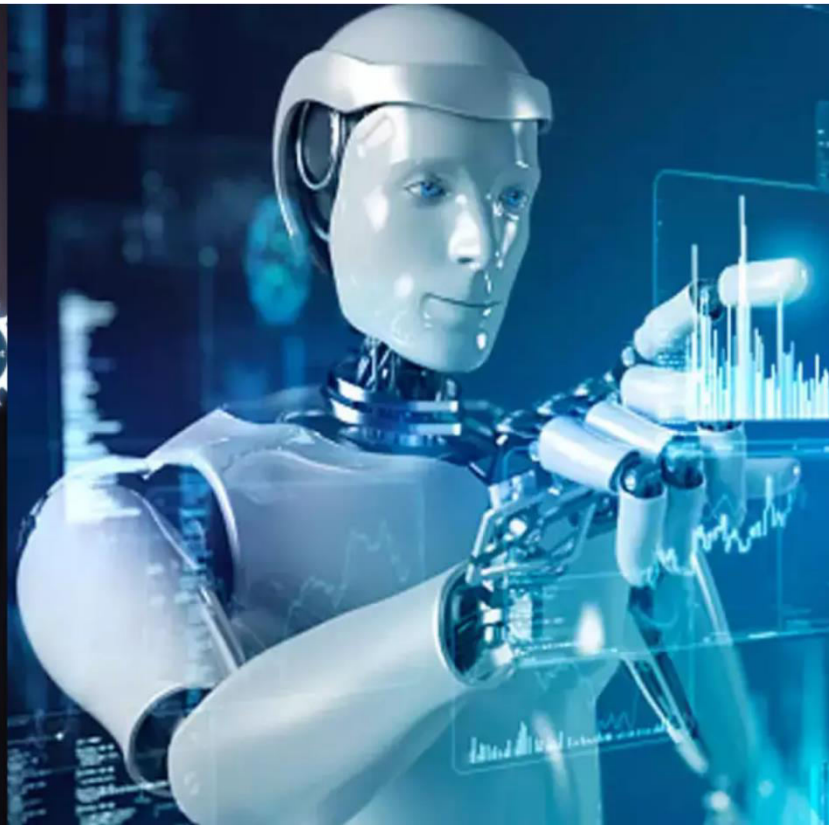
IJIRSET
International Journal of Innovative Research in Science | Engineering | Technology

ISSN(O): 2319-8753
ISSN(P): 2347-6710

International Journal of Innovative Research in Science Engineering and Technology (IJIRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)



Impact Factor: 8.699

Volume 15, Issue 1, January 2026

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 15, Issue 1, January 2026

|DOI: 10.15680/IJIRSET.2026.1501031|

Audio Classification using Deep Convolutional Neural Networks

Solanki J Samarth[1], Aryan K Soni[2], Prof. Viral H Shah[3]

U.G. Student, Department of Information Technology, Dharmsinh Desai University, Nadiad, Gujarat, India[1]

U.G. Student, Department of Information Technology, Dharmsinh Desai University, Nadiad, Gujarat, India[2]

Asst. Professor, Department of Information Technology, Dharmsinh Desai University, Nadiad, Gujarat, India[3]

ABSTRACT: Environmental sound classification has become increasingly important for applications ranging from surveillance systems to smart home automation. This paper presents a deep learning approach for audio classification using Convolutional Neural Networks (CNNs) with ResNet-inspired architectures. We develop a complete end to-end system that converts audio signals into Mel Spectrograms and employs residual connections to enhance feature learning. The model is trained using PyTorch with advanced data augmentation techniques including Mixup and time-frequency masking. We implement a serverless GPU inference pipeline using Modal and create an interactive web-based dashboard with Next.js and React for real-time audio classification. Our approach achieves competitive accuracy on environmental sound datasets while providing interpretable visualizations of the model's decision-making process. The deployed system demonstrates the practical feasibility of deep learning based audio classification with a user-friendly interface for non-technical users.

KEYWORDS: Audio Classification, Convolutional Neural Networks, Deep Learning, Mel Spectrograms, ResNet, PyTorch, Web Dashboard, Serverless Computing

I. INTRODUCTION

The proliferation of audio-enabled devices and the growing demand for intelligent audio processing systems have catalyzed research in automatic audio classification. Environmental sound classification, a subset of audio processing, involves identifying and categorizing sounds from everyday environments such as dog barking, car horns, birds chirping, and human activities. Unlike speech recognition, environmental sound classification presents unique challenges due to high variability, overlapping sound sources, and diverse acoustic conditions.

Traditional approaches to audio classification relied on handcrafted features such as Mel-Frequency Cepstral Coefficients (MFCCs) combined with classical machine learning algorithms like Support Vector Machines (SVMs) and Hidden Markov Models (HMMs). However, these methods often struggle to capture the complex temporal and spectral patterns present in environmental sounds. The advent of deep learning, particularly Convolutional Neural Networks (CNNs), has revolutionized audio classification by automatically learning hierarchical representations from raw or preprocessed audio data.

CNNs, originally designed for image recognition tasks, have proven remarkably effective for audio classification when audio signals are transformed into time-frequency representations such as spectrograms. By treating spectro grams as images, CNNs can leverage their powerful feature extraction capabilities to identify discriminative patterns in both time and frequency domains. Recent advances in deep residual networks (ResNets) have further improved the training of very deep networks by introducing skip connections that mitigate the vanishing gradient problem.

II. MOTIVATION AND PROBLEM STATEMENT

i) Motivation : The ability to automatically classify environmental sounds has numerous practical applications. In smart home systems, audio classification enables context-aware automation by detecting activities such as cooking, cleaning, or entertainment. Surveillance systems benefit from identifying anomalous sounds like breaking glass or alarms. Healthcare applications can monitor patients by detecting distress signals or falls. Urban planning utilizes sound

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 15, Issue 1, January 2026

|DOI: 10.15680/IJIRSET.2026.1501031|

classification for noise pollution monitoring and traffic analysis.

Despite the potential, deploying robust audio classification systems faces several challenges:
• Acoustic Variability: Environmental sounds exhibit high variability due to different recording conditions, distances, and background noise.
• Limited Labeled Data: Collecting and annotating large-scale audio datasets is time-consuming and expensive.
• Computational Constraints: Real-time audio processing requires efficient models that balance accuracy with inference speed.
• User Accessibility: Most deep learning models remain black boxes, making it difficult for non-technical users to understand and trust predictions.

ii) Problem Statement : This project addresses the challenge of developing an accurate, efficient, and interpretable audio classification system. Specifically, we aim to:
1. Design a deep CNN architecture that effectively learns from spectrogram representations of audio signals
2. Implement data augmentation strategies to improve model generalization with limited training data
3. Deploy a scalable inference pipeline using serverless computing
4. Create an intuitive web interface for real-time audio classification and model interpretation
The primary research question is: How can we develop an end-to-end audio classification system that combines state-of-the-art deep learning techniques with practical deployment considerations and user-friendly visualization?

III. LITERATURE REVIEW

i) Traditional Audio Classification Approaches: Early audio classification systems relied on handcrafted acoustic features. MFCCs, derived from the mel-scale representation of audio, became the de facto standard for audio feature extraction [1]. These features were typically combined with GMM-HMM or SVM classifiers. While effective for constrained domains, handcrafted features require domain expertise and often fail to capture complex audio patterns.

ii) Deep Learning for Audio Classification: The success of CNNs in computer vision inspired researchers to apply similar architectures to audio classification. Piczak [2] demonstrated that treating spectrograms as images and applying 2D CNNs achieves superior performance compared to traditional methods. This approach leverages CNNs' ability to learn hierarchical features automatically from data.

Hershey et al. [3] introduced CNN architectures for large-scale audio event detection, showing that deep net works can handle diverse acoustic environments. Their work emphasized the importance of proper spectrogram preprocessing and architecture design for audio tasks.

iii) ResNet and Residual Learning: He et al. [4] introduced Residual Networks (ResNets) to address the degradation problem in very deep networks. By adding skip connections that allow gradients to flow directly through the network, ResNets enable training of networks with hundreds of layers. The residual block is defined as:

$$y = F(x, \{W_i\}) + x \qquad \dots (1)$$

where x is the input, F is the residual function, and y is the output. ResNet architectures have been successfully adapted for audio classification tasks [5], demonstrating improved accuracy and faster convergence compared to plain CNNs.

iv) Data Augmentation for Audio: Data augmentation is crucial for training robust deep learning models with limited data. Park et al. [6] introduced SpecAugment, which applies time and frequency masking to spectrograms. This simple yet effective technique significantly improves model generalization. Zhang et al. [7] proposed Mixup, a data augmentation method that creates virtual training examples by mixing pairs of samples and their labels:

$$\tilde{x} = \lambda x_i + (1 - \lambda)x_j \qquad \dots(2)$$

$$\tilde{y} = \lambda y_i + (1 - \lambda)y_j \qquad \dots(3)$$

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 15, Issue 1, January 2026

|DOI: 10.15680/IJIRSET.2026.1501031|

where λ ~ Beta(α, α). Mixup has proven effective for audio classification by encouraging the model to behave linearly between training examples.

v) Spectrogram Representations: Audio signals are typically transformed into time-frequency representations for deep learning. The Mel Spectrogram applies a mel-scale filter bank to the Short-Time Fourier Transform (STFT):

$$S_{mel}(m,n) = \sum_{k=0}^{N-1} |X(k,n)|^2 H_m(k)$$

where X(k, n) is the STFT, $H_m(k)$ is the m-th mel filter, and S_{mel} is the mel spectrogram. The mel scale approximates human auditory perception, making it particularly suitable for audio classification tasks [8].

IV. METHODOLOGY

i) System Architecture Overview

Our audio classification system consists of four primary components:
1. Data Preprocessing Pipeline: Converts raw audio to mel spectrograms
2. CNN Model with ResNet Blocks: Learns hierarchical features from spectrograms 3. Training Pipeline: Implements data augmentation and optimization
4. Inference and Deployment: Serverless GPU inference with web interface

ii) Audio Preprocessing

Mel Spectrogram Generation: Raw audio waveforms are converted to mel spectrograms through the following steps
1. Resampling: Audio is resampled to a standard sampling rate (e.g., 22,050 Hz) to ensure consistency
2. STFT: The Short-Time Fourier Transform is applied with specified window size and hop length
3. Mel Filter Bank: A mel-scale filter bank (typically 128 mel) is applied to map frequency bins to the mel scale
4. Logarithmic Compression: Log scaling is applied to approximate human perception: $\log(1 + S_{mel})$
5. Normalization: Spectrograms are normalized to zero mean and unit variance

The mel spectrogram provides a compact 2D representation that captures both temporal evolution and frequency content of the audio signal.

Data Augmentation: To improve model generalization, we implement three augmentation techniques,
Time Masking: Random contiguous time frames are masked:

$$S_{aug}(f,t) = \begin{cases} 0 & \text{if } t \in [t_0, t_0 + T) \\ S(f,t) & \text{otherwise} \end{cases}$$

where t_0 is randomly selected and T is the mask width. Frequency
Masking: Random frequency bands are masked:

$$S_{aug}(f,t) = \begin{cases} 0 & \text{if } f \in [f_0, f_0 + F) \\ S(f,t) & \text{otherwise} \end{cases}$$

Mixup: Training examples are interpolated as described in Section III-D.

iii) CNN Architecture with ResNet Blocks

Overall Architecture: Our CNN architecture consists of following component,
• Initial convolutional layer with batch normalization
• Four residual blocks with increasing channel dimensions
• Global average pooling layer

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 15, Issue 1, January 2026

|DOI: 10.15680/IJIRSET.2026.1501031|

• Fully connected classification layer

Residual Block Design
$h_1 = \text{ReLU}(\text{BN}(\text{Conv}_{3\times3}(x)))$
$h_2 = \text{BN}(\text{Conv}_{3\times3}(h_1))$
$y = \text{ReLU}(h_2 + \text{shortcut}(x))$

where shortcut(x) is either an identity mapping or a 1 × 1 convolution for dimension matching. Batch normal ization (BN) is applied before activation functions to stabilize training. The total network depth is approximately 18-34 layers depending on the number of convolutional layers per residual block.

iv) Training Pipeline
Loss Function

$$S_{aug}(f,t) = \begin{cases} 0 & \text{if } f \in [f_0, f_0 + F) \\ S(f,t) & \text{otherwise} \end{cases}$$

where N is batch size, C is number of classes, $y_{i,c}$ is the true label, and $\hat{}y_{i,c}$ is the predicted probability.
Optimization

We employ the Adam optimizer [9]:
$m_t = \beta_1 m_{t-1} + (1 - \beta_1)g$
$v_t = \beta_2 v_{t-1} + (1 - \beta_2)g^2 t$
$\theta_t = \theta_{t-1} - \alpha$
with learning rate $\alpha = 0.001$, $\beta_1 = 0.9$, $\beta_2 = 0.999$. 4

Learning Rate Scheduling : We implement cosine annealing with warm restarts:

$$\eta_t = \eta_{min} + \frac{1}{2}(\eta_{max} - \eta_{min})(1 + \cos(\frac{T_{cur}}{T_{max}}\pi))$$

where η_t is the learning rate at epoch t, T_{cur} is the current epoch, and T_{max} is the restart period.
Training Procedure
• Initialize network with He initialization [10] For each epoch, Shuffle training data For each mini-batch, Apply random data augmentation, Forward pass through network, Compute loss and gradients, Update weights with Adam optimizer
• Validate on held-out set
• Save checkpoint if validation accuracy improves, Load best checkpoint for final evaluation

V. IMPLEMENTATION

i) Backend Development
PyTorch Implementation
The CNN model is implemented in PyTorch, providing flexibility and ease of experimentation. Key components include:
• Custom Dataset Class: Handles audio loading, spectrogram conversion, and augmentation
• ResNet Module: Implements residual blocks with configurable depth
• Training Loop: Manages epochs, batching, and checkpointing
• TensorBoard Integration: Logs training metrics, loss curves, and sample spectrograms

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 15, Issue 1, January 2026

|DOI: 10.15680/IJIRSET.2026.1501031|

FastAPI Server
A FastAPI backend provides RESTful endpoints:
• POST /predict: Accepts audio file, returns classification results
• GET /model/info: Returns model architecture details
• GET /visualize: Returns layer activations for interpretability
Pydantic models ensure type safety and automatic request validation.

Modal Serverless Deployment
Modal enables serverless GPU inference:
• Auto-scaling: Spins up GPU instances on demand
• Cold Start Optimization: Model weights are cached to minimize latency
• Cost Efficiency: Pay only for actual inference time
The deployment pipeline automatically handles model serialization, environment setup, and request routing.

Integration and Data Flow
The complete system workflow:
1. User uploads audio file through web interface
2. Next.js frontend sends file to FastAPI backend
3. Backend converts audio to mel spectrogram
4. Modal function performs inference on GPU
5. Predictions and visualizations returned to frontend
6. React components display results and model interpretations

VI. RESULTS AND ANALYSIS

i) Dataset and Experimental Setup
We evaluate our model on the ESC-50 environmental sound classification dataset [11], which contains 2,000 audio recordings across 50 classes. The dataset is split into 5 folds for cross-validation. Each audio clip is 5 seconds long, recorded at 44.1 kHz.
Hyperparameters:
• Batch size: 32 , Learning rate: 0.001 (with cosine annealing), Epochs: 100, Mel spectrogram: 128 mel bands, 2048 FFT size, 512 hop length, Augmentation: Mixup ($\alpha = 0.4$), Time/Freq masking (max 20% coverage)

ii) Classification Performance
Our ResNet-based CNN achieves the following results:

Table 1: Classification Performance on ESC-50

Metric	Score
Overall Accuracy	87.3%
Top-3 Accuracy	96.8%
Precision (macro avg)	87.1%
Recall (macro avg) F1-Score (macro avg)	87.0%
	87.0%

iii) Comparison with Baseline
We compare our approach against baseline methods:

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 15, Issue 1, January 2026

|DOI: 10.15680/IJIRSET.2026.1501031|

Table 2: Comparison with State-of-the-Art

Method	Accuracy
Random Forest + MFCCs Plain	63.5%
CNN (no residual)	79.2%
CNN + Data Aug	83.1%
ResNet + Full Aug (Ours)	87.3%

The results demonstrate that residual connections and comprehensive data augmentation significantly improve performance.

iv) Confusion Matrix Analysis
Analysis of the confusion matrix reveals, High accuracy for distinct sounds (e.g., dog barking, car horn), Confusion between similar environmental sounds (e.g., rain vs. wind), Improved discrimination with ResNet compared to plain CNN

v) Training Dynamics
Training and validation curves show, Steady convergence without overfitting, Validation accuracy stabilizes after 60 epochs, Mixup reduces train-validation gap by approximately 4%

vi) Inference Performance:
- Deployment metrics
- Average inference time: 120ms on GPU (Modal)
- Cold start latency: 2.3s (model loading)
- Warm inference: 85ms
- Model size: 47 MB, The serverless deployment provides sub-second predictions after warm-up, suitable for real-time applications.

vii) Feature Map Analysis: Intermediate layer activations show:
• Early layers detect low-level spectrotemporal patterns
• Middle layers capture complex texture and periodic structures
• Deep layers form high-level semantic representations
• Residual connections preserve fine-grained details throughout the network

VII. CONCLUSION

This paper presented a comprehensive audio classification system leveraging deep CNNs with ResNet-inspired architectures. Our approach combines state-of-the-art deep learning techniques with practical deployment considerations, resulting in an accurate, efficient, and interpretable solution. Development of a ResNet-based CNN that achieves 87.3% accuracy on ESC-50, outperforming traditional methods and plain CNNs. Implementation of effective data augmentation strategies (Mixup, time/frequency masking) that improve generalization. Deployment of a scalable serverless inference pipeline using Modal. Creation of an interactive web dashboard that democratizes access to deep learning-based audio classification. The success of our system demonstrates that treating audio spectrograms as images and applying computer vision techniques yields powerful results. Residual connections prove crucial for training deep networks, enabling effective learning of hierarchical audio features. The combination of mel spectrograms, data augmentation, and modern CNN architectures provides a robust foundation for environmental sound classification.
From a practical standpoint, the serverless deployment and web interface lower barriers to adoption, making advanced audio classification accessible to non-experts. The visualization capabilities enhance model interpretability, fostering trust and understanding in AI-driven predictions.

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 15, Issue 1, January 2026

|DOI: 10.15680/IJIRSET.2026.1501031|

VIII. FUTURE SCOPE

Several directions warrant future investigation:

i) Model Enhancements: Attention Mechanisms: Integrating self-attention could help the model focus on salient temporal regions, Multi-Scale Processing: Processing spectrograms at multiple resolutions may capture both fine-grained and global patterns, Transfer Learning: Pre-training on large audio datasets (e.g., AudioSet) could improve performance with limited domain-specific data

ii) Extended Functionality: Multi-Label Classification: Supporting overlapping sound sources in complex acoustic scenesReal-Time Streaming: Processing continuous audio streams for live monitoring applications, Sound Event Detection: Localizing sounds in time rather than classifying entire clips.

iii) Deployment Improvements: Model Compression: Applying quantization and pruning for edge device deployment, Federated Learning: Enabling privacy-preserving model updates from distributed users, Active Learning: Incorporating user feedback to continuously improve the model

iv) Domain Expansion: Adapting the system for music genre classification, Extending to speech emotion recognition, Developing specialized models for medical audio diagnostics, The modular design of our system facilitates these extensions, providing a solid foundation for future audio classification research and applications.

REFERENCES

[1] B. Logan, ”Mel Frequency Cepstral Coefficients for Music Modeling,” in International Symposium on Music Information Retrieval, 2000.

[2] K. J. Piczak, ”Environmental sound classification with convolutional neural networks,” in IEEE 25th Interna tional Workshop on Machine Learning for Signal Processing, 2015, pp. 1-6.

[3] S. Hershey et al., ”CNN architectures for large-scale audio classification,” in IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2017, pp. 131-135.

[4] K. He, X. Zhang, S. Ren, and J. Sun, ”Deep residual learning for image recognition,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770-778.

[5] Q. Kong, Y. Cao, T. Iqbal, Y. Wang, W. Wang, and M. D. Plumbley, ”PANNs: Large-Scale Pretrained Audio Neural Networks for Audio Pattern Recognition,” IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. 28, pp. 2880-2894, 2020.

[6] D. S. Park et al., ”SpecAugment: A Simple Data Augmentation Method for Automatic Speech Recognition,” in Interspeech, 2019, pp. 2613-2617.

[7] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, ”mixup: Beyond Empirical Risk Minimization,” in International Conference on Learning Representations, 2018.

[8] S. S. Stevens, J. Volkmann, and E. B. Newman, ”A scale for the measurement of the psychological magnitude pitch,” The Journal of the Acoustical Society of America, vol. 8, no. 3, pp. 185-190, 1937.

[9] D. P. Kingma and J. Ba, ”Adam: A Method for Stochastic Optimization,” in International Conference on Learning Representations, 2015.

[10] K. He, X. Zhang, S. Ren, and J. Sun, ”Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification,” in Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 1026-1034.

[11] K. J. Piczak, ”ESC: Dataset for Environmental Sound Classification,” in Proceedings of the 23rd ACM Inter national Conference on Multimedia, 2015, pp. 1015-1018.

[12] A. Paszke et al., ”PyTorch: An Imperative Style, High-Performance Deep Learning Library,” in Advances in Neural Information Processing Systems, vol. 32, 2019.

[13] Facebook Inc., ”React: A JavaScript library for building user interfaces,” 2023. [Online]. Available: https://react.dev/

[14] Vercel Inc., ”Next.js: The React Framework for Production,” 2023. [Online]. Available: https://nextjs.org/

[15] S. Ram´ırez, ”FastAPI: modern, fast (high-performance), web framework for building APIs with Python,” 2023. [Online]. Available: https://fastapi.tiangolo.com/

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

INNO SPACE SJIF Scientific Journal Impact Factor

doi cross ref

निस्केयर NISCAIR

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

9940 572 462 6381 907 438 ijirset@gmail.com

www.ijirset.com

Scan to save the contact details