyshev filter) to remove noises and enhance smoothness. The preprocessed signal is shown on the right.

4.3. Model Architecture

To address the challenges posed by the small dataset, from the perspective of curse of dimensionality, it is important to minimize the number of trainable parameters to avoid overfitting and ensure the model parameters are estimated with sufficient data. To achieve this, we design a shallow convolutional neural network (CNN) with a specific architecture, which consists of 3 convolution layers, 2 pooling layers, and 2 fully-connected layers, as shown in Table 1. As the data frame are 1-d series, we employ 1D convolutional kernel with varying lengths in different layers to capture the sequential patterns in the PPG signals. While

Layer	Type	Kernel Size	Input Channel	Output Channel	Trainable Parameters
1	Conv1d	30	1	64	2015
2	Conv1d	15	64	64	61504
-	MaxPool1d	7	64	64	0
3	Conv1d	5	64	128	41088
-	MaxPool1d	7	128	128	0
4	Linear	-	-	-	-
5	Linear	-	-	-	-

Table 1. The model architecutre of the proposed **(Shallow-PPGNet).** architecture.

the former layers of the CNN extract low-level features, the convolutions in later layers extract more high-level features. In addition to convolutional layers, we adopt max-pooling layers to reduce the data dimension and prevent overfitting and linear layers with non-linear activations to map the extracted features to the number of output categories. The network prediction can be formulated as

$$\mathbf{p} = Softmax(\text{Shallow-PPGNet}(\mathbf{x})) \tag{1}$$

$$\hat{y} = \arg\max_{k} p_k \tag{2}$$

where $\mathbf{p}$ denotes the output class probability. By using this architecture, we can effectively reduce the number of parameters in the network while still achieving high accuracy in our predictions, and leverage the power of convolutional neural networks while mitigating the risks associated with overfitting in the small dataset size.

4.4. Learning

We adopt a Softmax layer at the end of the linear layer to produce the classification probability for both two-class (NT/PHT and NT/HT) and three-class (NT/PHT/HT) classification tasks. The network is optimized with a cross-entropy loss, i.e.

$$l_{ce} = -\sum_{k=1}^{C} y_k \log \mathbf{p}_k \tag{3}$$

$$L_{ce} = \frac{1}{N}\sum_{i=1}^{N} l_{ce}(\mathbf{x}_i, y_i) \tag{4}$$

where C is the number of classes. Note that $C = 2$ for NT/PHT and NT/HT classification, and $C = 3$ for NT/PHT/HT classification.

5. Experiments

5.1. Datasets

We validate Shallow-PPGNet with two different datasets [13, 14], as introduced in the following.

PPG-BP Database [14] is collected from Guilins People's Hospital in China. The dataset contains 219 subjects ranging in age from 20 to 89, including their hypertension and diabetes histories, weight, height, age, diastolic blood pressure, systolic blood pressure, BMI, and heart rate. There are 657 data segments from 219 subjects, and the blood pressure and 3 PPG segments from each subject, shown in Figure 5, also contained the data quality of each segment. The Sampling rate is 1kHz, so we have 2100 sampling points for each 2.1-second segment.

After choosing high-quality data with positive quality scores, we use 648 data segments, which is a very small dataset size to train deep learning models. Afterward, we split the dataset into training, validation, and testing sets with the proportion [0.64, 0.16, 0.2], respectively, and the number of hypertension samples is less than those of the other two classes, as shown in Figure 6.

MIMIC-II Dataset [13] is a publicly available electronic health record database that provides a wealth of information about patients admitted to the intensive care units (ICUs) of the Beth Israel Deaconess Medical Center in Boston, Massachusetts, USA. The dataset contains over 20000 recording periods of PPG signals with ABP ranging from seconds to hours. The data was collected using electronic medical record systems, which allowed for detailed and accurate documentation of clinical variables. The Sampling rate is 125Hz, and we segment each signal into 5 seconds.

In MIMIC-II dataset, ABP signals recorded simultaneously with PPG signals are used as a ground truth value for hypertension classification. We implement Elgendi et al.'s peak detection approach [4] for each segment and obtain the ground truth SBP values from ABP. According to the standards of the US National Institutes of Health, Normal, Prehypertension, and Hypertension are labeled if the calculated SBP values $<$ 120mmHg, 120-140mmHg, $>$ 140mmHg respectively.

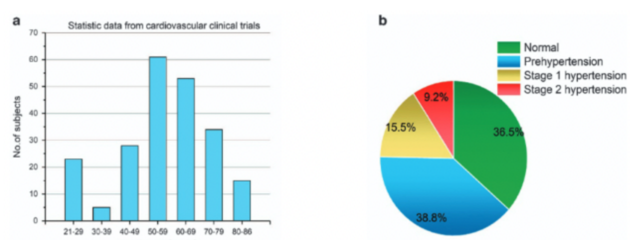

Figure 4. Data statistics of the PPG-BP dataset from [14].

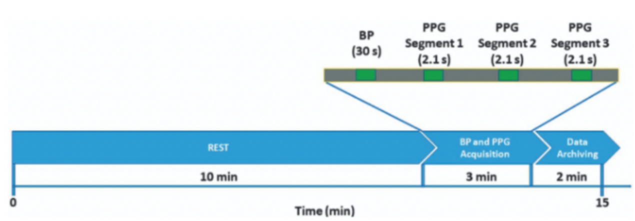

Figure 5. The measurement protocol from [14].

Model	Tasks	Accuracy (%)	Sensitivity (%)	Specificity (%)	Precision (%)	F1 score (%)
VGG16 [22]	NT/PHT	55.67	96.15	8.89	54.95	69.93
	NT/HT	62.03	46.67	71.43	50.00	48.28
ResNet18 [8]	NT/PHT	62.89	92.59	25.58	60.98	73.53
	NT/HT	65.82	33.33	85.71	58.82	42.55
CNN-LSTM [7]	NT/PHT	61.07	55.90	64.40	50.77	53.21
	NT/HT	67.76	68.40	66.60	75.76	71.89
Shallow-PPGNet (Ours)	NT/PHT	**71.13**	94.64	39.02	67.95	79.10
	NT/HT	**77.22**	67.86	82.35	67.86	67.86

Table 2. Comparisons to different CNN-based architectures on the PPG-BP [14] dataset. Models are evaluated with two tasks, NT/PHT classification and NT/HT classification.

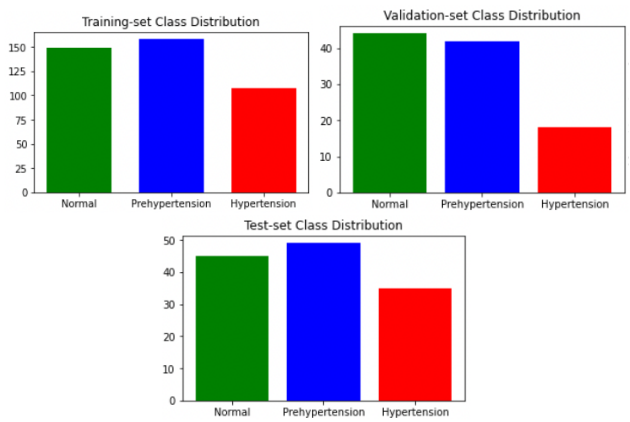

Figure 6. Class distributions in training, validation, and testing set.

5.2. Evaluation Metrics

We evaluate the models with 5 different metrics: accuracy, sensitivity, specificity, precision, and F1-score. These metrics are defined with different combinations of true positive rate (TP), true negative rate (TN), false positive rate (FP), and false negative rate (FN), which are defined as follows,

$$TP_k = \frac{\sum_i^N \mathbb{1}(y_i = k, \hat{y}_i = k)}{\sum_i^N \mathbb{1}(y_i = k)}$$

$$TN_k = \frac{\sum_i^N \mathbb{1}(y_i \neq k, \hat{y}_i \neq k)}{\sum_i^N \mathbb{1}(y_i \neq k)}$$

$$FP_k = \frac{\sum_i^N \mathbb{1}(y_i \neq k, \hat{y}_i = k)}{\sum_i^N \mathbb{1}(y_i \neq k)}$$

$$FN_k = \frac{\sum_i^N \mathbb{1}(y_i = k, \hat{y}_i \neq k)}{\sum_i^N \mathbb{1}(y_i = k)}$$

where $(\cdot)_k$ denotes the metric for a class k in C classes.

Accuracy

$$Accuracy = \frac{1}{C}\sum_{k=1}^{C} \frac{TP_k + TN_k}{TP_k + TN_k + FP_k + FN_k}$$

Accuracy is the most general evaluation metric, and it works if false positives (FP) and false negatives (FN) have similar costs. However, if the cost of FP and FN are very different, it is better to consider both Sensitivity and Specificity. In our task, the data is a little imbalanced since we have less HT data as mentioned above, which is a typical case in medical cases. Therefore, a model can just predict all results as NT and get high accuracy, but it is not useful for the task. Therefore, we have to consider the Sensitivity, Specificity, Precision, and F1-score at the same time to evaluate the model properly.

Sensitivity

$$Sensitivity = \frac{1}{C}\sum_{k=1}^{C} \frac{TP_k}{TP_k + FN_k}$$

Sensitivity, also known as true positive rate, measures the proportion of actual positive instances that are correctly identified by the model. It reflects the ability of the model to correctly identify instances that belong to the positive class.

Specificity

$$Specificity = \frac{1}{C}\sum_{k=1}^{C} \frac{TN_k}{TN_k + FP_k}$$

Specificity, also known as true negative rate, measures the proportion of actual negative instances that are correctly identified by the model. It reflects the ability of the model to correctly identify instances that belong to the negative class.

Precision Precision is important in situations where the cost of false positives is high, especially in medical diagnosis. In these scenarios, it is crucial to minimize the number of false positives to avoid misdiagnosis or incorrect accusations of fraud. In addition, precision can provide a better understanding of the model performance when the class distribution is imbalanced.

$$Precision = \frac{1}{C} \sum_{k=1}^{C} \frac{TP_k}{TP_k + FP_k}$$

F1-Score F1-score is a weighted average of precision and sensitivity. It is especially suitable for evaluating imbalanced datasets because it considers both precision and sensitivity simultaneously.

$$F1\text{-}Score = \frac{2 * Precision * Sensitivity}{Precision + Sensitivity}$$

5.3. Ablation Studies

In this section, we conduct experiments for three classification tasks (i.e. NT/PHT, NT/HT, and NT/PHT/HT classification) with two different datasets. We also compare the proposed model with other model architectures as discuss the difference.

Comparisons to Random Forest. Since the PPG-BP Database [14] is very small, we implement a random forest classifier as the very first baseline model, which is a traditional machine learning method leveraging multiple decision trees with boostraping and bagging strategies. We ablate different hyperparameter choices (e.g. number of decision trees in the random forest), and find that 15-20 estimators give optimal results. However, even the best model only provides 41% accuracy, recall, and F1-score. For a more detailed analysis, we plot the confusion matrix as shown in Figure 7 (left). From the plot, we can observe that the model has about 40% probability to classify a sample into prehypertension no matter what the ground truth label is, which means the model does not really learn the class. By contrast, the proposed Shallow-PPGNet is a deep learning-based approach, which achieves superior performance on the same dataset as shown in the confusion matrix in Figure 7 (right). The confusion matrix in this case is way better than the one for the random forest model, as we can clearly see the diagonal of the matrix. This is reasonable because the random forest baseline does not model the sequential information of the data frame, while our model (Shallow-PPGNet) leverages convolutional kernels to capture the patterns in time series, leading to superior performance.

Comparison to Different CNN-based Architectures. In this section, we further ablate different CNN-based architectures, including VGG16 [22] and ResNet [8], and the current state-of-the-art approach, i.e. CNN-LSTM [7], that adopts a combination of CNN and LSTM (long-short-term-memory). LSTM is a type of recurrent neural network (RNN), which is widely used for learning time-series data. Note that for VGG16 and ResNet, we replace the 2D convolutions with 1D convolutions but keep the kernel sizes and the number of channels the same. As we can see in Table 2, the proposed Shallow-PPGNet outperforms other model architectures, where Shallow-PPGNet achieves 71.13% and 77.22% accuracy in NT/PHT and NT/HT classification tasks on PPG-BP dataset [14] respectively. Compared to the CNN-LSTM model, we improve about 10% accuracy, and also improve overall performance such as sensitivity, precision, and F1-scores. In addition, we achieve 94.64% and 67.86% sensitivity for each classification task, which is helpful in medical applications because the doctor can get a high true positive rate to give early treatment to the patient. Besides, the precision and F1-score of Shallow-PPGNet are also the highest compared to other models in the NT/PHT classification task. These experiments show that a shallow network can have more power than deeper and more complex model architectures when solving a medical task which does not have a large number of training samples.

Experiments with Another Dataset. Despite the high performance of the proposed Shallow-PPGNet on the PPG-BP dataset, we further validate its power with MIMIC-II dataset [13]. Similar to the performance on PPG-BP, Shallow-PPGNet achieves high accuracy on all the tasks and has uniformly good performance in all the metrics, as shown in Table 3. In the confusion matrix plot in Figure 8, we can also observe that the diagonal of the matrix is very clear, which means that there is little confusion among the classes.

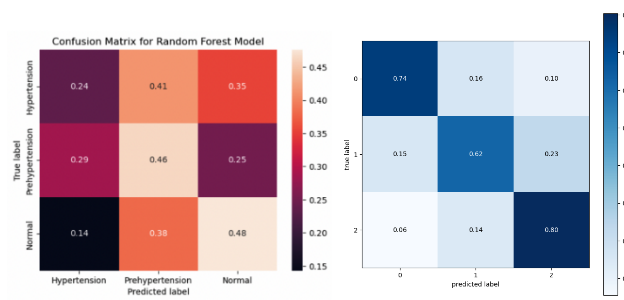

Figure 7. The confusion matrix of the random forest model (left) and the proposed Shallow-PPGNet (right) on the PPG-BP dataset. The accuracy of Shallow-PPGNet in the NT/PHT/HT classification task is 44.19%.

5.4. Transfer Learning to Diabetes Detection

In this section, we implement transfer learning between hypertension and diabetes. Transfer learning is a machine

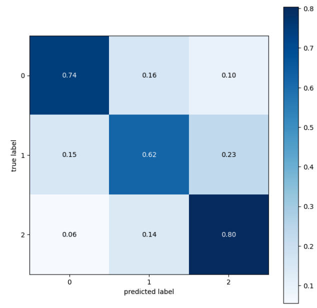

Figure 8. The confusion matrix for Our Model (Shallow-PPGNet) on MIMIC-II Dataset.

Tasks	NT/PHT	NT/HT	NT/PHT/HT
Accuracy (%)	80.27	89.51	72.86
Sensitivity (%)	81.54	90.73	-
Specificity (%)	79.03	88.04	-
Precision (%)	79.08	90.10	-
F1 score (%)	80.30	90.41	-

Table 3. Results of our model **(Shallow-PPGNet)** on the MIMIC-II dataset [13].

Approach	Training from scratch	Transfer learning
Accuracy (%)	80.62	83.46
Sensitivity (%)	0.0	27.78
Specificity (%)	98.11	92.66
Precision (%)	0.0	38.46
NPV (%)	81.89	88.60
F1-score (%)	-	32.26

Table 4. Results of our model **(Shallow-PPGNet)** training from scratch and transferring learning from blood pressure disease labels to diabetes labels.

learning technique that enables a model to apply the knowledge gained from one task to another. It has become increasingly popular in various domains by leveraging pre-existing knowledge and saving both time and computational resources [1]. Especially in medical fields, since medical data can be difficult to obtain due to privacy concerns, regulations, and the time-consuming nature of data collection and labeling, transfer learning is beneficial when working with small medical datasets [1]. Diabetes is a chronic metabolic condition characterized by high blood sugar levels, which increases the risk of developing various cardiovascular diseases including hypertension. We assessed whether transfer learning improves the performance of the model in diabetes by fine-tuning the model pre-trained on the hypertension task. We utilized Shallow-PPGNet pre-trained on MIMIC-II Dataset, which has shown the highest performance on hypertension (NT/HT) classification, and fine-tuned the model with diabetes classes from the PPG-BP Database. Our experimental results show that we can improve the overall performance of diabetes prediction via applying transfer learning on our shallow-PPGNet. As shown in Table 4, we can achieve 83.46% accuracy which is 2.84% higher than the model trained from the scratch. Other criteria including sensitivity, precision, and F1-score are also significantly increased for predicting diabetes of patients despite the large data imbalance on normal versus diabetes.

5.5. Implementation Details

The data preprocessing procedure is described in subsection 4.2. We train our model for 180 epochs, with batch size 10 and learning rate 0.01. The network is optimized with SGD optimizer with momentum 0.9 and weight decay 0.0005. The proposed network was implemented with PyTorch, and the random forest baseline was trained with the implementation in scikit-learn package.

6. Conclusion

In this paper, we explore the task of hypertension detection with PPG signals, which is formulated as a classification task. We propose a novel architecture, **Shallow-PPGNet**, which is shallow but effective as we leverage convolutional kernels to capture the temporal information in the PPG signals. As medical datasets are small, to overcome the curse of dimensionality, we adopt a shallow model design that can better estimate the model parameters with a limited number of samples. We evaluate the model with 3 different tasks, including NT/PHT, NT/HT, and NT/PHT/HT classification, on two different datasets, PPG-BP database and MIMIC-II dataset. The experiments show that the proposed **Shallow-PPGNet** outperforms standard machine learning models like random forest and other deep learning architectures, including the state-of-the-art CNN-LSTM model. We also transfer the knowledge from the hypertension detection to the diabetes detection task and show that Shallow-PPGNet can generalize to other medical tasks as well. These experiments suggest that a shallow network can be more powerful for medical tasks than more complex models. In conclusion, our approach has significant potential for use in medical applications, allowing doctors to provide early treatment to patients with blood pressure diseases.

7. Future Works

As the challenges discussed in the introduction section, one can improve the performance in this task by integrating training algorithms that compensate for data imbalance. For example, the model can be trained with Focal loss [16]. In addition, we can train the model on MIMIC-II dataset [13], which is a larger dataset, and adapt it to PPG-BP Database [14], which is a very small dataset, to validate whether our model works for domain adaptation.

References

[1] Laith Alzubaidi, Mohammed A Fadhel, Omran Al-Shamma, Jinglan Zhang, J Santamaría, Ye Duan, and Sameer R. Oleiwi. Towards a better understanding of transfer learning for medical imaging: a case study. *Applied Sciences*, 10(13):4523, 2020.

[2] Anand Chandrasekhar, Mohammad Yavarimanesh, Keerthana Natarajan, Jin-Oh Hahn, and Ramakrishna Mukkamala. Ppg sensor contact pressure should be taken into account for cuff-less blood pressure measurement. *IEEE Transactions on Biomedical Engineering*, 67(11):3134–3140, 2020.

[3] Sungoh Kwon Dahee Ban. Movement noise cancellation in ppg signals. 2016.

[4] Mohamed Elgendi, Ian Norton, Matt Brearley, Derek Abbott, and Dale Schuurmans. Systolic peak detection in acceleration photoplethysmograms measured from emergency responders in tropical conditions. *PloS one*, 8(10):e76585, 2013.

[5] Edward Rolando Núñez-Valdez Vicente García-Díaz Fernando López-Martínez, Aron Schwarcz.MD. Machine learning classification analysis for a hypertensive population as a function of several risk factors. *Expert Systems with Applications*, 110:206–215, 2018.

[6] Rubén González Crespo Vicente García-Díaz Fernando López-Martínez, Edward Rolando Núñez-Valdez. An artificial neural network approach for predicting hypertension using nhanes data. *Scientific Report*, 10, 2020.

[7] Ketan Gupta, Nasmin Jiwani, and Neda Afreen. Blood pressure detection using cnn-lstm model. In *2022 IEEE 11th International Conference on Communication Systems and Network Technologies (CSNT)*, pages 262–366, 2022.

[8] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[9] Hendri Murfi Hendrana Tjahjadi, Kalamullah Ramli. Noninvasive classification of blood pressure based on photoplethysmography signals using bidirectional long short-term memory and time-frequency analysis. *IEEE Access*, 8, 2020.

[10] Stenio Fernando Pimentel Duarte Cristiano Mauro Assis Gomes Telma de Jesus Soares Luciana Araujo dos Reis Hudson Fernandes Golino, Liliany Souza de Brito Amaral and Joselito Santos. Predicting increased blood pressure using machine learning. *Journel of Obesity*, 2014.

[11] Philip Joseph, Vellappillil Raman Kutty, Viswanathan Mohan, Rajesh Kumar, Prem Mony, Krishnapillai Vijayakumar, Shofiqul Islam, Romaina Iqbal, Khawar Kazmi, Omar Rahman, et al. Cardiovascular disease, mortality, and their associations with modifiable risk factors in a multi-national south asia cohort: a pure substudy. *European Heart Journal*, 43(30):2831–2840, 2022.

[12] Yogesh Kumar, Apeksha Koul, Ruchi Singla, and Muhammad Fazal Ijaz. Artificial intelligence in disease diagnosis: a systematic literature review, synthesizing framework and future research agenda. *Journal of Ambient Intelligence and Humanized Computing*, pages 1–28, 2022.

[13] Joon Lee, Daniel J Scott, Mauricio Villarroel, Gari D Clifford, Mohammed Saeed, and Roger G Mark. Open-access mimic-ii database for intensive care research. In *2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 8315–8318. IEEE, 2011.

[14] Yongbo Liang, Zhencheng Chen, Guiyong Liu, and Mohamed Elgendi. A new, short-recorded photoplethysmogram dataset for blood pressure monitoring in china. *Scientific data*, 5(1):1–7, 2018.

[15] Yongbo Liang, Mohamed Elgendi, Zhencheng Chen, and Rabab Ward. An optimal filter for short photoplethysmogram signals. *Scientific data*, 5(1):1–12, 2018.

[16] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988, 2017.

[17] Yue Luo; Yang Li; Yao Lu; Shuisheng Lin; Xinchun Liu. The prediction of hypertension based on convolution neural network. 2019.

[18] Andreas S Panayides, Amir Amini, Nenad D Filipovic, Ashish Sharma, Sotirios A Tsaftaris, Alistair Young, David Foran, Nhan Do, Spyretta Golemati, Tahsin Kurc, et al. Ai in medical imaging informatics: current challenges and future directions. *IEEE journal of biomedical and health informatics*, 24(7):1837–1857, 2020.

[19] Debleena Paul, Gaurav Sanap, Snehal Shenoy, Dnyaneshwar Kalyane, Kiran Kalia, and Rakesh K Tekade. Artificial intelligence in drug discovery and development. *Drug discovery today*, 26(1):80, 2021.

[20] Mustafa Sameer and Bharat Gupta. Beta band as a biomarker for classification between interictal and ictal states of epileptical patients. In *2020 7th International Conference on Signal Processing and Integrated Networks (SPIN)*, pages 567–570, 2020.

[21] Muhammad Shabaan, Kaleem Arshid, Muhammad Yaqub, Feng Jinchao, M Sultan Zia, Giridhar Reddy Bojja, Muazzam Iftikhar, Usman Ghani, Loknath Sai Ambati, and Rizwan Munir. Survey: smartphone-based assessment of cardiovascular diseases using ecg and ppg analysis. *BMC medical informatics and decision making*, 20:1–16, 2020.

[22] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.