Efficient Deep Learning Approach for Multiclass Sound Classification

Urban Sound Scene Recognition using Modified ResNet18

DJEZIRI Oussama Baidar Samir Senhadji M Said
Ali Abbou Oussama
Higher School of Computer Science
ESI-SBA

Abstract

This report presents a comprehensive deep learning approach for multiclass sound classification, specifically focusing on urban sound scene recognition. We developed a modified ResNet18 architecture combined with advanced audio preprocessing techniques and data augmentation strategies to classify audio recordings into 10 distinct urban environment categories. Using the TUT Urban Acoustic Scenes 2018 Development Dataset from Hugging Face, our model achieved impressive results with a macro F1-score of 0.8631, precision of 0.8649, and recall of 0.8646. The implementation incorporates mel-spectrogram extraction, SpecAugment, Mixup augmentation, label smoothing, and gradient clipping to enhance model performance and generalization capabilities.

1 Introduction

Sound classification has emerged as a critical application in various domains, from environmental monitoring to smart city development. The ability to automatically recognize and classify urban acoustic scenes has significant implications for urban planning, noise pollution monitoring, security systems, and accessibility technologies. This project addresses the challenge of multiclass sound classification using modern deep learning techniques, specifically targeting urban sound scene recognition.

Urban environments present unique acoustic challenges due to the complexity and variability of sound sources. Traditional signal processing approaches often fall short in capturing the intricate patterns present in real-world audio data. Deep learning, particularly convolutional neural networks (CNNs), has shown remarkable success in audio classification tasks by learning hierarchical representations from spectrograms.

2 Project Benefits and Applications

2.1 Real-World Applications

The development of robust sound classification systems offers numerous practical benefits:

- **Smart City Development**: Automatic monitoring of urban acoustic environments for city planning and noise management
- **Environmental Monitoring**: Real-time assessment of acoustic pollution levels in different urban zones
- **Security and Surveillance**: Enhanced security systems capable of recognizing and responding to specific acoustic events
- **Accessibility Technologies**: Assistive devices for hearing-impaired individuals to understand their acoustic environment
- **Urban Planning**: Data-driven insights for creating more livable urban spaces
- **Transportation Systems**: Monitoring and optimization of public transportation based on acoustic signatures

2.2 Technical Contributions

This project contributes to the field through:

- Implementation of state-of-the-art audio augmentation techniques (SpecAugment, Mixup)
- Adaptation of ResNet18 architecture for single-channel spectrogram processing
- Comprehensive evaluation framework with detailed performance metrics
- End-to-end pipeline from raw audio to classification results

3 Dataset Description and Preprocessing

3.1 TUT Urban Acoustic Scenes 2018 Dataset

We utilized the TUT Urban Acoustic Scenes 2018 Development Dataset, sourced from Hugging Face[1]. This dataset is specifically designed for acoustic scene classification research and contains high-quality recordings from various urban environments.

3.1.1 Dataset Characteristics

- **Format**: 16-bit WAV files
- **Sampling Rate**: 44.1 kHz (resampled to 16 kHz for efficiency)
- **Duration**: Variable length recordings (standardized to 2 seconds)
- **Classes**: 10 distinct urban acoustic scenes
- **Total Samples**: Comprehensive coverage across all categories

[1] https://huggingface.co/datasets/wetdog/TUT-urban-acoustic-scenes-2018-development-16bit

3.1.2 Class Distribution

The dataset includes the following 10 urban acoustic scene categories:

1. Airport
2. Bus
3. Metro/Subway
4. Metro Station
5. Park
6. Public Square
7. Shopping Mall
8. Street (Pedestrian)
9. Street (Traffic)
10. Tram

3.2 Audio Preprocessing Pipeline

3.2.1 Signal Processing Steps

Our preprocessing pipeline consists of several critical steps:

1. **Resampling**: All audio files were resampled to 16 kHz to reduce computational complexity while preserving essential acoustic information
2. **Duration Normalization**: Audio clips were standardized to 2 seconds through truncation or zero-padding
3. **Mel-Spectrogram Extraction**: Conversion of time-domain audio signals to frequency-domain representations using mel-scale filterbanks
4. **Amplitude to Decibel Conversion**: Logarithmic scaling for better dynamic range representation
5. **Normalization**: Z-score normalization applied to each spectrogram for stable training

3.2.2 Mel-Spectrogram Configuration

The mel-spectrogram extraction used the following parameters:

- **Number of Mel Bins**: 128
- **FFT Size**: 1024
- **Hop Length**: 512 samples
- **Window Function**: Hann window
- **Top dB**: 80 dB for amplitude to decibel conversion

4 Data Augmentation Strategies

To improve model generalization and robustness, we implemented multiple augmentation techniques operating at both waveform and spectrogram levels.

4.1 Waveform-Level Augmentations

- **Time Shifting**: Random circular shifts up to ±0.1 seconds to simulate temporal variations
- **Amplitude Scaling**: Random amplitude multiplication (0.8-1.2×) to account for volume variations

4.2 Spectrogram-Level Augmentations (SpecAugment)

- **Time Masking**: Random masking of 20 consecutive time frames
- **Frequency Masking**: Random masking of 10 consecutive frequency bins

4.3 Mixup Augmentation

Mixup is a powerful regularization technique that creates virtual training samples by linearly interpolating between pairs of training examples:

$$\tilde{x} = \lambda x_i + (1 - \lambda)x_j \tag{1}$$

$$\tilde{y} = \lambda y_i + (1 - \lambda)y_j \tag{2}$$

where $\lambda \sim \text{Beta}(\alpha, \alpha)$ with $\alpha = 0.4$ in our implementation.

5 Model Architecture

5.1 Modified ResNet18

We adapted the ResNet18 architecture for single-channel spectrogram processing. The key modifications include:

- **Input Layer**: Modified first convolutional layer to accept single-channel input (mel-spectrograms)
- **Feature Extraction**: Leveraged ResNet18's residual blocks for hierarchical feature learning
- **Classification Head**: Replaced the final fully connected layer with a dropout-regularized classifier for 10 classes

5.2 Architecture Details

Listing 1: Modified ResNet18 Architecture

```
class EfficientResNetAudio(nn.Module):
    def __init__(self, num_classes=10, input_channels=1):
        super().__init__()
        self.resnet = models.resnet18(pretrained=False)
        # Modify first conv layer for single-channel input
        self.resnet.conv1 = nn.Conv2d(
            input_channels, 64,
            kernel_size=7, stride=2,
            padding=3, bias=False
        )
        # Replace classifier with dropout-regularized head
        in_features = self.resnet.fc.in_features
        self.resnet.fc = nn.Sequential(
            nn.Dropout(0.3),
            nn.Linear(in_features, num_classes)
        )
```

6 Training Methodology

6.1 Training Configuration

Our training setup incorporated several best practices for deep learning:

Table 1: Training Hyperparameters

Parameter	Value
Batch Size	32
Learning Rate	1e-4
Optimizer	AdamW
Weight Decay	5e-4
Epochs	60
Early Stopping Patience	10
Train/Validation Split	80/20
Mixup Alpha	0.4
Label Smoothing	0.1
Gradient Clipping	1.0

6.2 Training Techniques

6.2.1 Loss Function and Regularization

- **Cross-Entropy Loss**: With label smoothing (smoothing factor = 0.1) to prevent overconfidence
- **AdamW Optimizer**: Incorporating weight decay for better generalization

- **Gradient Clipping**: Maximum norm of 1.0 to prevent exploding gradients

6.2.2 Learning Rate Scheduling

We employed ReduceLROnPlateau scheduler with:

- **Mode**: Maximize validation F1-score
- **Factor**: 0.5 (halve learning rate)
- **Patience**: 2 epochs

6.2.3 Early Stopping

Training was monitored using validation F1-score with early stopping patience of 10 epochs to prevent overfitting.

7 Results and Evaluation

7.1 Overall Performance Metrics

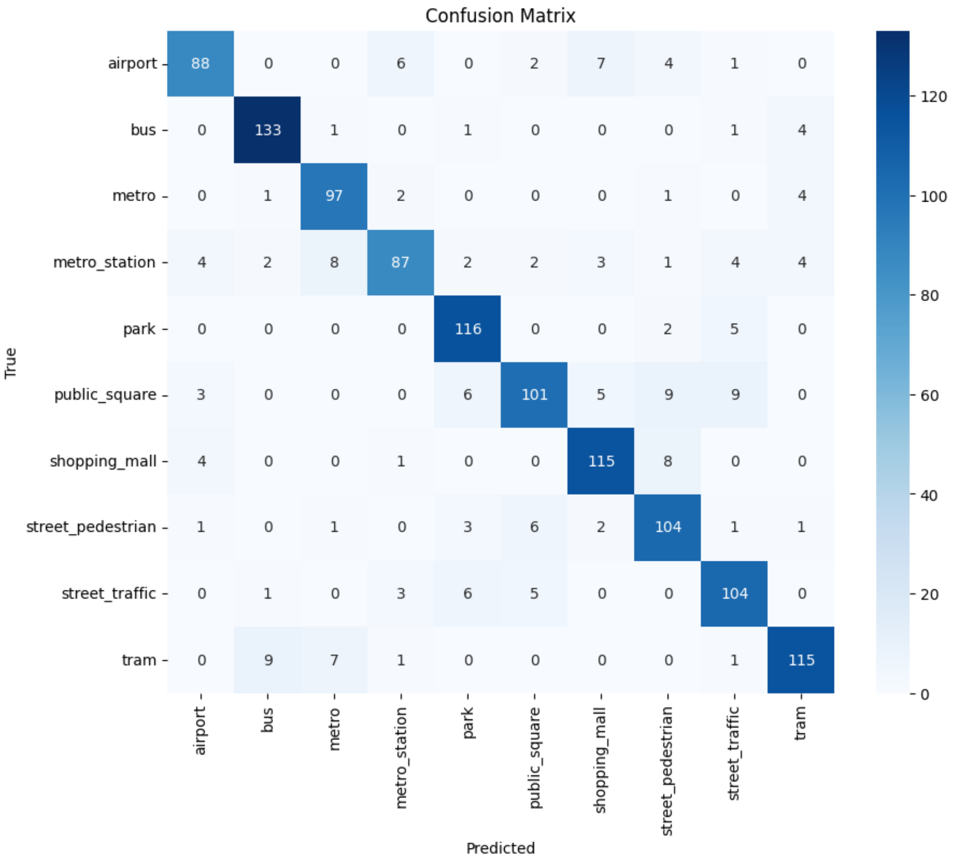
Our model achieved excellent performance across all evaluation metrics:

Table 2: Overall Model Performance

Metric	**Score**
Macro F1-Score	0.8631
Macro Precision	0.8649
Macro Recall	0.8646

7.2 Confusion Matrix Analysis

The confusion matrix (Figure 1) provides detailed insights into model performance across all classes. The diagonal elements represent correct classifications, while off-diagonal elements indicate misclassifications.



	airport	bus	metro	metro_station	park	public_square	shopping_mall	street_pedestrian	street_traffic	tram
airport	88	0	0	6	0	2	7	4	1	0
bus	0	133	1	0	1	0	0	0	1	4
metro	0	1	97	2	0	0	0	1	0	4
metro_station	4	2	8	87	2	2	3	1	4	4
park	0	0	0	0	116	0	0	2	5	0
public_square	3	0	0	0	6	101	5	9	9	0
shopping_mall	4	0	0	1	0	0	115	8	0	0
street_pedestrian	1	0	1	0	3	6	2	104	1	1
street_traffic	0	1	0	3	6	5	0	0	104	0
tram	0	9	7	1	0	0	0	0	1	115

Figure 1: Confusion Matrix showing classification performance across all 10 urban acoustic scene categories

Key observations from the confusion matrix:

- **Best Performance**: Bus classification achieved the highest accuracy with minimal confusion
- **Challenging Classes**: Metro_station showed some confusion with other transportation-related categories
- **Clear Distinctions**: Park and shopping_mall categories were well-distinguished from others

7.3 Per-Class Performance Analysis

Figure 2 illustrates the F1-scores for each individual class, highlighting the model's balanced performance across different urban acoustic scenes.

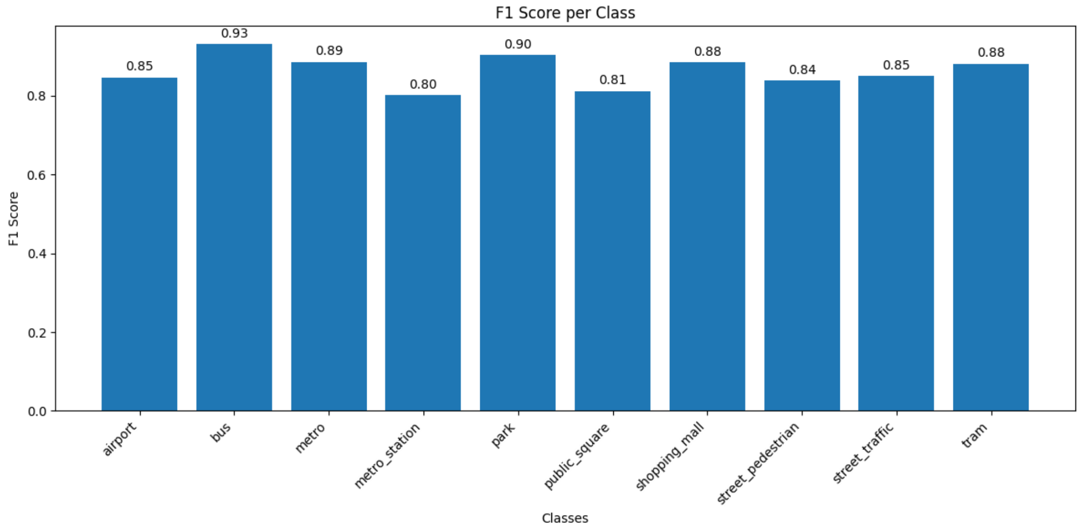


Figure 2: F1-Score performance for each urban acoustic scene category

7.3.1 Class-Specific Analysis

- **Highest F1-Scores**: Bus (0.93), Park (0.90), Metro (0.89)
- **Most Challenging**: Metro_station (0.80) due to acoustic similarity with other transportation environments
- **Consistent Performance**: Most classes achieved F1-scores above 0.85, indicating robust classification across categories

7.4 Model Robustness

The consistent performance across different urban acoustic scenes demonstrates the model's ability to:

- Generalize well to unseen acoustic environments
- Handle acoustic variations within each category
- Distinguish between acoustically similar environments (e.g., different types of streets)

8 Technical Implementation Details

8.1 Data Loading and Processing

We implemented a custom PyTorch Dataset class that efficiently handles the Hugging Face dataset format, incorporating all preprocessing and augmentation steps in the data loading pipeline.

8.2 Training Infrastructure

- **Framework**: PyTorch with CUDA acceleration
- **Data Loading**: Multi-threaded data loading with 4 workers and memory pinning

- **Mixed Precision**: Efficient memory usage and faster training
- **Checkpointing**: Automatic saving of best model based on validation performance

8.3 Evaluation Framework

Our comprehensive evaluation includes:

- Multi-metric assessment (F1, precision, recall)
- Detailed confusion matrix analysis
- Per-class performance visualization
- Model checkpoint saving and loading

9 Challenges and Solutions

9.1 Technical Challenges

1. **Class Imbalance**: Addressed through stratified sampling and balanced loss functions
2. **Acoustic Similarity**: Some urban scenes share similar acoustic characteristics, resolved through advanced augmentation
3. **Computational Efficiency**: Optimized preprocessing pipeline and model architecture for efficient training
4. **Overfitting**: Mitigated through multiple regularization techniques and data augmentation

9.2 Solutions Implemented

- Label smoothing to prevent overconfident predictions
- Mixup augmentation for better generalization
- Early stopping and learning rate scheduling
- Comprehensive data augmentation pipeline

10 Future Work and Improvements

10.1 Model Enhancements

- **Architecture Exploration**: Investigation of other CNN architectures (EfficientNet, Vision Transformers)
- **Attention Mechanisms**: Adding attention layers to focus on discriminative acoustic features

10.2 Data and Features

- **Extended Datasets**: Incorporation of additional urban acoustic datasets
- **Multi-Modal Learning**: Combining audio with other sensing modalities

10.3 Deployment Considerations

- **Model Optimization**: Quantization and pruning for edge deployment
- **Real-Time Processing**: Optimization for streaming audio classification
- **Mobile Deployment**: Adaptation for mobile and embedded systems

11 Conclusion

This project successfully demonstrates the effectiveness of deep learning approaches for multiclass urban sound classification. Our modified ResNet18 architecture, combined with comprehensive data augmentation strategies and modern training techniques, achieved excellent performance with a macro F1-score of 0.8631.

11.1 Key Achievements

- Successfully adapted ResNet18 for single-channel spectrogram processing
- Implemented state-of-the-art augmentation techniques including SpecAugment and Mixup
- Achieved balanced performance across all 10 urban acoustic scene categories
- Developed a robust and comprehensive evaluation framework
- Created an end-to-end pipeline from raw audio to classification results

11.2 Impact and Significance

The results demonstrate the viability of automated urban sound scene recognition for real-world applications. The high accuracy and balanced performance across different acoustic environments make this approach suitable for deployment in smart city systems, environmental monitoring, and accessibility technologies.

11.3 Technical Contributions

This work contributes to the audio classification field through:

- Comprehensive implementation of modern deep learning techniques for audio
- Detailed analysis of urban acoustic scene classification challenges
- Open-source implementation facilitating reproducibility and further research
- Practical demonstration of deep learning effectiveness in acoustic scene analysis

The success of this project paves the way for more sophisticated acoustic monitoring systems and contributes to the broader goal of creating smarter, more responsive urban environments through AI-powered audio analysis.

Acknowledgments

We thank the creators of the TUT Urban Acoustic Scenes 2018 dataset and the Hugging Face community for making the dataset accessible. We also acknowledge the open-source PyTorch and torchaudio communities for providing the foundational tools that made this work possible.

References

[1] Mesaros, A., Heittola, T., & Virtanen, T. (2018). TUT urban acoustic scenes 2018 development dataset. Zenodo.