Improving EmoDB Speech Emotion Recognition with Custom CNNs

Lai Le Dinh Duc SE192137 | Chau Le Quan SE192913 | Nguyen Minh Duc SE192059 | Nguyen Trung Kien SE192209 | Le Minh Khoi SE192396

***Index Terms*—SpeechRecognition, DeepLearning, AudioProcessing**

I. Introduction

This project aims to develop an end-to-end machine learning pipeline to achieve optimal performance on the EmoDB dataset. The proposed pipeline will encompass data preprocessing, feature extraction, model building and training, as well as model evaluation. In the introduction section, our objective is to provide a comprehensive overview of the speech emotion recognition (SER) task, along with relevant aspects of audio data — the primary data type with which this project will engage.

A. Speech Emotion Recognition

Speech Emotion Recognition (SER) is the process of analyzing, identifying, and classifying human emotions through voice signals using audio signal processing and artificial intelligence. It plays an essential role in various applications such as psychology, mental health, and automated customer service systems.

B. Audio Data

In the natural environment, audio exists as sound waves (sine-cosine waveforms). These are vibrations transmitted through mediums such as air, liquids, and solids. When audio is recorded, it is stored as a digital or electromagnetic representation of sound.

When stored digitally, sound is converted into audio through sampling and quantization. Key parameters include:

- Sampling Rate: The number of samples taken per second, commonly 16kHz, 44.1kHz, or 48kHz.
- Bit Depth: Determines how many bits are used to represent each audio sample, typically 16-bit or 24-bit.
- Channels: Mono (1 channel) or Stereo (2 channels: left and right).

C. Audio Features

To recognize emotions, specific audio features must be extracted. These features can be categorized as follows:

1) Time-Domain Features:

- RMS (Root Mean Square): Represents the average energy level of the signal, useful for identifying the loudness of the audio.
- Zero-Crossing Rate (ZCR): The rate at which the signal changes sign, helpful in identifying frequency-related characteristics, speech speed, and whether the emotion is tense or relaxed.

2) Cepstral-Domain Features:

- MFCC (Mel-Frequency Cepstral Coefficients): These coefficients map the short-term power spectrum of a signal onto the Mel scale to approximate the way the human ear perceives sound. They compactly represent timbral and phonetic content.

3) Time-Frequency Features:

- Spectrogram: Represents how the signal's spectral density evolves over time, showing the distribution of energy across frequencies at each moment.
- Mel Spectrogram: A variation of the standard spectrogram where the frequency axis is scaled according to the Mel scale, aligning more closely with human auditory perception.
- Chroma Features: Chroma or pitch class profiles capture the distribution of spectral energy across the 12 pitch classes of the musical octave. They are well-suited for extracting harmonic and tonal information, which is valuable in music analysis and key detection.

II. Approaches

In the Approaches section, we aim to present an overview of the existing methods for speech emotion recognition and to clarify the rationale behind our chosen approach.

A. Data Processing Approaches

1) Raw Audio Processing: This approach processes the raw audio waveform directly using deep learning models such as CNNs, RNNs, or attention-based networks. It allows the model to automatically learn representations without manual feature extraction but typically requires large datasets and high computational resources.

2) Feature-Based Processing: In this approach, acoustic features like MFCCs, chroma, pitch, or prosodic cues are first extracted, then fed into machine learning models such as SVMs, Random Forests, or CNNs. This is a well-established method, often combined with normalization and feature selection for better performance.

3) Hybrid Processing: Modern systems often combine both approaches by leveraging pre-trained models (e.g., wav2vec 2.0, HuBERT) to learn robust features directly from raw audio, then fine-tune them for emotion recognition. This improves performance, especially when labeled data is limited.

B. Machine Learning Approaches

1) Traditional Machine Learning: Algorithms like Support Vector Machines (SVM), Random Forest, and K-Nearest Neighbors (KNN) use pre-extracted features and are suitable for small to medium-sized datasets and require careful feature engineering and selection.

2) Deep Learning:

- CNN (Convolutional Neural Networks): Commonly trained on spectrograms or Mel-spectrograms to learn local time-frequency patterns.
- RNN (Recurrent Neural Networks), LSTM (Long Short-Term Memory), and Transformer: Effectively leverage the temporal structure of audio signals, especially useful for modeling continuous and complex emotional changes.
- Hybrid Models: Combining CNNs with RNNs or Transformers leverages the strengths of both local feature extraction and temporal modeling.

Regarding our approach, our work initially focused on extracting features in a two-dimensional representation, which were then used to train several simple CNN-based model variants. Subsequently, we extended our approach to extract and organize features in a three-dimensional form, enabling us to build and experiment with multiple custom 3D Convolutional Neural Network (3D CNN) architectures.

III. Method

Following a series of experiments with different feature extraction strategies and model architectures, we present here the approach that has achieved the best performance among those we have explored so far on the Emo-DB dataset.

A. Dataset

The dataset used in this project is the Berlin Database of Emotional Speech (EmoDB), which was accessed from Kaggle. It contains German-language audio recordings of 10 professional actors (5 male and 5 female) expressing 7 different emotions: anger, boredom, disgust, anxiety/fear, happiness, sadness, and neutrality.

The EmoDB dataset contains 535 audio files and each audio file is labeled with a corresponding emotion class. The class distribution is illustrated in the figure below. As shown in the bar and pie chart Figure 1:

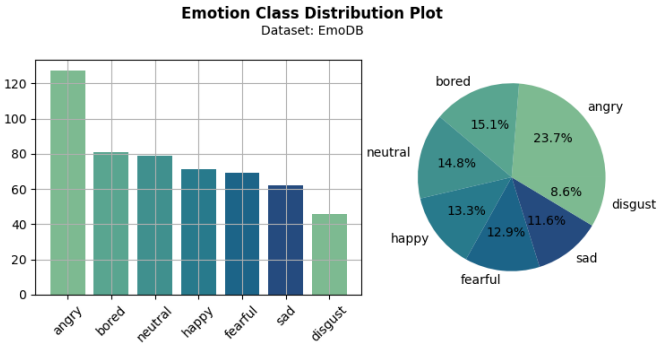

Figure 1. Emotion Class Distribution Plot

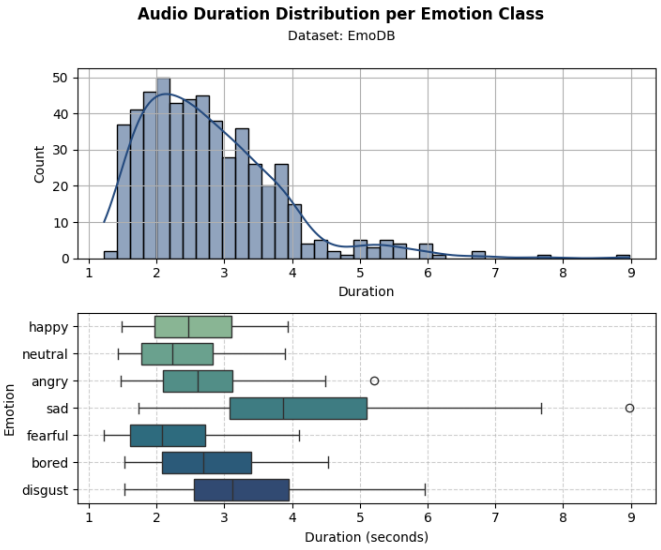

Figure 2. Audio Duration Distribution per Emotion Class

- Anger is the most represented emotion, comprising approximately 23.7% of the dataset (127 samples).
- Followed by boredom (15.1%), neutral (14.8%), happy (13.3%), fearful (12.9%), and sad (11.6%).
- Disgust is the least frequent, with only 8.6% of the total samples (46 samples).

In addition to class labels, the duration of each audio sample was analyzed in Figure 2 to understand variability across different emotional expressions.

The top histogram in the figure shows the overall distribution of audio durations. Most recordings range between 1.5 and 4 seconds, with a peak around 2 to 2.5 seconds, and a right-skewed tail extending up to nearly 9 seconds.

The bottom box plot breaks down audio durations by emotion class. Key observations include:

- Sad utterances tend to be longer on average, with several outliers extending beyond 6 seconds.
- Neutral, happy, and angry recordings have shorter and more consistent durations, with most values tightly dis-

tributed between 1.5 and 3.5 seconds.
- Disgust has the widest variability among the shorter-duration classes, with a few longer samples.

Each audio file is stored in .wav format, varying in length and down-sampled from 48-kHz to 16-kHz.

B. Feature Extraction

To transform raw audio data into a format suitable for deep learning, we applied a multi-step feature extraction process that captured both spectral and temporal information:

1) Resampling & Mono Conversion: All .wav audio files were loaded and resampled to a 44.1 kHz sampling rate. Stereo files were converted to mono by averaging across channels, ensuring consistent input dimensions.

2) Segmentation with Overlapping Windows: Each audio clip was divided into smaller overlapping segments using a sliding window approach. The window size was determined by the desired number of frames and hop length:

$$window\_size = hop\_length(frames - 1), \qquad (1)$$

and each window overlapped 50% with the previous one. This approach increased data quantity and regularized input dimensions.

3) MFCC Feature Extraction: For each segment, we extracted a 128-dimensional Mel-Frequency Cepstral Coefficient (MFCC) representation over 128 time frames, resulting in a (128, 128) matrix per window. This captured short-term frequency content of speech signals.

4) Delta and Delta-Delta Features: To capture temporal dynamics, we computed:

- First-order deltas (velocity) of MFCCs as the second channel.
- Second-order deltas (acceleration) as the third channel.

These were stacked to form a 3-channel spectrogram-like input of shape (128, 128, 3) for each segment.

5) Final Dataset Dimensions: The final feature tensor had the shape (1234, 128, 128, 3), representing 1234 segments extracted from the original 535 audio clips. Corresponding labels were stored as integer-encoded class indices (shape = (1234,)), based on a predefined mapping of emotion names to class indices.

C. Model Architecture

A custom Convolutional Neural Network was implemented using PyTorch to classify emotional speech based on MFCCbased feature maps. The model was designed to process 3-channel spectrogramlike inputs of shape (128, 128, 3) for each windowed segment.

The model consists of four convolutional blocks followed by a dense classifier as shown in Table I. This architecture enables the model to extract hierarchical spatial features from the input spectrograms and map them to the 7 emotion classes via a final softmax layer.

Table I
CUSTOM CNN ARCHITECTURE

Block	Layer Description	Output Shape
Input	3-channel tensor: - Channel 1: MFCC - Channel 2: First-order Delta - Channel 3: Second-order Delta	$3 \times 128 \times 128$
Conv 1	Conv2D ($3 \rightarrow 64$, kernel=3) Batch Normalization (64) ReLU activation MaxPooling2D (kernel=2)	$64 \times 64 \times 64$
Conv 2	Conv2D ($64 \rightarrow 128$, kernel=3) Batch Normalization (128) ReLU activation MaxPooling2D (kernel=2)	$128 \times 32 \times 32$
Conv 3	Conv2D ($128 \rightarrow 256$, kernel=3) Batch Normalization (256) ReLU activation MaxPooling2D (kernel=2) Dropout (p=0.3)	$256 \times 16 \times 16$
Conv 4	Conv2D ($256 \rightarrow 512$, kernel=3) Batch Normalization (256) ReLU activation Adaptive Average Pooling (output=1×1) Dropout (p=0.4)	$512 \times 1 \times 1$
Classifier	Flatten Linear ($512 \rightarrow 64$) ReLU activation Dropout (p=0.5) Linear ($64 \rightarrow 7$) LogSoftmax	7-class logits

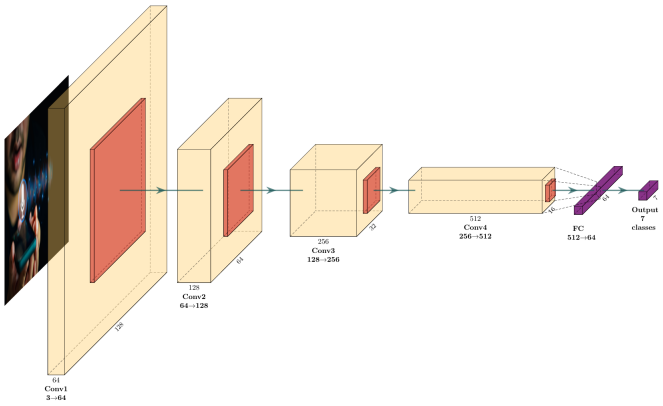

Figure 3. Model Visualization

IV. EXPERIMENT

A. Hardware Resources

Our model was trained on Kaggle using an NVIDIA Tesla P100 GPU with 16 GB VRAM, an Intel Xeon CPU at 2.30 GHz (2 cores), and 13 GB of RAM.

B. Training Configuration

- Batch Size: 4
- Epochs: 200
- Val Ratio: 0.15
- Test Ratio: 0.15
- Epochs: 200
- Loss Function: CrossEntropyLoss
- Optimizer: Adam (learning rate=1e-4)
- Scheduler: ReduceLROnPlateau (patience=5, factor=0.5)

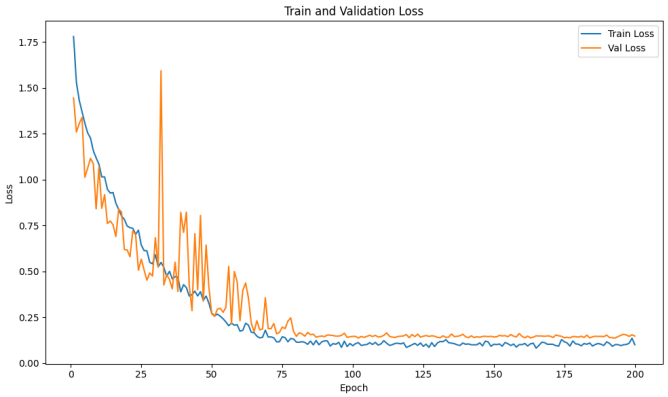

Figure 4. Train and Validation Loss Over Epochs

C. Logging Organization

Throughout the training phase, the hyperparameter configurations, loss, accuracy, F1 score, confusion matrix, and model artifacts were logged using Weights & Biases. The best-performing model was selected and saved based on the highest F1 score on the validation set.

D. Evaluation Metrics

The dataset EmoDB has imbalanced classes. Accuracy would be biased toward the dominant class. F1-macro averages the F1 score of each class, making it a fairer evaluation.

$$\text{Macro-F1} = \frac{1}{L}\sum_{i=1}^{L} F1_i \tag{2}$$

where:

- $F1_i$ – the F1 score of the i-th emotional label.
- L – total number of recordings.

V. Result

A. Overview

The training required approximately 6 minutes ($\sim$ 358.6 seconds) to complete 200 epochs. The best F1 score achieved on the validation set was 96%.

B. Metrics Visualization

To evaluate the learning behavior of the model, we plotted two key performance charts: Training & Validation Loss 4 and Training & Validation F1-score 5 over all epochs. These graphs help illustrate how well the model learns and generalizes throughout the training process.

In Figure 4, the training loss (blue curve) decreases rapidly during the initial epochs and converges to values below 0.1, indicating effective minimization of the objective function. The validation loss (orange curve) demonstrates higher variance early in training, including several spikes, but progressively stabilizes around 0.15.

In Figure 5, the training F1 score increases steadily to about 0.94. The validation F1 score fluctuates early but quickly stabilizes at comparable levels, indicating good generalization and minimal overfitting.

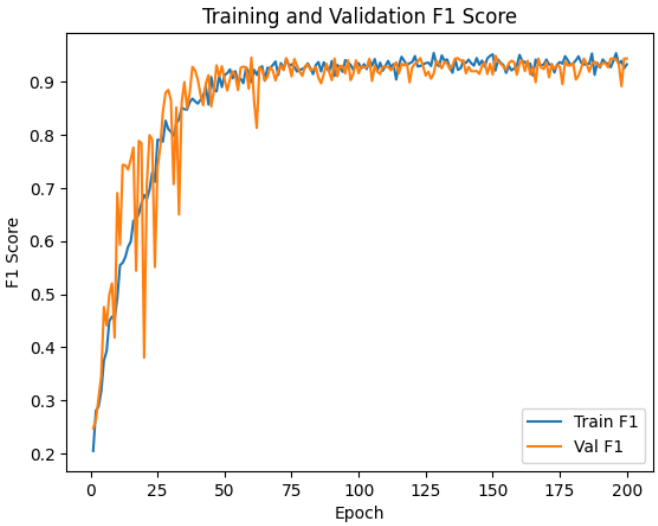

Figure 5. Train and Validation F1 Over Epochs

C. Classification Report

Table II
Classification Report by Emotion Labels

Class	Precision	Recall	F1-score	Support
fear	0.94	0.88	0.91	17
disgust	0.95	1.00	0.98	21
happy	0.91	0.95	0.93	21
angry	0.97	0.95	0.96	40
sad	1.00	1.00	1.00	38
neutral	1.00	0.95	0.98	21
boredom	0.97	1.00	0.98	28
Accuracy			0.97	186
Macro avg	0.96	0.96	0.96	186
Weighted avg	0.97	0.97	0.97	186

Interpretation:

- Per-class F1-scores: Most classes achieve F1 $>$ 0.90, showing good separation between emotion categories.
- Macro average F1-score: $\sim$ 0.96 $\rightarrow$ demonstrates the model's balanced performance across all classes.
- Perfect scores: Classes 4, 5, and 6 achieve near-perfect or perfect precision/recall/F1, indicating the model easily identifies these categories.
- Overall: Macro F1 of $\sim$ 0.96 $\rightarrow$ very high on EmoDB dataset, proving that the training pipeline is effective.

VI. Conclusion

In this work, we proposed a custom Convolutional Neural Network architecture for speech emotion recognition on the Berlin Database of Emotional Speech (EmoDB). By extracting MFCC-based time–frequency features along with their first- and second-order deltas, and feeding them into a four-block CNN, our model achieved a macro-F1 score of 0.97 on the

validation set and 0.96 on the held-out test set. These results demonstrate that the designed network effectively captures relevant spectral and temporal cues associated with different emotional states.

Future work will focus on adapting SER to Vietnamese and experimenting with hybrid CNN–attention architectures for enhanced real-time performance.