

Article

A Pulse-Feature LSTM Framework with Temporal Variance-Based Prediction Filtering for rPPG-Based Blood Pressure Estimation

Dogyun Park ^{1,†} , Hyojin Jo ^{1,†}  and Nam Kyu Kwon ^{2,*} 

¹ Department of Robotics and AI Convergence, Yeungnam University, Gyeongsan-si 38541, Republic of Korea; parkdk4634@yu.ac.kr (D.P.); hynn0254@yu.ac.kr (H.J.)

² Department of Electronic Engineering, Yeungnam University, Gyeongsan-si 38541, Republic of Korea

* Correspondence: namkyu@yu.ac.kr

† These authors contributed equally to this work.

Abstract

Non-contact blood pressure (BP) estimation from remote photoplethysmography (rPPG) can yield unstable window-level predictions. This study combines a pulse-feature long short-term memory (LSTM) estimator with temporal variance-based prediction filtering. Fifty-two features from the rPPG waveform and its first and second derivatives were arranged into sliding-window sequences for systolic BP (SBP) and diastolic BP (DBP) estimation. The filter selects locally stable predictions from temporally ordered prediction sequences without using ground-truth BP values. Performance was first assessed in 10 repeated subject-dependent randomized-block experiments. Filtering reduced the SBP mean absolute error (MAE) from 7.04 ± 1.52 to 5.34 ± 1.62 mmHg and the DBP MAE from 4.63 ± 1.32 to 3.07 ± 0.97 mmHg while retaining $76.39 \pm 7.59\%$ of predictions. A subject-specific mean predictor was also competitive under the subject-dependent setting and achieved lower DBP MAE at full coverage than the filtered LSTM. Chronological rolling-origin and leave-one-subject-out (LOSO) evaluations further showed that the improvement persisted for later intervals from the same subjects but not for unseen subjects. Because the filter operates only on model outputs, it requires neither additional trainable parameters nor model retraining. These findings show a low-overhead accuracy–coverage trade-off for within-subject rPPG-based BP estimation, while subject-independent generalization remains limited.

Keywords: remote photoplethysmography; blood pressure estimation; long short-term memory; temporal variance-based filtering; prediction-level post-processing; prediction coverage



Academic Editor: George Angelos Papadopoulos

Received: 20 July 2026

Revised: 25 August 2026

Accepted: 25 August 2026

Published: 27 August 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and

conditions of the [Creative Commons](https://creativecommons.org/licenses/by/4.0/)

[Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

1. Introduction

Blood pressure (BP) is a key physiological indicator used in various medical settings, including early diagnosis of cardiovascular diseases, chronic disease management, postoperative monitoring, and intensive care. In particular, changes in systolic blood pressure (SBP) and diastolic blood pressure (DBP) are closely associated with vascular elasticity, cardiac output, peripheral resistance, and autonomic nervous system responses. Therefore, it is important to monitor not only single-point BP measurements but also their temporal variations. However, cuff-based BP measurement methods, which are widely used in clinical and daily settings, provide measurements only at limited time points, cause user discomfort during repeated

measurements, and have limitations in long-term continuous monitoring or observing BP changes during natural daily activities [1,2]. For these reasons, there has been growing interest in non-invasive and non-contact BP monitoring technologies that can continuously estimate BP while reducing the burden of sensor attachment.

Photoplethysmography (PPG) is a physiological signal that measures changes in reflected or transmitted light from the skin caused by variations in blood volume, and it contains various types of information related to cardiovascular status [3]. Previous studies have extracted morphological features, temporal features, area-related features, and features based on the first and second derivatives from contact PPG signals, and have used them as inputs to machine learning or deep learning models for BP estimation [4–11]. Since the PPG waveform contains information related to pulse wave propagation, vascular elasticity, and reflected wave characteristics, it has been actively used as an input signal for estimating SBP and DBP [4,5,12,13]. However, contact PPG requires sensors to be attached to specific body sites, such as the finger, wrist, or earlobe. Long-term use may cause discomfort, variations in contact pressure, and motion-induced signal distortion. Therefore, for fully contactless physiological signal measurement and long-term monitoring in daily environments, new approaches that can replace or complement contact PPG are needed.

Remote photoplethysmography (rPPG) is a technique that recovers pulse wave signals associated with blood flow changes by analyzing subtle skin color variations in camera images [14–16]. Since rPPG enables physiological signals to be estimated from facial videos without requiring contact sensors, its potential has been investigated in various applications, including non-contact heart rate estimation, respiration monitoring, stress analysis, and BP estimation. In particular, camera-based measurement can reduce behavioral constraints on users and can utilize existing imaging devices, making it suitable for remote healthcare and non-contact physiological monitoring. However, compared with contact PPG, rPPG signals are more sensitive to measurement conditions such as illumination changes, facial motion, region of interest (ROI) selection, and video-related noise [17–20]. Consequently, rPPG-based BP estimation may suffer not only from degraded input signal quality but also from abnormal fluctuations in some of the continuous BP predictions generated by the model.

Recent studies on non-invasive BP estimation using optical pulse signals have explored both contact PPG-based and camera-based rPPG-based approaches. PPG-based studies have investigated handcrafted waveform features, derivative-based features, and deep learning models such as convolutional neural networks, recurrent neural networks, and attention-based architectures for SBP and DBP estimation [4,5,12,21–26]. In parallel, camera-based studies have attempted to estimate BP or BP-related physiological information directly from facial videos or rPPG signals, including facial video-based BP estimation, transdermal optical imaging, and transfer learning from PPG to rPPG [27–29]. These studies have mainly focused on which input representation to use, which neural network architecture to apply, or how long the input observation window should be. In addition, changing the observation time and estimation interval affects the temporal information and overlap of the input sequence, which can alter BP estimation performance even when the same features and model architecture are used. These studies have demonstrated the importance of input time-series construction in rPPG-based BP estimation.

However, from the perspective of continuous monitoring, the average estimation accuracy of a model alone is not sufficient. Even if a BP estimation model shows a low overall mean error, some predictions may deviate substantially at specific windows. In particular, rPPG-based BP estimation often generates predictions repeatedly from multiple overlapping windows. Therefore, the final output can be interpreted not as a single prediction but as a window-level prediction sequence arranged in temporal order. Since

adjacent windows share a considerable portion of the same or similar rPPG segments, predicted BP values are expected to show low temporal variance within a short time interval in which no large physiological change occurs. Conversely, if adjacent predictions fluctuate excessively within a short interval, the predictions in the corresponding segment may have reduced temporal stability due to degraded signal quality, motion, illumination changes, or unstable model responses.

From this perspective, rPPG-based BP estimation systems require not only input signal processing and model training but also a procedure for evaluating the temporal variability in predictions after model output. Conventional post-processing methods, such as simple moving average, low-pass filtering, and median filtering, can be considered to smooth prediction values. However, these methods transform all predictions uniformly, making it difficult to distinguish actual BP changes from unstable prediction errors. In addition, an abnormal prediction generated at a specific window may influence neighboring predictions. Although simple smoothing can reduce the visual variability in final predictions, it does not provide an explicit criterion for determining which predictions are temporally stable or which predictions should be excluded from evaluation. Therefore, for continuous rPPG-based BP estimation results, a temporal variance-based post-processing strategy is needed to select stable prediction segments, rather than simply correcting prediction values.

Selective prediction, also described as prediction with a reject option, evaluates estimation error together with the proportion of outputs retained [30,31]. In physiological-signal processing, PPG and rPPG signal-quality indices (SQIs) provide an input-level rejection strategy based on waveform morphology or template similarity [32,33]. Input-level SQIs assess signal quality before BP estimation, whereas output-level selection evaluates the temporal behavior of the prediction sequence after estimation.

Motivated by this distinction, this study proposes a prediction-level filtering method based on temporal variance for the window-level SBP and DBP predictions generated by the rPPG pulse-feature long short-term memory (LSTM) model. Rather than smoothing all predictions, the method arranges each subject's predictions in temporal order and examines the variance within short, overlapping chunks of consecutive predictions. Low-variance chunks contribute keep candidates, whereas high-variance chunks are examined through one-point removal analysis to identify removal candidates. Because each prediction may belong to several overlapping chunks, the keep and removal candidate counts are accumulated for each prediction, and a prediction is selected as temporally stable when its keep candidate count exceeds its removal candidate count. The procedure is applied separately to SBP and DBP, and only predictions selected for both are retained. The filtering process uses only the model outputs; ground-truth BP values are used only for performance evaluation. It introduces no additional trainable parameters and does not require modification or retraining of the estimator.

The main contributions of this study are twofold. First, a temporal variance-based filtering method is proposed for window-level SBP and DBP sequences by comparing accumulated keep and removal candidate counts across overlapping prediction chunks. The method uses only the model outputs and requires neither ground-truth BP values during selection nor model retraining. Second, the filtering method is evaluated under subject-dependent, chronological, and subject-independent protocols together with coverage-matched controls, input-level SQI rejection, and alternative regression backbones. Parameter sensitivity, per-subject behavior, agreement, and computational cost are also analyzed to clarify its accuracy–coverage trade-off and practical scope.

2. Materials and Methods

The overall framework of this study consists of two main components: an rPPG pulse-feature LSTM model for window-level SBP and DBP estimation, and temporal variance-based prediction filtering for selecting temporally stable predictions from the model outputs. In the first component, pulse-level features are extracted from rPPG signals and their first and second derivatives, arranged as sliding-window sequences, and used as inputs to the LSTM-based BP estimation model. In the second component, the resulting window-level predictions are rearranged into subject-specific prediction sequences, and temporally unstable predictions are selectively excluded using a temporal variance-based criterion.

Specifically, pulse-level features were extracted from the rPPG signal segments of each subject, and 52-dimensional feature vectors were constructed using the original rPPG signal and its first and second derivatives. The extracted feature vectors were arranged in temporal order to form a subject-specific rPPG feature sequence, which was then converted into input windows for the LSTM model using a sliding-window approach. Each input window was used as the basic unit for generating one SBP prediction and one DBP prediction, and the ground truth corresponding to each window was defined as the average of the SBP and DBP values included in that window. The predictions generated by the trained pulse-feature LSTM model for the test windows were defined as raw predictions, and these predictions were rearranged in temporal order for each subject to construct window-level prediction sequences. Finally, temporal variance-based filtering was applied to the same prediction sequences to select filtered predictions, and the performance of the raw and filtered predictions was compared. The overall procedure is illustrated in Figure 1.

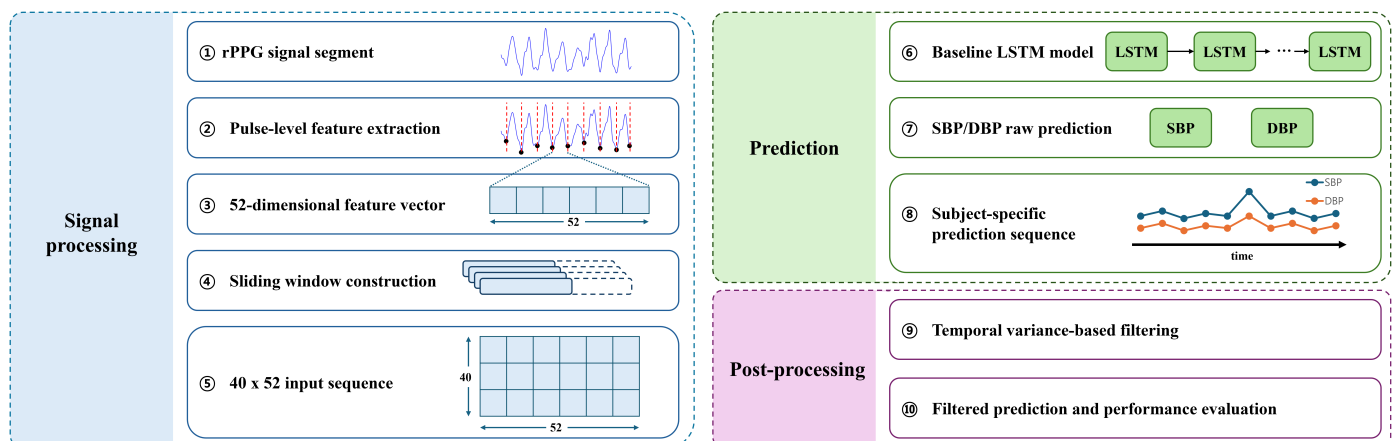


Figure 1. Overall framework of the proposed rPPG-based blood pressure estimation and post-processing pipeline. The framework consists of signal processing, prediction, and post-processing stages. Pulse-level features are extracted from rPPG signal segments and converted into sliding-window input sequences for the pulse-feature LSTM model. The resulting subject-specific SBP and DBP raw prediction sequences are then processed using temporal variance-based filtering to obtain the retained prediction subset for performance evaluation. Dashed borders delineate the signal-processing, prediction, and post-processing stages; dashed window outlines illustrate overlapping sliding windows, and point markers indicate waveform landmarks used for pulse-feature extraction.

2.1. Dataset and Preprocessing

This study used a public rPPG-based blood pressure dataset collected at Leipzig University Hospital [29,34,35]. The dataset was constructed from facial videos of patients, and RGB videos were acquired from regions of interest that included the forehead and cheek areas. The rPPG signals were extracted from the facial videos using the Plane-Orthogonal-to-Skin (POS) algorithm [16], and the corresponding SBP and DBP values were provided as ground truth for the same recording intervals [29,34]. The rPPG signals included in the

dataset were provided after basic signal processing. Specifically, a fourth-order Butterworth band-pass filter with a passband of 0.5–8.0 Hz was applied to the rPPG signals, and signal segments with a signal-to-noise ratio (SNR) below −7 dB were excluded from the analysis to remove low-quality signal intervals. The data used in this study contained 17 subjects and 7851 seven-second rPPG records after this screening. The preprocessed rPPG signals and their corresponding SBP and DBP values were used for feature extraction, BP estimation model training, and prediction-level post-processing evaluation.

Before feature extraction for BP estimation, normalization was applied at the rPPG signal segment level to reduce amplitude differences that may vary across subjects and recording intervals. The normalization was not performed independently for each pulse, but rather over the entire rPPG signal segment before pulse segmentation. This procedure was intended to reduce the excessive influence of absolute amplitude differences caused by recording conditions, ROI selection, and skin reflectance on the feature extraction process. The normalization of each rPPG signal segment was performed using min–max scaling with a small positive offset:

$$\tilde{r}_k[n] = \frac{r_k[n] - \min_x r_k[x] + \epsilon}{\max_x r_k[x] - \min_x r_k[x] + \epsilon}. \quad (1)$$

Here, $r_k[n]$ denotes the n -th sample of the k -th rPPG signal segment, and $\tilde{r}_k[n]$ denotes the normalized rPPG sample. $\min_x r_k[x]$ and $\max_x r_k[x]$ represent the minimum and maximum values, respectively, computed over all sample indices x included in the k -th rPPG signal segment. ϵ is a small positive offset and was set to $\epsilon = 10^{-5}$ in this study. The offset was added to both the numerator and denominator so that the minimum normalized sample remained slightly above zero, thereby avoiding exact-zero values in subsequent ratio-based feature calculations.

Because feature extraction was performed after segment-level normalization, temporal features such as pulse interval, waveform width, width ratio, and landmark timing were preserved. Features related to amplitude, area, and derivative amplitude represented relative waveform characteristics within each normalized segment rather than absolute optical intensity. Consequently, absolute optical-intensity information and between-segment amplitude-scale differences were not preserved by the segment-level normalization. Since the extracted features have different physical meanings and numerical ranges, feature-level min–max normalization was additionally applied before using them as inputs to the LSTM model. The target SBP and DBP values were also used in a normalized range during model training, and the normalized predictions generated by the model were restored to mmHg units during the performance evaluation stage.

2.2. rPPG Feature Extraction

In this study, pulse-level features for BP estimation were extracted from the normalized rPPG signal segments. Since an rPPG pulse reflects peripheral blood volume changes associated with cardiac activity, its morphology, temporal structure, area, and rising and falling characteristics may contain information related to BP variation. Accordingly, a total of 52 pulse-level morphological features were constructed for each pulse using the original rPPG waveform, its first derivative, and its second derivative, following the PPG feature extraction strategy reported in a previous PPG-based BP estimation study [12]. The original waveform, first derivative, and second derivative were used to describe pulse morphology and timing, rate of change, and waveform curvature, respectively. The main waveform landmarks and feature composition extracted from each signal are illustrated in Figure 2.

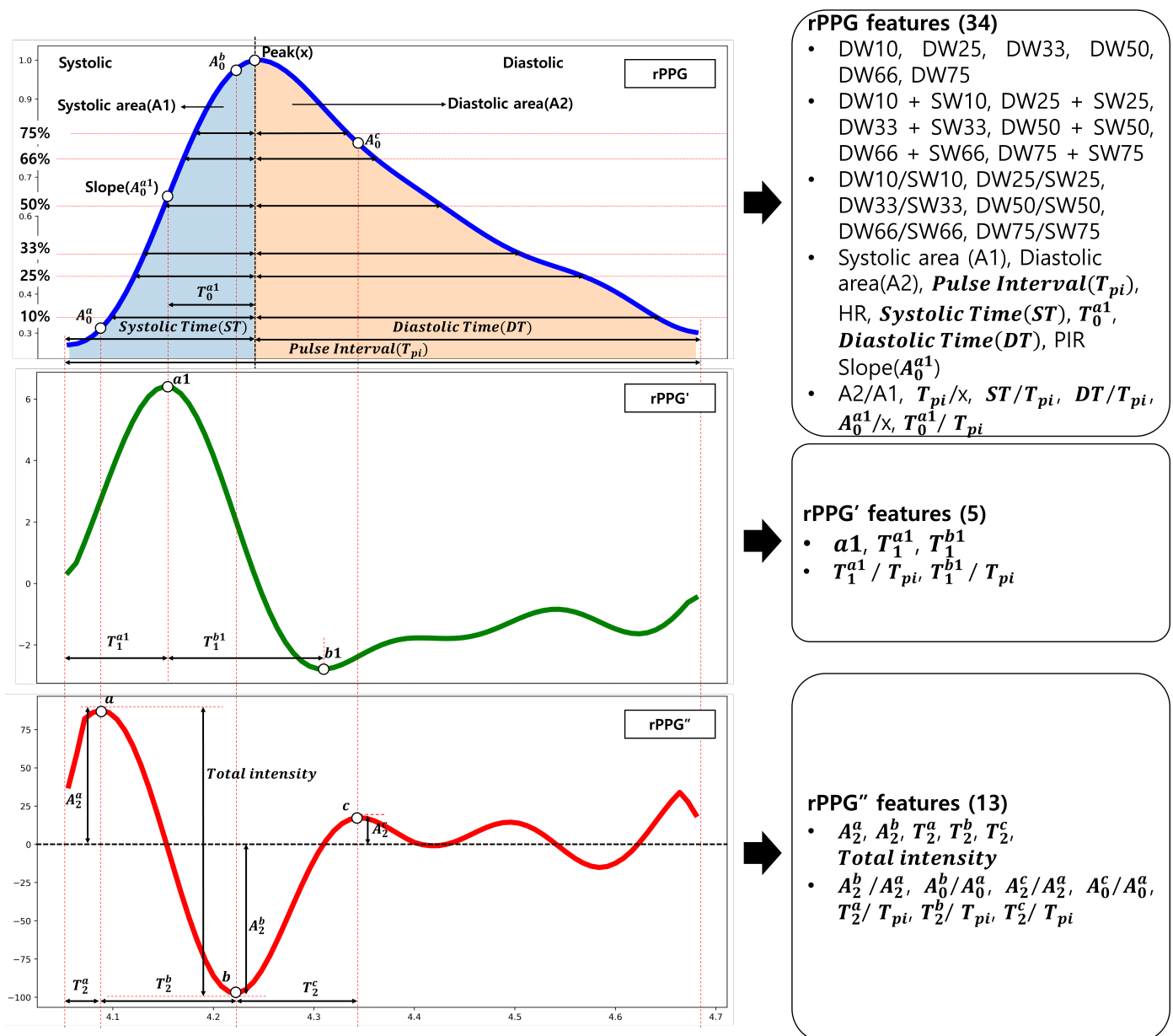


Figure 2. Illustration of pulse-level feature extraction from the normalized rPPG waveform and its first and second derivatives. The 52-dimensional feature vector consists of 34 features extracted from the original rPPG waveform, 5 features extracted from the first derivative, and 13 features extracted from the second derivative.

After detecting valleys in the normalized rPPG signal segment, the interval between two adjacent valleys was defined as one pulse. From each pulse, features representing the morphology and temporal structure of the rPPG waveform were extracted, including the systolic peak, pulse interval, systolic upstroke time, diastolic time, pulse area, and PPG intensity ratio. In addition, to quantify the rising and falling characteristics of each pulse, multiple amplitude levels were defined based on the amplitude difference between the systolic peak and the pulse minimum, and the systolic width, diastolic width, and width ratio at each level were calculated. To further incorporate information on the rate of change and curvature of the pulse waveform, additional features were extracted from the first and second derivative signals. From the first derivative signal, peak amplitude and timing information related to the rapid upstroke of the pulse were used. From the second derivative signal, the amplitudes, temporal locations, and ratios of major peaks and valleys

were used. Through this process, not only the overall morphology of the rPPG pulse but also its rising velocity and curvature changes were represented.

Finally, each valid pulse was represented as a 52-dimensional feature vector. This feature vector consisted of 34 features derived from the original rPPG waveform, 5 features derived from the first derivative, and 13 features derived from the second derivative. During feature extraction, pulses for which the systolic peak, valley, width crossing points, or major peaks and valleys in the derivative signals could not be reliably detected were excluded. This was performed to prevent abnormal features computed from incomplete pulses from affecting the subsequent LSTM model training and prediction-level filtering evaluation. Across the 17 subjects, 111,868 valley-to-valley candidate pulse intervals were detected. Among them, 27,135 intervals yielded complete 52-dimensional feature vectors, whereas 84,733 intervals (75.74%) were excluded because one or more landmarks required for feature extraction could not be detected. Because only pulses with complete landmark information were retained for feature extraction, the subsequent window construction and SQI evaluation were performed on a subset that had already been screened for landmark detectability. The reference SBP and DBP values corresponding to the 27,135 complete feature vectors ranged from 92.86 to 165.00 mmHg and from 41.00 to 85.00 mmHg, respectively.

2.3. Window-Based Sequence Construction

The pulse-level feature vectors obtained through rPPG feature extraction form a feature sequence arranged in temporal order. In this study, a sliding-window approach was applied to use this feature sequence as the input to the LSTM model. Each feature vector consists of 52-dimensional features extracted from one valid rPPG pulse, and each input window contains multiple temporally consecutive feature vectors. For subject u , the feature sequence arranged in temporal order is expressed as follows:

$$\mathbf{F}_u = [\mathbf{f}_{u,1}, \mathbf{f}_{u,2}, \dots, \mathbf{f}_{u,N_u}], \quad \mathbf{f}_{u,t} \in \mathbb{R}^{52}. \quad (2)$$

Here, $\mathbf{f}_{u,t}$ denotes the 52-dimensional feature vector extracted from the t -th pulse of subject u , and N_u denotes the total number of available feature vectors for that subject.

In this study, the window size W denotes the number of consecutive feature vectors included in one input window, which corresponds to the number of time steps in the LSTM input structure. In the remainder of this paper, the length of the input sequence is referred to as the window size. The stride S is defined as the interval, in units of feature vectors, between the starting positions of two adjacent input windows. In this study, $W = 40$ and $S = 10$ were used, resulting in a 75% overlap between adjacent windows. Based on the detected pulse intervals, a 40-pulse window and a 10-pulse stride corresponded to cumulative pulse durations of 30.26 ± 6.22 s and 7.56 ± 1.64 s, respectively. These settings were fixed for all subsequent analyses. Letting k denote the window index, the starting index a_k of the k -th input window is defined as follows:

$$a_k = 1 + kS. \quad (3)$$

Here, $k = 0, 1, \dots, \lfloor (N_u - W) / S \rfloor$, so the first window begins at $a_0 = 1$. Accordingly, the k -th input window of subject u is constructed as follows:

$$\mathbf{X}_{u,k} = [\mathbf{f}_{u,a_k}, \mathbf{f}_{u,a_k+1}, \dots, \mathbf{f}_{u,a_k+W-1}], \quad \mathbf{X}_{u,k} \in \mathbb{R}^{W \times 52}. \quad (4)$$

Here, $\mathbf{X}_{u,k}$ denotes the k -th window sequence used as the input to the LSTM model. Each input window consists of W consecutive feature vectors, and each feature vector is 52-dimensional. Therefore, under the settings used in this study, each input window is

represented as a 40×52 sequence. A window is generated only when $a_k + W - 1 \leq N_u$ is satisfied. The BP label corresponding to each input window was defined as the average of the SBP and DBP values included in that window:

$$\mathbf{y}_{u,k} = [y_{u,k}^{\text{SBP}}, y_{u,k}^{\text{DBP}}] = \frac{1}{W} \sum_{t=a_k}^{a_k+W-1} [y_{u,t}^{\text{SBP}}, y_{u,t}^{\text{DBP}}], \quad \mathbf{y}_{u,k} \in \mathbb{R}^2. \quad (5)$$

Here, $\mathbf{y}_{u,k}$ denotes the window-level BP label corresponding to the k -th input window. $y_{u,t}^{\text{SBP}}$ and $y_{u,t}^{\text{DBP}}$ denote the SBP and DBP values corresponding to the t -th feature vector of subject u , respectively. Thus, the LSTM model was trained to simultaneously predict one SBP value and one DBP value from each input window.

2.4. rPPG Pulse-Feature LSTM Model for Blood Pressure Estimation

In this study, an rPPG pulse-feature LSTM model was constructed to simultaneously estimate SBP and DBP using the feature sequences generated through window-based sequence construction. The model uses 52-dimensional pulse-level features extracted from the original rPPG waveform and its first and second derivatives, and generates one window-level SBP prediction and one window-level DBP prediction from each input sequence. These window-level predictions are subsequently rearranged into subject-specific prediction sequences and used as the input to temporal variance-based prediction filtering. The input to the model is an rPPG feature sequence corresponding to one input window. As described in the previous section, each input window consists of 40 consecutive feature vectors, and each feature vector is 52-dimensional. Thus, one sample is represented as a 40×52 sequence, and the actual input shape for batch-based model training is (batch size, 40, 52). Based on this input sequence, the model outputs one SBP value and one DBP value corresponding to the window. The overall architecture of the rPPG pulse-feature LSTM model is shown in Figure 3.

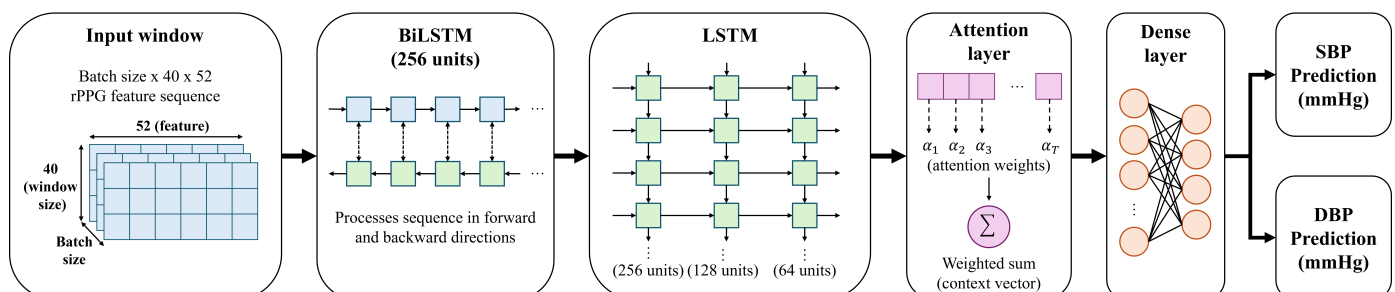


Figure 3. Architecture of the rPPG pulse-feature LSTM-based blood pressure estimation model. The model receives a 40×52 rPPG feature sequence as input and processes it through a bidirectional LSTM layer, stacked LSTM layers, an attention layer, and a dense layer to generate window-level SBP and DBP predictions. Colors distinguish the major network components, horizontal arrows indicate information flow between components, and vertical arrows within the recurrent blocks represent temporal recurrence.

The network first processes the input sequence in both forward and backward directions using a bidirectional LSTM (BiLSTM) layer [36,37]. This allows the model to incorporate information from both earlier and later pulses within the window. Subsequently, multiple unidirectional LSTM layers were sequentially stacked to progressively extract temporal patterns in the sequence. In the current implementation, LSTM layers with 256, 128, and 64 units were used after the BiLSTM layer. Dropout with a rate of 0.3 was applied after the BiLSTM layer and each subsequent LSTM layer to reduce overfitting [38]. The output of the LSTM stack was then passed to an attention layer. The attention layer computes weights that indicate the contribution of each sequence position within the window to the final BP estimation and summarizes the entire sequence into a context vector

using these weights [39]. Unlike using only the last LSTM output or averaging the entire sequence, the attention mechanism can assign greater importance to relatively informative time positions within the window. The context vector generated by the attention layer is passed through a dense layer and then to the final output layer. The final output layer generates two continuous values, corresponding to the window-level SBP prediction and DBP prediction. During training, normalized SBP and DBP values were used as targets, and during performance evaluation, the model outputs were restored to mmHg units.

For model training, mean squared error was used as the loss function, and Adam was used as the optimizer [40]. The learning rate was set to 0.0001, the batch size was 16, and the maximum number of training epochs was set to 300. During training, the model with the lowest validation loss was saved and used to generate test predictions. After training, the model predicted SBP and DBP for each input window in the test set. The predictions generated by the model before post-processing were defined as raw predictions. These raw predictions were then arranged in temporal order for each subject to construct prediction sequences, which were used as the input to the proposed temporal variance-based filtering. The estimator architecture and input settings were held fixed in all post-processing experiments.

2.5. Subject-Specific Prediction Sequence Construction

The proposed temporal variance-based filtering does not evaluate each window prediction independently; instead, it uses the variability in consecutive predictions arranged in temporal order. Therefore, the raw predictions generated by the LSTM model for each input window in the test set must first be reconstructed into subject-specific prediction sequences.

The trained pulse-feature LSTM model outputs one SBP prediction and one DBP prediction for each input window in the test set. Each prediction is associated with the temporal position of the corresponding window in the rPPG feature sequence. When generating the test windows, the subject ID, the block group to which the window belongs, and the start and end positions of the window were stored together. These values were used to arrange the model outputs in temporal order for each subject. For subject u , the window-level raw predictions generated from the test set can be arranged in temporal order to construct the following prediction sequence:

$$\hat{\mathbf{P}}_u = [\hat{\mathbf{y}}_{u,1}, \hat{\mathbf{y}}_{u,2}, \dots, \hat{\mathbf{y}}_{u,M_u}]. \quad (6)$$

Here, $\hat{\mathbf{P}}_u$ denotes the window-level prediction sequence of subject u , and M_u denotes the number of test predictions generated for that subject. $\hat{\mathbf{y}}_{u,j}$ denotes the raw prediction for the j -th test window and consists of the SBP and DBP predictions:

$$\hat{\mathbf{y}}_{u,j} = [\hat{y}_{u,j}^{\text{SBP}}, \hat{y}_{u,j}^{\text{DBP}}]. \quad (7)$$

Here, $\hat{y}_{u,j}^{\text{SBP}}$ and $\hat{y}_{u,j}^{\text{DBP}}$ denote the SBP and DBP predictions for the j -th window, respectively. These predictions are the raw predictions before post-processing and are used as the input to the subsequent temporal variance-based filtering.

2.6. Temporal Variance-Based Prediction Filtering

Temporal variance-based filtering was applied to the subject-specific window-level prediction sequences. The proposed method is based on the assumption that if consecutive BP predictions fluctuate excessively within a short interval, the corresponding segment is likely to contain temporally unstable predictions. In general, BP is not expected to change abruptly by a large amount over a very short time interval. Therefore, when temporally adjacent window-level predictions show a large variance, the corresponding segment can

be regarded as an unstable prediction interval. Let $q \in \{\text{SBP}, \text{DBP}\}$ denote the BP indicator. For subject u , the prediction sequence for BP indicator q can be expressed as follows:

$$\hat{\mathbf{p}}_u^q = [\hat{y}_{u,1}^q, \hat{y}_{u,2}^q, \dots, \hat{y}_{u,M_u}^q]. \quad (8)$$

Here, $\hat{\mathbf{p}}_u^q$ denotes the sequence of window-level raw predictions arranged in temporal order, and M_u denotes the number of test predictions generated for the subject. Filtering was performed using predictions restored to mmHg units rather than normalized values.

The overall procedure of the proposed temporal variance-based prediction filtering method is illustrated in Figure 4. Overlapping chunks are generated from the temporally ordered prediction sequence, and keep candidates and removal candidates are determined based on the variance of each chunk. The candidate counts accumulated across all chunks are then compared to select stable predictions. Finally, the intersection of the stable prediction indices selected independently for SBP and DBP is used as the final filtered prediction set.

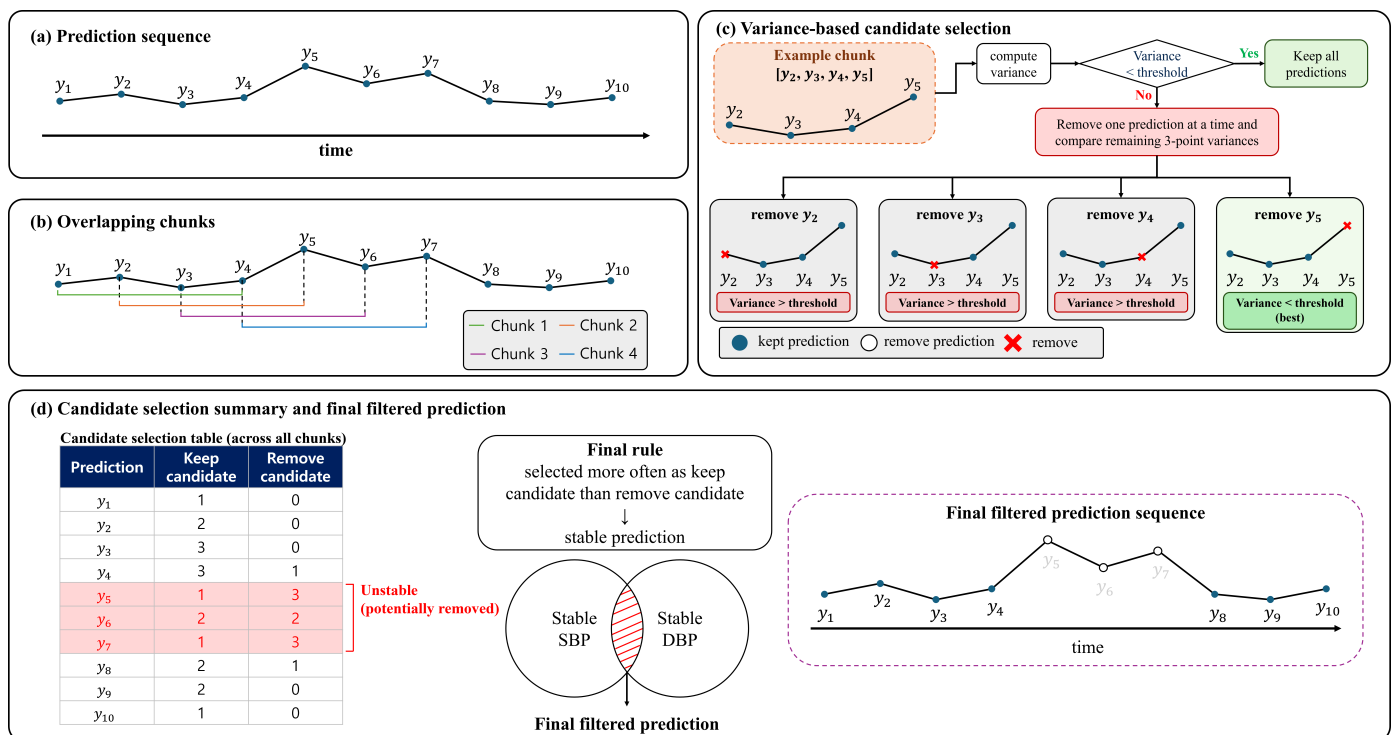


Figure 4. Framework of the proposed temporal variance-based prediction filtering method. (a) A subject-specific prediction sequence is arranged in temporal order. (b) Overlapping chunks of size C are generated with a stride of one prediction. (c) The variance of each chunk is evaluated, and one-point removal analysis is performed when the chunk variance exceeds the threshold to identify keep and removal candidates. (d) Candidate counts are accumulated across all chunks, and predictions selected as stable for both SBP and DBP are included in the final filtered predictions.

First, C consecutive predictions in the prediction sequence were defined as one chunk. The chunk size was set to $C = 4$, so one-point removal analysis evaluates the remaining three predictions when a candidate is excluded. The l -th chunk is expressed as follows:

$$\mathbf{c}_{u,l}^q = [\hat{y}_{u,l}^q, \hat{y}_{u,l+1}^q, \dots, \hat{y}_{u,l+C-1}^q]. \quad (9)$$

Since the chunks are generated by moving one prediction at a time along the prediction sequence, a single window-level prediction can be included in multiple overlapping chunks.

For each chunk, the prediction variance was calculated. Letting $\bar{y}_{u,l}^q$ denote the mean of the l -th chunk, the chunk variance is defined as follows:

$$V_{u,l}^q = \frac{1}{C} \sum_{m=0}^{C-1} \left(\hat{y}_{u,l+m}^q - \bar{y}_{u,l}^q \right)^2. \quad (10)$$

To keep the stability assessment local while allowing one-point removal to leave three predictions for reevaluation, the chunk size was set to $C = 4$. The variance threshold was set to $\tau_q = 10 \text{ (mmHg)}^2$ for both SBP and DBP, corresponding to a within-chunk standard deviation of approximately 3.16 mmHg and serving as the local stability criterion in the primary analysis. If $V_{u,l}^q < \tau_q$, the predictions included in the chunk were considered to remain stable within a short interval, and all predictions in the chunk were recorded as keep candidates. In contrast, if $V_{u,l}^q \geq \tau_q$, the chunk was considered likely to contain a transiently unstable prediction. In this case, the entire chunk was not immediately discarded. Instead, each prediction in the chunk was removed one at a time, and the variance of the remaining $C - 1$ predictions was recalculated. To compute the variance after removing the r -th prediction, the reduced chunk obtained by excluding the prediction corresponding to local index r from the l -th chunk is defined as follows:

$$\mathbf{c}_{u,l}^{q,(-r)} = \left[\hat{y}_{u,l+m}^q \mid m = 0, 1, \dots, C-1, m \neq r \right], \quad r = 0, 1, \dots, C-1. \quad (11)$$

Here, $\mathbf{c}_{u,l}^{q,(-r)}$ denotes the reduced chunk consisting of the remaining $C - 1$ predictions after excluding the prediction at the r -th position in the l -th chunk. In other words, the removed prediction is determined not by the prediction value itself, but by the local index r within the chunk. The r -th local index corresponds to the $(l + r)$ -th prediction in the entire prediction sequence. Letting $\bar{y}_{u,l}^{q,(-r)}$ denote the mean of the reduced chunk, the variance after removing the r -th prediction is calculated as follows:

$$\bar{y}_{u,l}^{q,(-r)} = \frac{1}{C-1} \sum_{\substack{m=0 \\ m \neq r}}^{C-1} \hat{y}_{u,l+m}^q \quad (12)$$

$$V_{u,l}^{q,(-r)} = \frac{1}{C-1} \sum_{\substack{m=0 \\ m \neq r}}^{C-1} \left(\hat{y}_{u,l+m}^q - \bar{y}_{u,l}^{q,(-r)} \right)^2. \quad (13)$$

The same threshold was applied before and after one-point removal because the remaining predictions were evaluated using the same operational definition of local stability; the two variances were not compared as sample-size-adjusted statistical estimators.

Since the chunk size was set to $C = 4$, the mean and variance were recalculated using the remaining three predictions after one prediction was removed. If the variance of the remaining predictions decreased to or below the threshold after removing a specific prediction, the removed prediction was recorded as a removal candidate for that chunk. The set of feasible removal candidates is defined as follows:

$$\mathcal{R}_{u,l}^q = \left\{ r \in \{0, 1, \dots, C-1\} \mid V_{u,l}^{q,(-r)} \leq \tau_q \right\}. \quad (14)$$

If $\mathcal{R}_{u,l}^q$ is not empty, the removal candidate is selected as follows:

$$r^* = \arg \min_{r \in \mathcal{R}_{u,l}^q} V_{u,l}^{q,(-r)}. \quad (15)$$

Here, r^* denotes the local index selected as the removal candidate in the chunk. If $\mathcal{R}_{u,l}^q$ is not empty, $\hat{y}_{u,l+r^*}^q$ is recorded as a removal candidate and the remaining predictions are recorded as keep candidates. If $\mathcal{R}_{u,l}^q$ is empty, no single prediction can reduce the remaining variance to the stability threshold; therefore, all predictions in the chunk are recorded as removal candidates. This chunk-level decision was performed independently for the SBP prediction sequence and the DBP prediction sequence. Accordingly, the keep candidate count and removal candidate count were accumulated for each prediction. Let $K_{u,j}^q$ and $R_{u,j}^q$ denote the keep candidate count and removal candidate count of the j -th prediction, respectively. The prediction was selected as a stable prediction when the following condition was satisfied:

$$K_{u,j}^q > R_{u,j}^q. \quad (16)$$

That is, a prediction was regarded as temporally stable when it was selected as a keep candidate more often than as a removal candidate across the chunks in which it was included.

Finally, the stable predictions selected for SBP and DBP were jointly considered. Let $\mathcal{I}_u^{\text{SBP}}$ and $\mathcal{I}_u^{\text{DBP}}$ denote the sets of stable prediction indices selected based on SBP and DBP, respectively. The final filtered prediction index set is defined as follows:

$$\mathcal{I}_u^{\text{final}} = \mathcal{I}_u^{\text{SBP}} \cap \mathcal{I}_u^{\text{DBP}}. \quad (17)$$

Thus, the final filtered predictions consist only of window-level predictions judged to be stable for both SBP and DBP:

$$\hat{\mathbf{p}}_u^{\text{filtered}} = \left\{ \hat{\mathbf{y}}_{u,j} \mid j \in \mathcal{I}_u^{\text{final}} \right\}. \quad (18)$$

Here, $\hat{\mathbf{y}}_{u,j}$ denotes the prediction vector containing both the SBP and DBP raw predictions for the j -th window. In other words, if a window-level prediction was judged to be stable for only SBP or only DBP, it was not included in the final filtered prediction set.

The proposed filtering process does not use ground-truth BP values. Whether a prediction is kept or excluded is determined solely based on the temporal variability in the SBP and DBP prediction sequences generated by the model. Ground-truth values are used only in the evaluation stage after post-processing is completed to compare the performance of raw and filtered predictions. Therefore, the proposed method can be applied in practical settings using only continuously generated window-level predictions. Its output is a binary keep-or-discard decision rather than a continuous reliability score or calibrated confidence value.

3. Results

3.1. Experimental Setup and Evaluation Protocol

3.1.1. Primary Subject-Dependent Protocol

To evaluate the effect of temporal variance-based filtering, raw predictions generated by the trained pulse-feature LSTM model, predictions processed by conventional smoothing-based post-processing methods, and predictions selected by the proposed filtering method were compared. The experiments were conducted by dividing the rPPG feature sequence of each subject into blocks and constructing training, validation, and test sets. The entire feature sequence of each subject was divided into 20 blocks in temporal order. The block indices were then randomly shuffled, and 16 blocks were assigned to the training set, 2 blocks to the validation set, and 2 blocks to the test set. This block-based partitioning was designed to separate the training, validation, and evaluation intervals within the same subject while preventing feature sequences from different sets from directly

overlapping during sliding-window generation. Temporally consecutive blocks assigned to each set were connected as one segment. Sliding windows were then generated only within the segments belonging to each set, and no window was generated across the boundaries between the training, validation, and test sets. Therefore, the test predictions consisted only of window-level predictions generated from test blocks that were not used during model training. The normalization parameters for the features and targets were computed only from the training set. The same normalization criteria computed from the training set were then applied to the validation and test sets. This procedure was used to prevent information from the validation and test data from being reflected in the preprocessing stage of training. The model was trained using the training set, and the model parameters with the lowest validation loss were saved and used to generate test predictions. Because the training and test sets contained different temporal blocks from the same subjects, this protocol is referred to as subject-dependent evaluation.

Because the proposed filtering method uses the temporal variability in subject-specific prediction sequences, subjects with too few test predictions are not suitable for stable evaluation. Since the chunk size was set to $C = 4$, at least four test predictions are required to construct one prediction chunk. Therefore, subjects with fewer than C test predictions generated from the test blocks were excluded from the final evaluation. After training, the model predicted SBP and DBP for each input window in the test set, and these predictions were defined as raw predictions before post-processing. The raw predictions were then arranged in temporal order for each subject, and temporal variance-based filtering was applied to the same prediction sequence to construct the filtered predictions. Thus, the performance difference between the raw and filtered predictions can be interpreted as the effect of applying the post-processing step to the same model outputs, rather than as a result of differences in model architecture or training conditions. Across the 10 repetitions, 13–15 subjects contributed evaluable test prediction sequences, and 164.2 ± 4.7 test predictions were evaluated per repetition. Across all 10 repetitions, 15 of the 17 subjects contributed at least one evaluable test prediction sequence.

Performance evaluation was conducted separately for SBP and DBP. BP estimation accuracy was evaluated using MAE, and the variability in the prediction error was evaluated using error SD. Since the proposed filtering method selects a subset of the raw predictions, the proportion of predictions finally retained after filtering among all raw predictions was calculated as the prediction retention ratio. To compare the proposed method with conventional post-processing approaches, three smoothing-based methods were additionally applied to the same subject-specific raw prediction sequences: simple moving average, median filtering, and low-pass filtering. The simple moving average and median filtering were applied using the same window length as the proposed chunk size, $C = 4$. The low-pass filtering was implemented using a second-order Butterworth low-pass filter with a cutoff frequency of 0.10 cycles per prediction. These conventional smoothing-based methods were applied independently to the SBP and DBP prediction sequences arranged in temporal order for each subject. Unlike the proposed temporal variance-based filtering, the conventional methods modify the prediction values while retaining all predictions; therefore, their retention ratio was set to 100%. The proposed method, in contrast, does not modify the prediction values but selects a subset of temporally stable predictions. To account for variability caused by random block partitioning, the entire experiment was repeated 10 times using different random splits. In each repeated experiment, block partitioning, normalization parameter computation, model training, test prediction generation, post-processing, and performance evaluation were performed using the same procedure. The final results are reported as mean $\pm$ between-run SD over the 10 repeated experiments. For each repetition, the mean and SD of the reference BP values

were calculated over the test windows. Across the 10 repetitions, the test-set mean was 139.27 ± 2.02 mmHg for SBP and 64.93 ± 1.05 mmHg for DBP, while the within-test-set SD was 14.63 ± 1.16 mmHg for SBP and 11.53 ± 0.54 mmHg for DBP.

3.1.2. Chronological and Subject-Independent Protocols

To evaluate prediction on chronologically later intervals, a four-fold rolling-origin protocol was used. Fold 1 used blocks 0–9 for training, blocks 10–11 for validation, and blocks 12–13 for testing. In the subsequent folds, the training interval was expanded to blocks 0–11, 0–13, and 0–15, the next two blocks were used for validation, and blocks 14–15, 16–17, and 18–19 were used for testing, respectively. Scaling and model selection were performed independently within each fold using only its training and validation data. Because the same subjects contributed earlier training data and later test data, the rolling-origin protocol remained subject-dependent.

Subject-independent performance was evaluated using 17-fold leave-one-subject-out (LOSO) cross-validation. In each outer fold, one subject was held out for testing. Among the remaining 16 subjects, 14 were used for training and two for inner validation to determine the training duration. A new model was then fitted using all 16 non-test subjects for the selected number of epochs and evaluated on the held-out subject. The outer test subject was excluded from scaling, model selection, and model fitting. LOSO results were summarized as subject-macro averages.

3.1.3. Comparison Methods and Additional Baselines

To distinguish the effect of prediction selection from prediction correction, the moving-average, median-filter, and low-pass outputs were evaluated at the indices retained by the proposed method. To examine the effect of reduced prediction coverage alone, a random-rejection control retained the same number of predictions as the proposed method but selected them uniformly at random. Random rejection was repeated 1000 times within each repetition. A centered Hampel identifier with a five-prediction window was included as a simple local outlier-selection baseline [41], with its retained count matched to that of the proposed method.

For comparison with naive and lightweight estimators, a common set of 10 subject-dependent train/validation/test splits was applied to the global training-mean predictor, the subject-specific training-block mean predictor, Ridge regression, gradient-boosted tree regression, and the pulse-feature LSTM. The global training-mean predictor assigned the mean training SBP and DBP to every test window, whereas the subject-specific training-block mean predictor assigned each subject's mean training BP to that subject's test windows. Ridge regression [42] and gradient-boosted tree regression [43] used each 40×52 sequence flattened into a 2080-dimensional vector, with separate models for SBP and DBP. Ridge regularization was selected from $\{10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 1, 10, 100, 1000\}$. For gradient boosting, the number of estimators was selected from $\{100, 300\}$, the learning rate from $\{0.03, 0.10\}$, and the maximum depth from $\{2, 3\}$, with a minimum of 10 samples per leaf. Hyperparameters were selected using validation MAE. The temporal variance filter was then applied to the Ridge, gradient-boosted tree, and LSTM prediction sequences.

Two input-level SQI baselines were also constructed from the available rPPG waveforms. For template correlation, valid training pulses were resampled to 100 points, centered, normalized to unit length, and averaged to form a training-only pulse template. The mean pulse-to-template correlation across the 40 pulses in each input window was used as the window-level SQI [32]. A second SQI used the median pulse skewness within each window [33].

For the primary comparison, each SQI rejection threshold was determined exclusively from the validation-score distribution using 76.39% as the target retention ratio, corresponding to the retention level of the primary subject-dependent experiment. The selected threshold was then fixed and applied to the corresponding test set. To characterize the accuracy–coverage behavior of the SQI methods, additional validation-target retention levels ranging from 50% to 100% were evaluated. The resulting curves therefore represent different validation-derived thresholds, while the horizontal axis reports the observed retention ratio on the test set.

For a consistent comparison, all estimators and SQI baselines were evaluated using a common set of 10 subject-dependent data splits so that the comparison methods shared identical training, validation, and test partitions. The splits contained 1825.6 ± 29.9 training windows, 163.2 ± 4.2 validation windows, and 165.5 ± 5.1 test windows per repetition.

3.1.4. Sensitivity, Statistical, and Computational Analyses

The temporal variance of the reference window labels was compared with that of the raw predictions using the same four-prediction chunk definition. MAE and error SD were also calculated separately for the retained and discarded subsets. Parameter sensitivity was evaluated without retraining the estimator by reapplying the filter for $C = 3, \dots, 7$ and 41 logarithmically spaced thresholds from 1 to 100 (mmHg)².

Prediction error was defined as predicted BP minus reference BP. In addition to MAE and error SD, signed mean error (bias) and 95% limits of agreement, calculated as bias ± 1.96 times the error SD, were reported and visualized using Bland–Altman analysis [44]. Paired comparisons between methods evaluated on identical splits used two-sided Wilcoxon signed-rank tests [45], followed by Holm correction within each analysis family [46]. Per-subject retention and MAE were also calculated to examine variation across subjects.

The GPU-based LSTM experiments were conducted in Python 3.10 using TensorFlow 2.16.2 and Keras 3.9.2 with CUDA 12.3 and cuDNN 8. Ridge and gradient-boosted tree regression were implemented using scikit-learn 1.6.1. Independent reproduction of the temporal filtering procedure and the filter-timing analysis were performed in Python 3.9.13 using NumPy 1.21.5, pandas 1.4.4, and SciPy 1.9.1.

Estimator complexity was summarized using parameter count, serialized model size, and batch-size-one inference time. LSTM inference was measured on an NVIDIA RTX A4500 GPU (NVIDIA Corporation, Santa Clara, CA, USA), whereas Ridge and gradient-boosted tree timing used a single CPU thread. Filter-core processing time was measured for complete test prediction sequences, excluding file input/output. Because the estimators were timed on different devices, the reported values describe the tested implementations rather than a hardware-normalized speed ranking.

3.2. Overall Performance Comparison

In this section, BP estimation performance was compared among raw predictions, conventional smoothing-based post-processing methods, and the proposed temporal variance-based filtering. Raw predictions refer to the unfiltered predictions generated by the trained pulse-feature LSTM model for all input windows in the test set. The conventional post-processing methods include simple moving average, median filtering, and low-pass filtering, which smooth the subject-specific prediction sequences while retaining all predictions. In contrast, the proposed method selects temporally stable predictions without directly modifying the prediction values. The performance was reported as the average MAE and SD obtained over 10 repeated experiments. In addition, the prediction retention ratio was reported to indicate the proportion of predictions retained after each post-processing method.

As shown in Table 1, all conventional smoothing-based methods reduced MAE and error SD compared with the raw predictions while retaining all outputs. The temporal variance-based method achieved lower MAE and error SD for the retained subset, with SBP MAE and error SD of 5.34 and 8.04 mmHg and DBP MAE and error SD of 3.07 and 5.02 mmHg, respectively. A retention ratio of 76.39% means that 23.61% of the original window-level predictions were not included in the final subset. Thus, the error reductions should be interpreted together with the corresponding prediction coverage.

Table 1. Comparison of conventional post-processing methods and temporal variance-based filtering over 10 repeated subject-dependent experiments. Values are mean $\pm$ between-run SD.

Method	SBP MAE (mmHg)	SBP Error SD (mmHg)	DBP MAE (mmHg)	DBP Error SD (mmHg)	Retention (%)
Raw	7.04 $\pm$ 1.52	10.72 $\pm$ 2.13	4.63 $\pm$ 1.32	7.62 $\pm$ 2.14	100.00
Moving average	6.61 $\pm$ 1.52	9.67 $\pm$ 2.30	4.38 $\pm$ 1.35	6.74 $\pm$ 2.19	100.00
Median filtering	6.65 $\pm$ 1.64	9.82 $\pm$ 2.54	4.33 $\pm$ 1.46	6.82 $\pm$ 2.36	100.00
Low-pass filtering	6.46 $\pm$ 1.46	9.46 $\pm$ 2.15	4.38 $\pm$ 1.33	6.73 $\pm$ 2.17	100.00
Temporal variance filtering	5.34 $\pm$ 1.62	8.04 $\pm$ 2.31	3.07 $\pm$ 0.97	5.02 $\pm$ 1.93	76.39 $\pm$ 7.59

Because the proposed method achieved lower MAE and error SD for the retained subset, the distribution between true blood pressure and predicted blood pressure was further examined using scatter plots. Figure 5 shows, for both SBP and DBP, the distributions of the predictions included in the final retained subset and the predictions discarded during filtering. Colored markers represent retained predictions, whereas gray open markers indicate discarded predictions. The dashed line denotes the identity line between true and predicted blood pressure. In the SBP results, some discarded predictions were distributed relatively far from the identity line, whereas the retained predictions generally showed a more compact distribution around the identity line. A similar pattern was observed for DBP. These visual results are consistent with the reductions in MAE and error SD shown in Table 1.

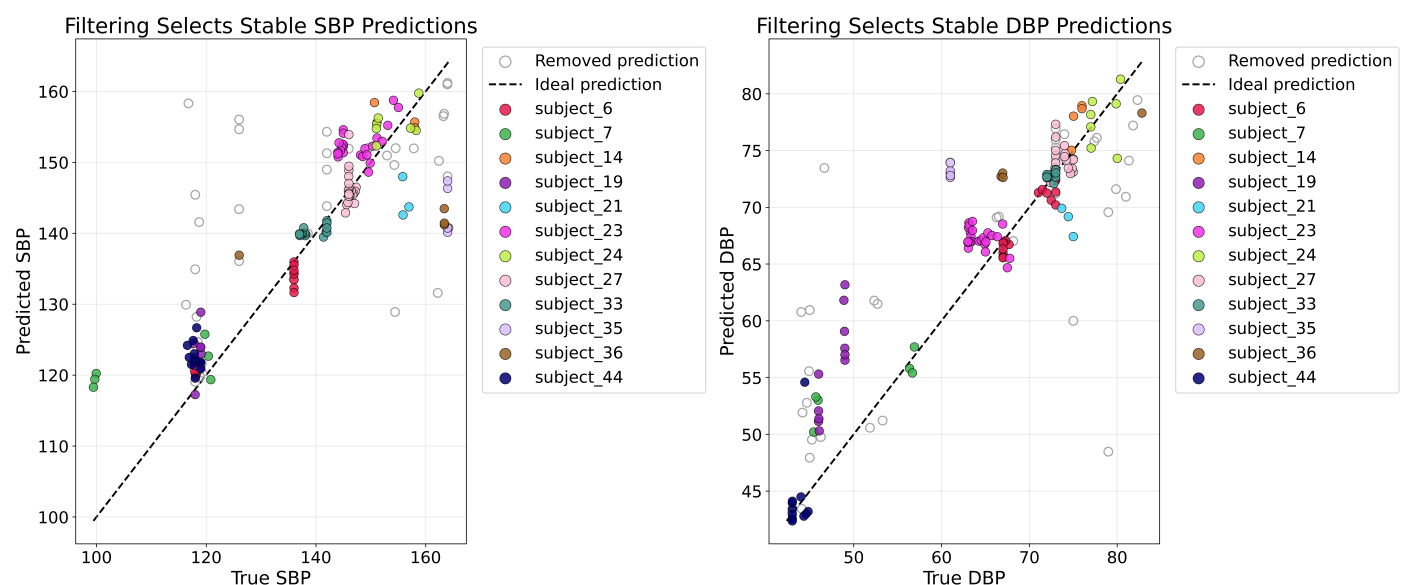


Figure 5. Scatter plots of SBP and DBP predictions for the repetition with performance closest to the mean across the repeated experiments. Predictions retained by temporal variance-based filtering are shown with colored markers, discarded predictions with gray open markers, and the dashed line denotes the identity line.

3.3. Coverage-Matched Controls and Retained/Discarded Subsets

Table 2 compares the proposed selection with controls evaluated at the same retention ratio. Random rejection produced results close to the complete raw prediction set, and the Hampel identifier retained higher errors than the proposed method. Compared with both controls, the proposed method showed significantly lower SBP and DBP MAEs and error SDs after Holm correction ($p_{\text{Holm}} = 0.0176$ for all four outcomes). Compared with the unsmoothed retained predictions, moving-average and median filtering produced significantly lower SBP MAE and error SD ($p_{\text{Holm}} = 0.0176$ for both outcomes). Low-pass filtering also reduced SBP MAE and error SD, with $p_{\text{Holm}} = 0.0488$ and $p_{\text{Holm}} = 0.0371$, respectively. None of the corresponding DBP differences was significant ($p_{\text{Holm}} = 1.0000$).

Table 2. Coverage-matched control results and the discarded subset in the primary experiment. Values are mean $\pm$ between-run SD.

Method or Subset	SBP MAE	SBP Error SD	DBP MAE	DBP Error SD	Retention (%)
Temporal variance filtering	5.34 $\pm$ 1.62	8.04 $\pm$ 2.31	3.07 $\pm$ 0.97	5.02 $\pm$ 1.93	76.39 $\pm$ 7.59
Moving average, same indices	5.17 $\pm$ 1.61	7.69 $\pm$ 2.23	3.10 $\pm$ 0.91	4.94 $\pm$ 1.74	76.39 $\pm$ 7.59
Median filter, same indices	5.18 $\pm$ 1.65	7.72 $\pm$ 2.32	3.02 $\pm$ 0.98	4.91 $\pm$ 1.87	76.39 $\pm$ 7.59
Low-pass filter, same indices	5.13 $\pm$ 1.62	7.64 $\pm$ 2.22	3.07 $\pm$ 0.91	4.81 $\pm$ 1.78	76.39 $\pm$ 7.59
Hampel identifier	6.82 $\pm$ 1.69	10.42 $\pm$ 2.64	4.35 $\pm$ 1.45	7.26 $\pm$ 2.44	76.39 $\pm$ 7.59
Random rejection	7.04 $\pm$ 1.52	10.69 $\pm$ 2.11	4.63 $\pm$ 1.31	7.60 $\pm$ 2.13	76.39 $\pm$ 7.59
Discarded subset	12.09 $\pm$ 2.16	15.74 $\pm$ 2.52	9.49 $\pm$ 2.43	12.37 $\pm$ 2.65	23.61 $\pm$ 7.59

The discarded subset had substantially higher errors than the retained subset, with SBP and DBP MAEs of 12.09 and 9.49 mmHg, respectively. This result confirms that the excluded predictions were, on average, more difficult cases in the present dataset, while the random-rejection result shows that reduced coverage alone did not account for the observed improvement.

3.4. Parameter Sensitivity and Reference-Sequence Variability

The reference window-label sequences were smoother than the raw prediction sequences. For SBP, the mean chunk variance was 6.83 ± 3.47 (mmHg)² for the reference labels and 18.89 ± 11.83 (mmHg)² for the raw predictions; the corresponding median values were 0.11 and 2.79 (mmHg)². For DBP, the mean values were 1.44 ± 0.63 and 12.91 ± 8.75 (mmHg)², with median values of 0.04 and 0.98 (mmHg)². The lower reference variance is consistent with averaging BP values within each 40-pulse window and the 75% overlap between adjacent windows.

As shown in Figure 6, both the chunk size C and variance threshold τ affected the error characteristics of the retained predictions. Among the evaluated chunk sizes, $C = 4$ generally showed the lowest or near-lowest MAE and error SD over a broad range of variance thresholds. For all chunk sizes, the error metrics tended to increase as the variance threshold became less restrictive, although the magnitude of this change differed according to C . These results indicate that both parameters influence the behavior of the temporal variance filter.

Figure 7 further shows the accuracy–coverage relationship for the primary chunk size $C = 4$. As the variance threshold τ decreased, stricter filtering generally reduced SBP and DBP MAEs and error SDs, but also decreased the proportion of retained predictions. For example, decreasing τ from 100 to 1 (mmHg)² reduced the retention ratio from approximately 95.00% to 44.89%, while SBP MAE decreased from 6.39 to 4.12 mmHg and DBP MAE decreased from 4.02 to 1.98 mmHg. This trend demonstrates a clear trade-off between prediction error and coverage. At $C = 4$, a variance threshold of $\tau = 10$ (mmHg)² pro-

vided an intermediate balance between prediction error and coverage, retaining 76.39% of the predictions with SBP and DBP MAEs of 5.34 and 3.07 mmHg, respectively. Thus, the primary setting of $C = 4$ and $\tau = 10$ (mmHg)² was located at an intermediate point in the observed error–coverage trade-off.

Subject-sequence temporal-filter threshold sensitivity

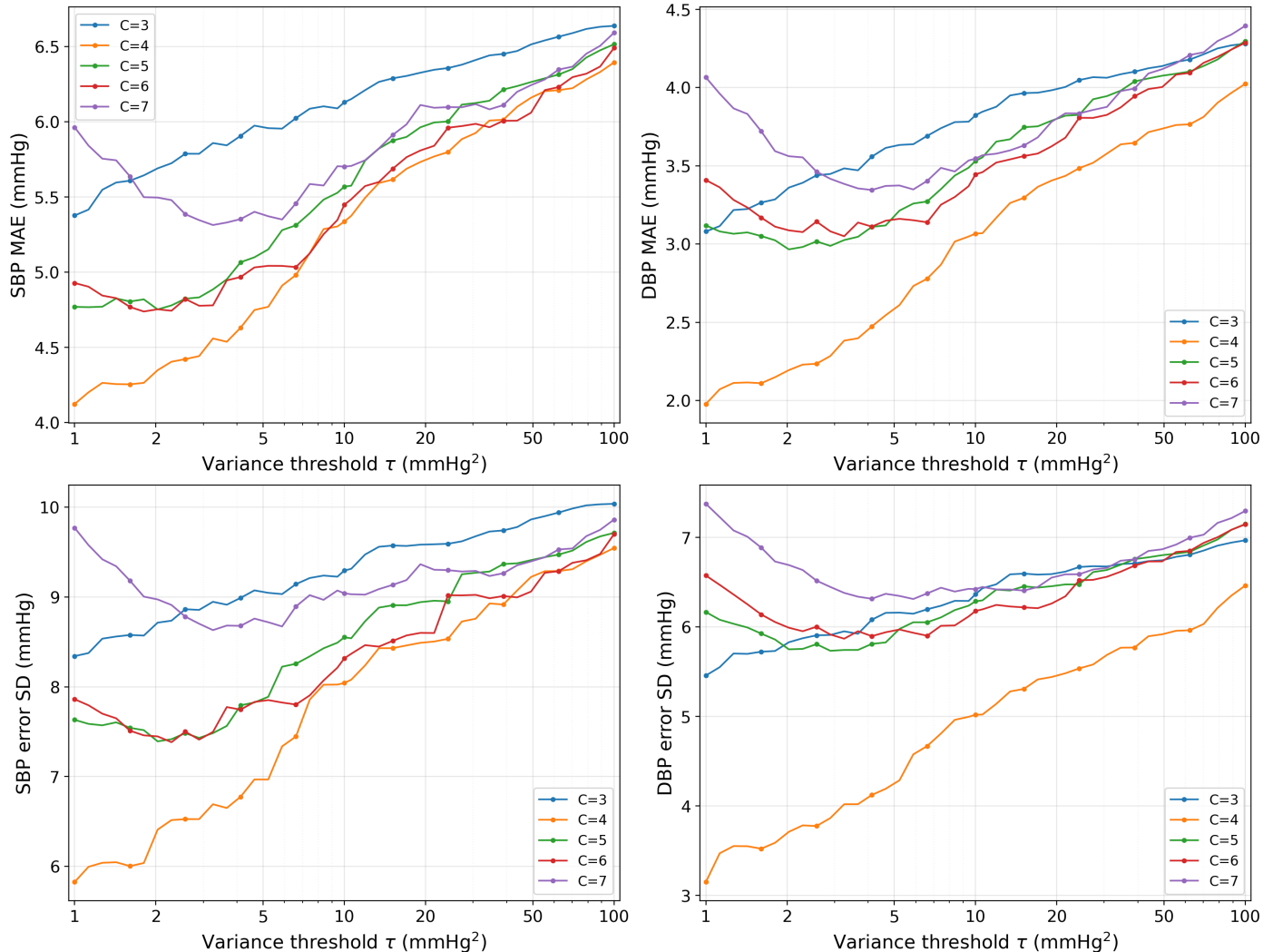


Figure 6. Sensitivity of temporal variance-based filtering to the chunk size C and variance threshold τ . SBP and DBP MAEs and error SDs are shown for chunk sizes $C = 3$ – 7 over variance thresholds ranging from 1 to 100 (mmHg)².

3.5. Chronological and Subject-Independent Evaluation

The rolling-origin protocol produced higher errors than the randomized-block protocol, but filtering reduced both MAE and error SD for chronologically later intervals from the same subjects. Under LOSO evaluation, the raw SBP and DBP MAEs increased to 15.96 and 12.11 mmHg, respectively, and filtering did not improve MAE. These results demonstrate a clear difference between within-subject performance and generalization to unseen subjects. The corresponding results for the three evaluation protocols are summarized in Table 3.

Table 3. Comparison of the primary, rolling-origin, and LOSO protocols. Primary values are mean $\pm$ between-run SD, rolling-origin values are mean $\pm$ between-fold SD, and LOSO values are subject-macro mean $\pm$ between-subject SD.

Protocol	Output	SBP MAE	SBP Error SD	DBP MAE	DBP Error SD	Retention (%)
Subject-dependent randomized blocks	Raw	7.04 $\pm$ 1.52	10.72 $\pm$ 2.13	4.63 $\pm$ 1.32	7.62 $\pm$ 2.14	100.00
Subject-dependent randomized blocks	Filtered	5.34 $\pm$ 1.62	8.04 $\pm$ 2.31	3.07 $\pm$ 0.97	5.02 $\pm$ 1.93	76.39 $\pm$ 7.59
Rolling-origin	Raw	9.04 $\pm$ 1.59	12.17 $\pm$ 2.24	6.57 $\pm$ 0.90	9.16 $\pm$ 1.11	100.00
Rolling-origin	Filtered	7.69 $\pm$ 1.82	9.92 $\pm$ 2.39	5.47 $\pm$ 1.14	7.60 $\pm$ 1.21	79.31 $\pm$ 6.84
LOSO	Raw	15.96 $\pm$ 7.81	8.59 $\pm$ 4.91	12.11 $\pm$ 7.84	5.28 $\pm$ 2.51	100.00
LOSO	Filtered	16.33 $\pm$ 8.23	8.29 $\pm$ 4.92	12.33 $\pm$ 7.90	5.22 $\pm$ 2.90	83.62 $\pm$ 23.68

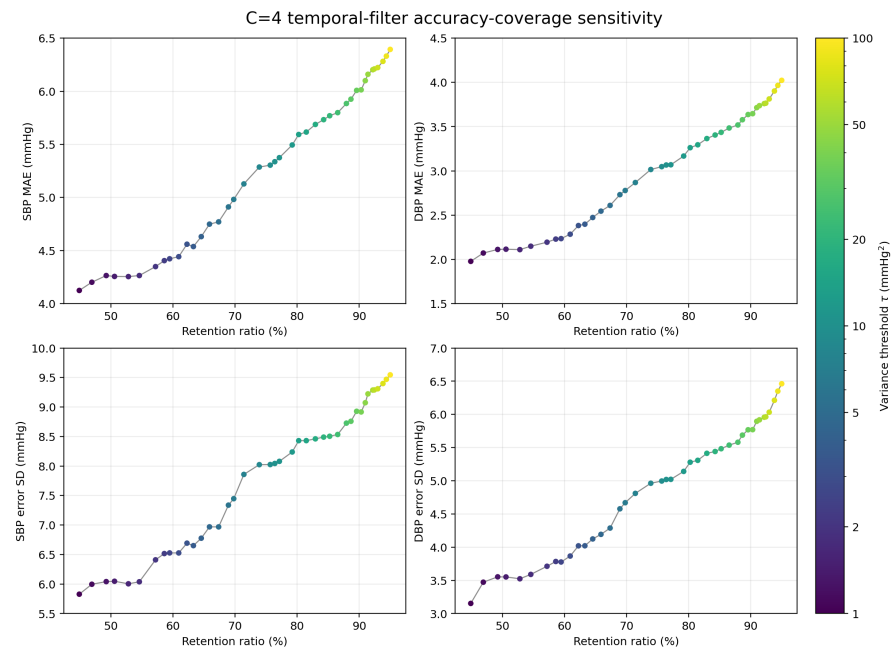


Figure 7. Accuracy-coverage sensitivity of temporal variance-based filtering for $C = 4$. SBP and DBP MAEs and error SDs are plotted against the prediction retention ratio as the variance threshold τ varies from 1 to 100 $(\text{mmHg})^2$. Marker color represents the corresponding variance threshold.

3.6. Naive Predictors, Alternative Regressors, and SQI Rejection

To examine the proposed filtering method relative to naive predictors, alternative regression models, and input-level SQI rejection, all methods were compared using the common set of 10 subject-dependent splits described above. The corresponding results are summarized in Table 4.

Table 4. Comparison of naive predictors, alternative regressors, and SQI-based selection methods using a common set of 10 subject-dependent splits. Values are mean $\pm$ between-run SD.

Predictor or Selection Method	Output	SBP MAE	SBP Error SD	DBP MAE	DBP Error SD	Retention (%)
Global training mean	Raw	12.23 $\pm$ 1.32	14.72 $\pm$ 1.37	9.54 $\pm$ 0.67	11.53 $\pm$ 0.73	100.00
Subject-specific training-block mean	Raw	5.26 $\pm$ 1.03	7.08 $\pm$ 1.29	2.39 $\pm$ 0.40	3.37 $\pm$ 0.64	100.00
Ridge	Raw	10.17 $\pm$ 1.31	12.78 $\pm$ 1.16	8.14 $\pm$ 0.68	10.15 $\pm$ 0.83	100.00
Ridge with temporal filtering	Filtered	9.94 $\pm$ 1.41	12.39 $\pm$ 1.35	7.98 $\pm$ 0.95	9.97 $\pm$ 1.13	87.72 $\pm$ 12.40
Gradient-boosted tree	Raw	9.14 $\pm$ 1.28	11.72 $\pm$ 1.33	6.66 $\pm$ 0.80	8.52 $\pm$ 0.92	100.00
Gradient-boosted tree with temporal filtering	Filtered	8.47 $\pm$ 1.31	10.68 $\pm$ 1.46	5.99 $\pm$ 0.94	7.71 $\pm$ 1.08	80.89 $\pm$ 8.05
Pulse-feature LSTM	Raw	6.33 $\pm$ 1.30	9.53 $\pm$ 1.76	4.32 $\pm$ 1.08	6.95 $\pm$ 1.59	100.00
Pulse-feature LSTM with temporal filtering	Filtered	4.56 $\pm$ 1.66	6.38 $\pm$ 2.20	2.83 $\pm$ 1.18	4.33 $\pm$ 1.90	75.13 $\pm$ 5.15
Template-correlation SQI	Selected	6.28 $\pm$ 1.09	9.50 $\pm$ 1.77	3.77 $\pm$ 1.01	6.19 $\pm$ 1.91	76.41 $\pm$ 4.50
Skewness SQI	Selected	6.93 $\pm$ 1.50	10.09 $\pm$ 1.64	4.74 $\pm$ 1.22	7.44 $\pm$ 1.71	78.90 $\pm$ 7.27

The raw LSTM outperformed the global training mean, Ridge, and gradient-boosted tree predictors, but the subject-specific training-block mean achieved lower errors than the raw LSTM. The difference was significant for SBP MAE ($p_{\text{Holm}} = 0.0488$) and for DBP MAE and both error SD outcomes ($p_{\text{Holm}} \leq 0.0078$). Thus, subject-specific BP offsets contributed substantially to performance under the subject-dependent protocol. Temporal variance filtering reduced the errors of Ridge, gradient-boosted tree, and LSTM outputs, although the amount of improvement differed by estimator. The filtered LSTM remained better than the filtered Ridge and gradient-boosted tree models for all four outcomes ($p_{\text{Holm}} = 0.0039$).

Figure 8 compares the accuracy–coverage characteristics of the two input-level SQI methods with the fixed temporal variance filter. As the SQI rejection thresholds were varied, both template-correlation and skewness SQIs exhibited trade-offs between prediction coverage and estimation error. At the primary operating points, temporal variance filtering retained $75.13 \pm 5.15\%$ of the test predictions, compared with $76.41 \pm 4.50\%$ for template-correlation SQI and $78.90 \pm 7.27\%$ for skewness SQI. Despite the similar prediction coverage, temporal variance filtering achieved lower MAE and error SD for both SBP and DBP. Compared with template-correlation SQI, the differences were significant for SBP MAE, DBP MAE, and both error SD outcomes ($p_{\text{Holm}} \leq 0.0137$). Compared with skewness SQI, all four outcomes were significant ($p_{\text{Holm}} = 0.0039$). These results suggest that prediction-level temporal variability provides information complementary to input-level waveform quality measures.

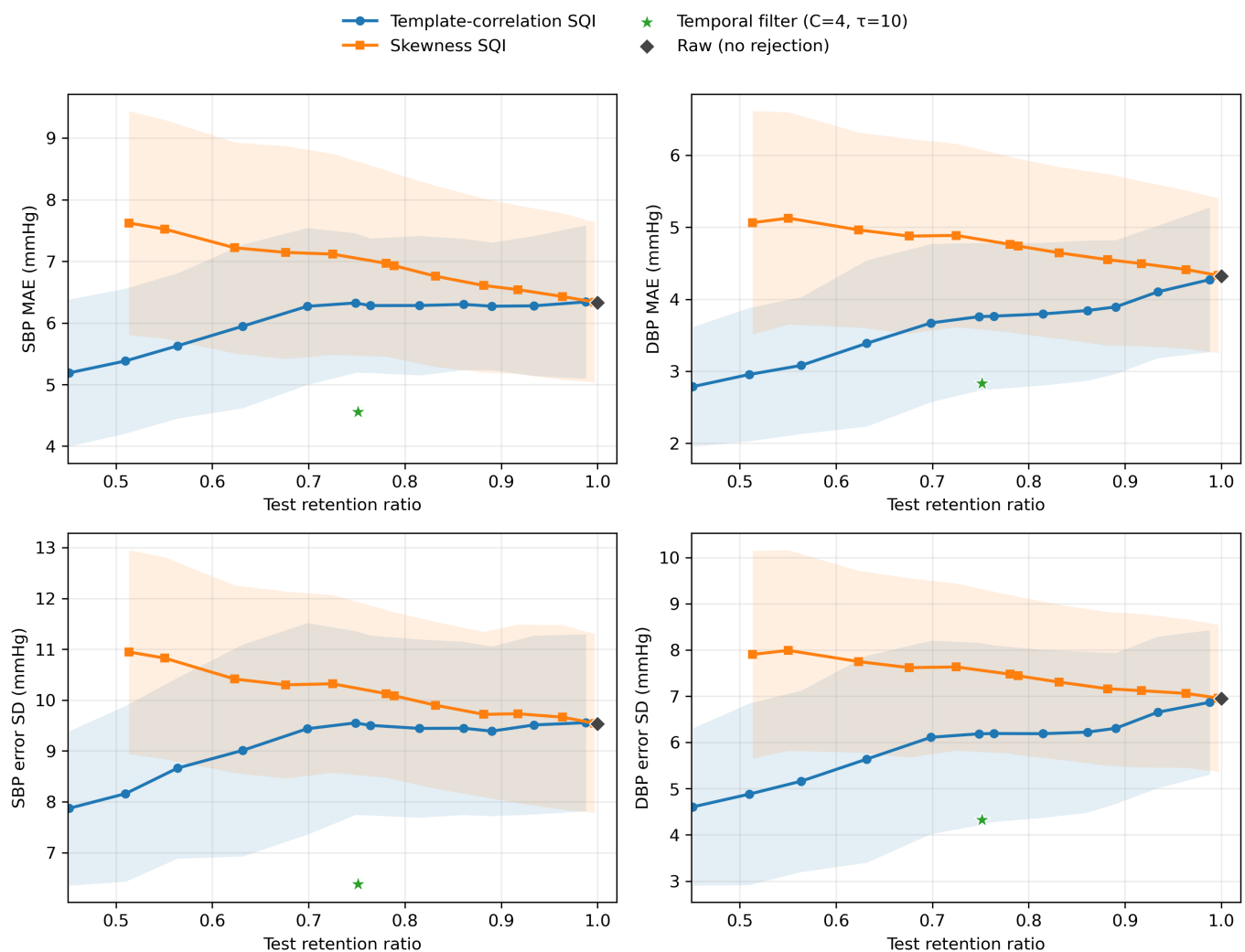


Figure 8. Accuracy–coverage comparison of input-level SQI rejection and output-level temporal variance filtering over 10 subject-dependent repetitions. Template-correlation and skewness SQI curves

were obtained by varying validation-derived rejection thresholds, and the horizontal axis represents the observed test retention ratio. The green star denotes temporal variance filtering at the fixed setting $C = 4$ and $\tau = 10 \text{ (mmHg)}^2$, while the black diamond denotes raw predictions without rejection. Shaded bands indicate mean $\pm$ one between-run standard deviation across the 10 repetitions.

3.7. Residual, Bias, and Agreement Analyses

The signed mean errors were small and positive for both raw and filtered predictions. The average limits of agreement narrowed after filtering, consistent with the lower error SD of the retained subset. The signed-bias and agreement results are summarized in Table 5. The corresponding residual distributions are shown in Figure 9.

Table 5. Signed bias and 95% limits of agreement (LoA) in the primary experiment. Bias values are mean $\pm$ between-run SD; limits are the mean run-level endpoints.

Output	SBP Bias	SBP Error SD	SBP 95% LoA	DBP Bias	DBP Error SD	DBP 95% LoA
Raw	0.64 ± 1.74	10.72 ± 2.13	−20.38, 21.66	0.21 ± 1.63	7.62 ± 2.14	−14.72, 15.15
Filtered	1.06 ± 1.52	8.04 ± 2.31	−14.70, 16.82	0.33 ± 1.71	5.02 ± 1.93	−9.51, 10.17

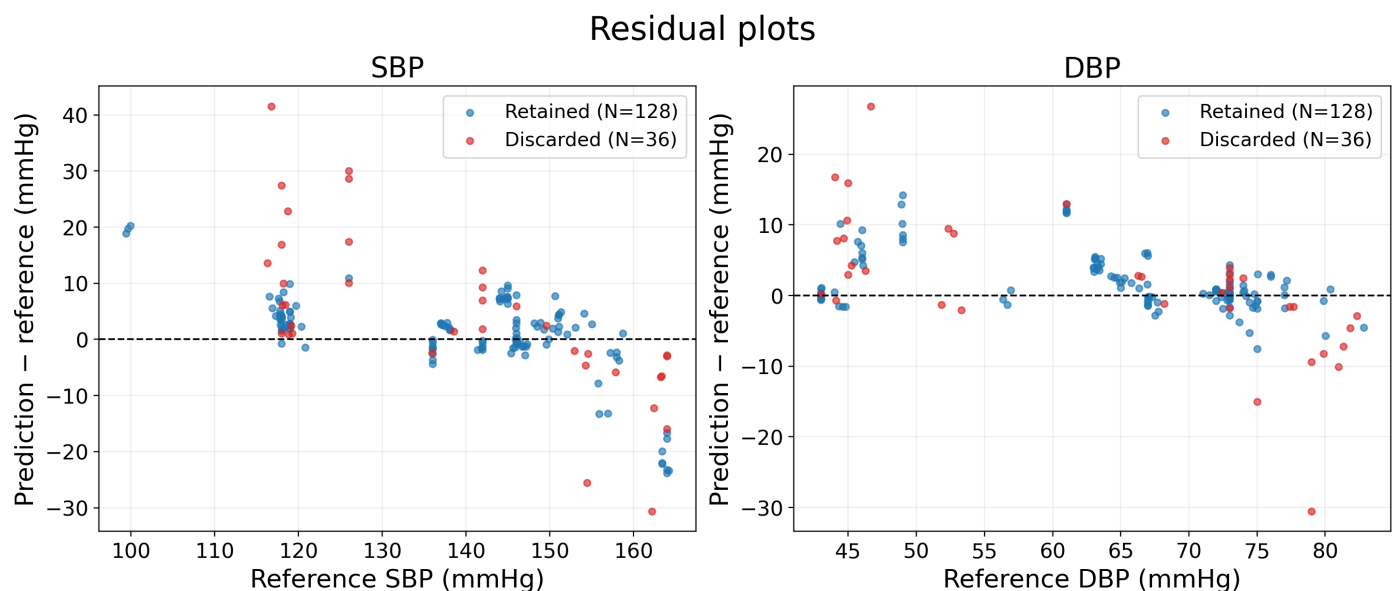


Figure 9. Residual plots for SBP and DBP for the same repetition used in Figure 5. Residuals were defined as predicted BP minus reference BP.

The residual plots show that several discarded predictions had relatively large positive or negative errors, although nonzero residuals also remained among the retained predictions. Temporal stability therefore does not guarantee accuracy for every individual window.

3.8. Per-Subject Results

Per-subject retention ranged from 22.5% to 97.5%, and filtering did not improve both targets for every subject. This variation indicates that the aggregate retention ratio does not imply uniform prediction coverage across subjects. The corresponding Bland–Altman plots are shown in Figure 10. The per-subject retention and MAE results are summarized in Table 6.

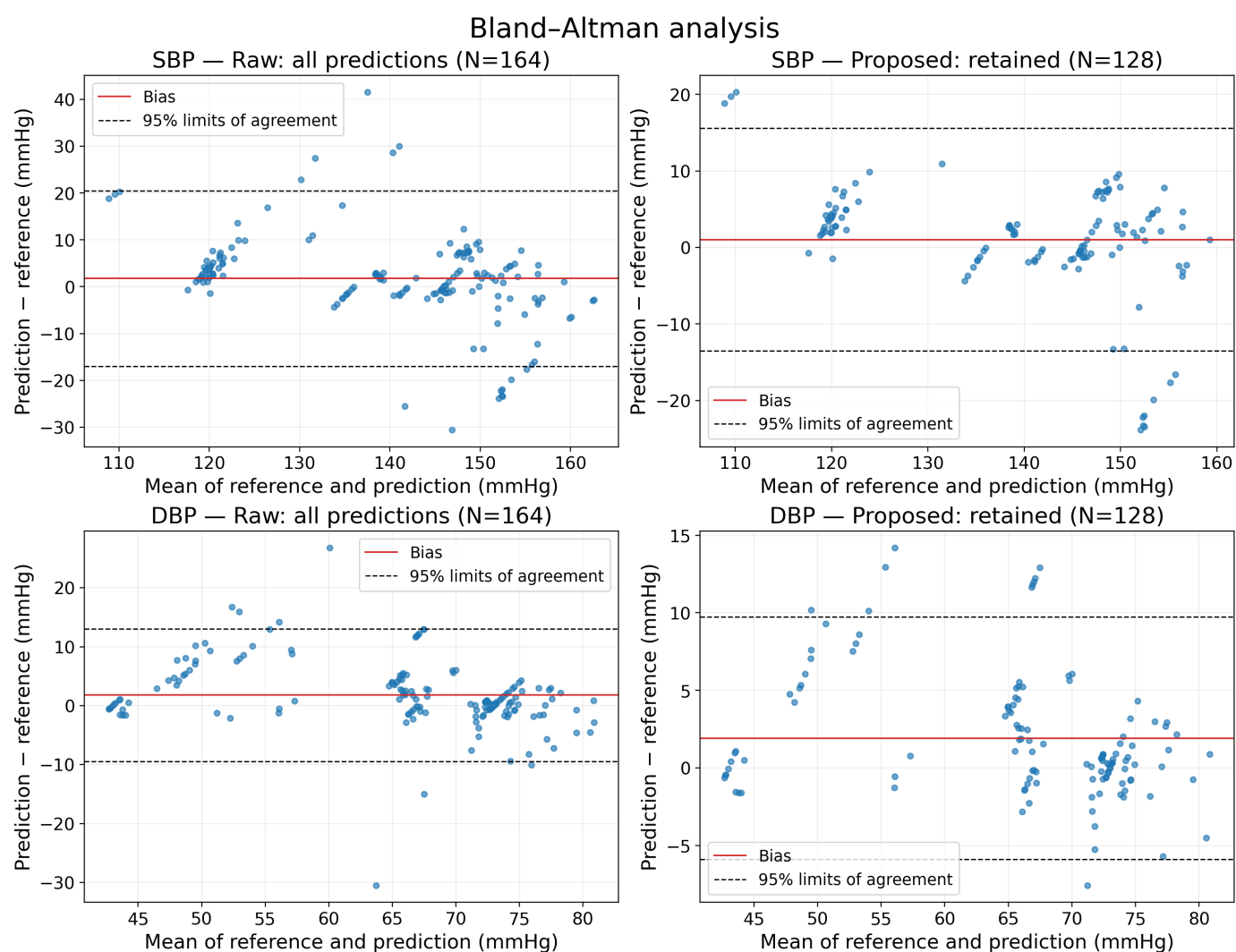


Figure 10. Bland–Altman plots for raw and filtered SBP and DBP predictions for the same repetition used in Figures 5 and 9. Solid lines indicate mean signed error, and dashed lines indicate the 95% limits of agreement.

Table 6. Per-subject results in the primary experiment. n_{runs} is the number of repetitions in which the subject contributed an evaluable test sequence; retention and MAE are averaged over those repetitions.

Subject	n_{runs}	Retention (%)	SBP MAE (mmHg)		DBP MAE (mmHg)	
			Raw	Filtered	Raw	Filtered
6	10	77.2	3.67	2.94	3.34	2.90
7	10	44.4	16.74	16.76	9.82	11.69
10	2	37.5	9.45	10.52	14.81	16.43
13	10	22.5	7.67	4.71	10.70	2.17
14	10	95.5	3.74	3.56	3.00	2.38
19	10	64.2	6.61	6.09	6.83	7.42
21	10	45.8	10.68	5.13	6.80	3.67
23	10	97.5	5.93	5.83	2.65	2.61
24	10	79.1	8.48	7.68	4.58	3.93
27	10	90.9	3.20	2.41	2.56	1.50
33	10	85.7	6.55	5.96	3.25	2.65
35	2	79.2	13.93	13.46	7.86	6.78
36	10	66.6	12.94	11.12	6.31	5.53
44	10	72.1	7.23	5.06	4.24	2.46
48	10	27.5	15.59	14.06	13.72	11.02

3.9. Computational Characteristics

The pulse-feature LSTM contained 1,669,026 trainable parameters and had a serialized model size of 20.11 MB. Its batch-size-one inference time was 41.85 ± 31.89 ms per window on the GPU used in this study. The temporal variance filtering stage introduced no additional trainable parameters and required no model retraining. Processing a complete primary test prediction sequence required 3.73 ± 0.60 ms, corresponding to 22.8 ± 3.9 μ s per prediction. For online use, four consecutive window-level predictions are required before the first filtering decision can be made. Based on the mean stride duration of 7.56 s, the requirement for four consecutive predictions corresponds to approximately 22.7 s of additional temporal latency after the first window-level prediction before the first filtering decision becomes available. The computational characteristics are summarized in Table 7.

Table 7. Computational characteristics of the evaluated estimators and the temporal variance filtering procedure. The reported processing times are implementation-specific because CPU and GPU measurements were obtained on different devices.

Estimator or Procedure	Complexity	Serialized Size	Processing Time
Ridge	4162 coefficients	16.18 kB	0.200 ± 0.004 ms/window, CPU
Gradient-boosted tree	520 ± 103 trees	295.1 ± 67.5 kB	0.680 ± 0.036 ms/window, CPU
Pulse-feature LSTM	1,669,026 trainable parameters	20.11 MB	41.85 ± 31.89 ms/window, GPU
Temporal variance filtering	0 trainable parameters	Not applicable	3.73 ± 0.60 ms/sequence

3.10. Comparison with Published rPPG Results

Table 8 summarizes published results on the same or closely related Leipzig rPPG dataset. The values are provided for context because preprocessing, subject separation, personalization, input representation, and prediction coverage differ across studies.

Table 8. Contextual comparison with published rPPG-based BP results on the Leipzig dataset. NR indicates that computational information was not reported.

Study	Input/Model	Evaluation Protocol	SBP MAE	DBP MAE	Coverage	Computation
Schrumpf et al. [34]	rPPG transfer LSTM	Subject-wise, no personalization	13.6	10.3	100%	NR
Schrumpf et al. [34]	rPPG transfer ResNet	Held-out subject with personalization	12.7	10.8	100%	NR
Kim et al. [47]	5-s rPPG, U-Net-based model	Random sample-level split	6.90	4.43	100%	NR
Present study	Pulse-feature LSTM, raw	Subject-dependent randomized blocks	7.04	4.63	100%	Table 7
Present study	Pulse-feature LSTM, filtered	Subject-dependent randomized blocks	5.34	3.07	76.39%	Table 7
Present study	Pulse-feature LSTM, raw	LOSO	15.96	12.11	100%	Table 7
Present study	Pulse-feature LSTM, filtered	LOSO	16.33	12.33	83.62%	Table 7

Compared with the subject-wise rPPG transfer-LSTM result reported by Schrumpf et al. [34], the raw LOSO evaluation in the present study yielded higher MAEs for both SBP (15.96 vs. 13.6 mmHg) and DBP (12.11 vs. 10.3 mmHg). However, these values should not be interpreted as a direct ranking because the preprocessing procedures, model configurations, and evaluation protocols differ across studies. In particular, subject-dependent or sample-level splits represent a different prediction setting from subject-independent evaluation, and the filtered results additionally reflect reduced prediction coverage. Direct computational comparison was also limited because the cited studies did not report parameter count and inference time in a form comparable with the present implementation.

4. Discussion

In this study, temporal variance-based filtering was applied to the window-level prediction sequences generated by the proposed rPPG pulse-feature LSTM model. The primary subject-dependent experiments showed lower MAE and error SD for the retained

SBP and DBP predictions than for the complete raw prediction set and the full-coverage smoothing results. The coverage-matched controls further clarified this result. Random rejection produced errors similar to those of the raw predictions, and the Hampel identifier was less effective than the proposed selection at the same coverage. Conventional smoothing applied to the same retained indices produced a small additional improvement for SBP, indicating that prediction selection and prediction correction can contribute separately.

The proposed method selects a subset of the original predictions rather than correcting the complete sequence. A retention ratio of 76.39% means that approximately three of every four window-level predictions satisfied the temporal-stability criterion. This represents an accuracy–coverage trade-off: excluding unstable outputs reduced error for the retained subset, while fewer estimates remained available for continuous monitoring. In a practical system, discarded windows could be flagged for continued video acquisition or repeated measurement instead of being displayed as valid BP estimates. The filtering process uses no ground-truth BP values and does not alter the retained prediction values.

The comparison across evaluation protocols highlights the difference between within-subject prediction and unseen-subject generalization. Error increased when temporal order was preserved in the rolling-origin evaluation, although filtering still reduced MAE and error SD for chronologically later intervals from the same subjects. In the controlled experiment, the subject-specific training-block mean was competitive with the full filtering pipeline. It achieved a DBP MAE of 2.39 ± 0.40 mmHg at 100% coverage, compared with 2.83 ± 1.18 mmHg for the filtered LSTM at $75.13 \pm 5.15\%$ coverage, although the filtered LSTM achieved lower SBP MAE (4.56 ± 1.66 vs. 5.26 ± 1.03 mmHg). This finding further indicates that subject-specific BP offsets contributed substantially under the subject-dependent protocol. Under LOSO evaluation, error increased markedly and filtering did not improve MAE. Temporal stability of an output sequence therefore does not resolve systematic estimation error for unseen subjects.

The ground-truth window labels were smoother than the raw predictions because each label was averaged over 40 pulses and adjacent windows overlapped by 75%. The larger MAE of the discarded subset supports the usefulness of the proposed criterion in the present dataset, but it does not establish that every discarded prediction is erroneous. A genuine rapid physiological BP change may also produce high temporal variance and could be discarded. Validation using recordings with verified rapid BP transitions is therefore needed before applying the method to continuous clinical monitoring.

The sensitivity analysis showed that C and τ control the balance between coverage and error rather than defining a single universally optimal setting. The SQI comparison further highlights the distinction between input-level signal-quality screening and output-level temporal selection. In the present dataset, which had already undergone SNR-based screening and landmark-based pulse exclusion before the SQI comparison, the template-correlation and skewness SQIs did not identify subsets with errors as low as those obtained by temporal variance filtering at comparable prediction coverage. This suggests that waveform quality and temporal prediction stability capture different aspects of the estimation process. However, the relative effectiveness of SQI-based and prediction-level selection may depend on the dataset, preprocessing procedure, and available range of signal quality. Future studies may therefore investigate combined input- and output-level selection strategies. The wide range of per-subject retention further suggests that adaptive subject-level settings should also be investigated.

The signed-error and Bland–Altman analyses provide descriptive information about the direction and spread of the prediction errors. The filtered predictions had average biases of 1.06 mmHg for SBP and 0.33 mmHg for DBP. However, the present study does not constitute validation under AAMI/ESH/ISO or British Hypertension Society protocols.

The AAMI/ESH/ISO framework requires at least 85 subjects and a prescribed reference-measurement design, whereas BHS grading is based on cumulative error percentages within 5, 10, and 15 mmHg. The present cohort included 17 subjects and the primary evaluation was subject-dependent; therefore, no AAMI/ESH/ISO pass claim or BHS grade is assigned [48,49].

From a computational perspective, the proposed filtering stage adds little complexity to the underlying estimator because it introduces no trainable parameters and requires no model retraining. Its application to Ridge, gradient-boosted tree, and LSTM outputs also indicates that the procedure is not restricted to one estimator. The LSTM was substantially larger than the lightweight regressors, while the attention layer, feature groups, window size, and stride were held fixed in the present study. Additional limitations include the small number of subjects, uneven per-subject coverage, and the absence of external-dataset validation. Future work should investigate alternative estimator designs, adaptive filtering conditions, and larger datasets with more diverse subjects, environments, cameras, and BP ranges.

5. Conclusions

In this study, temporal variance-based filtering was applied to the window-level prediction sequences generated by a pulse-feature LSTM estimator. The method compares accumulated keep and removal candidate counts across overlapping prediction chunks, uses no ground-truth BP values during selection, and requires neither additional trainable parameters nor model retraining. In the primary subject-dependent experiments, SBP MAE decreased from 7.04 ± 1.52 to 5.34 ± 1.62 mmHg and DBP MAE decreased from 4.63 ± 1.32 to 3.07 ± 0.97 mmHg, while $76.39 \pm 7.59\%$ of predictions were retained. Random rejection and a coverage-matched Hampel identifier did not reproduce this improvement, and the discarded subset had substantially larger errors.

Results from the different evaluation protocols further clarify the scope of these findings. A subject-specific training-block mean outperformed the raw LSTM in the controlled subject-dependent comparison, and filtering did not improve LOSO MAE for unseen subjects. The proposed method should therefore be interpreted as a low-overhead temporal-stability selection procedure that adjusts the accuracy–coverage characteristics of an existing prediction sequence, rather than as a calibrated confidence estimator or evidence of subject-independent BP accuracy. Further validation is required under causal monitoring conditions, during verified rapid BP changes, and on larger external datasets.

Author Contributions: Conceptualization, D.P., H.J. and N.K.K.; methodology, D.P. and H.J.; software, D.P.; validation, D.P. and H.J.; formal analysis, D.P.; investigation, D.P. and H.J.; resources, N.K.K.; data curation, D.P.; writing—original draft preparation, D.P. and H.J.; writing—review and editing, D.P., H.J. and N.K.K.; visualization, D.P.; supervision, N.K.K.; project administration, N.K.K.; funding acquisition, N.K.K. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the Gyeongbuk ANCHOR system through the Gyeongbuk ANCHOR CENTER, funded by the Ministry of Education (MOE) and the Gyeongsangbuk-do, Republic of Korea (2026-ANCHOR-15-115).

Institutional Review Board Statement: The original data collection was approved by the Ethics Committee of the University of Leipzig. This study used a previously collected rPPG-based BP dataset and did not involve new data collection from human participants.

Informed Consent Statement: Written informed consent was obtained from all subjects involved in the original study. This study did not involve new participant recruitment.

Data Availability Statement: The rPPG-based BP dataset analyzed in this study is available in the GitHub repository provided by Schruppf et al. at <https://github.com/Fabian-Sc85/non-invasive-bp-estimation-using-deep-learning> (accessed on 4 June 2026). The repository includes the rPPG data used for camera-based BP prediction in an HDF5 file in the data subfolder. The processed features and experimental results generated in the present study are available from the corresponding author upon reasonable request.

Conflicts of Interest: The authors declare no conflicts of interest. The funder had no role in the design of the study; the collection, analysis, or interpretation of data; the writing of the manuscript; or the decision to publish the results.

Abbreviations

The following abbreviations are used in this manuscript:

rPPG	Remote photoplethysmography
PPG	Photoplethysmography
BP	Blood pressure
SBP	Systolic blood pressure
DBP	Diastolic blood pressure
ROI	Region of interest
LSTM	Long short-term memory
BiLSTM	Bidirectional long short-term memory
MAE	Mean absolute error
SD	Standard deviation
SNR	Signal-to-noise ratio
SQI	Signal quality index
LOSO	Leave-one-subject-out
LoA	Limits of agreement

References

1. Ding, X.-R.; Zhao, N.; Yang, G.-Z.; Pettigrew, R.I.; Lo, B.; Miao, F.; Li, Y.; Liu, J.; Zhang, Y.-T. Continuous Blood Pressure Measurement from Invasive to Unobtrusive: Celebration of 200th Birth Anniversary of Carl Ludwig. *IEEE J. Biomed. Health Inform.* **2016**, *20*, 1455–1465. [[CrossRef](#)] [[PubMed](#)]
2. Mukherjee, R.; Ghosh, S.; Gupta, B.; Chakravarty, T. A Literature Review on Current and Proposed Technologies of Noninvasive Blood Pressure Measurement. *Telemed. J. e-Health* **2018**, *24*, 185–193. [[CrossRef](#)] [[PubMed](#)]
3. Allen, J. Photoplethysmography and Its Application in Clinical Physiological Measurement. *Physiol. Meas.* **2007**, *28*, R1–R39. [[CrossRef](#)] [[PubMed](#)]
4. Kachuee, M.; Kiani, M.M.; Mohammadzade, H.; Shabany, M. Cuffless Blood Pressure Estimation Algorithms for Continuous Health-Care Monitoring. *IEEE Trans. Biomed. Eng.* **2017**, *64*, 859–869. [[CrossRef](#)] [[PubMed](#)]
5. Slapničar, G.; Mlakar, N.; Luštrek, M. Blood Pressure Estimation from Photoplethysmogram Using a Spectro-Temporal Deep Neural Network. *Sensors* **2019**, *19*, 3420. [[CrossRef](#)] [[PubMed](#)]
6. Elgendi, M.; Fletcher, R.; Liang, Y.; Howard, N.; Lovell, N.H.; Abbott, D.; Lim, K.; Ward, R. The Use of Photoplethysmography for Assessing Hypertension. *npj Digit. Med.* **2019**, *2*, 60. [[CrossRef](#)] [[PubMed](#)]
7. Kurylyak, Y.; Lamonaca, F.; Grimaldi, D. A Neural Network-Based Method for Continuous Blood Pressure Estimation from a PPG Signal. In Proceedings of the 2013 IEEE International Instrumentation and Measurement Technology Conference, Minneapolis, MN, USA, 6–9 May 2013; pp. 280–283. [[CrossRef](#)]
8. Hsu, Y.-C.; Li, Y.-H.; Chang, C.-C.; Harfiya, L.N. Generalized Deep Neural Network Model for Cuffless Blood Pressure Estimation with Photoplethysmogram Signal Only. *Sensors* **2020**, *20*, 5668. [[CrossRef](#)] [[PubMed](#)]
9. Tanveer, M.S.; Hasan, M.K. Cuffless Blood Pressure Estimation from Electrocardiogram and Photoplethysmogram Using Waveform Based ANN-LSTM Network. *Biomed. Signal Process. Control* **2019**, *51*, 382–392. [[CrossRef](#)]
10. Hasanzadeh, N.; Ahmadi, M.M.; Mohammadzade, H. Blood Pressure Estimation Using Photoplethysmogram Signal and Its Morphological Features. *IEEE Sens. J.* **2020**, *20*, 4300–4310. [[CrossRef](#)]
11. Chowdhury, M.H.; Shuzan, M.N.I.; Chowdhury, M.E.H.; Mahbub, Z.B.; Uddin, M.M.; Khandakar, A.; Reaz, M.B.I. Estimating Blood Pressure from the Photoplethysmogram Signal and Demographic Features Using Machine Learning Techniques. *Sensors* **2020**, *20*, 3127. [[CrossRef](#)] [[PubMed](#)]

12. El-Hajj, C.; Kyriacou, P.A. Cuffless Blood Pressure Estimation from PPG Signals and Its Derivatives Using Deep Learning Models. *Biomed. Signal Process. Control* **2021**, *70*, 102984. [\[CrossRef\]](#)
13. Finnegan, E.; Davidson, S.; Harford, M.; Watkinson, P.; Tarassenko, L.; Villarroel, M. Features from the Photoplethysmogram and the Electrocardiogram for Estimating Changes in Blood Pressure. *Sci. Rep.* **2023**, *13*, 986. [\[CrossRef\]](#) [\[PubMed\]](#)
14. Poh, M.-Z.; McDuff, D.J.; Picard, R.W. Non-Contact, Automated Cardiac Pulse Measurements Using Video Imaging and Blind Source Separation. *Opt. Express* **2010**, *18*, 10762–10774. [\[CrossRef\]](#) [\[PubMed\]](#)
15. de Haan, G.; Jeanne, V. Robust Pulse Rate from Chrominance-Based rPPG. *IEEE Trans. Biomed. Eng.* **2013**, *60*, 2878–2886. [\[CrossRef\]](#) [\[PubMed\]](#)
16. Wang, W.; den Brinker, A.C.; Stuijk, S.; de Haan, G. Algorithmic Principles of Remote PPG. *IEEE Trans. Biomed. Eng.* **2017**, *64*, 1479–1491. [\[CrossRef\]](#) [\[PubMed\]](#)
17. de Haan, G.; van Leest, A. Improved Motion Robustness of Remote-PPG by Using the Blood Volume Pulse Signature. *Physiol. Meas.* **2014**, *35*, 1913–1926. [\[CrossRef\]](#) [\[PubMed\]](#)
18. Kamshilin, A.A.; Nippolainen, E.; Sidorov, I.S.; Vasilev, P.V.; Erofeev, N.P.; Podolian, N.P.; Romashko, R.V. A New Look at the Essence of the Imaging Photoplethysmography. *Sci. Rep.* **2015**, *5*, 10494. [\[CrossRef\]](#) [\[PubMed\]](#)
19. Moço, A.V.; Stuijk, S.; de Haan, G. New Insights into the Origin of Remote PPG Signals in Visible Light and Infrared. *Sci. Rep.* **2018**, *8*, 8501. [\[CrossRef\]](#) [\[PubMed\]](#)
20. Lin, Y.-C.; Lin, Y.-H. A Study of Color Illumination Effect on the SNR of rPPG Signals. In Proceedings of the 2017 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, Seogwipo, Republic of Korea, 11–15 July 2017; pp. 4301–4304. [\[CrossRef\]](#) [\[PubMed\]](#)
21. Li, Y.-H.; Harfiya, L.N.; Purwandari, K.; Lin, Y.-D. Real-Time Cuffless Continuous Blood Pressure Estimation Using Deep Learning Model. *Sensors* **2020**, *20*, 5606. [\[CrossRef\]](#) [\[PubMed\]](#)
22. El Hajj, C.; Kyriacou, P.A. Cuffless and Continuous Blood Pressure Estimation From PPG Signals Using Recurrent Neural Networks. In Proceedings of the 2020 42nd Annual International Conference of the IEEE Engineering in Medicine and Biology Society, Montreal, QC, Canada, 20–24 July 2020; pp. 4269–4272. [\[CrossRef\]](#) [\[PubMed\]](#)
23. Baek, S.; Jang, J.; Yoon, S. End-to-End Blood Pressure Prediction via Fully Convolutional Networks. *IEEE Access* **2019**, *7*, 185458–185468. [\[CrossRef\]](#)
24. Eom, H.; Lee, D.; Han, S.; Hariyani, Y.S.; Lim, Y.; Sohn, I.; Park, K.; Park, C. End-To-End Deep Learning Architecture for Continuous Blood Pressure Estimation Using Attention Mechanism. *Sensors* **2020**, *20*, 2338. [\[CrossRef\]](#) [\[PubMed\]](#)
25. Yang, S.; Sohn, J.; Lee, S.; Lee, J.; Kim, H.C. Estimation and Validation of Arterial Blood Pressure Using Photoplethysmogram Morphology Features in Conjunction with Pulse Arrival Time in Large Open Databases. *IEEE J. Biomed. Health Inform.* **2021**, *25*, 1018–1030. [\[CrossRef\]](#) [\[PubMed\]](#)
26. Su, P.; Ding, X.-R.; Zhang, Y.-T.; Liu, J.; Miao, F.; Zhao, N. Long-Term Blood Pressure Prediction with Deep Recurrent Neural Networks. In Proceedings of the 2018 IEEE EMBS International Conference on Biomedical & Health Informatics, Las Vegas, NV, USA, 4–7 March 2018; pp. 323–328. [\[CrossRef\]](#)
27. Jain, M.; Deb, S.; Subramanyam, A.V. Face Video Based Touchless Blood Pressure and Heart Rate Estimation. In Proceedings of the 2016 IEEE 18th International Workshop on Multimedia Signal Processing, Montreal, QC, Canada, 21–23 September 2016; pp. 1–5. [\[CrossRef\]](#)
28. Luo, H.; Yang, D.; Barszczyk, A.; Vempala, N.; Wei, J.; Wu, S.J.; Zheng, P.P.; Fu, G.; Lee, K.; Feng, Z.-P. Smartphone-Based Blood Pressure Measurement Using Transdermal Optical Imaging Technology. *Circ. Cardiovasc. Imaging* **2019**, *12*, e008857. [\[CrossRef\]](#) [\[PubMed\]](#)
29. Schrumpp, F.; Frenzel, P.; Aust, C.; Osterhoff, G.; Fuchs, M. Assessment of Deep Learning Based Blood Pressure Prediction from PPG and rPPG Signals. In Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, Nashville, TN, USA, 19–25 June 2021; pp. 3815–3825. [\[CrossRef\]](#)
30. Geifman, Y.; El-Yaniv, R. SelectiveNet: A Deep Neural Network with an Integrated Reject Option. In *Proceedings of the 36th International Conference on Machine Learning, Long Beach, CA, USA, 9–15 June 2019*; Proceedings of Machine Learning Research: Cambridge, MA, USA, 2019; Volume 97, pp. 2151–2159.
31. Zaoui, A.; Denis, C.; Hebiri, M. Regression with Reject Option and Application to kNN. In *Advances in Neural Information Processing Systems 33*; Curran Associates, Inc.: Red Hook, NY, USA, 2020.
32. Lim, P.K.; Ng, S.-C.; Lovell, N.H.; Yu, Y.P.; Tan, M.P.; McCombie, D.; Lim, E.; Redmond, S.J. Adaptive Template Matching of Photoplethysmogram Pulses to Detect Motion Artefact. *Physiol. Meas.* **2018**, *39*, 105005. [\[CrossRef\]](#) [\[PubMed\]](#)
33. Elgendi, M.; Martinelli, I.; Menon, C. Optimal Signal Quality Index for Remote Photoplethysmogram Sensing. *npj Biosens.* **2024**, *1*, 5. [\[CrossRef\]](#)
34. Schrumpp, F.; Frenzel, P.; Aust, C.; Osterhoff, G.; Fuchs, M. Assessment of Non-Invasive Blood Pressure Prediction from PPG and rPPG Signals Using Deep Learning. *Sensors* **2021**, *21*, 6022. [\[CrossRef\]](#) [\[PubMed\]](#)

35. Schrumppf, F. Non-Invasive Blood Pressure Estimation Using Deep Learning. Available online: <https://github.com/Fabian-Sc85/non-invasive-bp-estimation-using-deep-learning> (accessed on 4 June 2026).
36. Hochreiter, S.; Schmidhuber, J. Long Short-Term Memory. *Neural Comput.* **1997**, *9*, 1735–1780. [CrossRef] [PubMed]
37. Schuster, M.; Paliwal, K.K. Bidirectional Recurrent Neural Networks. *IEEE Trans. Signal Process.* **1997**, *45*, 2673–2681. [CrossRef]
38. Srivastava, N.; Hinton, G.; Krizhevsky, A.; Sutskever, I.; Salakhutdinov, R. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *J. Mach. Learn. Res.* **2014**, *15*, 1929–1958.
39. Bahdanau, D.; Cho, K.; Bengio, Y. Neural Machine Translation by Jointly Learning to Align and Translate. In Proceedings of the International Conference on Learning Representations, San Diego, CA, USA, 7–9 May 2015.
40. Kingma, D.P.; Ba, J. Adam: A Method for Stochastic Optimization. In Proceedings of the International Conference on Learning Representations, San Diego, CA, USA, 7–9 May 2015.
41. Hampel, F.R. The Influence Curve and Its Role in Robust Estimation. *J. Am. Stat. Assoc.* **1974**, *69*, 383–393. [CrossRef]
42. Hoerl, A.E.; Kennard, R.W. Ridge Regression: Biased Estimation for Nonorthogonal Problems. *Technometrics* **1970**, *12*, 55–67. [CrossRef]
43. Friedman, J.H. Greedy Function Approximation: A Gradient Boosting Machine. *Ann. Stat.* **2001**, *29*, 1189–1232. [CrossRef]
44. Bland, J.M.; Altman, D.G. Statistical Methods for Assessing Agreement between Two Methods of Clinical Measurement. *Lancet* **1986**, *1*, 307–310. [CrossRef]
45. Wilcoxon, F. Individual Comparisons by Ranking Methods. *Biom. Bull.* **1945**, *1*, 80–83. [CrossRef]
46. Holm, S. A Simple Sequentially Rejective Multiple Test Procedure. *Scand. J. Stat.* **1979**, *6*, 65–70.
47. Kim, S.; Lim, H.; Baek, J.; Lee, E.C. RGB Camera-Based Blood Pressure Measurement Using U-Net Basic Generative Model. *Electronics* **2023**, *12*, 3771. [CrossRef]
48. Stergiou, G.S.; Alpert, B.; Mieke, S.; Asmar, R.; Atkins, N.; Eckert, S.; Frick, G.; Friedman, B.; Graßl, T.; Ichikawa, T.; et al. A Universal Standard for the Validation of Blood Pressure Measuring Devices: Association for the Advancement of Medical Instrumentation/European Society of Hypertension/International Organization for Standardization (AAMI/ESH/ISO) Collaboration Statement. *Hypertension* **2018**, *71*, 368–374. [CrossRef] [PubMed]
49. O'Brien, E.; Petrie, J.; Littler, W.; de Swiet, M.; Padfield, P.L.; O'Malley, K.; Jamieson, M.; Altman, D.; Bland, M.; Atkins, N. The British Hypertension Society Protocol for the Evaluation of Automated and Semi-Automated Blood Pressure Measuring Devices with Special Reference to Ambulatory Systems. *J. Hypertens.* **1990**, *8*, 607–619. [CrossRef] [PubMed]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.