p-ISSN 2089 – 0133 e-ISSN 2477-6416 Indonesian Journal of Applied Physics (IJAP) Vol. 15 No.2 page 330
URL: https://jurnal.uns.ac.id/ijap/article/view//102421 DOI:https://doi.org/10.13057/ijap.v15i2.102421

IMPLEMENTATION OF CNN-SVM WITH INDEX PATTERN-BASED FEATURE SELECTION ON PPG SIGNALS FOR CUFFLESS HYPERTENSION DETECTION

Dafina Nazhifah, Wiharto*, Fajar Muslim
Department of Informatics, Faculty of Information Technology and Data Science, Sebelas Maret University
Surakarta, Indonesia
*wiharto@staff.uns.ac.id

Received 2025-05-18, Revised 2025-05-25, Accepted 2025-06-04,
Available Online 2025-10-1, Published Regularly October 2025

ABSTRACT

Hypertension is one of the leading causes of death worldwide and often goes undetected due to its minimal symptoms. Early detection is crucial, and one non-invasive method involves the use of photoplethysmogram (PPG) signals. PPG-based analysis also supports cuffless blood pressure monitoring, offering a more comfortable and continuous alternative to traditional cuff-based methods. However, PPG signals contain a large number of features, which can lead to information redundancy and decreased model performance. This study proposes a hypertension detection system based on a CNN-SVM combination, preceded by feature selection using position-based indices (odd, even, specific multiples) to reduce data dimensionality and accelerate computation. This simple feature selection approach, which is rarely explored in previous studies, aims to reduce dimensionality without requiring complex computations. The PPG signal dataset was obtained from 216 patients at UNS Hospital. After preprocessing and feature selection, feature extraction was performed using a Convolutional Neural Network (CNN), followed by classification using a Support Vector Machine (SVM). The model was evaluated under three classification scenarios: normal vs. prehypertension-hypertension, normal-prehypertension vs. hypertension, and three-class classification. The best classification accuracy achieved was 93.10% for the normal vs. prehypertension-hypertension scenario. The result shows that the method used in this study is practical because it is simple and computationally efficient, but still gives good accuracy. This suggests that simple feature selection strategies can effectively enhance PPG-based hypertension detection.

Keywords: Hypertension Detection; Photoplethysmogram (PPG); CNN-SVM; Index-Based Selection; Cuffless Monitoring

INTRODUCTION

Hypertension is one of the leading causes of death worldwide and is often referred to as the "silent killer" because it usually has no symptoms in the early stages[1]. According to the World Health Organization, approximately 1.28 billion adults aged 30 to 79 are estimated to be living with hypertension globally, with 46% of them unaware of their condition. This lack of awareness is due to the fact that hypertension typically shows no symptoms until it reaches a critical stage[2]. Regular blood pressure monitoring plays an essential role in early prevention. However, current measurement techniques, both invasive and non-invasive (cuff-based), have some drawbacks, including patient discomfort, risk of infection, and impracticality for continuous monitoring[3,4].

With advancements in biomedical technology, photoplethysmogram (PPG) signals have emerged as a promising non-invasive and cuffless alternative for monitoring cardiovascular parameters, including blood pressure[5]. PPG sensors measure variations in blood volume through light absorption at the skin surface, commonly at the fingertip or wrist[6]. These signals are easy to acquire and inexpensive, making them suitable for wearable devices. From these signals, various physiological parameters can be derived, such as heart rate, blood pressure, and blood flow[7]. To ensure accurate interpretation, PPG signals must be properly processed and classified using machine learning (ML) techniques, which serve as the primary focus of this research. Traditional methods such as Pulse Transit Time (PTT), Pulse Arrival Time (PAT), and Pulse Wave Velocity (PWV) estimate blood pressure using mathematical models. However, these approaches have notable limitations, such as needing multiple sensors, complex modelling, and frequent calibration[3]. ML offers a more practical, adaptive, and accurate approach for estimating blood pressure using only PPG signals.

Previous studies have used different machine learning methods to detect hypertension from PPG signals, such as CNN-LSTM models[8] and regression models on FPGA hardware[5]. These models gave good results, with the CNN-LSTM achieving an accuracy of 76% and the regression model reaching up to 92.42%. However, many studies still face challenges in effectively handling high-dimensional data. To address this, feature selection is often employed. Yet, many studies rely on complex methods, such as Wavelet Scattering Transform (WST)[2], achieving 71.42% accuracy, or deep spectral-morphological networks (e.g., MTFF[9] and CS-NET[10], both achieving 98.7% accuracy), which demand high computational resources and are often trained on public datasets that lack population diversity. Using local data is important due to the characteristics of body signals can vary between populations such as skin pigmentation. The research results showed that people with darker skin tone may get less accurate PPG readings because their skin absorbs more infrared light before it reaches the blood vessels. Many public datasets mostly contain data from lighter-skinned individuals, which can cause bias and make the models less accurate for other groups[11].

This study focuses on developing a hypertension detection model based on local PPG signal data from UNS Hospital. The approach utilizes a simple feature selection technique based on positional indices (e.g., odd, even, specific multiple positions) to reduce dimensionality and processing time. This method is rarely explored in previous research. The selected features are then processed through a hybrid CNN-SVM architecture, which has demonstrated strong performance in biomedical signal analysis[10]. The combination of CNN and SVM leverages the strengths of both methods: CNN excels at extracting noise-resilient features, while SVM provides robust classification by maximizing the decision margin. Replacing the CNN's classification layer with an SVM, can enhance the model's accuracy and generalization capabilities[12]. The proposed model also aims to provide better relevance and generalizability to local populations, where physiological differences such as skin pigmentation may affect PPG signal quality. Overall, this approach offers an alternative that is both accurate and computationally efficient, and more suitable for Indonesian population.

METHOD

The workflow of this study for detecting hypertension using PPG signals is illustrated in Figure 1. This research is an experimental quantitative study, as it involves designing, testing, and evaluating a computational model based on measured numerical data. It consists of several main stages: dataset collection, preprocessing, feature selection, feature extraction, classification, and model evaluation. The initial stage begins with collecting PPG signal data from local sources, followed by preprocessing to enhance data quality. This process includes handling missing values, removing duplicates, and grouping the data into several classification scenarios. Next, feature selection is performed based on positional indices, with several experimental schemes such as selecting odd indices, even indices, multiples of 3, and multiples of 4. The selected feature data is then used as input for a hybrid CNN-SVM architecture, where CNN is responsible for extracting important features from the signals, and SVM acts as the classifier. Various configurations of CNN (number of layers, number of filters, kernel sizes) and SVM (kernel types, C parameter) are tested to obtain the best parameter combination. Finally, the model's performance is evaluated using metrics such as accuracy, specificity, sensitivity, and F1-score to assess the system's effectiveness in detecting hypertension based on PPG signals. The computational experiments were conducted using Google Colaboratory, a cloud-based Python programming environment that provides GPU support and facilitates real-time collaboration and reproducibility of the research workflow.

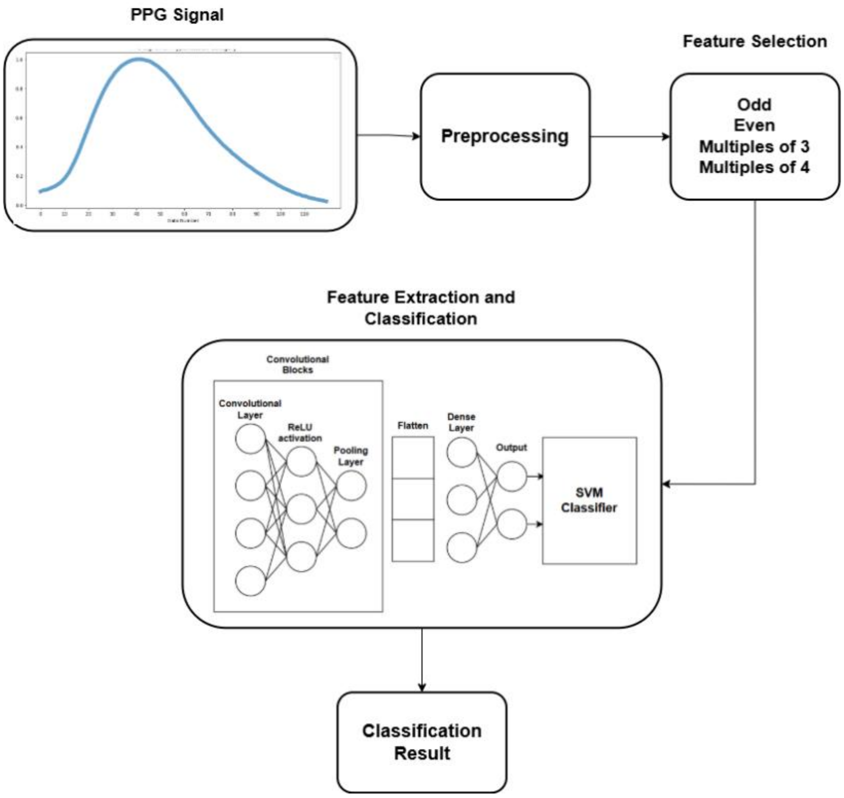

Figure 1. Proposed system for hypertension detection using PPG signals

PPG Dataset

The PPG data used in this study were collected from patients at UNS Hospital using a device developed by Nuryani[13] using Arduino and Android-based device. The system consists of several main components: the Easy Pulse Plugin PPG sensor, Arduino Nano, HC-05 Bluetooth module, and an Android application as the user interface. As shown in Figure 2, the PPG sensor detects changes in light intensity reflected by blood flow in the fingertip. It emits light using an LED and captures it with a photodiode, detecting changes in blood volume caused by heartbeats and converting them into signals. The signal is sent to the Arduino Nano, where a peak detection algorithm using thresholding and time interval constraints. The processed PPG signal is then wirelessly transmitted via the HC-05 Bluetooth module to an Android smartphone. Data is sent every 5 ms, allowing real-time and practical heart rate monitoring through the mobile app.

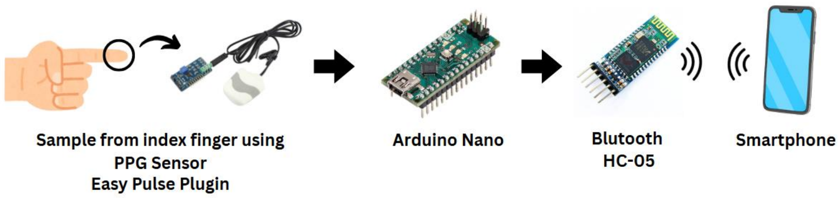

Figure 2. Structure of the PPG Measurement System[13]

Recordings were conducted for 2 minutes per patient across 216 individuals. There were no specific inclusion or exclusion criteria applied in selecting the patients; the data were collected randomly from available patients. The PPG signal was segmented into one-second intervals, each representing a single pulse and consisting of 120 data points, resulting in a total of 30,694 samples. Of these, 21,884 samples were from hypertensive patients, 5,766 from prehypertensive patients, and 3,045 from individuals with normal blood pressure.

As illustrated in Figure 3, the three blood pressure categories have clear visual differences. In normal patients, as shown in Figure 3a, the signal descends gradually. In prehypertensive individuals, as illustrated in Figure 3b, the signal drops more quickly and then flattens, creating a slight notch. Meanwhile, in hypertensive patients, as can be seen in Figure 3c, the signal shows a sharper decline, making it steeper than the other two.

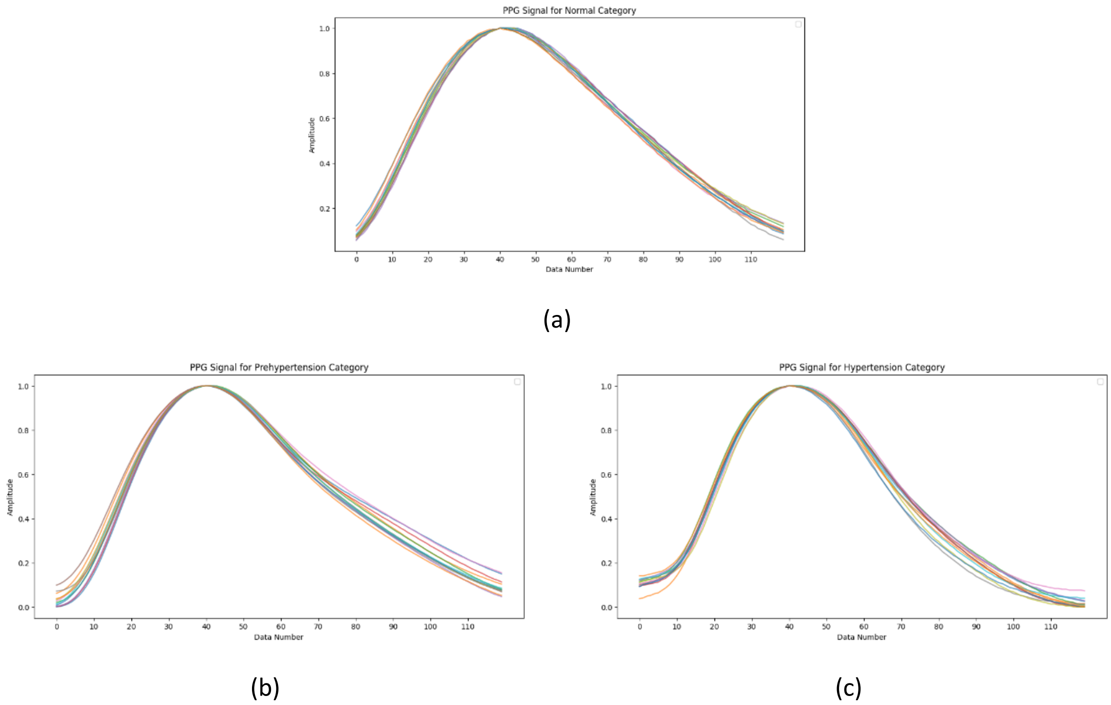

Figure 3. PPG data for each category (a) Normal, (b) Prehypertension, (c) Hypertension

Preprocessing

Before training, the dataset underwent several preprocessing steps to ensure data quality and consistency. First, missing values were identified and removed to prevent potential bias and learning disruption during model training. Duplicate records were also eliminated to avoid redundancy and ensure that each sample contributed uniquely to the learning process. After cleaning, the data were grouped into different experimental settings based on classification scenarios. The first experiment was a binary classification between normal vs. prehypertensive-hypertensive, to separate healthy people from those at risk or already having high blood pressure. The second experiment also used binary classification, between normal-prehypertension vs. hypertension, to tell apart people still in the early stage from those who already have hypertension. The third experiment used three classes—normal, prehypertension, and hypertension—for a more detailed classification that clearly separates all three conditions. This grouping allowed for comprehensive evaluation of model performance across different levels of diagnostic granularity. Table 1 presents the distribution of data across the experimental groups after preprocessing.

Table 1. The distribution of data in each Experiment after preprocessing

Experiment	Blood Pressure Category	Total Data
Experiment A	Normal	3,045
	Prehypertension - Hypertension	13,949
Experiment B	Normal - Prehypertension	8,811
	Hypertension	8,183
Experiment C	Normal	3,045
	Prehypertension	5,766
	Hypertension	8,183

Feature Selection

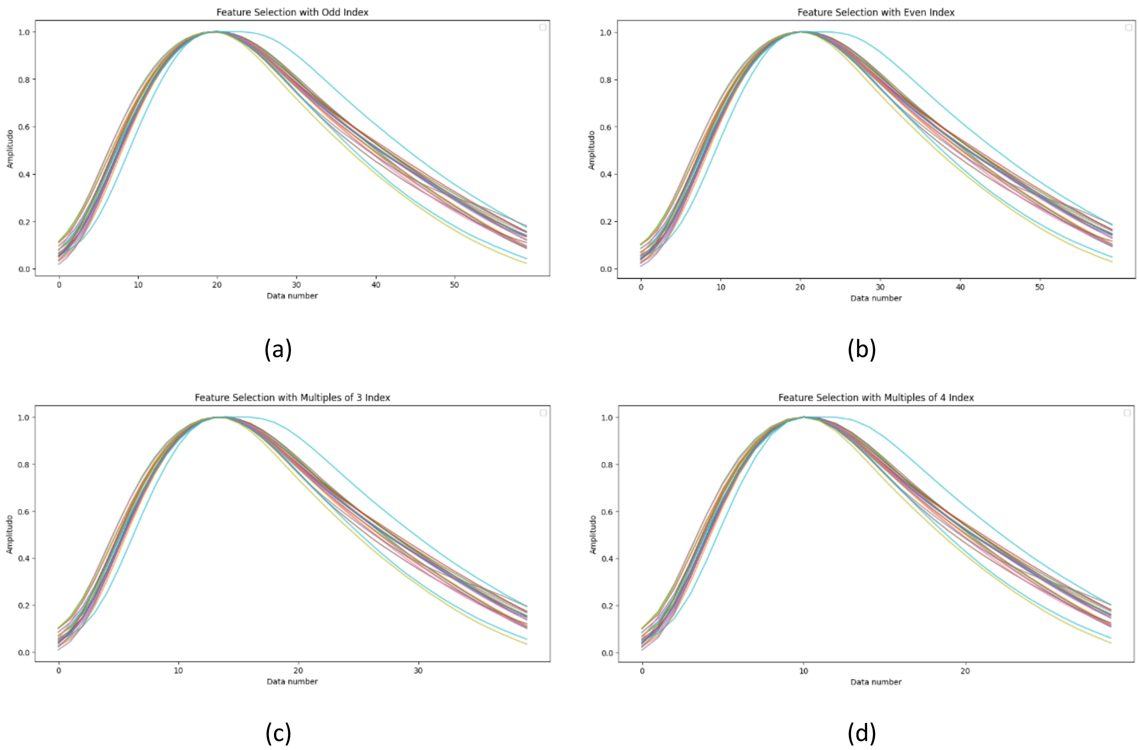

Figure 4. Visualization of the selected features (a) Odd Index, (b) Even Index, (c) Multiples of 3 Index, (d) Multiples of 4 Index

After the duplicate data removal process, feature selection was carried out. Four feature selection schemes were applied, based on positional indices: odd indices, even indices, multiples of 3, and multiples of 4. A visualization of the selected features for each scheme is shown in Figure 4. Visually, the overall shape of the signals did not change significantly; however, the number of features was substantially reduced. From the original 120 features, the count was reduced to 60 features for the odd and even index schemes (Figure 4a and 4b), 40 features for the multiple-of-3 scheme (Figure 4c), and 30 features for the multiple-of-4 scheme (Figure 4d).

CNN-SVM Architecture

In this study, a hybrid Convolutional Neural Network–Support Vector Machine (CNN-SVM) model was employed to perform hypertension classification based on PPG signals. The CNN component functions as a feature extractor, while the SVM acts as a classifier. The CNN architecture receives raw or pre-processed PPG data and extracts features, which are subsequently flattened and passed into the SVM for classification as shown in Figure 5.

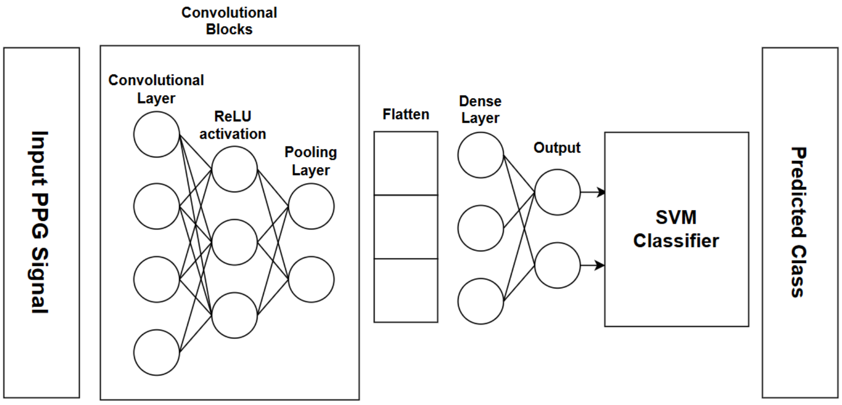

Figure 5. CNN-SVM Architecture

Rather than using a fixed CNN architecture, this study tested several experimental experiments involving different CNN configurations to identify the most effective setup for each classification scenario. These configurations varied in the number of convolutional layers, filter sizes, and kernel sizes. The convolutional layers function to extract spatial features from input data, while pooling layers reduce the dimensionality and help retain the most important information[14]. In each experiment, the CNN model that gave the highest validation accuracy was chosen as the best model for that experiment.

After feature extraction, the output from the CNN was flattened and passed to the SVM classifier. The SVM then performed classification by finding the optimal hyperplane that separates classes with the maximum margin [15]. To determine the most suitable SVM configuration, several kernel types were tested, including linear, polynomial, and radial basis function (RBF). Hyperparameters such as the regularization parameter C and kernel-specific parameters (such as gamma for RBF and degree for polynomial) were also tuned during experimentation. The best-performing SVM model for each experiment was selected based on validation performance metrics.

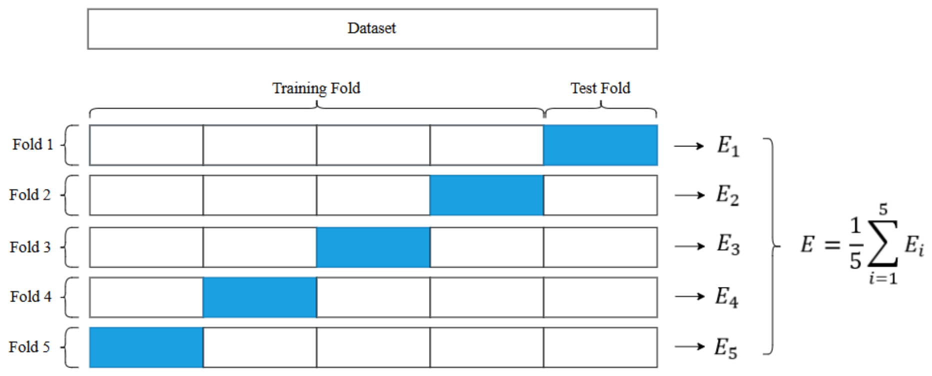

Figure 6. Illustration of 5-fold cross-validation

To validate the classification performance of the model, the k-fold cross-validation method is used with 5-fold. Each fold involved training the CNN feature extractor and then training an SVM model using the extracted features. In this method, the data is divided into 5 equally sized parts. In each iteration, one part is used as testing data, while the remaining parts are used as training data. This process is repeated 5 times so that each part is used as testing data once[16]. After all iterations are completed, the accuracy, precision, recall, and F1-Score of each fold is calculated and then averaged to obtain the final accuracy of the system. An illustration of 5-fold cross-validation is shown in Figure 6.

To evaluate the model's performance, this study uses several key evaluation metrics; accuracy, sensitivity, specificity, and F1-Score[17]. Accuracy indicates how often the model's prediction are correct for both positive and negative classes. Sensitivity measures the model's ability to correctly detect all positive cases of hypertension. On the other hand, specificity shows the model's ability to correctly identify negative cases, referring to individuals who do not have hypertension. F1-Score assesses the balance between the model's precision and its ability to comprehensively identify positive cases[18]. By using these metrics, the model's performance can be assessed more comprehensively, especially in medical applications.

RESULTS AND DISCUSSION

In this study, a series of experiments was carried out to find the best model for detecting hypertension using PPG signals. The experiment was divided into several stages. The first step was to test different numbers of CNN layers, consisting of 3, 4, and 5 to see which gave the best results. The best number of layers was chosen based on how well each performed during validation. Once the number of layers was decided, the next step was to test other parts of the model, such as the number of filters, kernel sizes, and stride values. Each configuration was evaluated using 5-fold cross-validation to identify the most effective architecture for extracting relevant features from the PPG data.

After obtaining the optimal CNN architecture, the features it produced were tested with different machine learning classifiers. The classifiers tested included Support Vector Machine (SVM), Logistic Regression (LR), Decision Tree (DT), Random Forest (RF), Gradient Boosting (GB), and Extreme Gradient Boosting (XGBoost). Among them, SVM gave the best results across all three experiment experiments and was therefore selected for further optimization. The performance of each classifier is shown in Table 2.

Table 2. Comparison of CNN performance with classifiers

Model	Experiment A	Experiment B	Experiment C
CNN-SVM	**92.10%**	**86.82%**	**79.72%**
CNN-LR	91.63%	86.43%	79.52%
CNN-DT	89.04%	85.81%	77.05%
CNN-RF	91.70%	86.49%	77.74%
CNN-GB	91.73%	86.42%	77.97%
CNN-XGBoost	91.53%	86.58%	79.19%

For the SVM, additional tuning was performed to find the best combination of parameters. This included experimenting with various kernel functions (linear, RBF, and polynomial), as well as adjusting hyperparameters such as the regularization parameter C, kernel coefficient gamma, and degree. The best-performing configuration was selected individually for each scenario, as the optimal setup varied depending on the data distribution and classification task. Table 3 summarizes the performance of different CNN architectures and classifier combinations across the three experiments.

Table 3. The optimal set of parameters for CNN-SVM after tuning

Experiment	CNN Layers	Filters	Kernel Size	Strides	SVM Kernel	C	Accuracy
Experiment A	4	16-128	7x7	3	linear	100	92.10%
Experiment B	4	16-128	7x7	2	linear	10	86.87%
Experiment C	4	16-128	9x9	2	linear	100	79.72%

In each experiment, the CNN architecture with 4 layers using 16, 32, 62, and 128 filters proved to be the most effective. This means that this architecture shows the consistency of effectiveness in extracting features from PPG signals. Additionally, all experiments employed a linear kernel in the SVM, indicating that the features extracted by the CNN were sufficiently linearly separable.

After identifying the best CNN architecture and classifier combination for each experiment, the next step involved applying feature selection to further enhance the model's performance and efficiency. Initially, a simple index-based feature selection method was applied, which reduced the input dimension by selecting features based on positional patterns such as odd-indexed, even-indexed, multiples of 3, and multiples of 4. The performance results of the feature selection can be found in Table 4.

Table 4. The performance of the feature selection methods across all experiments

Feature Selection Method	Experiment A	Experiment B	Experiment C
odd-indexed	92.81%	**88.38%**	81.64%
even-indexed	92.12%	86.34%	80.06%
Multiples of 3	**93.10%**	88.15%	**82.79%**
Multiples of 4	91.91%	85.44%	78.86%

In Experiment A and Experiment C, the feature selection method using multiples of 3 gave the highest accuracy, which were 93.10% and 82.79%, respectively. This shows that selecting features at every third index can still retain important information from the PPG signal, even with fewer input features. In Experiment B, the odd-indexed feature selection method produced the best accuracy at 88.38%. Meanwhile, in all experiments, the method using multiples of 4 consistently gave the lowest accuracy compared to the other methods. These results suggest that features at odd indexes or at multiples of 3 captured important patterns for differentiating blood pressure conditions. This shows that feature subsets taken periodically with a certain interval are able to capture quite representative information from the PPG signal.

Table 5. Performance metrics of the best parameter combination for each experiment

Experiment	Model	Accuracy	Specificity	Sensitivity	F1-Score
Experiment A	Multiples of 3 feature selection – CNN-SVM	93.10%	72.41%	97.17%	95.85%
Experiment B	Odd indexed feature selection – CNN-SVM	88.38%	87.85%	88.83%	88.04%
Experiment C	Multiples of 3 feature selection – CNN-SVM	82.79%	90.63%	82.79%	82.86%

Table 5 presents the performance evaluation of the CNN-SVM model using the best parameter combinations for each Experiment, based on accuracy, specificity, sensitivity, and F1-score. In Experiment A, which used features based on multiples of 3, the model achieved the highest accuracy of 93.10% and sensitivity of 97.17%, indicating strong performance in detecting positive cases. However, its specificity was relatively low at 71.41%, suggesting weaker performance in identifying negative cases. Experiment B, using odd-indexed features, showed a more balanced performance with 88.38% accuracy, 87.85% sensitivity, and 88.44% specificity, indicating consistent detection of both positive and negative cases, though slightly less effective than Experiment A in identifying positives. Experiment C, using features of multiples of 3, achieved the highest specificity of 90.63%, but with lower accuracy and sensitivity at 81.79%, implying that the model is better at avoiding false positives but may miss actual hypertension cases.

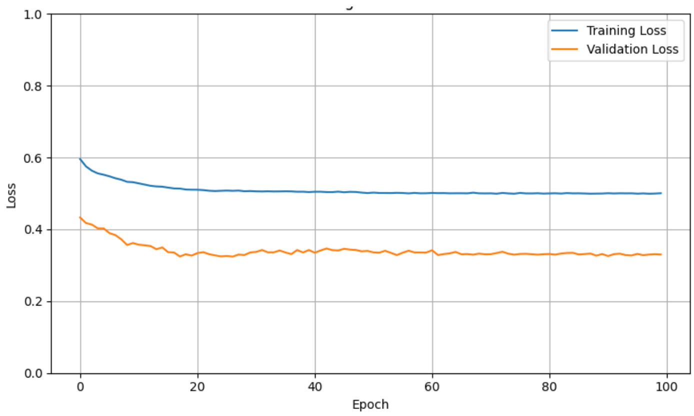

Figure 7. Learning Curve

Figure 7 shows the learning curve graph during the model training process. Both the training loss and validation loss show a decreasing trend as the number of epochs increases, indicating that the model is progressively learning the data patterns. The validation loss decreases during the early stages of training and begins to stabilize around the 40th epoch, suggesting that the model has reached a point of convergence. There is a slight gap between the training and validation loss, but this gap remains consistent and does not widen, indicating that the model's generalization ability is still within acceptable limits. Moreover, there are no prominent signs of overfitting, as the validation loss does not increase even though the training loss continues to gradually decline.

Besides using the index-based method, this study also tested other well-known techniques for reducing the number of features, such as Principal Component Analysis (PCA), Kernel PCA (KPCA), and Linear Discriminant Analysis (LDA). Table 6 presents the performance comparison between the index-based feature selection methods across other feature selection method.

Table 6. The performance of alternative feature selection and extraction methods

Feature Selection/Extraction Method	Experiment A	Experiment B	Experiment C
PCA	91.94%	85.87%	79.62%
KPCA	88.90%	80.65%	71.74%
LDA	82.08%	66.52%	54.91%

The results show that even though index-based feature selection is a simple approach that does not require complicated calculations like PCA, KPCA, or LDA, it can still give comparable or even better performance. This means that choosing features based on certain position, such as, odd, even, or multiple of certain indices can still keep essential information in the PPG signals for classification.

Table 7. Performance comparison of the proposed model with other studies

Experiment	Study	Dataset	Feature	Classifier	F1-Score
Experiment A	(Martinez-Ríos et al., 2022)[2]	PPG-BP	19 WST Features	SVM	76.00%
Experiment A	(Nuryani et al., 2024)[19]	MIMIC	PPG and ECG	CNN	95.27%
Experiment B	(Liang et al., 2018)[20]	MIMIC	CWT Scalogram	GoogleNet	82.95%
Experiment C	(Kuzmanov et al., 2022)[8]	UCI Machine Learning Dataset	PPG and ECG	CNN-LSTM	66.00%
Experiment A	This study	Local Dataset	Multiples of 3 indices from the PPG signal	CNN-SVM (This study)	95.85%
Experiment B	This study	Local Dataset	Odd indices from the PPG signal	CNN-SVM (This study)	88.04%
Experiment C	This study	Local Dataset	Multiples of 3 indices from the PPG signal	CNN-SVM (This study)	82.86%

Tabel 7 presents a performance comparison between the proposed model in this study and several previous works. Earlier studies employed feature extraction methods, such as Wavelet Scattering Transform (WST), Continuous Wavelet Transform (CWT), or a combination of PPG and ECG signals. In contrast, this study applied a simple feature selection method based on positional indices using a PPG signal from a local dataset. Nevertheless, the results are quite competitive, achieving F1-Scores of 95.85% in Experiment A, 88.04% in Experiment B, and 82.86% in Experiment C. These results indicate that the CNN-SVM approach, despite its simplicity in feature processing, delivers good performance compared with several previous studies that used public datasets and more complex feature extraction techniques.

CONCLUSION

This study developed a hypertension detection model based on PPG signals using a hybrid CNN-SVM approach, combined with index-based feature selection. The dataset used was local data collected from 216 patients at UNS Hospital, so the constructing model is more relevant and suitable for the Indonesian people with specific skin color. Evaluation results showed that the CNN-SVM model performed best in Experiment A, achieving an accuracy of 93.10% using features selected from multiples of 3. Moreover, in Experiment B, the highest accuracy of 88.38% was obtained using features from odd indices, while in Experiment C, the multiples of 3 index method produced the highest accuracy at 82.79%. These findings indicate that a simple feature reduction technique can effectively keep important information from PPG signals for classification. Compared to other more complex feature selection methods like PCA, KPCA, and LDA, the index-based approach showed competitive performance, and in some cases, it outperformed them, despite requiring less computational effort. However, it is important to acknowledge that index-based feature selection methods do not explicitly consider the statistical importance of each feature in relation to the target variable. This means that some important features might be missed. However, due to its simplicity and efficiency, this method remains worth considering, especially for developing systems that require fast and lightweight processing.

REFERENCES

1 Gupta, K., Bajaj, V., & Ansari, I. A. (2023). A support system for automatic classification of hypertension using BCG signals. *Expert Systems with Applications*, *214*, 119058. https://doi.org/https://doi.org/10.1016/j.eswa.2022.119058

2 Martinez-Ríos, E., Montesinos, L., & Alfaro-Ponce, M. (2022). A machine learning approach for hypertension detection based on photoplethysmography and clinical data. *Computers in Biology and Medicine*, *145*, 105479. https://doi.org/https://doi.org/10.1016/j.compbiomed.2022.105479

3 El-Hajj, C., & Kyriacou, P. A. (2020). A review of machine learning techniques in photoplethysmography for the non-invasive cuff-less measurement of blood pressure. *Biomedical Signal Processing and Control*, *58*, 101870. https://doi.org/https://doi.org/10.1016/j.bspc.2020.101870

4 Koparır, H. M., & Arslan, Ö. (2024). Cuffless blood pressure estimation from photoplethysmography using deep convolutional neural network and transfer learning. *Biomedical Signal Processing and Control*, *93*, 106194. https://doi.org/https://doi.org/10.1016/j.bspc.2024.106194

5 Chowdhury, A., Das, D., Eldaly, A. B. M., Cheung, R. C. C., & Chowdhury, M. H. (2024). Photoplethysmogram-based heart rate and blood pressure estimation with hypertension

classification. *IPEM-Translation*, *9*, 100024. https://doi.org/https://doi.org/10.1016/j.ipemt.2024.100024
6 Castaneda, D., & Ghamari, M. (n.d.). *A review on wearable photoplethysmography sensors and their potential future applications in health care*.
7 El-Dahshan, E.-S. A., Bassiouni, M. M., Khare, S. K., Tan, R.-S., & Rajendra Acharya, U. (2024). ExHyptNet: An explainable diagnosis of hypertension using EfficientNet with PPG signals. *Expert Systems with Applications*, *239*, 122388. https://doi.org/https://doi.org/10.1016/j.eswa.2023.122388
8 Kuzmanov, I., Bogdanova, A. M., Kostoska, M., & Ackovska, N. (2022). Fast Cuffless Blood Pressure Classification with ECG and PPG signals using CNN-LSTM Models in Emergency Medicine. *2022 45th Jubilee International Convention on Information, Communication and Electronic Technology (MIPRO)*, 362–367. https://doi.org/10.23919/MIPRO55190.2022.9803582
9 Rong, M., & Li, K. (2021). A multi-type features fusion neural network for blood pressure prediction based on photoplethysmography. *Biomedical Signal Processing and Control*, *68*, 102772. https://doi.org/https://doi.org/10.1016/j.bspc.2021.102772
10 Pankaj, Kumar, A., Komaragiri, R., & Kumar, M. (2023). A novel CS-NET architecture based on the unification of CNN, SVM and super-resolution spectrogram to monitor and classify blood pressure using photoplethysmography. *Computer Methods and Programs in Biomedicine*, *240*, 107716. https://doi.org/https://doi.org/10.1016/j.cmpb.2023.107716
11 Sinaki, F. Y., Ward, R., Abbott, D., Allen, J., Fletcher, R. R., Menon, C., & Elgendi, M. (2022). Ethnic disparities in publicly-available pulse oximetry databases. *Communications Medicine*, *2*(1), 59. https://doi.org/10.1038/s43856-022-00121-8
12 Botros, J., Mourad-Chehade, F., & Laplanche, D. (2022). CNN and SVM-Based Models for the Detection of Heart Failure Using Electrocardiogram Signals. *Sensors*, *22*(23). https://doi.org/10.3390/s22239190
13 Nuryani, N., Akshya, M. F., & Wiyono, N. (2023). Sistem Pengukuran Detak Jantung Menggunakan Arduino Dan Android Berbasis Fotopletismogram. *Indonesian Journal of Applied Physics*, *13*(1), 147–157.
14 Rastegar, S., Gholam Hosseini, H., & Lowe, A. (2023). Hybrid CNN-SVR Blood Pressure Estimation Model Using ECG and PPG Signals. *Sensors*, *23*(3). https://doi.org/10.3390/s23031259
15 Cervantes, J., Garcia-Lamont, F., Rodríguez-Mazahua, L., & Lopez, A. (2020). A comprehensive survey on support vector machine classification: Applications, challenges and trends. *Neurocomputing*, *408*, 189–215. https://doi.org/https://doi.org/10.1016/j.neucom.2019.10.118
16 Sontakke, S. A., Lohokare, J., Dani, R., & Shivagaje, P. (2019). Classification of Cardiotocography Signals Using Machine Learning. In K. Arai, S. Kapoor, & R. Bhatia (Eds.), *Intelligent Systems and Applications* (pp. 439–450). Springer International Publishing.
17 Gupta, K., Bajaj, V., Ansari, I. A., & Rajendra Acharya, U. (2022). Hyp-Net: Automated detection of hypertension using deep convolutional neural network and Gabor transform techniques with ballistocardiogram signals. *Biocybernetics and Biomedical Engineering*, *42*(3), 784–796. https://doi.org/https://doi.org/10.1016/j.bbe.2022.06.001
18 Hicks, S. A., Strümke, I., Thambawita, V., Hammou, M., Riegler, M. A., Halvorsen, P., & Parasa, S. (2022). On evaluation metrics for medical applications of artificial intelligence. *Scientific Reports*, *12*(1), 5979. https://doi.org/10.1038/s41598-022-09954-8
19 Nuryani, N., Utomo, T. P., Prabowo, N. A., Aripriharta, Yazid, M., & Yunianto, M. (2024). Advancing Non-Cuff Hypertension Detection: Leveraging 1D Convolutional Neural Network

and Time Domain Physiological Signals. *International Journal of Online and Biomedical Engineering (IJOE)*, *20*(05), 78–100. https://doi.org/10.3991/ijoe.v20i05.45547
20 Liang, Y., Chen, Z., Ward, R., & Elgendi, M. (2018). Photoplethysmography and Deep Learning: Enhancing Hypertension Risk Stratification. *Biosensors*, *8*(4). https://doi.org/10.3390/bios8040101