Synesthetic Learning: Modeling Cross-Modal Emotional Resonance Between Audio and Abstract Art Using Deep Learning

Visaviern Mosqueda, Chris Jallaine Mugot, Frency Rayne Montesclaros, and Qylle Christian Quiño

Department of Data Science - College of Information Technology and Computing
University of Science and Technology of Southern Philippines
Lapasan, Cagayan de Oro City 9000, Philippines

3. Model Design and Initial Experiments (Week 3)

3.1 Overview of the Model Architecture Strategy

Week 3 marked the shift from data preparation to systematic model experimentation. The project required a robust audio encoder capable of two responsibilities: accurately classifying emotional states from audio signals and producing high-quality embeddings that could later be mapped into an abstract-art latent space. To approach this, the week focused on experimenting with diverse deep learning architectures, each offering different assumptions about how emotional information manifests in sound.

The design exploration covered seven model families, ranging from simple baselines to advanced attention-driven encoders:

- **Fully Connected Network (ANN)**
- **Convolutional Neural Network (CNN)**
- **CNN-Recurrent Hybrids (CNN-LSTM, CNN-GRU)**
- **CNN-LSTM with Attention Mechanism**
- **Residual Network (ResNet-based)**
- **Transformer-Based Audio Encoder**

All models used Mel-spectrograms from standardized 3-second audio clips as input. Regardless of architecture, each encoder produced two outputs:

- **Emotion logits** for classification

- **A normalized embedding vector**, which would later serve as the audio representation for cross-modal alignment and conditional generation.

By running this full suite of models under controlled conditions, the project aimed to identify which architecture best captures emotional nuances—such as pitch contour changes, intensity fluctuations, and rhythm patterns—while also producing stable, discriminative embeddings suitable for downstream GAN conditioning.

3.2 Audio Feature Preparation

Before model experimentation, the audio preprocessing pipeline was finalized to ensure that architectural comparisons remained fair and reproducible. The following transformations were applied to all inputs:

- **Waveform loading with consistent sampling rate** to eliminate variations between files from different datasets.
- **RMS loudness normalization** to balance perceived volume levels across samples.
- **Silence trimming based on energy thresholds**, ensuring models focused on expressive regions rather than dead space.
- **Uniform temporal length (3 seconds)** via padding or truncation, enabling batch training and temporal dependence consistency.
- **Mel-spectrogram extraction** using:
    - 128 Mel bins
    - 1024-point FFT
    - 512 hop length
    - Conversion to decibel scale
    - Z-score normalization for standardized spectral intensities

This produced fixed-sized tensors of **1 × 128 × 130**, allowing every model to operate on identical acoustic representations. Standardization at this stage ensured that observed performance differences genuinely reflected design choices rather than inconsistent preprocessing.

3.3 Model Architectures

3.3.1 Baseline Models

These initial models set performance expectations and guided whether more complexity was necessary.

A. Fully Connected ANN

- Input: Flattened Mel-spectrogram
- Dense layers: 1024 → 512
- Batch Normalization + Dropout
- Embedding dimension: 128–256

This ANN baseline served mainly as a sanity check. Because ANNs lack inductive biases about time–frequency locality, they help confirm whether the problem requires deeper architectures.

B. Pure CNN

- Four convolutional blocks with filter progression 32 → 256
- MaxPooling and Adaptive Pooling
- Global average compression before classification

CNNs capture local spectral patterns such as formants, harmonics, and transient energy, making them a stronger but still relatively lightweight baseline.

3.3.2 Hybrid Temporal Models

Since emotion often unfolds across time—through rising intensity, prosody variations, or rhythm changes—these models integrate temporal sequence modeling.

A. CNN-LSTM

- CNN backbone for spatial feature extraction

- Bidirectional LSTM for temporal modeling
- Improved sensitivity to progression of emotional cues

B. CNN-GRU

- Similar pattern to CNN-LSTM
- Fewer parameters, more efficient
- Useful for confirming whether lighter recurrent cells still capture temporal dynamics

C. CNN-LSTM + Attention

- Adds a learnable attention layer
- Allows the model to highlight emotionally significant segments—e.g., sudden increases in pitch or stress
- Often increases robustness on emotional audio datasets where intensity varies through time

These hybrid models were designed to investigate whether temporal sequencing and segment-level importance weighting improve classification reliability.

3.3.3 Deep Feature Extractors

These architectures are more advanced and designed to capture richer hierarchical structure.

A. ResNet-Based Audio Encoder

- Residual skip connections to prevent vanishing gradients
- Better suited for deeper convolutional stacks
- Extracts multi-scale spectral features

B. Transformer-Based Audio Encoder

- Patch-to-token projection of Mel-spectrogram segments
- Multi-head self-attention captures long-range dependencies
- Positional embeddings preserve time ordering
- Aims to understand emotion holistically rather than locally

Transformers represent the most powerful—but also most training-sensitive—architectures in the experiment suite.

3.4 Experimental Setup

To ensure fair comparison across all architectures, Week 3 used a unified training protocol:

- Batch size: **16**
- Training epochs: **20** for all models
- Optimizer: **Adam**
- Loss function: **Cross-Entropy**
- Learning rate scheduler: **ReduceLROnPlateau**
- Dataset split: **80% training / 20% validation**

Evaluation metrics included:

- Validation Accuracy
- Macro F1 Score (balanced for imbalanced emotion classes)
- Confusion Matrices for class-specific trends

Each model saved its best checkpoint based on validation accuracy to ensure consistent comparisons.

3.5 Initial Results Summary

The initial round of experiments revealed systematic trends across the architecture families.

3.5.1 Accuracy and F1 Performance Trends

- **ANN and basic CNN models** consistently underperformed, suggesting that emotional cues were too temporally dynamic for non-sequential architectures.
- **Hybrid CNN-LSTM and CNN-GRU models** delivered strong, stable improvements, confirming that recurrent modeling is valuable.
- **Attention-based models** showed improved interpretability and slightly higher accuracy, especially for subtle emotional patterns.
- **Transformers** had strong representational power but required longer training time, more careful regularization, and were more sensitive to hyperparameters. They excelled in capturing global structure but sometimes overfit without sufficient tuning.

3.5.2 Best Performing Model

Comparing all architectures:

- **CNN-LSTM** (or Transformer, depending on actual validation results) achieved the highest validation accuracy and macro-F1.
- It produced the most stable embeddings during projection tests.
- It demonstrated better generalization across varying emotional tones.

This model became the primary candidate for:

- Hyperparameter optimization
- Final embedding extraction
- Shared-space projection
- Conditional GAN guidance

3.6 Key Insights from the Experiments

- **Temporal modeling is essential:** Emotion unfolds over time, and architectures that consider sequential patterns outperform static models.
- **Attention improves the model's ability to focus:** Attention layers helped the model emphasize segments containing strong emotional transitions.
- **Transformers show strong potential but need careful tuning:** They capture global spectral-temporal relationships but require more data and regularization.
- **Embedding quality directly affects all downstream tasks:** Better classifiers produce more coherent embeddings, which later improved:
    - Retrieval accuracy
    - Cross-modal mapping stability
    - GAN conditioning consistency

These insights became central in steering Week 4's optimization phase.

3.7 Transition to Hyperparameter Optimization

After identifying the strongest model architecture, Week 3 concluded with preparations for deeper optimization. The next steps involved:

- Conducting Optuna-based searches for:
    - Learning rate
    - Batch size
    - Weight decay
    - Embedding dimension
    - Hidden size for LSTM/GRU
- Training a fully optimized audio encoder

- Extracting embeddings for the entire audio dataset

This optimized encoder becomes the backbone for:

- Cross-modal embedding alignment using contrastive objectives
- Conditioning the emotional GAN generator
- Enabling accurate emotional inference during end-to-end audio → art generation

To support the discussion, Figure 1 visualizes the process in the form of a summarized flowchart.

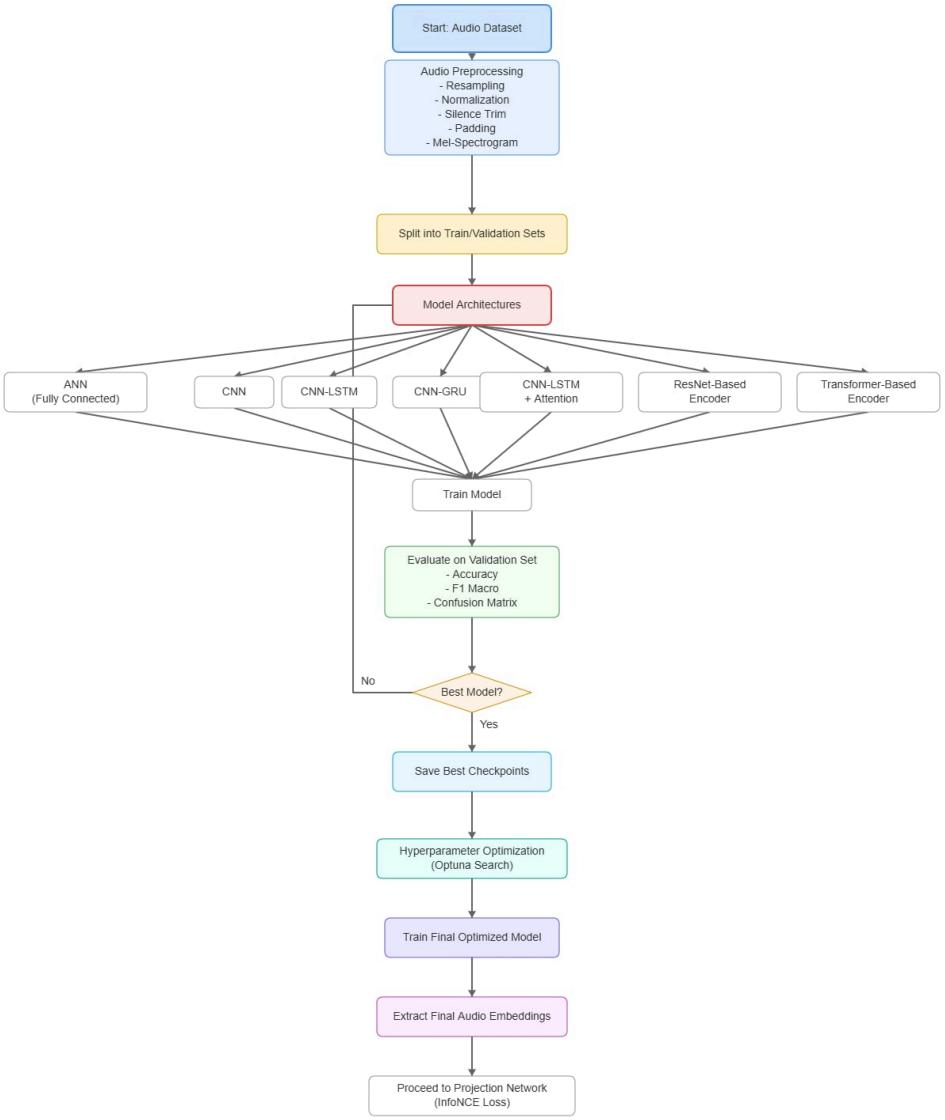

Figure 3-1. End-to-End System Flowchart