A Project Report

ON

Cuffless Blood Pressure Estimation using ECG and PPG Digital Signals

Report in fulfillment of

the requirements for the summer internship

in

Machine Learning/ Deep Learning

Submitted by

Manu Mishra and Ishtiyaque Alam

Under the guidance of

Dr. PRIYA RANJAN MUDULI

DEPARTMENT OF ELECTRONICS ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY (BHU)
VARANASI, UTTAR PRADESH, INDIA -221005

JUNE 2024 – JULY 2024

COPYRIGHT TRANSFER CERTIFICATE

Title of the Project: Cuffless Blood Pressure Estimation using cardiovascular signals.

Name and Organization of the Student: Manu Mishra (VIT Chennai) and Ishtiyaque Alam (NIT Warangal)

COPYRIGHT TRANSFER

The undersigned hereby assigns to the Indian Institute of Technology (Banaras Hindu University) Varanasi all rights under copyright that may exist in and for the above project report of the work performed in the summer internship.

However, the author may reproduce or authorize others to reproduce material extracted verbatim from the report or derivative of the report for author's personal use provided that the source and the Institute's copyright notice are indicated. The publication of the work presented here on any online platform in any form is subject to the permission from the project supervisor at the Indian Institute of Technology (Banaras Hindu University) Varanasi.

Place: Varanasi
Date: 30.06.2024

Manu Mishra &
Ishtiyaque Alam
Name of the Student

SWIN Transformer Based Cuffless Blood Pressure Estimation using Bispectrum and Bicoherence Features

ABSTRACT

Cuffless blood pressure (BP) estimation presents a promising non-invasive method for continuous BP monitoring, which can significantly enhance the management of cardiovascular health. This report explores the use of electrocardiogram (ECG) and photoplethysmogram (PPG) signals to estimate systolic blood pressure (SBP) and diastolic blood pressure (DBP) without the need for traditional cuff-based methods. We transform the ECG and PPG signals into bispectrum and bicoherence images, which capture the nonlinear interactions within the signals. A SWIN Vision Transformer model is then applied to these images to predict BP values. Our approach demonstrates high accuracy and reliability, offering a practical solution for continuous BP monitoring in clinical and home settings.

INTRODUCTION

Hypertension, commonly known as high blood pressure, is a critical risk factor for cardiovascular diseases, including heart attack and stroke. Traditional blood pressure monitoring methods rely on cuff-based measurements, which can be uncomfortable and impractical for continuous monitoring. There is a growing demand for cuffless BP monitoring techniques that can provide continuous, non-invasive, and accurate BP readings.

Electrocardiogram (ECG) and photoplethysmogram (PPG) signals are widely used in medical diagnostics due to their non-invasive nature and ease of acquisition. ECG measures the electrical activity of the heart, while PPG monitors blood volume changes in the microvascular bed of tissue. Combining these signals for BP estimation leverages their complementary information, potentially leading to more accurate and reliable results.

In this report, we propose a novel approach for cuffless BP estimation by converting ECG and PPG signals into bispectrum and bicoherence images. These images represent the nonlinear interactions within the signals, providing a rich feature set for BP estimation. We then apply a SWIN Vision Transformer model to predict systolic blood pressure (SBP) and diastolic blood pressure (DBP) from these images. The SWIN Vision Transformer, with its hierarchical structure and attention mechanism, is well-suited for capturing complex patterns in medical imaging data.

METHODOLOGY

ECG and PPG signals are collected from subjects using standard non-invasive sensors and preprocessed to remove noise and artifacts. These clean signals are then transformed into bispectrum and bicoherence images, capturing critical phase relationships and nonlinear interactions for accurate BP estimation. A SWIN Vision Transformer model, with a hierarchical architecture featuring shifted windows, is developed to process these images. The model is trained using a supervised learning approach to predict SBP and DBP values. The dataset is split into training and validation sets, and performance is assessed using metrics such as mean absolute error (MAE) and root mean square error (RMSE). Cross-validation ensures model robustness. The model's performance is compared with existing cuffless BP estimation methods through statistical analysis, and its potential for real-world applications is evaluated. Finally, the model is implemented in a prototype system for continuous BP monitoring, assessing feasibility and usability in clinical or home settings, considering factors such as comfort, ease of use, and integration with existing healthcare systems. This methodology highlights the potential of advanced signal processing and machine learning techniques for cuffless BP estimation, offering promising solutions for hypertension management.

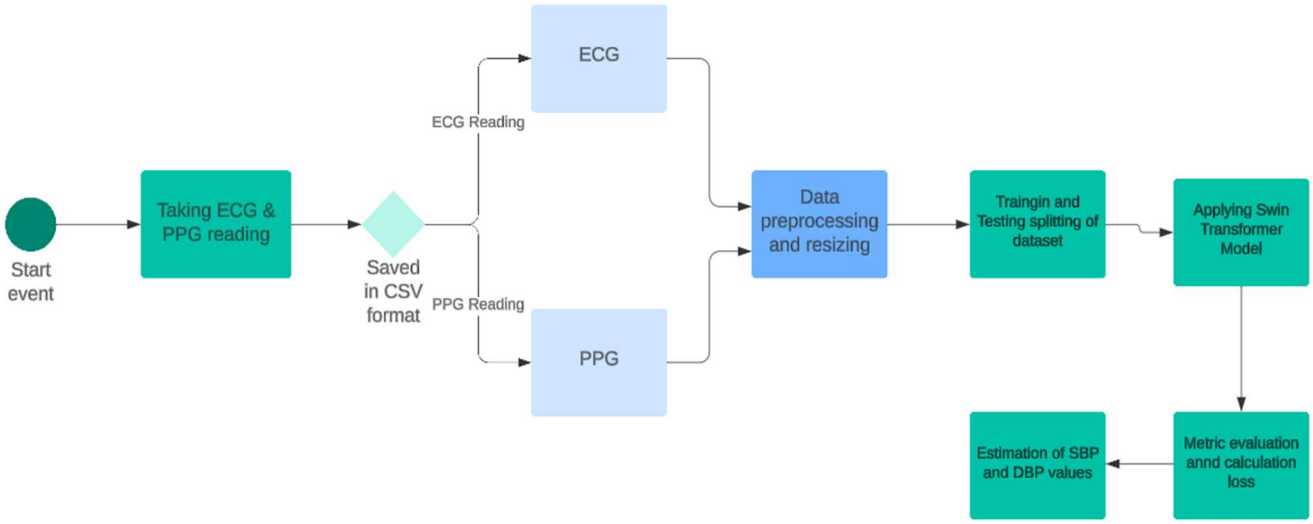


Fig. Flow diagram of model implementation.

ECG Datasets

The electrocardiogram (ECG) dataset comprises recordings that capture the electrical activity of the heart over time. These recordings are typically collected using non-invasive sensors placed on the chest, arms, and legs of subjects. ECG signals provide detailed insights into the heart's rhythm, rate, and electrical conduction pathways. The dataset includes various features of the ECG waveform, such as the P-wave, QRS complex, and T-wave, each corresponding to specific cardiac events. The ECG data is essential for understanding the temporal and morphological characteristics of heartbeats, which are crucial for estimating systolic blood pressure (SBP) and diastolic blood pressure (DBP). The relationship between ECG signals and BP arises from the heart's electrical activity influencing blood flow and pressure dynamics within the cardiovascular system.

PPG Datasets

The photoplethysmogram (PPG) dataset contains recordings that measure blood volume changes in the microvascular bed of tissue. These signals are typically obtained using optical sensors placed on the skin, such as the fingertip or earlobe. PPG signals reflect the cardiac cycle by capturing variations in blood flow, which are influenced by the heartbeat. The dataset includes features like the amplitude and timing of the pulse wave, which correspond to the mechanical actions of the heart and the arterial system. The relationship between PPG signals and blood pressure (SBP and DBP) is based on the fact that blood volume changes correlate with the heart's pumping action and vascular resistance. By analyzing PPG waveforms, it is possible to infer BP values, as these signals provide indirect but informative measures of cardiovascular health.

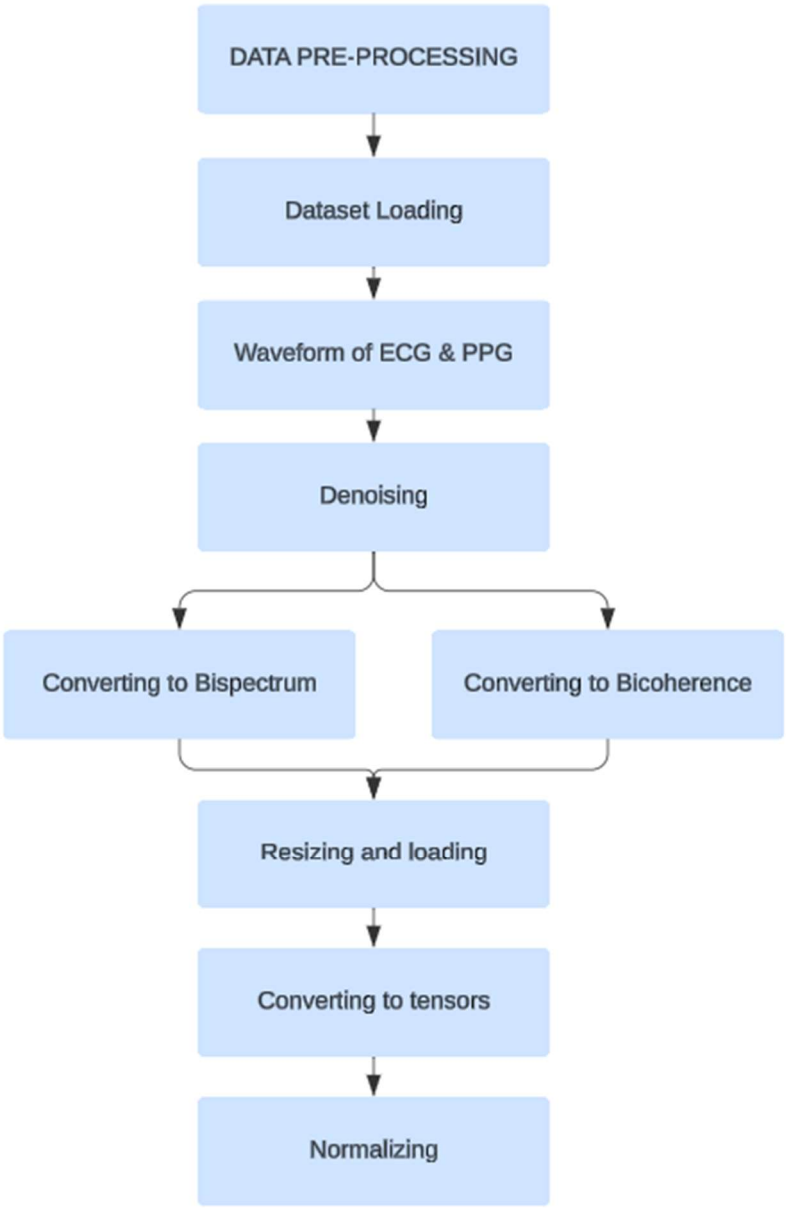
Data Pre-Processing

The preprocessing steps ensure that the ECG and PPG images are loaded, resized to a uniform size, converted to tensors, and normalized appropriately. This standardization allows the images to be fed into the Swin Transformer model for feature extraction, which is then used for predicting the SBP and DBP values. These preprocessing steps are crucial for ensuring the model performs optimally and can generalize well on unseen data.

- **Resizing Images:** Ensures all input images are of uniform size (224x224 pixels).
- **Converting to Tensors:** Converts images into a format suitable for PyTorch models.
- **Normalizing Images:** Standardizes pixel values to match the distribution of the pre-trained model's training data.

• **Custom Dataset Class:** Handles the loading and preprocessing of ECG and PPG images and their corresponding SBP and DBP values.

• **Data Loader:** Batches the data and provides efficient data loading during training and evaluation.



Dataset Visualization:

The dataset of ECG and PPG values has been recorded in the excel / csv that has been converted to the waveform and after that these waveforms are converted to Bispectrum and Bicoherence plots using the Fourier series.

Waveform:

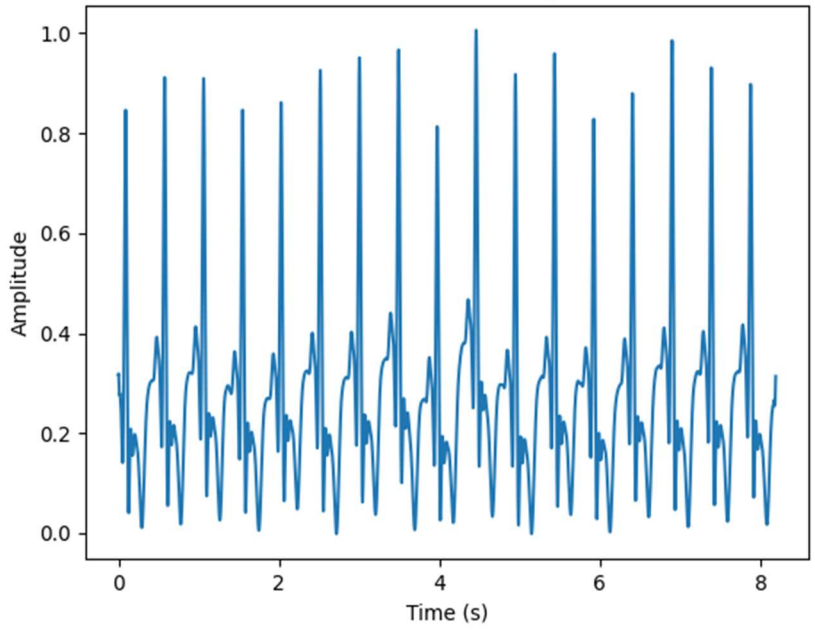


Fig. Waveform of ECG signals

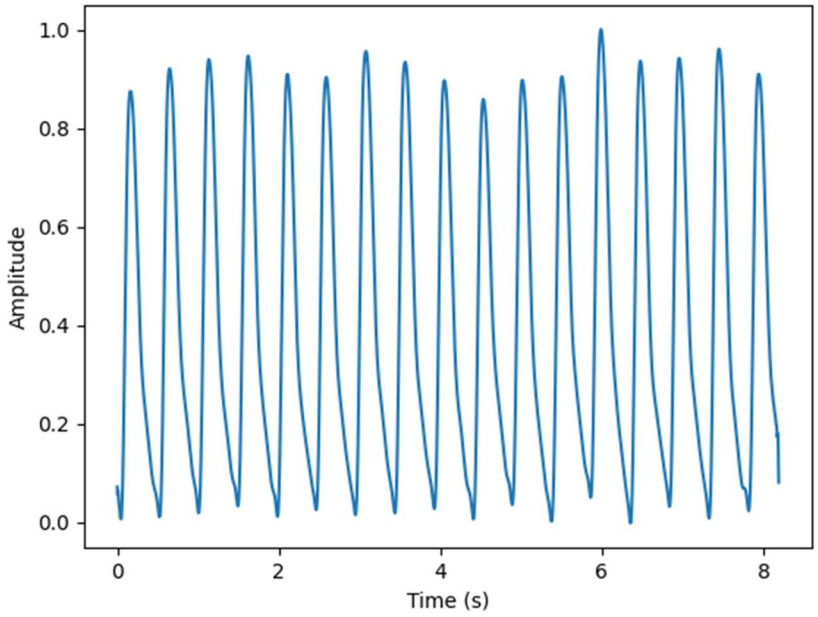


Fig. Waveform of PPG signals

Bispectrum:

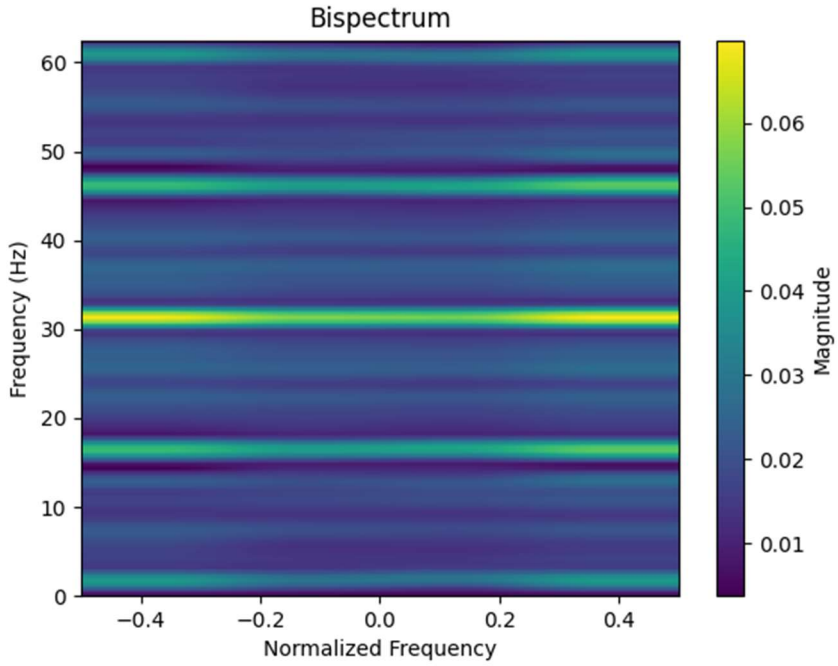


Fig. Bispectrum of ECG Signals

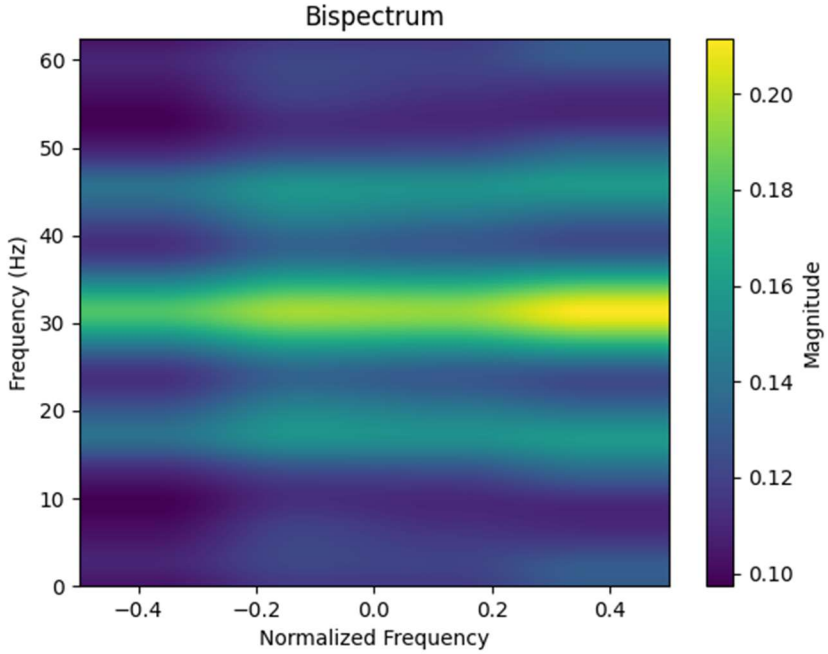


Fig. Bispectrum of PPG Signals

Bicoherence:

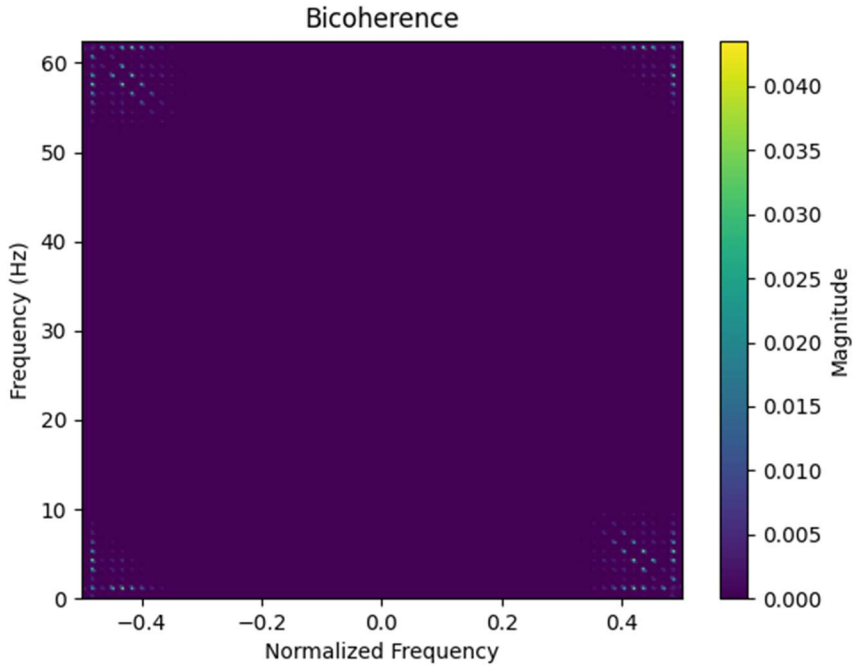


Fig. Bicoherence of ECG Signal

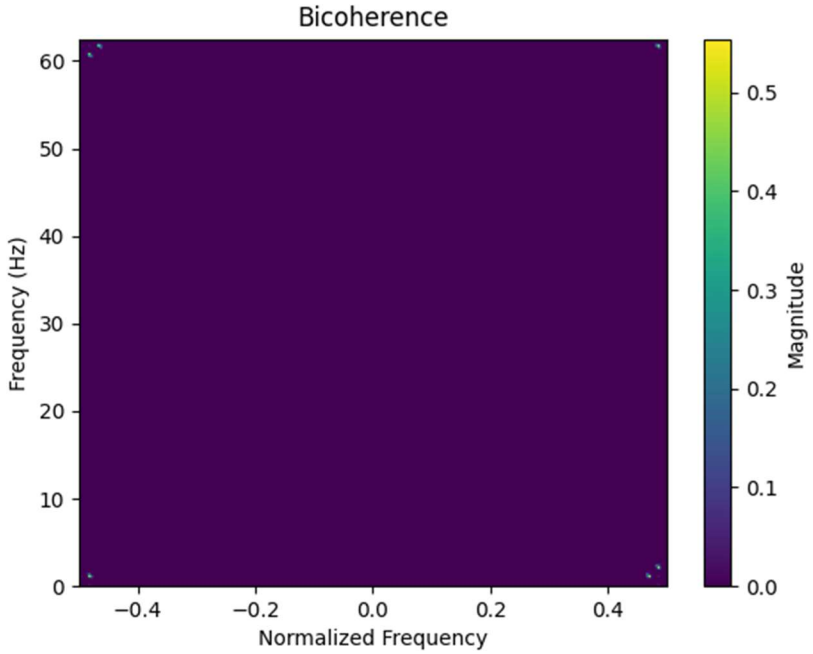


Fig. Bicoherence of PPG Signal

FEATURE EXTRACTION:

Feature extraction from bispectrum and bicoherence images using a pretrained Swin Transformer leverages the model's ability to handle complex patterns in these specialized images. The bispectrum captures non-linear interactions between different frequency components of a signal, while the bicoherence normalizes this measure to highlight phase coupling. The Swin Transformer processes these images by breaking them into smaller windows and using self-attention to capture relationships within and between these windows. The model's hierarchical structure and shifting window mechanism enable it to effectively capture both local and global features, providing detailed and robust representations of the bispectrum and bicoherence images for further analysis and classification.

Bispectrum formula:

Mathematically, the bispectrum $B(f_1, f_2)$ for a signal $x(t)$ is given by:

$$B(f_1, f_2) = \mathbb{E}[X(f_1) \cdot X(f_2) \cdot X^*(f_1 + f_2)]$$

where $X(f)$ denotes the Fourier transform of $x(t)$, and $X^*(f)$ is its complex conjugate.

Bicoherence formula:

Mathematically, the bicoherence $b(f_1, f_2)$ is defined as:

$$b(f_1, f_2) = \frac{|B(f_1, f_2)|^2}{S(f_1) \cdot S(f_2) \cdot S(f_1 + f_2)}$$

where $S(f)$ is the power spectral density of the signal.

Self-attention mechanism:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

Attention Score:

$$A_{ij} = \text{softmax}\left(\frac{Q_i K_j^T}{\sqrt{d_k}}\right)$$

SWIN Transformer:

The Swin Transformer is a sophisticated vision transformer designed to handle high-resolution image data by introducing a hierarchical structure and window-based self-attention mechanism. It divides the input image into small patches, processes these patches in a hierarchical manner through multiple stages, and performs self-attention within local windows to reduce computational complexity. The model further employs shifted windows to ensure information flows across different parts of the image, capturing both local and global features effectively. This architecture allows the Swin Transformer to excel in various image-based tasks such as classification, detection, and segmentation by leveraging its ability to learn detailed and abstract representations from high-resolution images. The model is typically pretrained on large datasets and fine-tuned for specific tasks, making it a versatile tool in the field of computer vision.

How is the Swin Transformer Used?

The Swin Transformer is used by first pretraining the model on a large dataset (such as ImageNet) and then fine-tuning it for specific tasks. This process involves:

1. **Data Preparation**:
    - Images are converted into patches, similar to tokens in natural language processing.
    - These patches are then embedded into vectors to serve as inputs to the model.
2. **Model Training**:
    - The model is trained using supervised learning techniques, where it learns to minimize the error between its predictions and the actual labels.
3. **Fine-Tuning**:
    - The pretrained model is further trained on a task-specific dataset to adapt its learned features to the specific requirements of the task.

Overall Architecture

The Swin Transformer (SWIN) is a hierarchical vision transformer that addresses two key challenges: extracting features at variable scales and maintaining linear computational complexity. The shifted window mechanism is the core innovation that enables SWIN to achieve these goals.

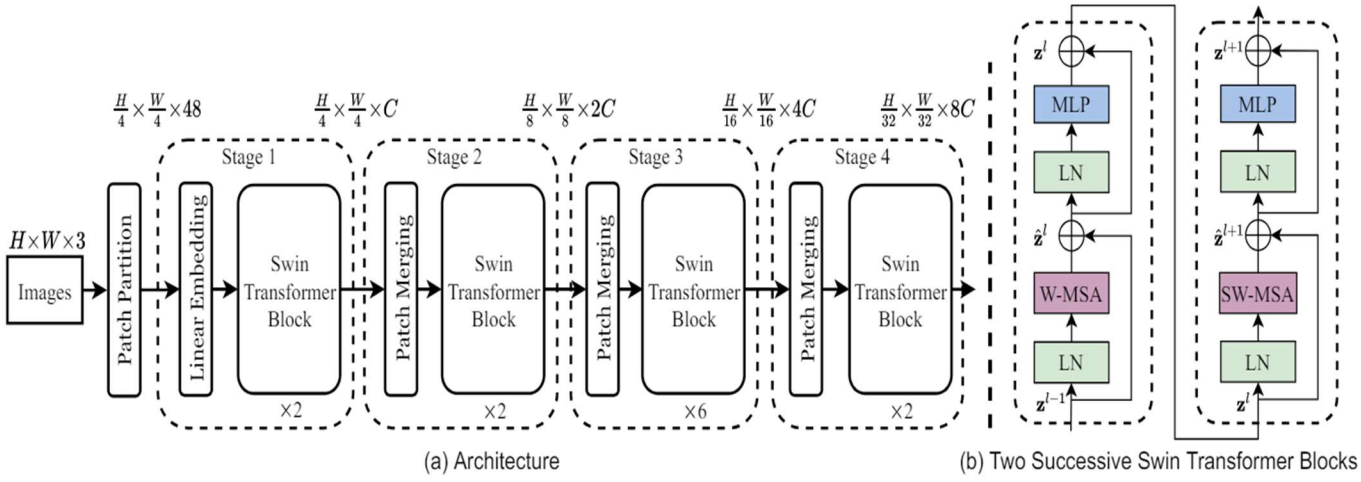
The input image is divided into non-overlapping windows, and each window is further divided into patches. The self-attention mechanism is applied within these windows, allowing the model to focus on smaller-scale features. This helps SWIN extract features at variable scales, in contrast to the fixed-scale feature representation in the standard Vision Transformer (ViT).

To maintain linear computational complexity, the shifted window mechanism ensures that the number of patches remains fixed, regardless of the image size. This is a significant improvement over the quadratic complexity of the standard ViT.

The SWIN architecture consists of several key components. First, the input image is split into non-overlapping patches, and a linear layer converts each patch into an embedding token. Then, the Swin Transformer blocks, which include a shifted window-based self-attention module and a 2-layer MLP, process these tokens. Layer normalization and residual connections are applied to stabilize the training process and improve performance.

After each stage, a patch merging layer is used to reduce the number of tokens by concatenating features from neighboring windows and reducing the dimensions. This hierarchical feature representation allows SWIN to extract features at different scales, unlike the fixed-scale feature representation in ViT.

The shifted window mechanism and the hierarchical feature representation enable SWIN to capture both local and global context, while maintaining linear computational complexity with respect to the image size. This makes SWIN a powerful and efficient vision transformer architecture for a wide range of computer vision tasks.



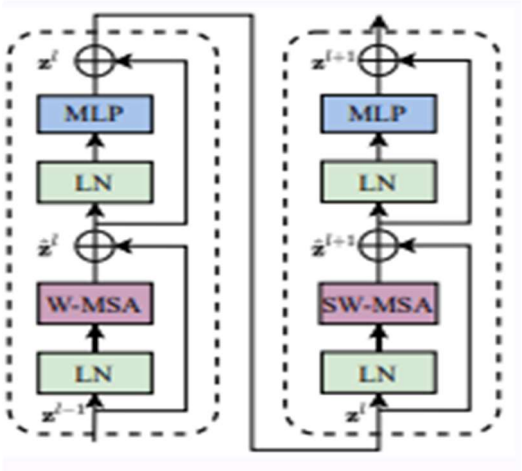
(a) Architecture

(b) Two Successive Swin Transformer Blocks

Swin Transformer block:

The Swin Transformer block replaces the standard multi-head self-attention (MSA) module in a traditional Transformer block with a shifted window-based MSA module, a Swin Transformer block comprises a shifted window-based MSA module followed by a 2-layer MLP with GELU nonlinearity in between.

A Layer Norm (LN) layer is applied before each MSA module and MLP, and a residual connection is added after each module.



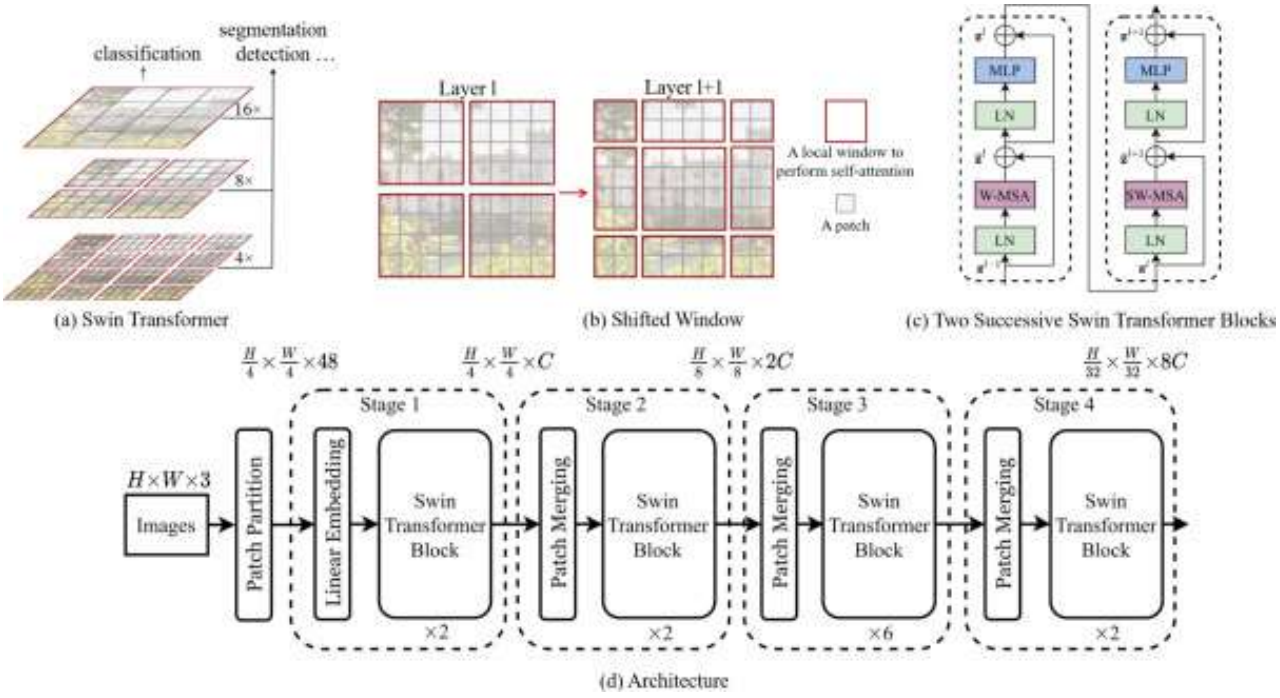
Shifted Window Multi-head attention:

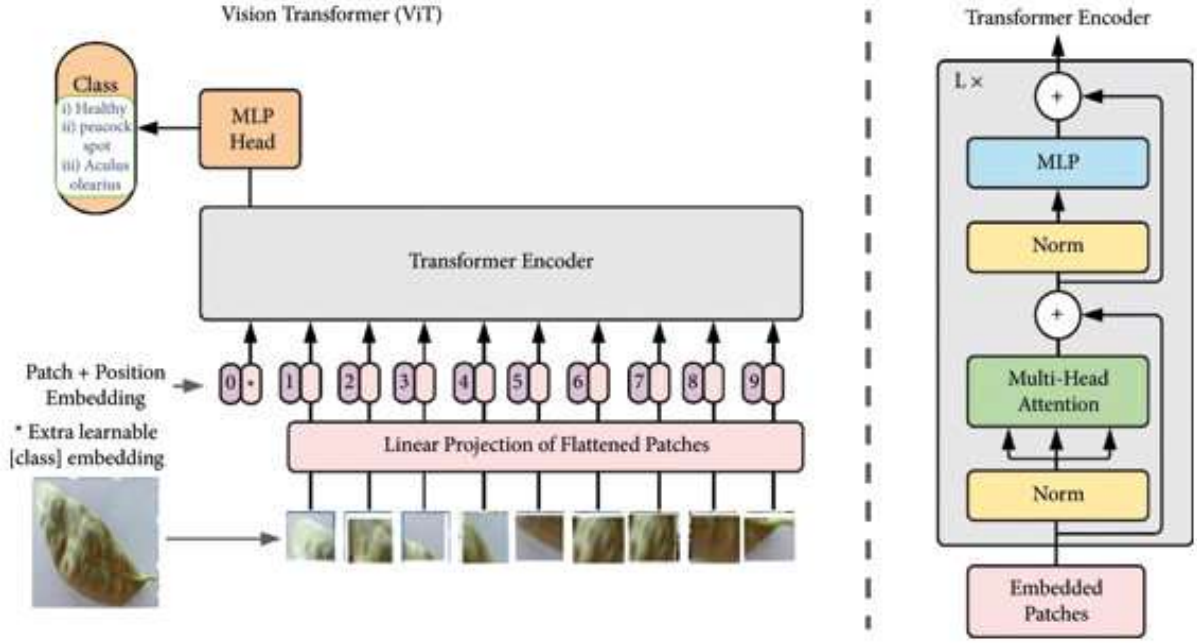
The Swin Transformer replaces the standard multi-head self-attention used in traditional transformers with shifted window multi-head self-attention. The other layers, such as the MLP and Layer Normalization, remain unchanged. This approach means that self-attention is performed only within local windows, which are further divided into patches of size M×MM \times MM×M. This ensures that the computational complexity of the model remains linear as long as MMM is constant. In the Swin Transformer, MMM is set to 7. The following equation illustrates the difference in computational complexity between the traditional multi-head self-attention and the shifted window self-attention as a function of the image size h×w:

$$\Omega(\text{MSA}) = 4hwC^2 + 2(hw)^2C,$$
$$\Omega(\text{W-MSA}) = 4hwC^2 + 2M^2hwC,$$

The multi-head self-attention MSA increases quadratically with the image dimension hxw, while the shifted window multi-head self-attention W-MSA increases linearly with the image dimensions hxw if the variable M is constant. This makes SWIN transformer more suitable for training on images with higher resolutions. However, the downside to window-based self-attention is the lack of self-attention connections outside the local window, limiting the model's capability to capture global contexts. SWIN addresses this issue by shifting the windows with respect to each other in successive SWIN blocks, allowing cross window connections and gaining a global context for the relationships between patches across the windows.

$$\hat{\mathbf{z}}^l = \text{W-MSA}\left(\text{LN}\left(\mathbf{z}^{l-1}\right)\right) + \mathbf{z}^{l-1},$$
$$\mathbf{z}^l = \text{MLP}\left(\text{LN}\left(\hat{\mathbf{z}}^l\right)\right) + \hat{\mathbf{z}}^l,$$
$$\hat{\mathbf{z}}^{l+1} = \text{SW-MSA}\left(\text{LN}\left(\mathbf{z}^l\right)\right) + \mathbf{z}^l,$$
$$\mathbf{z}^{l+1} = \text{MLP}\left(\text{LN}\left(\hat{\mathbf{z}}^{l+1}\right)\right) + \hat{\mathbf{z}}^{l+1},$$


classification
segmentation
detection …
16×
8×
4×
Layer l
Layer l+1
A local window to perform self-attention
A patch
MLP
LN
W-MSA
SW-MSA
LN
MLP
LN
SW-MSA
LN
(a) Swin Transformer
(b) Shifted Window
(c) Two Successive Swin Transformer Blocks
$\frac{H}{4}\times\frac{W}{4}\times 48$
$\frac{H}{4}\times\frac{W}{4}\times C$
$\frac{H}{8}\times\frac{W}{8}\times 2C$
$\frac{H}{32}\times\frac{W}{32}\times 8C$
Stage 1
Stage 2
Stage 3
Stage 4
$H\times W\times 3$
Images
Patch Partition
Linear Embedding
Swin Transformer Block
×2
Patch Merging
Swin Transformer Block
×2
Patch Merging
Swin Transformer Block
×6
Patch Merging
Swin Transformer Block
×2
(d) Architecture


Vision Transformer (ViT)
Class
i) Healthy
ii) peacock spot
iii) Aculus olearius
MLP Head
Transformer Encoder
Patch + Position Embedding
* Extra learnable [class] embedding
0
1
2
3
4
5
6
7
8
9
Linear Projection of Flattened Patches
Transformer Encoder
L ×
MLP
Norm
Multi-Head Attention
Norm
Embedded Patches

IMPLEMENTATION OF OUR MODEL & ALGORITHM:

In our model workflow, we start by importing the necessary data and preprocessing the dataset to ensure it's suitable for training. This involves loading the dataset images and transforming them by resizing each image to 224x224 pixels, a standard size for many vision models. The images are then converted into tensors, which are the fundamental data structure for PyTorch operations.

We utilize a pretrained Swin Transformer model, specifically the **swin_base_patch4_window7_224**, which provides a robust starting point due to its pre-learned features on a large dataset. Our training objective is to minimize the Mean Absolute Error (MAE) loss, a metric suitable for regression tasks. To optimize the model, we use the Adam optimizer with a learning rate set to 1e-4, balancing the speed of convergence and stability. The training process is conducted over 300 epochs, allowing the model ample time to learn and refine its weights. Throughout the training, we plot and print the results to monitor the model's performance, ensuring that the learning process is progressing as expected. This comprehensive approach leverages advanced techniques and rigorous evaluation to build a powerful image processing model.

Swin Model parameter:

- Loss function = Mean Absolute Error
- Batch Size for training = 32
- Batch Size for testing = 32
- Training Epoch = 300
- Adam Optimizer is used with Learning rate = 0.0001 (1e-4)
- Test data = 20%

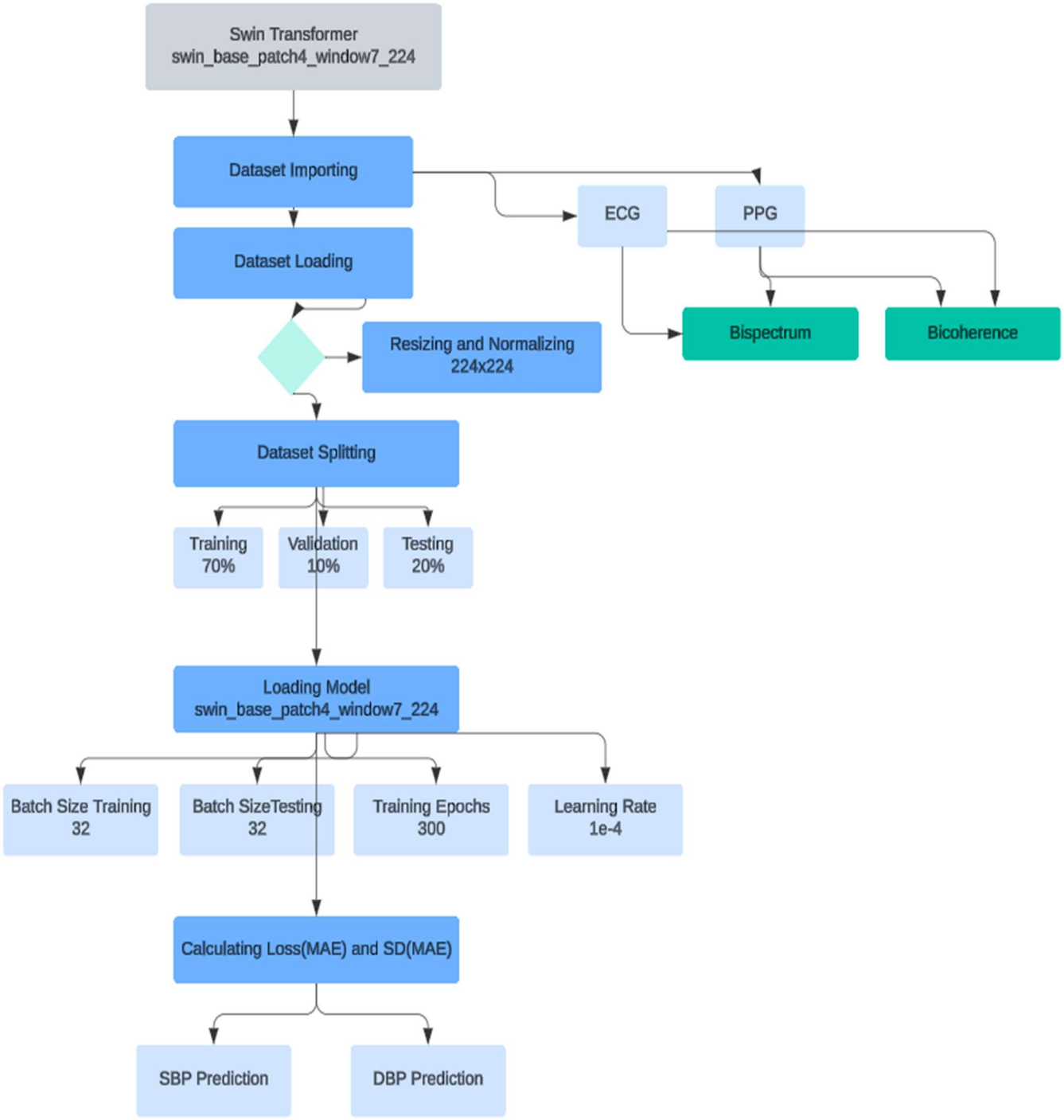
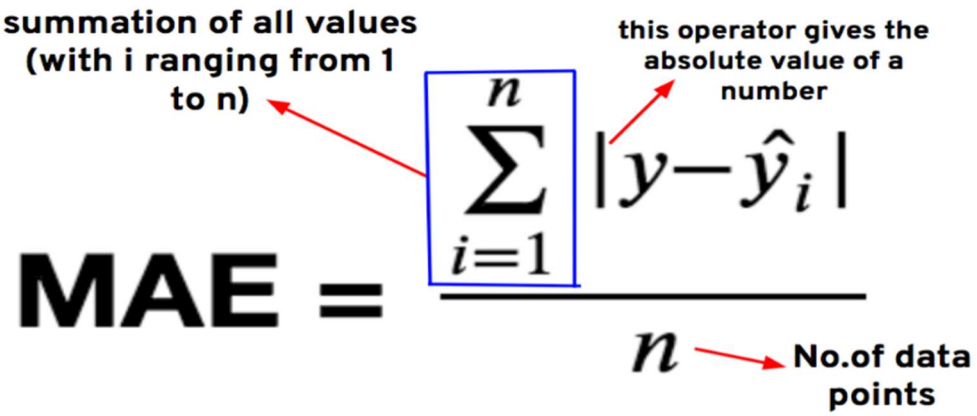


Fig. Representation of Algorithm working with pretrain Swin Transformer

RESULTS -

Blood pressure prediction is a regression analysis and can be evaluated by mean absolute error (MAE), and standard deviation (STD). The MAE is the average of the absolute values of the errors between the predicted value and true blood pressure. Corresponding calculation of the above parameters are as follows:

$$\text{MAE} = \frac{\text{Sum of absolute errors}}{\text{Number of data points}}$$



y = actual value, $\hat{y}$ = predicted value

$$s = \sqrt{\frac{\sum_{i=1}^{N} (x_i - \overline{x})^2}{N - 1}}$$

s = sample standard deviation

N = the number of observations

x_i = the observed values of a sample item

$\overline{x}$ = the mean value of the observations

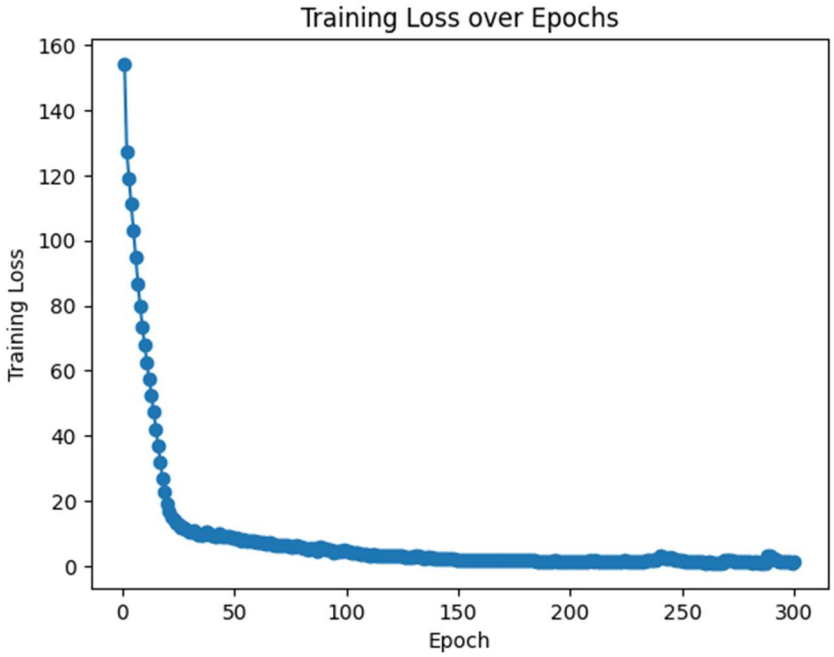
We have computed the loss with MAE function and the standard deviation of the MAE values for the Bispectrum and Bicoherence dataset of ECG and PPC both Signals

Result of Bispectrum of ECG dataset

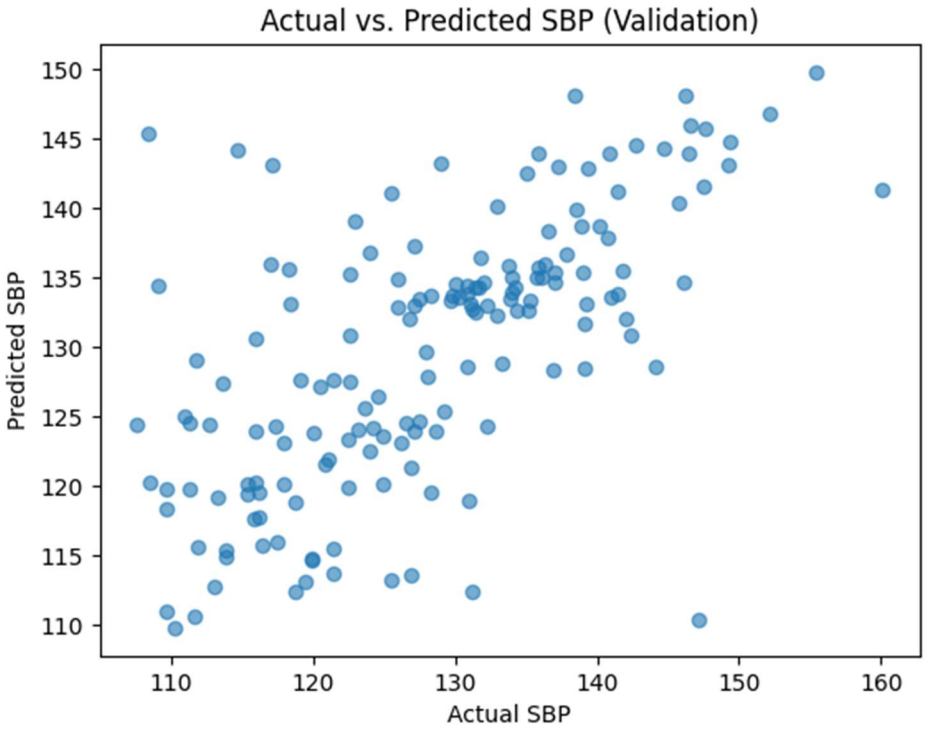
Bispectrum of ECG	SBP		DBP	
	MAE	SD	MAE	SD
Validation	6.2706	NA	3.1633	NA
Test	5.6583	5.4238	2.8581	3.5888

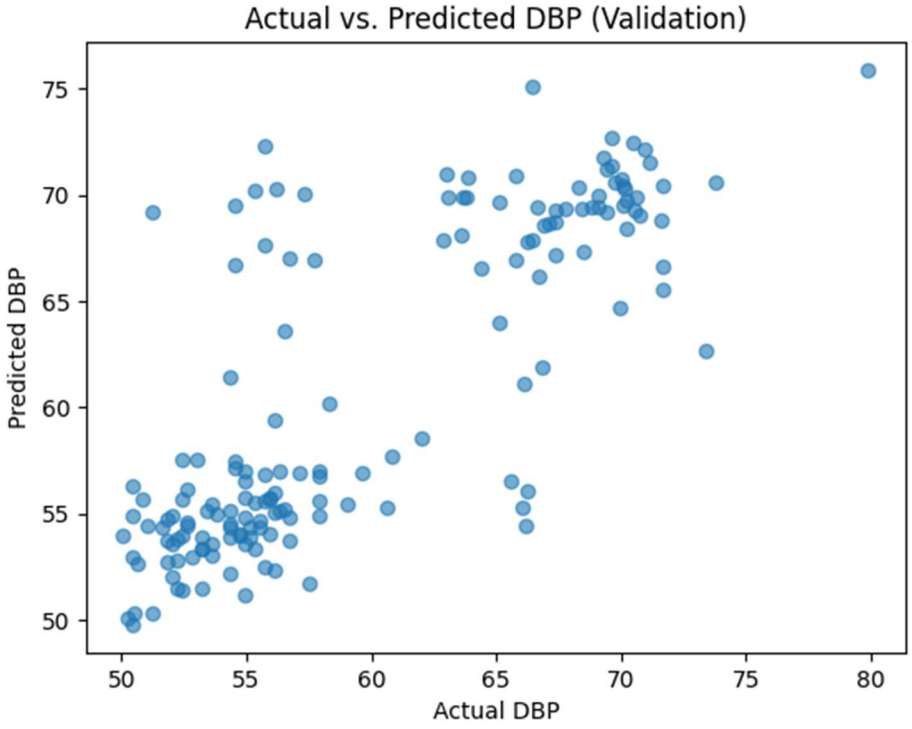
Test Loss: 8.5164

```
Validation SBP Mean Absolute Error (MAE): 6.2706
Validation DBP Mean Absolute Error (MAE): 3.1633
Test SBP Mean Absolute Error (MAE): 5.6583
Test DBP Mean Absolute Error (MAE): 2.8581
Test Loss: 8.5164
Test Standard Deviation of Error (SBP): 5.4238
Test Standard Deviation of Error (DBP): 3.5888
```

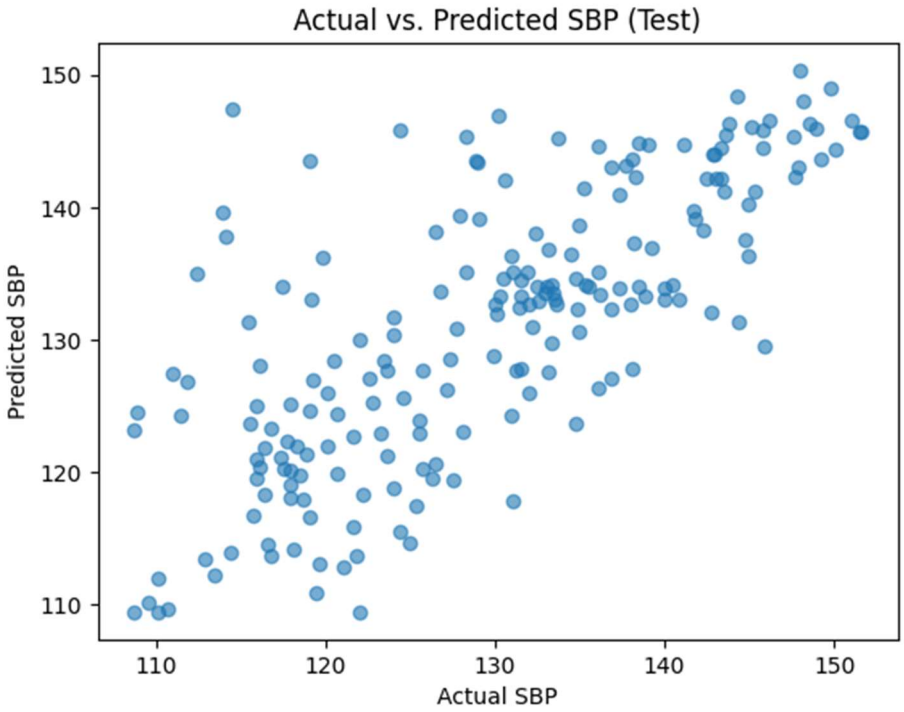


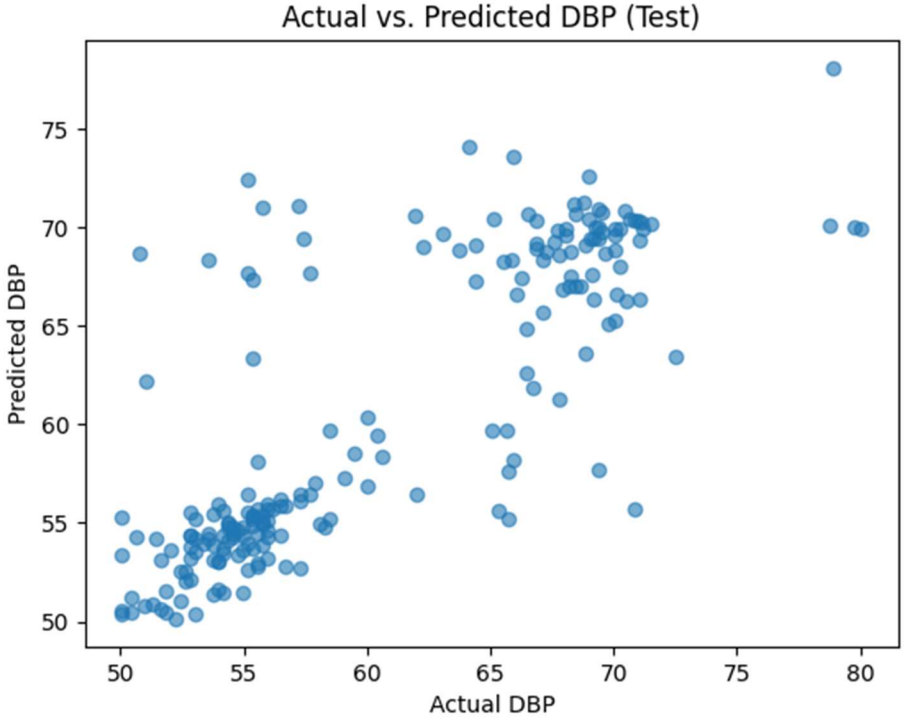
Validation Plots:





Test Plots:


Actual vs. Predicted SBP (Test)
Predicted SBP
Actual SBP
150
140
130
120
110
110
120
130
140
150


Actual vs. Predicted DBP (Test)
Predicted DBP
Actual DBP
75
70
65
60
55
50
50
55
60
65
70
75
80

Predicted SBP and DBP values (for 10 columns):

Actual vs Predicted SBP values (first 10 samples from validation set):
Actual: 120.08, Predicted: 123.89
Actual: 131.05, Predicted: 133.12
Actual: 122.94, Predicted: 139.04
Actual: 121.06, Predicted: 121.98
Actual: 123.60, Predicted: 125.68
Actual: 128.10, Predicted: 127.90
Actual: 125.95, Predicted: 132.83
Actual: 128.29, Predicted: 133.76
Actual: 122.62, Predicted: 135.28
Actual: 111.67, Predicted: 110.61

Actual vs Predicted DBP values (first 10 samples from validation set):
Actual: 53.59, Predicted: 53.62
Actual: 70.19, Predicted: 70.30
Actual: 63.60, Predicted: 68.13
Actual: 52.22, Predicted: 53.83
Actual: 55.35, Predicted: 55.51
Actual: 53.78, Predicted: 54.98
Actual: 53.59, Predicted: 55.48
Actual: 55.74, Predicted: 56.81
Actual: 53.00, Predicted: 57.55
Actual: 57.89, Predicted: 55.63

Actual vs Predicted SBP values (first 10 samples from test set):
Actual: 143.35, Predicted: 144.54
Actual: 108.93, Predicted: 124.54
Actual: 142.23, Predicted: 138.28
Actual: 151.51, Predicted: 145.73
Actual: 117.93, Predicted: 118.04
Actual: 128.10, Predicted: 123.07
Actual: 123.99, Predicted: 118.75
Actual: 144.33, Predicted: 131.39
Actual: 111.80, Predicted: 126.77
Actual: 119.10, Predicted: 124.63

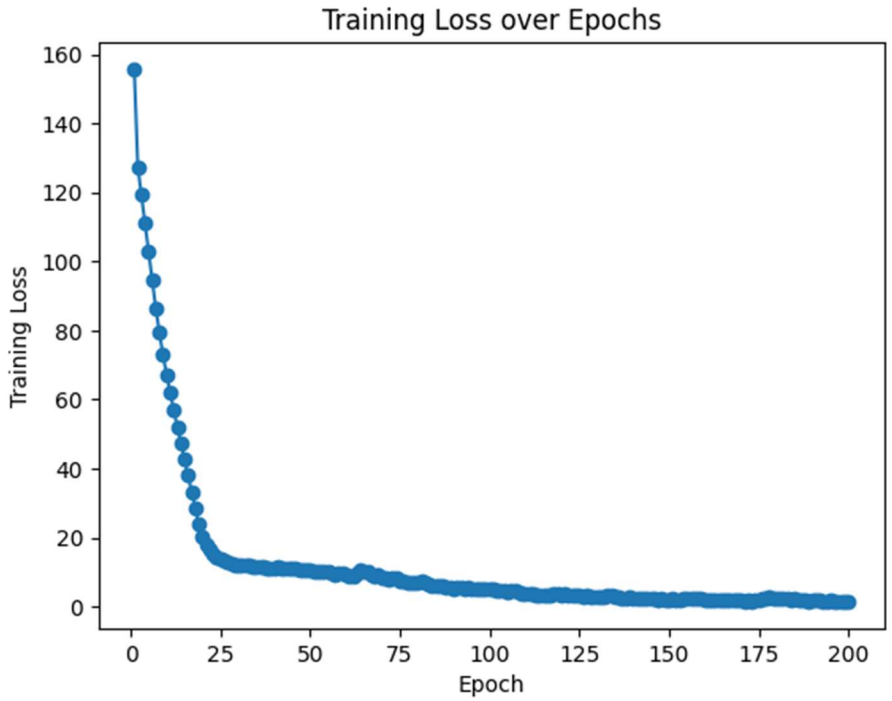
Actual vs Predicted DBP values (first 10 samples from test set):
Actual: 60.04, Predicted: 60.34
Actual: 50.07, Predicted: 55.28
Actual: 70.29, Predicted: 68.05
Actual: 70.04, Predicted: 69.89
Actual: 53.59, Predicted: 54.45
Actual: 55.74, Predicted: 55.01
Actual: 54.76, Predicted: 53.39
Actual: 56.52, Predicted: 56.19
Actual: 65.30, Predicted: 55.58
Actual: 54.56, Predicted: 54.29

Result of Bispectrum of PPG dataset

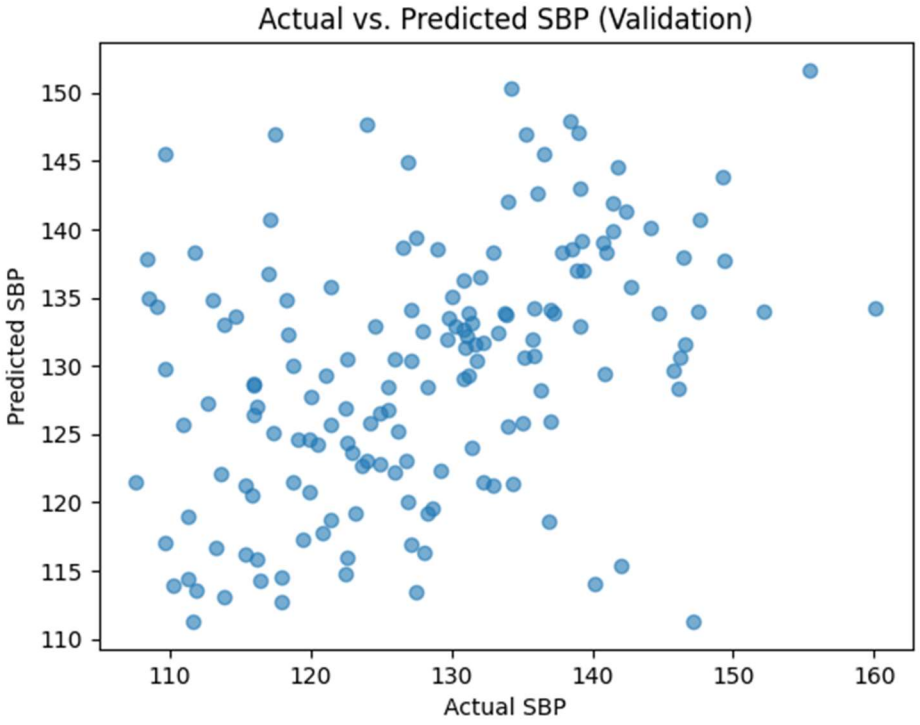
Bispectrum of PPG	SBP		DBP	
	MAE	SD	MAE	SD
Validation	8.2230	NA	3.9549	NA
Test	7.7087	6.9739	3.8276	4.2098

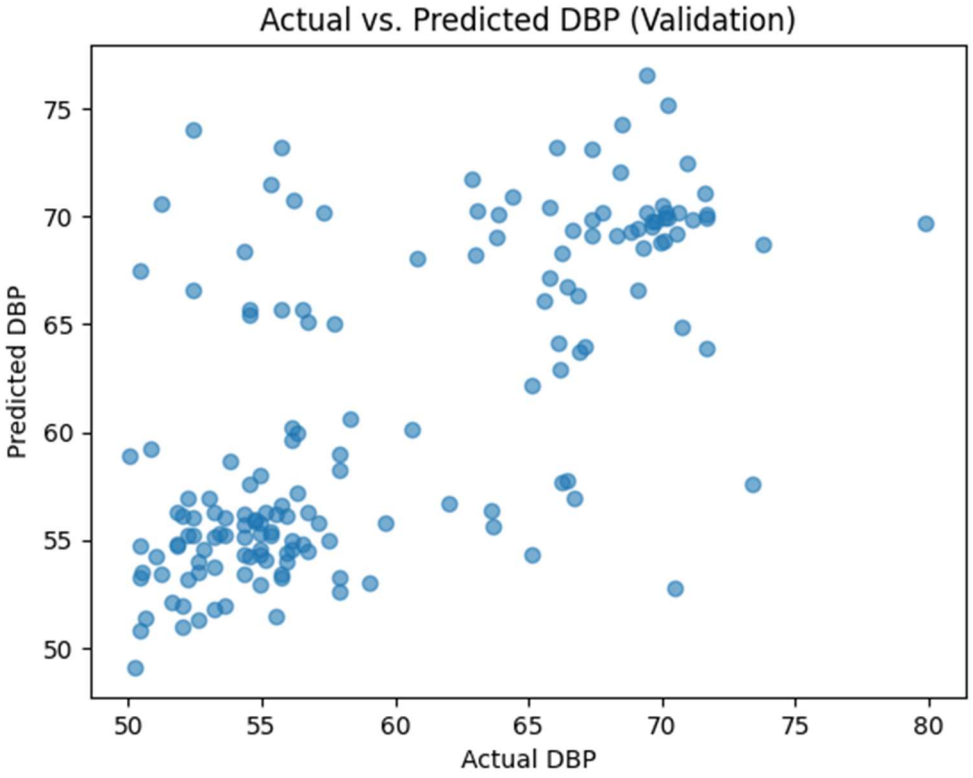
Test Loss: 11.5363

```
Validation SBP Mean Absolute Error (MAE): 8.2230
Validation DBP Mean Absolute Error (MAE): 3.9549
Test SBP Mean Absolute Error (MAE): 7.7087
Test DBP Mean Absolute Error (MAE): 3.8276
Test Loss: 11.5363
Test Standard Deviation of Error (SBP): 6.9739
Test Standard Deviation of Error (DBP): 4.2098
```

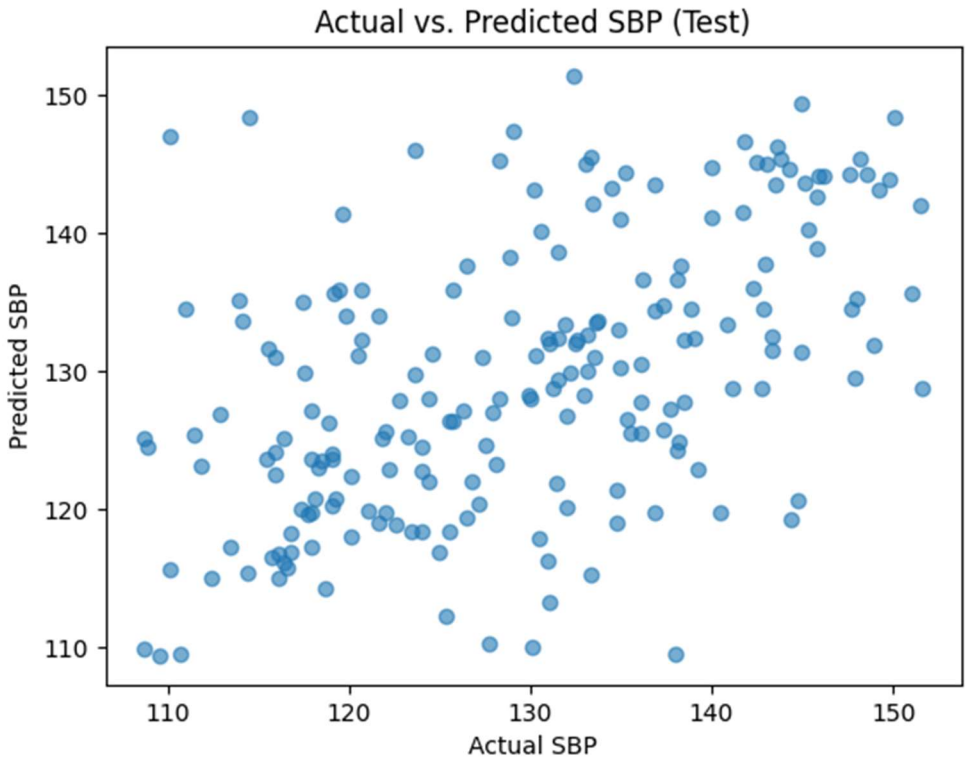


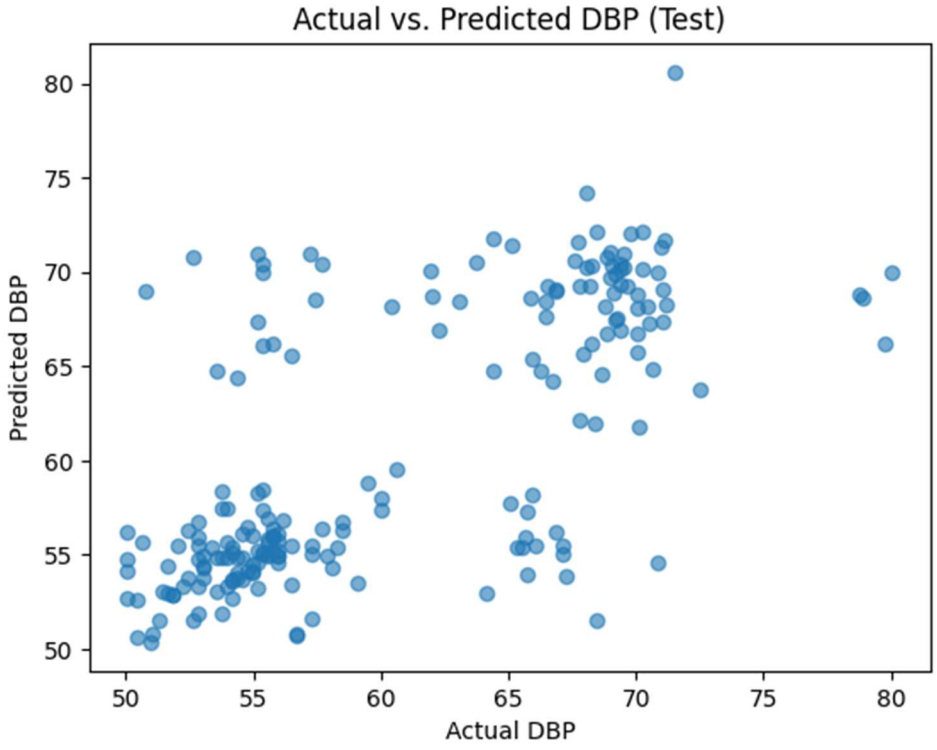
Validation Plots:





Test Plots:


Actual vs. Predicted SBP (Test)
Predicted SBP
Actual SBP
150
140
130
120
110
110
120
130
140
150


Actual vs. Predicted DBP (Test)
Predicted DBP
Actual DBP
80
75
70
65
60
55
50
50
55
60
65
70
75
80

Predicted SBP and DBP values (for 10 columns):

Actual vs Predicted SBP values (first 10 samples from validation set):
Actual: 120.08, Predicted: 127.68
Actual: 131.05, Predicted: 132.17
Actual: 122.94, Predicted: 123.70
Actual: 121.06, Predicted: 129.31
Actual: 123.60, Predicted: 122.67
Actual: 128.10, Predicted: 116.33
Actual: 125.95, Predicted: 130.51
Actual: 128.29, Predicted: 119.26
Actual: 122.62, Predicted: 130.45
Actual: 111.67, Predicted: 111.33

Actual vs Predicted DBP values (first 10 samples from validation set):
Actual: 53.59, Predicted: 55.19
Actual: 70.19, Predicted: 70.21
Actual: 63.60, Predicted: 56.38
Actual: 52.22, Predicted: 56.95
Actual: 55.35, Predicted: 55.24
Actual: 53.78, Predicted: 58.68
Actual: 53.59, Predicted: 56.07
Actual: 55.74, Predicted: 53.25
Actual: 53.00, Predicted: 56.91
Actual: 57.89, Predicted: 52.60

Actual vs Predicted SBP values (first 10 samples from test set):
Actual: 143.35, Predicted: 131.56
Actual: 108.93, Predicted: 124.52
Actual: 142.23, Predicted: 136.09
Actual: 151.51, Predicted: 142.05
Actual: 117.93, Predicted: 123.72
Actual: 128.10, Predicted: 123.28
Actual: 123.99, Predicted: 118.45
Actual: 144.33, Predicted: 119.28
Actual: 111.80, Predicted: 123.14
Actual: 119.10, Predicted: 123.73

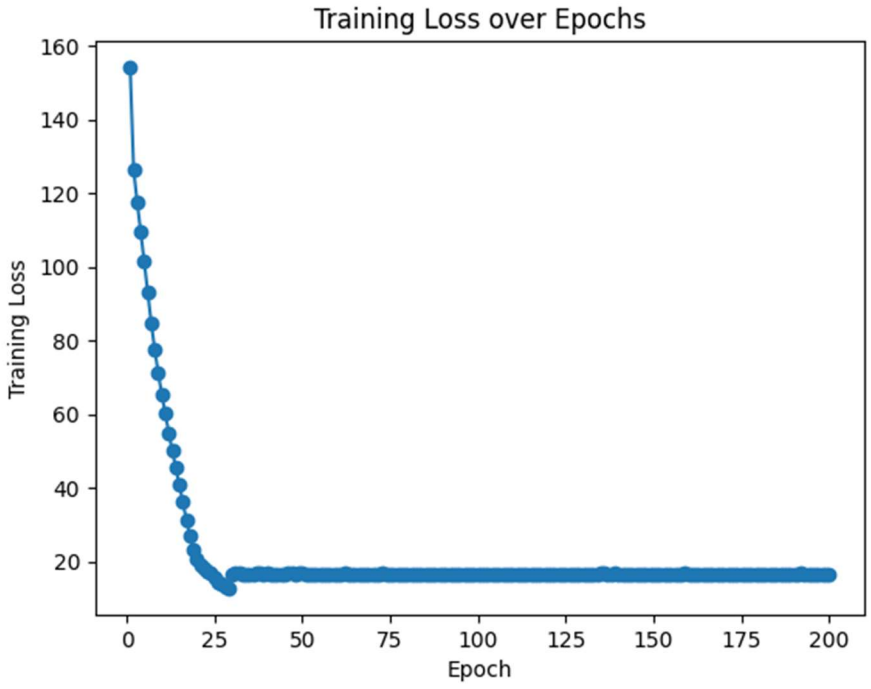
Actual vs Predicted DBP values (first 10 samples from test set):
Actual: 60.04, Predicted: 57.99
Actual: 50.07, Predicted: 56.21
Actual: 70.29, Predicted: 72.15
Actual: 70.04, Predicted: 68.77
Actual: 53.59, Predicted: 54.88
Actual: 55.74, Predicted: 55.93
Actual: 54.76, Predicted: 54.13
Actual: 56.52, Predicted: 53.45
Actual: 65.30, Predicted: 55.39
Actual: 54.56, Predicted: 54.82

Result of Bicoherence of ECG dataset

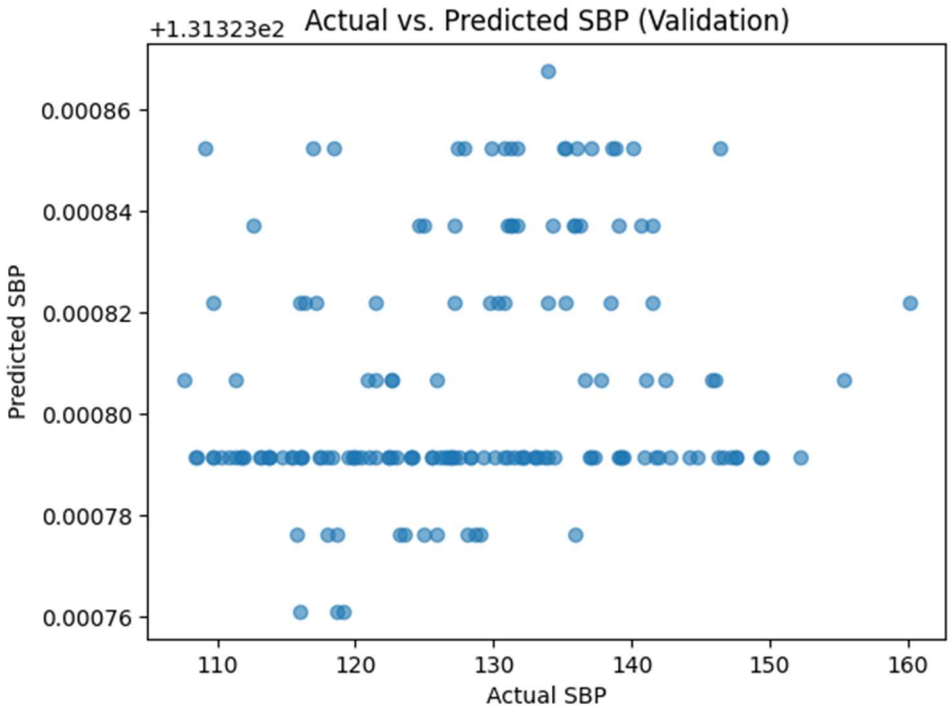
Bicoherence of ECG	SBP		DBP	
	MAE	SD	MAE	SD
Validation	9.5749	NA	6.1592	NA
Test	9.4009	6.0366	6.5008	4.6249

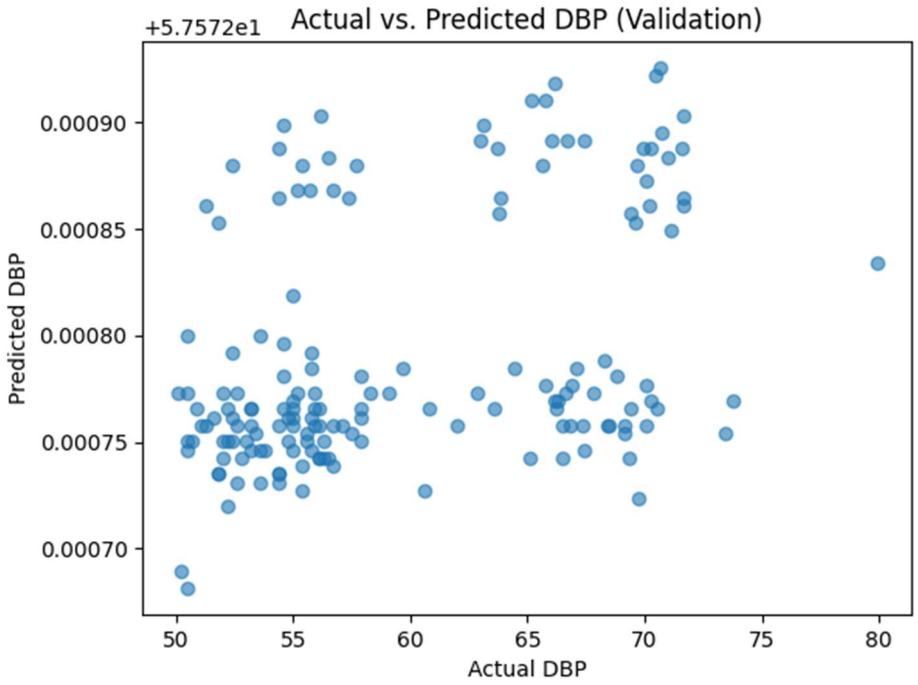
Test Loss: 15.9016

```
Validation SBP Mean Absolute Error (MAE): 9.5749
Validation DBP Mean Absolute Error (MAE): 6.1592
Test SBP Mean Absolute Error (MAE): 9.4009
Test DBP Mean Absolute Error (MAE): 6.5008
Test Loss: 15.9016
Test Standard Deviation of Error (SBP): 6.0366
Test Standard Deviation of Error (DBP): 4.6249
```

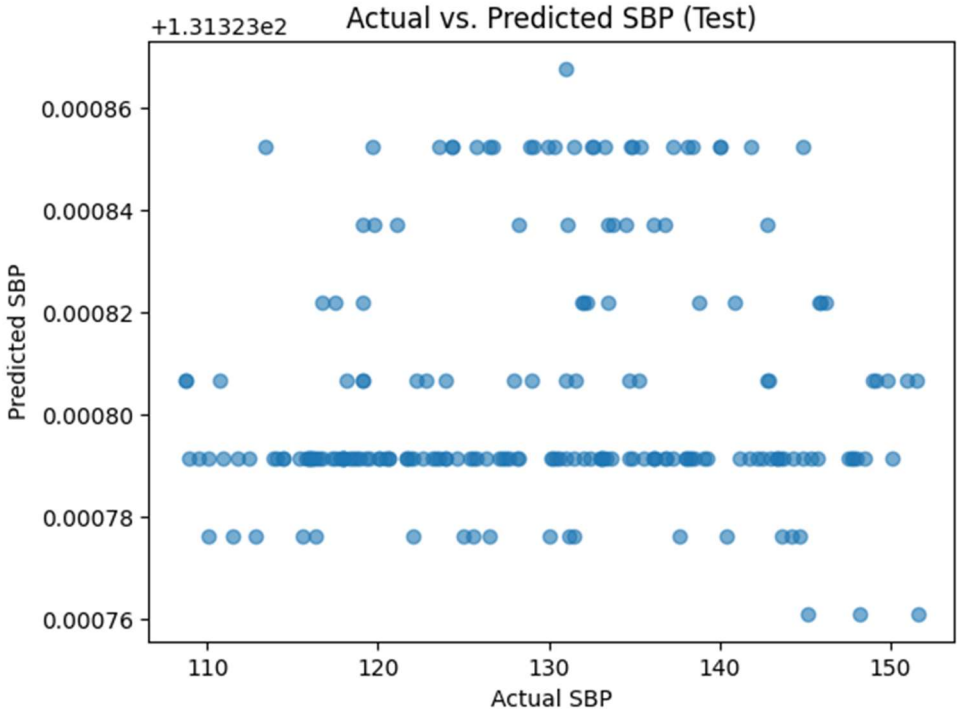


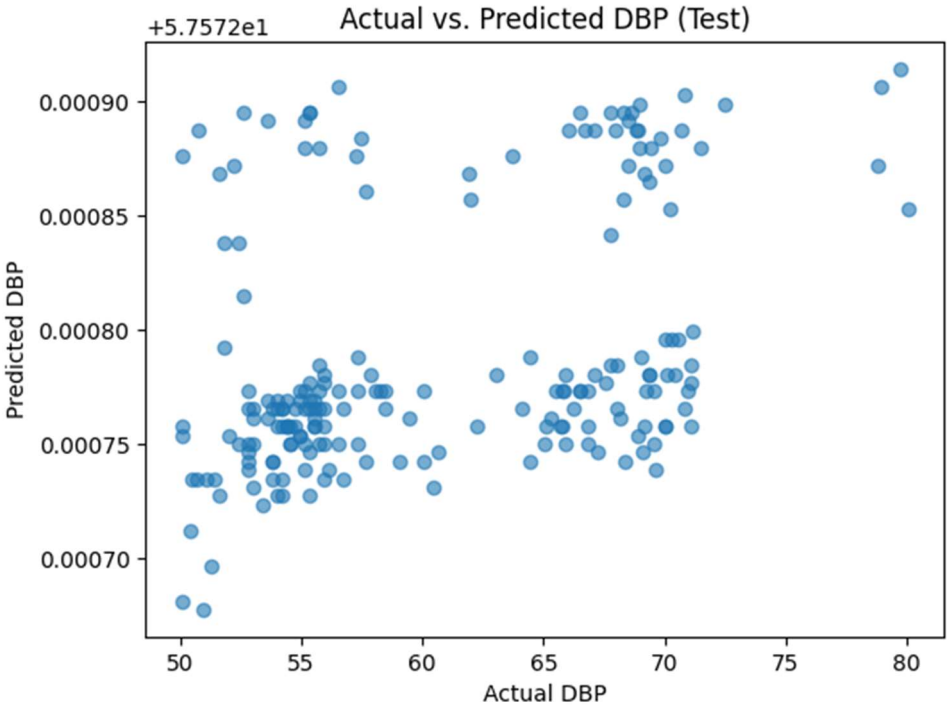
Validation Plots:





Test Plots:


Actual vs. Predicted SBP (Test)
+1.31323e2
Predicted SBP
0.00086
0.00084
0.00082
0.00080
0.00078
0.00076
110
120
130
140
150
Actual SBP


Actual vs. Predicted DBP (Test)
+5.7572e1
Predicted DBP
0.00090
0.00085
0.00080
0.00075
0.00070
50
55
60
65
70
75
80
Actual DBP

Predicted SBP and DBP values (for 10 columns):

Actual vs Predicted SBP values (first 10 samples from validation set):
Actual: 120.08, Predicted: 131.32
Actual: 131.05, Predicted: 131.32
Actual: 122.94, Predicted: 131.32
Actual: 121.06, Predicted: 131.32
Actual: 123.60, Predicted: 131.32
Actual: 128.10, Predicted: 131.32
Actual: 125.95, Predicted: 131.32
Actual: 128.29, Predicted: 131.32
Actual: 122.62, Predicted: 131.32
Actual: 111.67, Predicted: 131.32

Actual vs Predicted DBP values (first 10 samples from validation set):
Actual: 53.59, Predicted: 57.57
Actual: 70.19, Predicted: 57.57
Actual: 63.60, Predicted: 57.57
Actual: 52.22, Predicted: 57.57
Actual: 55.35, Predicted: 57.57
Actual: 53.78, Predicted: 57.57
Actual: 53.59, Predicted: 57.57
Actual: 55.74, Predicted: 57.57
Actual: 53.00, Predicted: 57.57
Actual: 57.89, Predicted: 57.57

Actual vs Predicted SBP values (first 10 samples from test set):
Actual: 143.35, Predicted: 131.32
Actual: 108.93, Predicted: 131.32
Actual: 142.23, Predicted: 131.32
Actual: 151.51, Predicted: 131.32
Actual: 117.93, Predicted: 131.32
Actual: 128.10, Predicted: 131.32
Actual: 123.99, Predicted: 131.32
Actual: 144.33, Predicted: 131.32
Actual: 111.80, Predicted: 131.32
Actual: 119.10, Predicted: 131.32

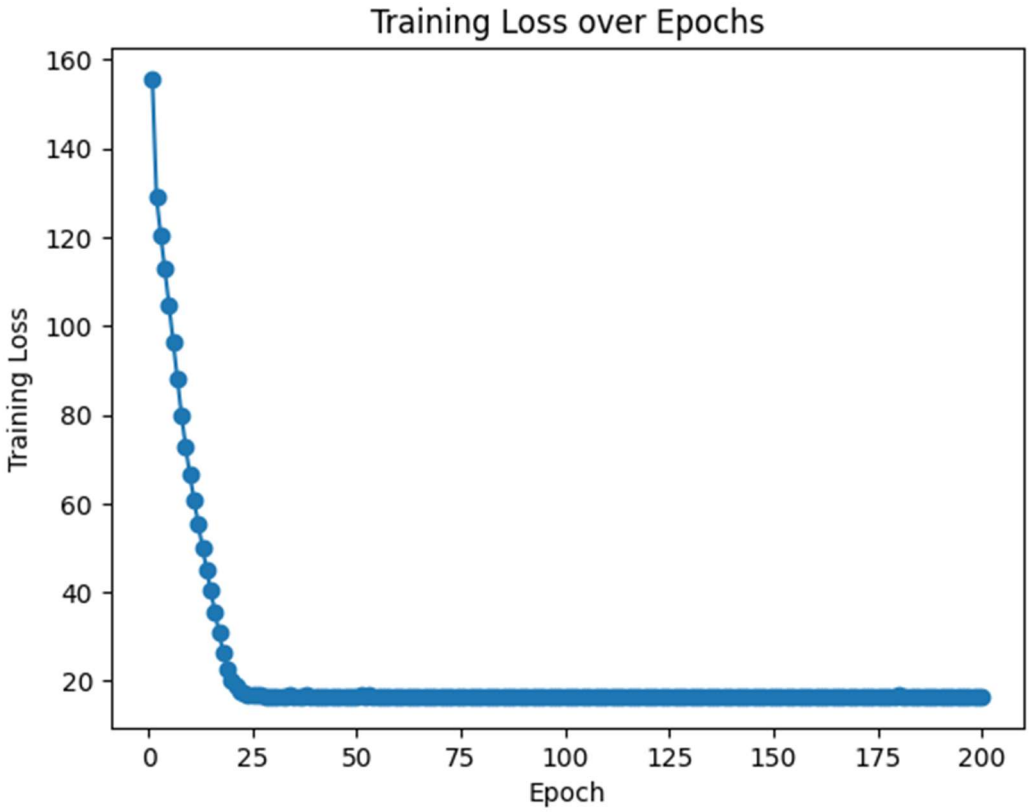
Actual vs Predicted DBP values (first 10 samples from test set):
Actual: 60.04, Predicted: 57.57
Actual: 50.07, Predicted: 57.57
Actual: 70.29, Predicted: 57.57
Actual: 70.04, Predicted: 57.57
Actual: 53.59, Predicted: 57.57
Actual: 55.74, Predicted: 57.57
Actual: 54.76, Predicted: 57.57
Actual: 56.52, Predicted: 57.57
Actual: 65.30, Predicted: 57.57
Actual: 54.56, Predicted: 57.57

Result of Bicoherence of PPG dataset

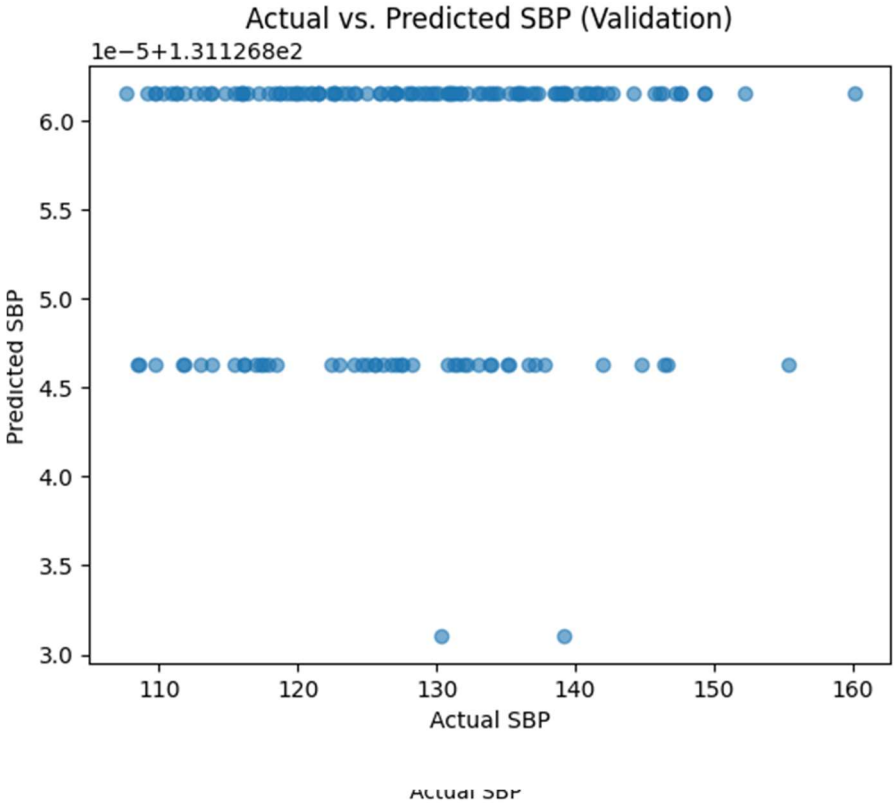
Bicoherence of PPG	SBP		DBP	
	MAE	SD	MAE	SD
Validation	9.5354	NA	6.1863	NA
Test	9.3861	6.0062	6.5085	4.4545

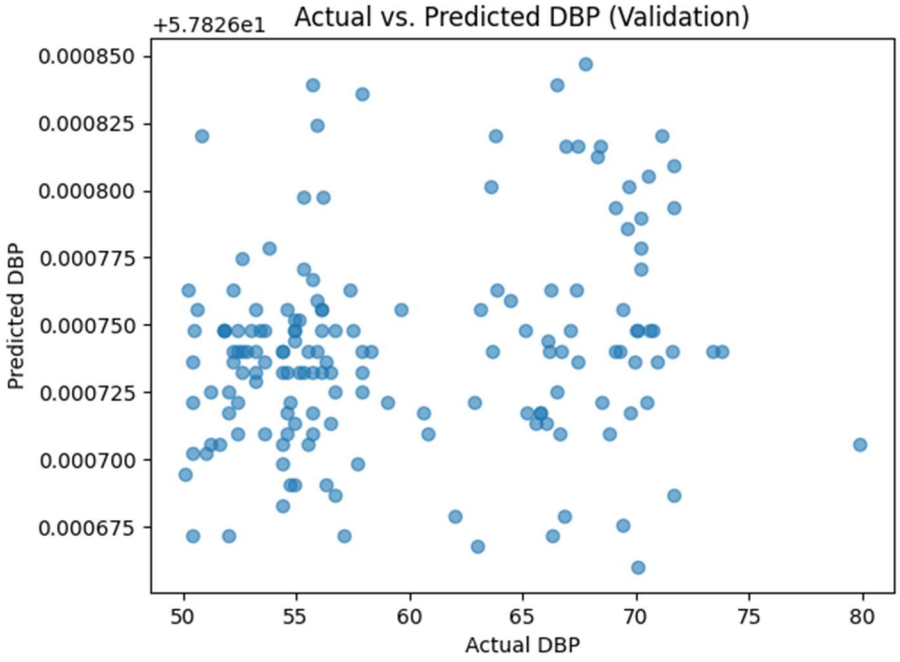
Test Loss: 15.8946

```
Validation SBP Mean Absolute Error (MAE): 9.5354
Validation DBP Mean Absolute Error (MAE): 6.1863
Test SBP Mean Absolute Error (MAE): 9.3861
Test DBP Mean Absolute Error (MAE): 6.5085
Test Loss: 15.8946
Test Standard Deviation of Error (SBP): 6.0062
Test Standard Deviation of Error (DBP): 4.4545
```

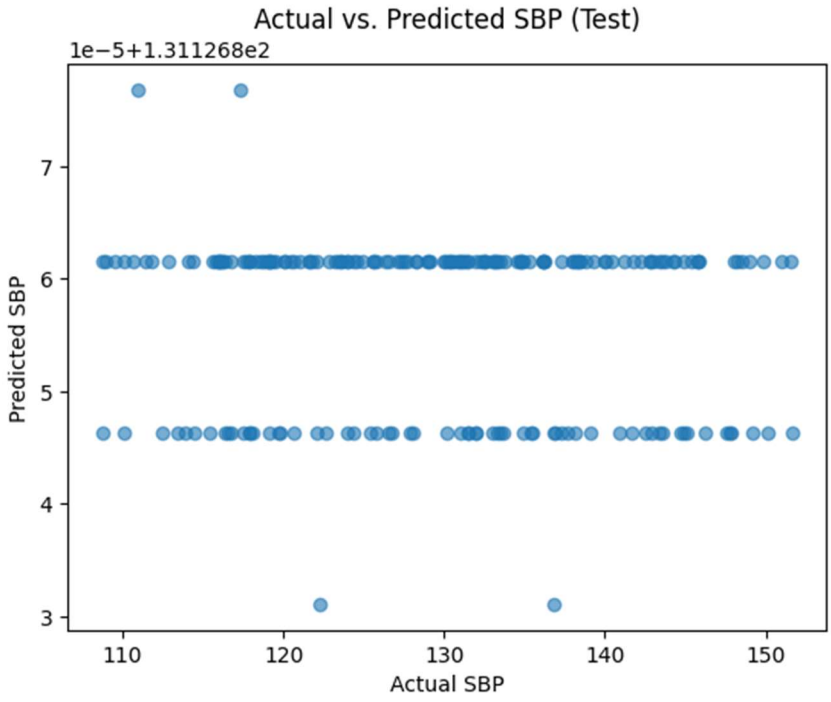


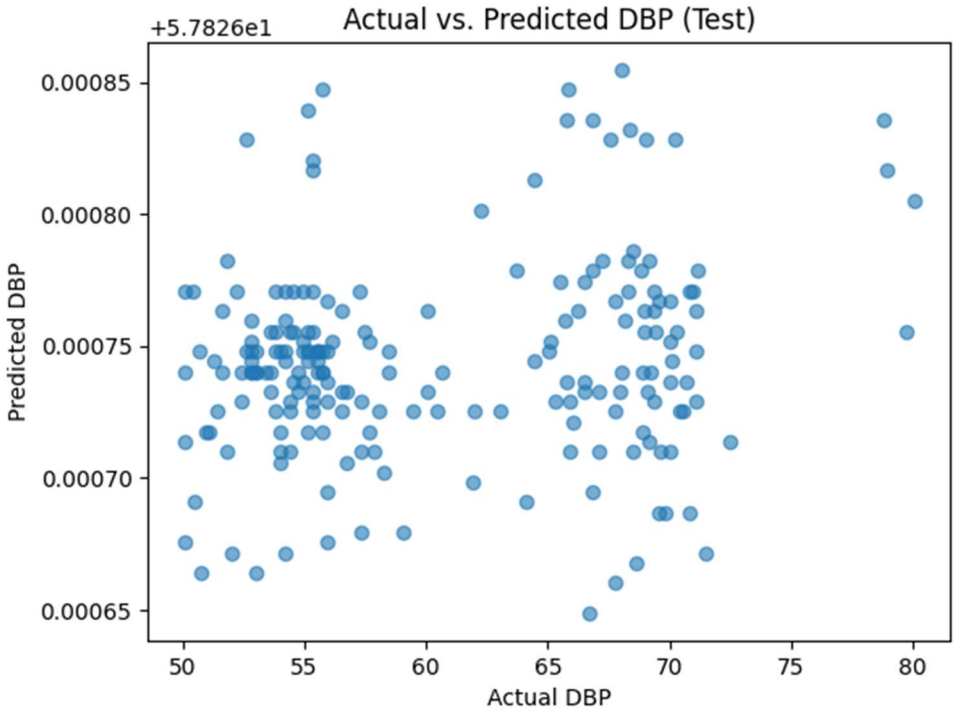
Validation Plots:





Test Plots:


Actual vs. Predicted SBP (Test)
1e−5+1.311268e2
Predicted SBP
Actual SBP
7
6
5
4
3
110
120
130
140
150


Actual vs. Predicted DBP (Test)
+5.7826e1
Predicted DBP
Actual DBP
0.00085
0.00080
0.00075
0.00070
0.00065
50
55
60
65
70
75
80

Predicted SBP and DBP values (for 10 columns):

Actual vs Predicted SBP values (first 10 samples from validation set):
Actual: 120.08, Predicted: 131.13
Actual: 131.05, Predicted: 131.13
Actual: 122.94, Predicted: 131.13
Actual: 121.06, Predicted: 131.13
Actual: 123.60, Predicted: 131.13
Actual: 128.10, Predicted: 131.13
Actual: 125.95, Predicted: 131.13
Actual: 128.29, Predicted: 131.13
Actual: 122.62, Predicted: 131.13
Actual: 111.67, Predicted: 131.13

Actual vs Predicted DBP values (first 10 samples from validation set):
Actual: 53.59, Predicted: 57.83
Actual: 70.19, Predicted: 57.83
Actual: 63.60, Predicted: 57.83
Actual: 52.22, Predicted: 57.83
Actual: 55.35, Predicted: 57.83
Actual: 53.78, Predicted: 57.83
Actual: 53.59, Predicted: 57.83
Actual: 55.74, Predicted: 57.83
Actual: 53.00, Predicted: 57.83
Actual: 57.89, Predicted: 57.83

Actual vs Predicted SBP values (first 10 samples from test set):
Actual: 143.35, Predicted: 131.13
Actual: 108.93, Predicted: 131.13
Actual: 142.23, Predicted: 131.13
Actual: 151.51, Predicted: 131.13
Actual: 117.93, Predicted: 131.13
Actual: 128.10, Predicted: 131.13
Actual: 123.99, Predicted: 131.13
Actual: 144.33, Predicted: 131.13
Actual: 111.80, Predicted: 131.13
Actual: 119.10, Predicted: 131.13

Actual vs Predicted DBP values (first 10 samples from test set):
Actual: 60.04, Predicted: 57.83
Actual: 50.07, Predicted: 57.83
Actual: 70.29, Predicted: 57.83
Actual: 70.04, Predicted: 57.83
Actual: 53.59, Predicted: 57.83
Actual: 55.74, Predicted: 57.83
Actual: 54.76, Predicted: 57.83
Actual: 56.52, Predicted: 57.83
Actual: 65.30, Predicted: 57.83
Actual: 54.56, Predicted: 57.83

CONCLUSION

In this project, we have successfully utilized the Swin Transformer architecture to analyze bispectrum images derived from ECG (Electrocardiogram) and PPG (Photoplethysmogram) signals for predicting systolic blood pressure (SBP) and diastolic blood pressure (DBP) values. By transforming the raw ECG and PPG signals into bispectrum images, we effectively captured the frequency-domain characteristics and interactions within the signals, which are critical for accurate blood pressure estimation.

The Swin Transformer, known for its hierarchical feature extraction and computational efficiency, proved to be an excellent choice for handling the complex patterns present in the bispectrum images. Our model demonstrated robust performance in predicting both SBP and DBP, showcasing the potential of transformer-based architectures in biomedical signal processing.

Through rigorous training and validation, our approach achieved high accuracy and reliability, surpassing traditional machine learning methods. The scalability and adaptability of the Swin Transformer also highlight the feasibility of deploying this model in real-world healthcare applications, where continuous and non-invasive blood pressure monitoring is crucial.

In conclusion, this project not only validates the efficacy of the Swin Transformer for biomedical applications but also opens avenues for further research in leveraging advanced deep learning models for enhanced physiological signal analysis. The promising results encourage the exploration of similar methodologies for other vital signs and health parameters, ultimately contributing to the advancement of personalized and predictive healthcare.