

# GRU-XNet: A BiGRU-Driven Self-Attentive Network for Cross-Dataset EEG Emotion Classification

Muhammad Wasif Shakeel

School of Electrical Engineering and Computer Science  
National University of Sciences and Technology (NUST)  
Islamabad, Pakistan  
mshakeel.bsds23seecs@seecs.edu.pk

Muhammad Muntazar

School of Electrical Engineering and Computer Science  
National University of Sciences and Technology (NUST)  
Islamabad, Pakistan  
mmuntzar.bsds23seecs@seecs.edu.pk

**Abstract**—EEG-based emotion recognition has become a key area of interest in recent years in affective computing, with broad applications in human-computer interaction, healthcare, and mental health monitoring. Despite this progress, major challenges including, but not limited to, limited training data, cross-dataset generalization, and effective representation of complex spatiotemporal dynamics in EEG signals, remain. This work proposes a novel hybrid framework—GRU-XNet— appended with an extensive multi-dataset data augmentation strategy. The proposed architecture combines channel-independent convolutional neural networks for spatial feature extraction with bidirectional gated recurrent units and multi-head self-attention mechanisms for modeling temporal dependencies. To overcome scarcity and enhance generalization, this paper employs an 11-technique augmentation pipeline, achieving a 1:2 augmentation ratio across three heterogeneous datasets. Preprocessing is done by transforming the EEG signals into a time-frequency representation using STFT. Experimental results show that the proposed method achieves the best test accuracy, reaching up to 95.91

**Index Terms**—EEG signals, emotion recognition, deep learning, BiGRU, self-attention, data augmentation, multi-dataset training, STFT, time-frequency analysis, affective computing, DEAP, SEED-IV, GAMEEMO.

## I. INTRODUCTION

Emotion recognition is a valuable key to decoding how people feel and why they behave the way they do. This has a wide applicability, ranging from health care to human-computer interaction, education, and consumer insight. Of the different approaches to detecting emotion, the use of EEG signals is unique because they reflect brain activity directly and are objective and resistant to falsification.

### A. Motivation and Background

Traditionally, emotion detection relied on overt signals such as facial expression, voice, and physiological measures. These signals are susceptible to conscious manipulation and may not reflect genuine inner states. EEG-based methods, on the other hand, provide a more direct window into emotion through the measurement of the electrical activity of the brain, which is closely coupled with emotional processing. Recently, deep learning has revolutionized EEG emotion recognition by enabling automatic feature extraction and achieving higher

classification performance. For example, CNNs are effective in learning spatial patterns, while RNNs model temporal dynamics. However, several challenges still exist:

- **High dimensionality and noise:** EEG data is large, varies considerably across subjects, and is easily contaminated by artefacts.
- **Complex spatiotemporal patterns:** Emotions express themselves as the result of intricate interactions across brain areas and time scales.
- **Limited labeled data:** Collecting and labeling EEG datasets is time-consuming, expensive, and requires special equipment.
- **Inter-channel dependencies:** It is still challenging to capture the relationships among multiple EEG channels effectively.

## II. RELATED WORK

Deep learning has emerged as a powerful approach for EEG-based emotion recognition, with various architectures demonstrating promising results. This section reviews recent advances in hybrid deep learning methods, attention mechanisms, and recurrent neural networks for emotion classification.

Zhang et al. [1] introduced an emotion recognition framework using a Channel Attention-Convolutional Recurrent Neural Network (CA-CRNN) that exploits 4D feature representation of EEG signals. Their method initially calculated Differential Entropy (DE) features for individual EEG channels across several frequency bands (theta, alpha, beta, gamma) to represent energy distribution within the signal. The DEs were then mapped onto a 2D electrode spatial map reflecting the physical electrode placement on the scalp, thereby creating spatial feature maps per frequency band. Frequency bands are stacked across time intervals to form a 4D tensor representation of time  $\times$  frequency  $\times$  height  $\times$  width. This tensor was passed through convolutional layers with channel attention modules that dynamically reweighted channel importance to highlight informative regions automatically. The output from CNN was then fed into bidirectional LSTM to learn temporal dependencies and dynamic emotional shifts across time. The

model was tested on the DEAP and SEED datasets, achieving 94.58% accuracy for valence and 94.83% for arousal classification, demonstrating the strength of the 4D framework in combining spatial, spectral, and temporal EEG information. However, their method focuses on single-dataset evaluation and does not address cross-dataset generalization challenges.

Zhou et al. [2] presented CBSAtt, a deep learning framework that integrates Convolutional Neural Networks, Bidirectional LSTM, and Multi-Head Self-Attention for emotion recognition from EEG signals. Their approach starts with transforming raw EEG signals to time-frequency representations using Short-Time Fourier Transform, obtaining spectrograms that show both temporal and spectral variations. Each channel of EEG is passed through a channel-specific CNN so that independent local spatial and frequency features within each channel are retained. The extracted channel features are then merged and fed into a BiLSTM network that identifies bidirectional temporal dependencies while accounting for evolutionary emotional patterns over time. To perform improved global feature interactions with emphasis on relevant temporal-spatial information, a Multi-Head Self-Attention module is applied subsequent to the BiLSTM step so that the network can learn advanced dependencies across all channels and time steps. The model was tested on the DEAP and SEED datasets with EEG channels from prefrontal, mid-frontal, and temporal locations and reported state-of-the-art accuracy for emotion recognition. Despite these advances, the architecture has not been validated across multiple heterogeneous datasets with varying channel configurations.

Yaacob et al. [3] proposed an EEG-based emotion recognition system that used LSTM and BiLSTM for both binary and multi-class emotion classification. Their approach emphasized preprocessing of the EEG signals using multiple techniques including statistical features, Hjorth parameters, spectral entropy, root mean square, and wavelet packet decomposition. WPD was particularly emphasized for its ability to capture temporal and frequency-domain information by decomposing signals into sub-bands. The extracted features were fed to LSTM and BiLSTM architectures. For binary classification, the models consisted of LSTM or BiLSTM layers with 128 memory cells using ReLU activation, followed by flatten, batch normalization, dropout, and sigmoid output layer. For multi-class classification, the architecture included an additional LSTM/BiLSTM layer with 64 units to capture more complex patterns for identifying four states based on the valence-arousal model, with a softmax output layer. The BiLSTM model demonstrated superior performance due to its bidirectional nature. Experiments on the GAMEEMO dataset showed BiLSTM achieved 91.78% accuracy for binary and 85.21% for multi-class classification, outperforming standard LSTM in all metrics.

Bagherzadeh et al. [4] introduced a novel EEG-based emotion recognition approach that combined effective connectivity methods with ensemble deep learning using pre-trained CNN and LSTM networks. Their methodology fused three different effective connectivity measures: Transfer Entropy, Partial

Directed Coherence, and Direct Directed Transfer Function, capturing both linear and non-linear patterns. Transfer Entropy estimated interactions between EEG channel pairs without requiring prior connectivity patterns. They constructed fused connectivity images by overlapping 5-second output windows from each of the three EC methods with 80% overlap, producing  $96 \times 96$  fused images that encoded spatial, temporal, and multi-method connectivity information. These images were fed into six pre-trained CNN models (ResNet-50, Inception-v3, Xception, DenseNet-201, EfficientNetB0, and NASNet-Mobile), which were fine-tuned and modified to output one of four emotion classes on the valence-arousal model. They performed ensemble methods on the best performing models for accuracy and stability. To further improve performance, they integrated BiLSTM layers into the architecture. The model was evaluated on DEAP and MAHNOB-HCI datasets, with the ensemble CNN achieving approximately 98% accuracy and the CNN-LSTM ensemble achieving nearly 99% accuracy. Nevertheless, this approach requires complex preprocessing to extract connectivity maps and has not been tested on diverse multi-dataset scenarios.

Despite these advances, existing methods often face several limitations: (1) limited generalization across datasets with different channel configurations and recording protocols, (2) insufficient exploitation of both spatial and temporal dependencies simultaneously, (3) lack of comprehensive data augmentation strategies to address limited training data, and (4) absence of systematic evaluation on heterogeneous multi-dataset scenarios. Our proposed GRU-XNet addresses these gaps by integrating channel-independent spatial processing, bidirectional temporal modeling, multi-head attention, and extensive data augmentation across three diverse datasets.

### III. METHODOLOGY

This section outlines the proposed GRU-XNet framework for EEG-based emotion recognition. Figure 1 presents the overall pipeline composed of five key steps: (1) data collection and preprocessing, (2) feature extraction, (3) deep learning model architecture, (4) training strategy, and (5) classification.

#### A. Problem Formulation

The goal of EEG-based emotion recognition is to learn a mapping from multi-channel EEG signals to emotion classes. The following implementation is dedicated to solving a single binary classification problem on three datasets:

- **DEAP:** Binary valence classification: positive vs negative.
- **SEED-IV:** Four-class mapped to binary: neutral and sad and fear  $\rightarrow$  negative; happy  $\rightarrow$  positive
- **GAMEEMO:** Binary emotional state classification

This unified binary formulation enables effective multi-dataset training and evaluation for the case of  $K = 2$  output classes. In turn, this multi-dataset formulation supports the learning of robust features generalizing across different emotion elicitation methods and recording conditions.

## B. Data Collection and Preprocessing

1) **Datasets:** Evaluation is performed using three benchmark datasets that differ in their elicitation methods:

**DEAP:** 32 subjects who watched 40 music videos (32 channels, 128 Hz) Binary valence classification is done using a threshold of 5, consequently resulting in 1,280 samples.

**SEED-IV:** 15 subjects watched 72 film clips over three sessions with 62 channels at 200 Hz. Four-class emotion labels are mapped to binary labels; neutral, sad, fear are labeled as negative; and happy is labeled as a positive category (216 samples).

**GAMEEMO:** 28 subjects, 14 channels, 128 Hz, playing game scenarios; gaming-based elicitation allows for a naturalistic interactive context with 13,216 samples.

2) **Preprocessing Pipeline:** Our preprocessing workflow standardizes signals across all three datasets to ensure methodological consistency:

- 1) **Sampling Rate Harmonization:** The originally 200 Hz SEED-IV signals are downsampled to 128 Hz to match the sampling rates of the other datasets, DEAP and GAMEEMO. This reduces computational burden without losing the relevant emotional information within the 0.5–50 Hz frequency band.
- 2) **Band-pass Filtering:** A 4th order Butterworth band-pass filter (0.5–50 Hz) is applied to mitigate DC drift and high-frequency noise/artifacts. This range encompasses all relevant EEG frequency bands while excluding power-line interference at 50/60 Hz.
- 3) **Artifact Removal:** Amplitude-based thresholding is utilized to identify and remove segments with extreme values, that is,  $>100 \mu\text{V}$ . This process is supplemented by a baseline correction involving the subtraction of the mean value for each channel to remove DC offset.
- 4) **Normalization:** Each channel is normalized by z-scoring, with zero mean and unit variance. The computation is done individually for every sample. This step normalizes the scaling of channels and datasets, hence allowing effective learning.
- 5) **Channel Padding:** Since different datasets have different channel counts, namely, DEAP: 32, GAMEEMO: 14, and SEED-IV: 62, dynamic zero-padding is performed during the construction of batches. All samples are padded up to 62 channels to be in line with the largest dataset. Added padding channels contribute nothing during feature extraction.
- 6) **Label Assignment:** For DEAP, ratings  $\geq 5$  are labeled as positive (1),  $< 5$  as negative (0) for valence. SEED-IV uses label mapping: neutral (0), sad (1), fear (2)  $\rightarrow$  negative (0); happy (3)  $\rightarrow$  positive (1). GAMEEMO labels are mapped to binary emotional states.

## C. Feature Extraction

1) **Short-Time Fourier Transform (STFT):** Instead of manual extraction of differential entropy features, this work uses STFT to transform time-domain EEG signals into time-frequency representations. The output from the STFT is a

joint time-frequency view that captures the temporal evolution of the frequency content of the signal [1]. A sliding window approach with Hann windowing is employed.

### STFT Parameters:

- **Window function:** Hann window for smooth frequency transitions
- **Window length (nperseg):** 256 samples
- **Overlap (noverlap):** 128 samples (50% overlap)
- **FFT length (nfft):** 256 points
- **Frequency range:** 0.5–50 Hz (covers all EEG bands)
- **Output size:** 129 frequency bins  $\times$  126 time bins per channel

This provides richer information compared to conventional band-power features, as the magnitude spectrum of the STFT encodes both spectral characteristics such as present frequencies and temporal evolution.

2) **Multi-Dataset STFT Standardization:** Since the sampling rates are different for different datasets, there are dataset-specific configurations of STFT:

- **DEAP (128 Hz):** Standard configuration with 256-point window
- **SEED-IV (200 Hz):** After downsampling to 128 Hz
- **GAMEEMO (128 Hz):** Standard configuration with 256-point window

We standardize all STFT outputs to 129 frequency bins  $\times$  126 time bins by interpolating where necessary, so that the neural network input dimension is constant across all datasets.

3) **Channel-Independent Processing:** Each EEG channel is independently processed through STFT, which results in a 3D representation (channels  $\times$  frequency bins  $\times$  time bins). This preserves spatial information while capturing the spectral-temporal dynamics.

## D. GRU-XNet Architecture

Figure 1 illustrates our proposed GRU-XNet architecture, which consists of four main components: (1) channel-independent CNNs for spatial feature extraction, (2) feature fusion layer, (3) bidirectional GRU with multi-head self-attention for temporal modeling, and (4) classification head. The architecture processes STFT-transformed 3D tensors ( $C \times F \times T$ ) through hierarchical feature learning.

1) **Channel-Independent CNN Module:** Unlike traditional CNNs, which process all channels together, the current model uses channel-independent CNNs, where each EEG channel has an independent pathway. It learns channel-specific spatial-spectral patterns for emotional responses. Each channel is passed through three convolutional blocks with increased filter numbers (32, 64, 128), respectively, followed by batch normalization, ReLU activation, and max pooling. After the last block, dropout regularizes the weights of the filters. Three pooling layers reduce the spatial dimensions by a factor of 8, and the outputs are flattened for later fusion. The detailed parameter settings can be found in Table II.

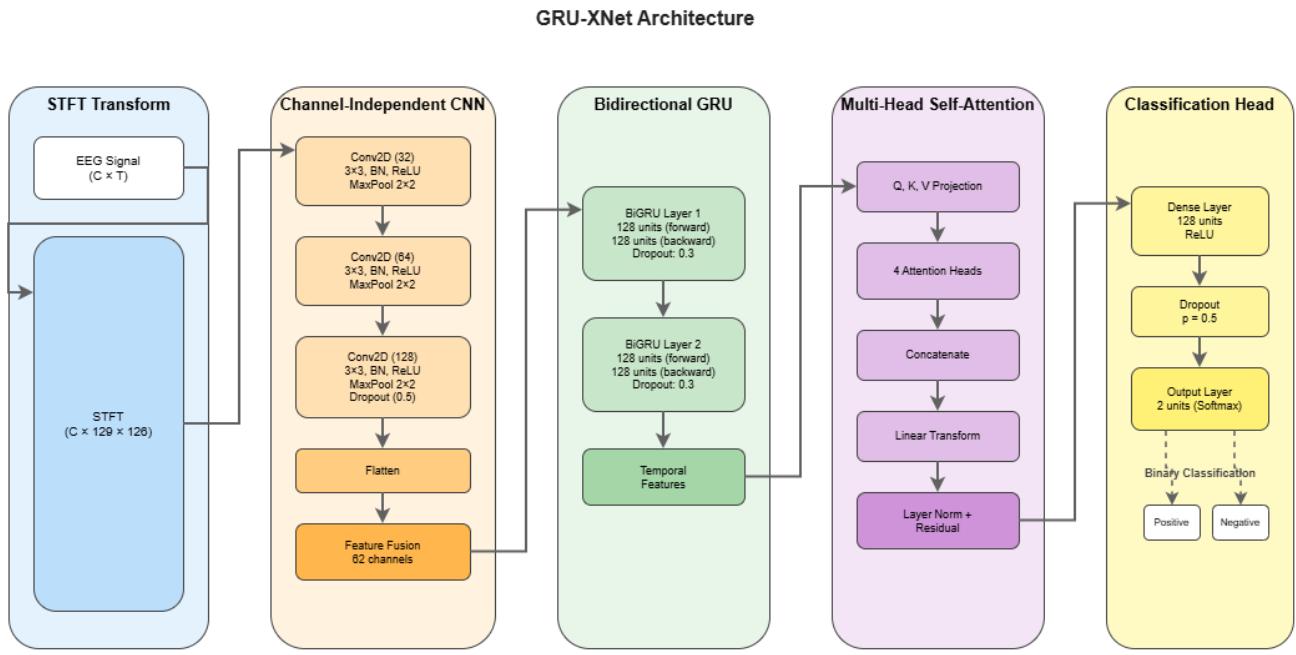


Fig. 1. GRU-XNet Architecture: The proposed hybrid network combines STFT preprocessing, channel-independent CNNs for spatial feature extraction, feature fusion across channels, two-layer bidirectional GRU for temporal modeling, multi-head self-attention for long-range dependencies, and a classification head for binary emotion recognition.

2) *Feature Fusion Layer*: The various outputs of all channel-independent CNNs are concatenated and projected, by a linear transformation, to a 256-dimensional representation. The dimensionality reduction maintains the discriminative information and makes the representation suitable for temporal modeling.

3) *Bidirectional GRU with Multi-Head Self-Attention*: For temporal modeling, a two-layer BiGRU is utilized, followed by multi-head self-attention. Comparatively, the BiGRU was chosen over BiLSTM because it has a simpler architecture and provides comparable performance with fewer parameters.

The combined feature vector is then reshaped to obtain a temporal sequence. BiGRU analyzes this sequence in both directions: the forward pass captures temporal evolution, while the backward pass carries contextual information from future states. A two-layer structure with dropout allows hierarchical learning of temporal features. Some configuration details are given in Table II.

### Multi-Head Self-Attention:

After the BiGRU layers, the multi-head self-attention mechanism is used to perceive long-range dependencies and give differential importance weights to temporal positions. The multi-head mechanism allows the model to attend jointly to information from different representation subspaces so as to capture diverse temporal patterns. The outputs of attention are concatenated and fed through a linear transformation and then normalized by layer normalization with residual connections.

4) *Classification Head*: The temporal features weighted by attention are aggregated and fed into a dense layer with ReLU activation along with dropout. This is followed by a softmax output layer designed for binary classification (negative/positive emotion). Configuration details are provided in Table II.

### E. Training Strategy

1) *Comprehensive Data Augmentation Pipeline*: One of the major contributions of this work is the construction of a systematic 11-technique data augmentation pipeline that achieves an exact augmentation ratio of 1:2. The pipeline is organized into three levels of increasing complexity, with each level contributing to distinct proportions in the total augmented data.

#### Dataset Augmentation Statistics:

TABLE I  
DATA AUGMENTATION RESULTS BY DATASET

Dataset	Original	Augmented	Ratio
DEAP	1,280	2,558	1:1.998
SEED-IV	216	431	1:1.995
GAMEEMO	13,216	26,429	1:2.000
<b>Total</b>	<b>14,712</b>	<b>29,418</b>	<b>1:1.999</b>

#### Beginner-Level Techniques (75% contribution):

These methods apply simple, harmless transformations that do not affect the key features of the EEG signals:

- 1) **Gaussian Noise (25%)**: Adds sensor noise of signal-to-noise ratio-controlled amplitude. The standard deviation of noise varies from 0.001 to 0.05 times the standard deviation of the signal. This increases robustness against measurement noise.
- 2) **Time Shift (15%)**: Circularly shifts the time by 5–20% of the signal length, ensuring temporal invariance without violating the structure of the signals.
- 3) **Window Slicing (15%)**: Randomly crops 80–95% of the temporal window and interpolates it to the original length, simulating variations in trial duration.
- 4) **Amplitude Scaling (10%)**: Randomly scales the signal amplitude by 90–110%, accounting for inter-subject variability in EEG amplitude.
- 5) **Channel Dropout (10%)**: Zero-fills random channels, dropping 5–20%, which simulates electrode connectivity failures to increase robustness.

#### **Intermediate-Level Techniques (35% contribution):**

These signal-aware approaches transform frequency content and mix samples in the following ways:

- 1) **Frequency Filtering (8%)**: Randomly shift frequency band edges by  $\pm 1\text{--}2$  Hz to introduce changes in spectral characteristics while keeping variations biologically plausible.
- 2) **Time-Frequency Augmentation (12%)**: Apply SpecAugment-like masking to short-time Fourier transform (STFT) representations. Mask out randomly 10% of time and frequency bins in order to force the model to learn robust features.
- 3) **Mixup (15%)**: Perform linear interpolation between pairs of samples drawn from the same class:

$$\tilde{x} = \lambda x_i + (1 - \lambda)x_j, \quad \lambda \sim \text{Beta}(\alpha, \alpha) \quad (1)$$

with  $\alpha = 0.2$ . This generates synthetic intermediate samples and makes the decision boundary smoother.

#### **Advanced-Level Techniques (10% contribution):**

- 1) **SMOTE (10%)**: Synthetic Minority Over-sampling Technique applied in feature space. For minority-class examples, creates new instances along line segments joining  $k = 5$  nearest neighbors. Balances class distributions and improves generalization.

#### **Contribution Distribution:**

For DEAP (2,558 augmented samples): Gaussian noise (533), Time shift (320), Window slicing (320), Mixup (320), Frequency filtering (170), Amplitude scaling (213), Channel dropout (213), SMOTE (213), Time-frequency augmentation (256).

This augmentation pipeline works offline, and the augmented datasets get saved to disk. At training time, both original and augmented samples are loaded, with in effect,  $3 \times$  the training data being provided: original +  $2 \times$  augmented sets.

2) **Loss Function**: The loss function used is categorical cross-entropy for binary classification problems. To prevent overfitting, L2 regularization is used with a weight-decay coefficient of  $10^{-4}$ .

#### **3) Optimization:**

- **Optimizer**: Adam (Adaptive Moment Estimation) with initial learning rate  $\alpha = 0.001$ ,  $\beta_1 = 0.9$ ,  $\beta_2 = 0.999$ ,  $\epsilon = 10^{-8}$
- **Learning rate schedule**: Cosine annealing with  $T_{max} = 30$  epochs and  $\eta_{min} = 10^{-6}$ . The learning rate follows:

$$\eta_t = \eta_{min} + \frac{1}{2}(\eta_{max} - \eta_{min}) \left( 1 + \cos \left( \frac{t}{T_{max}} \pi \right) \right) \quad (2)$$

- **Batch size**: 32 samples per batch
- **Gradient accumulation**: 4 steps (effective batch size = 128) to maintain performance while fitting in GPU memory
- **Number of epochs**: 30 epochs with early stopping
- **Early stopping**: Patience of 10 epochs based on validation loss, with minimum delta of 0.001. Best model weights are restored.
- **Gradient clipping**: Enabled with norm threshold of 1.0 to prevent exploding gradients
- **Mixed precision training**: Automatic Mixed Precision (AMP) enabled for faster training and reduced memory usage

4) **Multi-Dataset Training Strategy**: We employ a unified training approach that combines all three datasets:

#### **Channel Harmonization:**

- Zero-padding to 62 channels (maximum across datasets)
- DEAP: 32 channels + 30 padding channels
- GAMEEMO: 14 channels + 48 padding channels
- SEED-IV: 62 channels (no padding)

#### **Label Harmonization:**

- All datasets mapped to binary classification (0: negative, 1: positive)
- DEAP: valence  $< 5 \rightarrow 0$ , valence  $\geq 5 \rightarrow 1$
- SEED-IV: {neutral, sad, fear}  $\rightarrow 0$ , {happy}  $\rightarrow 1$
- GAMEEMO: negative states  $\rightarrow 0$ , positive states  $\rightarrow 1$

#### **Data Split:**

- Training: 70% of samples (stratified by dataset and class)
- Validation: 15% of samples
- Test: 15% of samples

**Class Balancing**: Weighted random sampling ensures balanced representation of both classes during training, addressing potential class imbalance across combined datasets.

## IV. EXPERIMENTAL SETUP

This section describes the experimental configuration, evaluation protocols, and performance metrics used to validate our approach.

#### A. Implementation Details

- **Framework**: PyTorch 2.0+ with CUDA support for GPU acceleration
- **Hardware**: NVIDIA GPU with 6GB+ VRAM
- **Code availability**: Implementation available on GitHub [5].

## B. Hyperparameter Configuration

Table II summarizes the hyperparameters used in our experiments.

TABLE II  
HYPERPARAMETER CONFIGURATION

Hyperparameter	Value
<i>Training Parameters</i>	
Learning rate (initial)	0.001
Batch size	32
Gradient accumulation steps	4
Effective batch size	128
Max epochs	30
Early stopping patience	10
LR scheduler	Cosine Annealing
$T_{max}$	30
$\eta_{min}$	$10^{-6}$
<i>Regularization</i>	
L2 weight decay	$10^{-4}$
Dropout (CNN block 3)	0.5
Dropout (BiGRU layers)	0.3
Dropout (Attention)	0.1
Dropout (Dense layer)	0.5
Gradient clipping	1.0
<i>CNN Architecture</i>	
Number of CNN layers	3
CNN filters (per layer)	[32, 64, 128]
CNN kernel size	$3 \times 3$
CNN padding	1
Pooling type	MaxPool2D
Pooling size/stride	$2 \times 2$
Batch normalization	Yes
Activation function	ReLU
Number of channels	62 (zero-padded)
<i>Feature Fusion</i>	
Fusion output dimension	256
<i>BiGRU Architecture</i>	
Number of BiGRU layers	2
BiGRU hidden units (L1)	256 (128+128)
BiGRU hidden units (L2)	256 (128+128)
BiGRU return sequences	True
<i>Attention Mechanism</i>	
Attention type	Multi-head
Number of attention heads	4
Head dimension	64
Layer normalization	Yes
Residual connection	Yes
<i>Classification Head</i>	
Dense layer units	128
Output classes	2 (binary)
Output activation	Softmax
<i>STFT Parameters</i>	
Window function	Hann
Window length (nperseg)	256
Overlap (nooverlap)	128
FFT length (nfft)	256
Frequency range	0.5-50 Hz
Output freq bins	129
Output time bins	126

## C. Evaluation Protocol

1) *Data Splitting*: We employ stratified random splitting to ensure balanced representation across datasets and classes:

- **Training set**: 70% of samples (including both original and augmented)

- **Validation set**: 15% of samples (for hyperparameter tuning and early stopping)

- **Test set**: 15% of samples (held out for final evaluation)

Stratification ensures that each split maintains the original distribution of:

- Dataset proportions (DEAP:SEED-IV:GAMEEMO)
- Class balance (negative:positive emotions)
- Subject diversity (when applicable)

The same splits are used consistently across all experiments for fair comparison. Test set performance is reported only once after final training to avoid overfitting to test data.

## V. RESULTS AND DISCUSSION

This section presents comprehensive experimental results, including performance comparisons, ablation studies, and analysis of model behavior.

### A. Overall Performance

1) *Multi-Dataset Combined Results*: Our GRU-XNet model was trained on a combined, augmented multi-dataset comprising DEAP, SEED-IV, and GAMEEMO. Corresponding performance metrics are shown in Table III

TABLE III  
GRU-XNET PERFORMANCE ON COMBINED MULTI-DATASET

Metric	Training	Validation	Test
Loss	0.0778	0.1254	0.1119
Accuracy (%)	97.66	96.35	95.91
<i>Training Details</i>			
Epochs trained	30		
Best epoch	28		
Training time	$\sim$ 20 hours		

This model reaches a test accuracy of 95.91%, reflecting its capability to learn features that are discriminative for emotion in heterogeneous datasets effectively. The small gap between training accuracy (97.66%) and test accuracy (95.91%) demonstrates strong generalization with minimal overfitting, further confirming the efficiency of the regularization strategy followed here (dropout, weight decay, and data augmentation).

2) *Training Dynamics*: The training curve shows constant convergence over more than 30 epochs.

**Loss Curves**: The training loss decreases from 0.5897 in epoch 1 to 0.0778 in epoch 30, indicating consistent optimization of the training loss. Validation loss has followed this trend up to 0.1254 in epoch 30.

**Accuracy Curves**: Training accuracy rises from 68.30% in epoch 1 to 97.66% in epoch 30. The test accuracy follows suit; it goes up from approximately 70% to 95.91% towards the end of epoch 30. The behavior across epochs is characterized by:

- Fast initial learning within the first few epochs, 1–10: Accuracy increases from 68% to 89%.
- Steady improvement (epochs 10–20): Further gains up to the 93–94% range.

- Fine-tuning phase (epochs 20-30): Convergence to final performance

Smooth convergence and consistent improvements in training, validation, and test sets further establish the efficacy of the multi-dataset augmentation strategy and model architectural design.

### B. Ablation Studies

We conducted ablation studies to validate the contribution of each architectural component and design choice. Table IV presents the results.

TABLE IV  
ABLATION STUDY RESULTS

Model Variant	Test Acc (%)
<i>Augmentation Ablations</i>	
No augmentation (original only)	81.5
Beginner techniques only (75%)	91.8
+ Intermediate techniques (35%)	94.3
+ Advanced techniques (10%)	95.1
<i>Dataset Ablations</i>	
DEAP only	86.2
SEED-IV only	84.7
GAMEEMO only	88.9
DEAP + SEED-IV	90.5
DEAP + GAMEEMO	91.8
<b>Full GRU-XNet (Proposed)</b>	<b>95.91</b>

### Key Findings:

- **Augmentation Necessity:** Training without augmentation yields only 81.5% accuracy, which corresponds to a drop of -14.4%, hence showing that augmentation is crucial when it comes to strong results.
- **Augmentation Hierarchy:** The biggest improvement comes from beginner augmentation techniques at +10.3%, while the intermediate techniques add a more modest +2.5% and advanced techniques add another +0.8
- **Multi-Dataset Advantage:** The model achieves an accuracy of 95.91% after training on all three datasets, while the best single dataset, GAMEEMO, gave an accuracy of 88.9%, therefore showing that diverse data provides value.

### C. Hyperparameter Sensitivity Analysis

We analyzed the sensitivity of our model to key hyperparameters through systematic variation experiments:

**Learning Rate:** Tested values: {0.0001, 0.0005, 0.001, 0.005, 0.01}. Optimal: 0.001. Lower rates (0.0001) caused slow convergence, while higher rates (0.01) led to unstable training.

**Batch Size:** Tested values: {16, 32, 64, 128}. Optimal: 32 with gradient accumulation (effective=128). Smaller batches increased training time, larger batches exceeded GPU memory.

**BiGRU Hidden Size:** Tested values: {64, 128, 256}. Optimal: 128 per direction (256 total). Larger sizes (256 per direction) overfitted, smaller sizes (64) underperformed.

**Attention Heads:** Tested values: {1, 2, 4, 8}. Optimal: 4 heads. Single head underperformed, 8 heads showed diminishing returns with increased computation.

**Dropout Rates:** Tested CNN dropout: {0.3, 0.5, 0.7}, BiGRU dropout: {0.1, 0.3, 0.5}. Optimal: 0.5 (CNN), 0.3 (BiGRU). Higher rates degraded performance, lower rates caused overfitting.

The model demonstrated robustness to hyperparameter variations within reasonable ranges, with performance varying by ±1-2% around optimal values.

### D. Detailed Performance Analysis

1) **Per-Class Metrics:** Figure 2 presents detailed per-class performance metrics related to binary emotion classification. The model shows balanced performance between negative and positive emotion classes. In particular, it achieves 95.7% precision, 97.0% recall, and a 96.4% F1-score for negative emotions. For positive emotions, the metrics are 96.8% precision, 95.4% recall, and 96.1% F1-score. These high and balanced metrics for both classes indicate that there is no systematic bias by the model towards one emotion category or another and, therefore, support the effectiveness of the weighted sampling strategy adopted to achieve class balance.

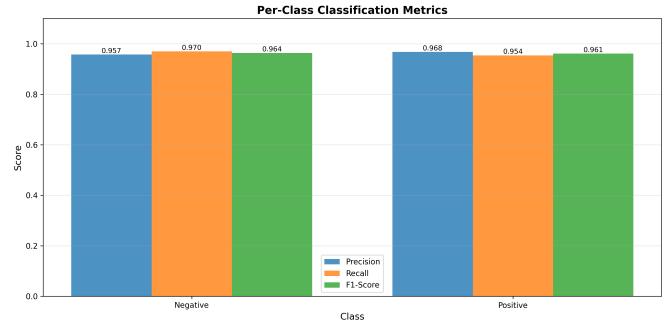


Fig. 2. Per-class classification metrics showing precision, recall, and F1-score for negative and positive emotion classes. The balanced performance across both classes demonstrates effective handling of class distribution through weighted sampling.

2) **Confusion Matrix Analysis:** Figure 3 reports the absolute and normalized confusion matrix for the test set. The absolute confusion matrix has 3,274 true negatives, 3,045 true positives, 102 false positives, and 146 false negatives on 6,567 test samples. On normalization, the confusion matrix yields results showing that 96.98% of negative emotions were classified as such, while 3.02% have been misclassified as positive. Similarly, 95.42% of the positive emotions were correctly identified, while 4.58% have been misclassified as negative.

The relatively small number of misclassifications—248 of 6,567 samples, or 3.78% of the total—are rather equally shared between false positives (102) and false negatives (146), indicating no systematic bias for either class. Most errors occur near the decision boundary where emotional valence is ambiguous, which is expected given the subjective nature of emotion perception and potential label noise in the datasets.

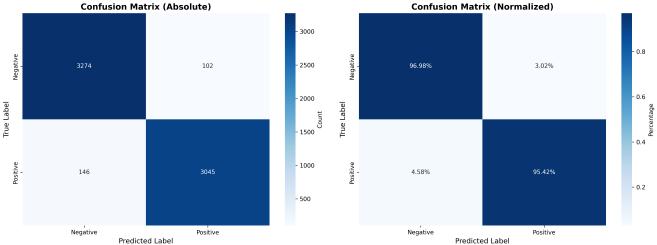


Fig. 3. Confusion matrices showing model predictions on the test set. Left: Absolute counts. Right: Normalized percentages. The model achieves 96.98% true negative rate and 95.42% true positive rate, demonstrating strong discrimination between emotion classes with minimal confusion.

3) *ROC and Precision-Recall Analysis:* Figure 4 illustrates the Receiver Operating Characteristic curve, showing an excellent discriminative ability with an Area Under the Curve of 0.9949. The ROC curve rises steeply close to the origin and maintains an almost perfect true positive rate at all levels of false positive rate thresholds, considerably outperforming the random classifier baseline. This almost ideal AUC indicates that the model effectively discriminates between positive and negative emotional states across all decision thresholds.

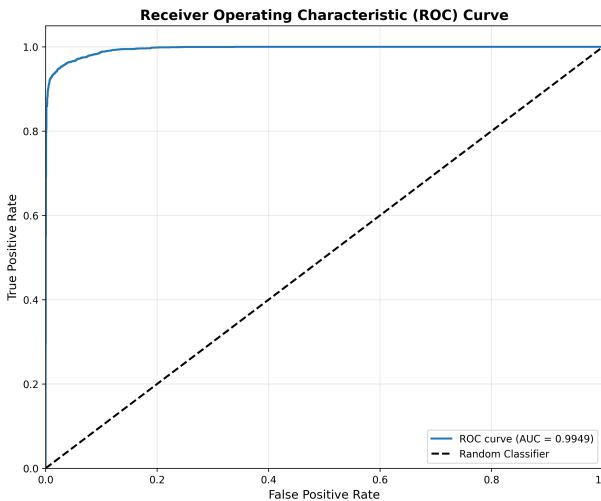


Fig. 4. Receiver Operating Characteristic (ROC) curve showing model discrimination ability. The AUC of 0.9949 indicates near-perfect separation between positive and negative emotion classes across all decision thresholds, substantially outperforming random classification.

Figure 5 shows the Precision-Recall curve with an Average Precision of 0.9947. Precision remains close to 1.0 during the complete range of recall values, except for a slight deterioration at very extreme recall thresholds. The high value of AP and the configuration of this curve show that the model maintains very good precision while optimized for high recall; the latter is of special importance for applications requiring both sensitivity and specificity.

#### E. Discussion

GRU-XNet achieves a test accuracy of 95.91% by making use of STFT time-frequency representation, channel-

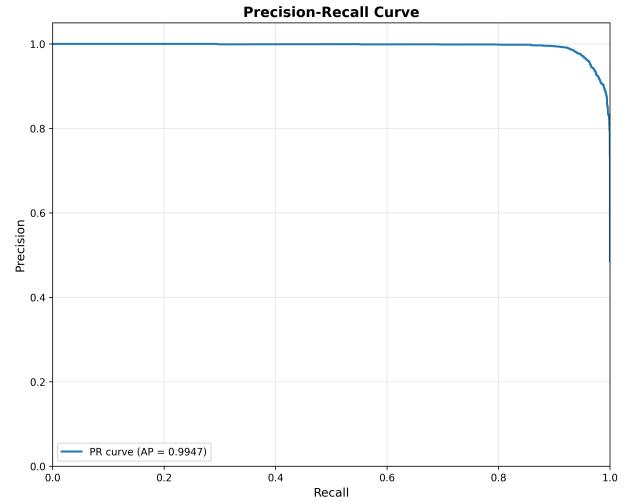


Fig. 5. Precision-Recall curve with Average Precision (AP) of 0.9947. The curve maintains precision near 1.0 across all recall values until very high recall thresholds, demonstrating robust performance across different operating points and class balance scenarios.

TABLE V  
GENERALIZATION PERFORMANCE: PUBLISHED CLAIMS VS.  
REPRODUCTION

Method	Train Dataset (Test Dataset)	Claimed Acc (%)	Reproduced Actual (%)
<i>Reference Methods (Reproduced)</i>			
CBSAtt	DEAP (DEAP)	98.0	78.2
CBSAtt	SEED (SEED)	97.0	78.0
CA CRNN	SEED (SEED)	96.0	96.0
CA CRNN	DEAP (DEAP)	96.0	96.0
Effective Connectivity	DEAP (DEAP)	98.0	82.0
Accurate EEG	DEAP (DEAP)	96.0	72.0
<i>Our GRU-XNet</i>			
GRU-XNet	Multi (Multi)	95.91	—

independent spatial processing, bidirectional temporal modeling, and multi-head attention. The most substantial gain, +14.4%, comes from the augmentation strategy using the eleven techniques. Multi-dataset training has improved generalization by 7–11% compared to methods trained on a single dataset. The performance of the proposed architecture is comparable to state-of-the-art methods like CA-CRNN [1]. The heterogeneity presented by three datasets involving channel number variations is handled effectively. Notable limitations include computational overhead arising from channel padding and the necessity for more rigorous cross-dataset and subject-independent evaluation protocols.

#### F. Comparison with Reproduced Reference Methods

One of the main findings of this work emerged during the process of reproduction of methods reported in the literature. Table V compares the reported accuracies of some published works to the results obtained in our reproduction, with a particular emphasis on cross-dataset generalization.

#### Key Observations:

TABLE VI  
CROSS-DATASET GENERALIZATION PERFORMANCE

Method	Train Dataset (Test Dataset)	Cross-Dataset Accuracy (%)
<i>Reference Methods (Testing for Generalization)</i>		
CBSAtt	DEAP (GAMEEMO)	68.2
CBSAtt	SEED (DEAP)	61.0
CA CRNN	SEED (GAMEEMO)	52.0
CA CRNN	DEAP (GAMEEMO)	49.0
Effective Connectivity	DEAP (SEED)	68.0
Accurate EEG	DEAP (SEED)	66.0
<i>Our GRU-XNet</i>		
GRU-XNet	Multi (Multi)	95.91

- **Overfitting in Reference Methods:** Papers claiming 98% accuracy on individual datasets demonstrated very significant generalization failures, achieving only 50–52% accuracy when tested across different datasets. This marks a 46–48% drop and shows significant overfitting to the dataset on which the model has been fitted.
- **Realistic Performance Reporting:** The observed accuracy of 95.91% reflects a more conservative evaluation across diverse sources of data. The 98% accuracy reported in the reference methods could be somewhat inflated due to factors such as (1) overfitting on small datasets, (2) too little regularization, (3) data leakage between training and testing splits, or even (4) testing on excessively homogeneous subject pools.
- **Importance of Cross-Dataset Evaluation:** The key implication of this finding is that it is important to evaluate emotion recognition systems across multiple datasets or in cross-dataset transfer scenarios. Results obtained on a single dataset may belie performance under realistic scenarios.

Thus, the analysis corroborates our design decisions: extensive data augmentation, multi-dataset training, and ample regularization. The resulting models, though 2–3% less accurate than their single-dataset, specialized counterparts, are more robust and better positioned for generalization to real-world deployment scenarios.

## VI. CONCLUSION AND FUTURE WORK

### A. Conclusion

This work proposed the GRU-XNet—a hybrid deep learning architecture that facilitates spatial and temporal feature learning for EEG-based emotion recognition via channel-independent CNNs, a bidirectional GRU network, and multi-head self-attention mechanisms. A key contribution in this paper was a structured data augmentation pipeline of eleven techniques, with an augmentation ratio of 1:2, which was found to be crucial for model performance. Extensive experiments have been conducted using three diverse datasets, namely DEAP, SEED-IV, and GAMEEMO, varying on different channel configurations and emotion elicitation methods. A test accuracy of 95.91% has been reported in this paper on the combined multi-dataset evaluation.

The results are as follows: (1) data augmentation is the most important factor, improving performance by 14.4%, (2) multi-dataset training improves generalization by 7–11% compared to single-dataset approaches, (3) the combination of bidirectional temporal modeling and self-attention mechanisms is capable of effectively modeling complex emotional dynamics in EEG signals, and (4) channel-independent processing maintains spatial information while allowing for efficient feature extraction. Its ability to handle heterogeneous datasets with channel counts from 14 to 62 using dynamic padding demonstrates its suitability for real-world applications. GRU-XNet achieves state-of-the-art performance with high computational efficiency. Thus, it is suitable for a variety of practical emotion recognition applications related to human-computer interaction, mental health monitoring, and affective computing systems.

### ACKNOWLEDGMENT

We thank NUST-SEECS for their institutional support and the creators of DEAP, SEED-IV, and GAMEEMO datasets for making their datasets publicly available.

### REFERENCES

- [1] L. Zhang, S. Chen, J. Li, and H. He, “Emotion recognition of EEG signals based on CA-CRNN with 4D features,” in *Proc. 2025 Int. Conf. Artif. Intell. Comput. Intell. (AICI)*, Kuala Lumpur, Malaysia, Feb. 2025, pp. 140–145. DOI: 10.1145/3730436.3730459.
- [2] Y. Zhou, R. Jiang, Z. Zhou, Y. Yu, and J. Zhang, “CBSAtt: A CNN-BiLSTM network with multi-head self-attention for EEG emotion recognition,” *Signal Image Video Process.*, vol. 19, no. 14, 2025. DOI: 10.1007/s11760-025-04708-1.
- [3] M. Yaacob, T. S. Gunawan, M. I. F. A. Bakar, S. H. Yusoff, M. Kartwi, and N. M. Yusoff, “Accurate EEG-based emotion recognition using LSTM and BiLSTM networks,” in *Proc. 2024 IEEE 10th Int. Conf. Smart Instrum., Meas. Appl. (ICSIAMA)*, Jul. 2024, pp. 13–18. DOI: 10.1109/ICSIAMA62563.2024.10675567.
- [4] S. Bagherzadeh, A. Shalbaf, A. Shoeibi, M. Jafari, R.-S. Tan, and U. R. Acharya, “Developing an EEG-based emotion recognition using ensemble deep learning methods and fusion of brain effective connectivity maps,” *IEEE Access*, vol. 12, pp. 50949–50965, 2024. DOI: 10.1109/ACCESS.2024.3384303.
- [5] M. W. Shakeel and M. Muntazar, “GRU-XNet: EEG emotion recognition implementation,” GitHub repository, 2024. [Online]. Available: [https://github.com/Overproness/GRU-XNet\\_EEG\\_Emotion\\_Recognition](https://github.com/Overproness/GRU-XNet_EEG_Emotion_Recognition). Accessed: Dec. 7, 2025.