AKADEMIA GÓRNICZO-HUTNICZA
IM. STANISŁAWA STASZICA W KRAKOWIE

AGH UNIVERSITY OF SCIENCE
AND TECHNOLOGY

Music Genres Classification

Computational Intelligence, Kacper Motyka

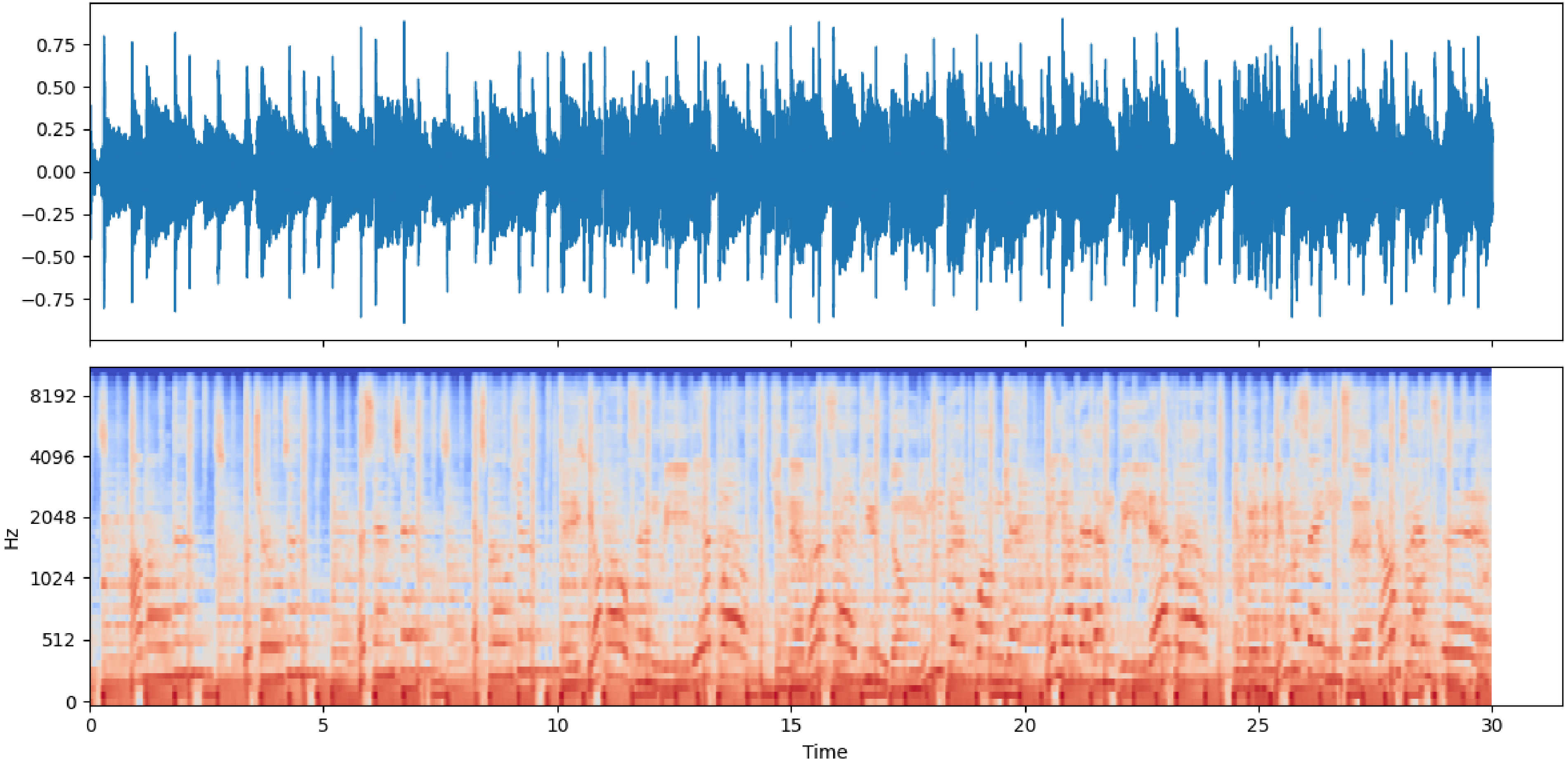
11.05.2023

About the dataset

- GTZAN Dataset: https://www.kaggle.com/datasets/andradaolteanu/gtzan-dataset-music-genre-classification
- Collection of 10 genres of music – 100 audio files each
- 30 seconds long
- 2 *.csv files containing features of the audio files
- MEL Spectrograms of the audio

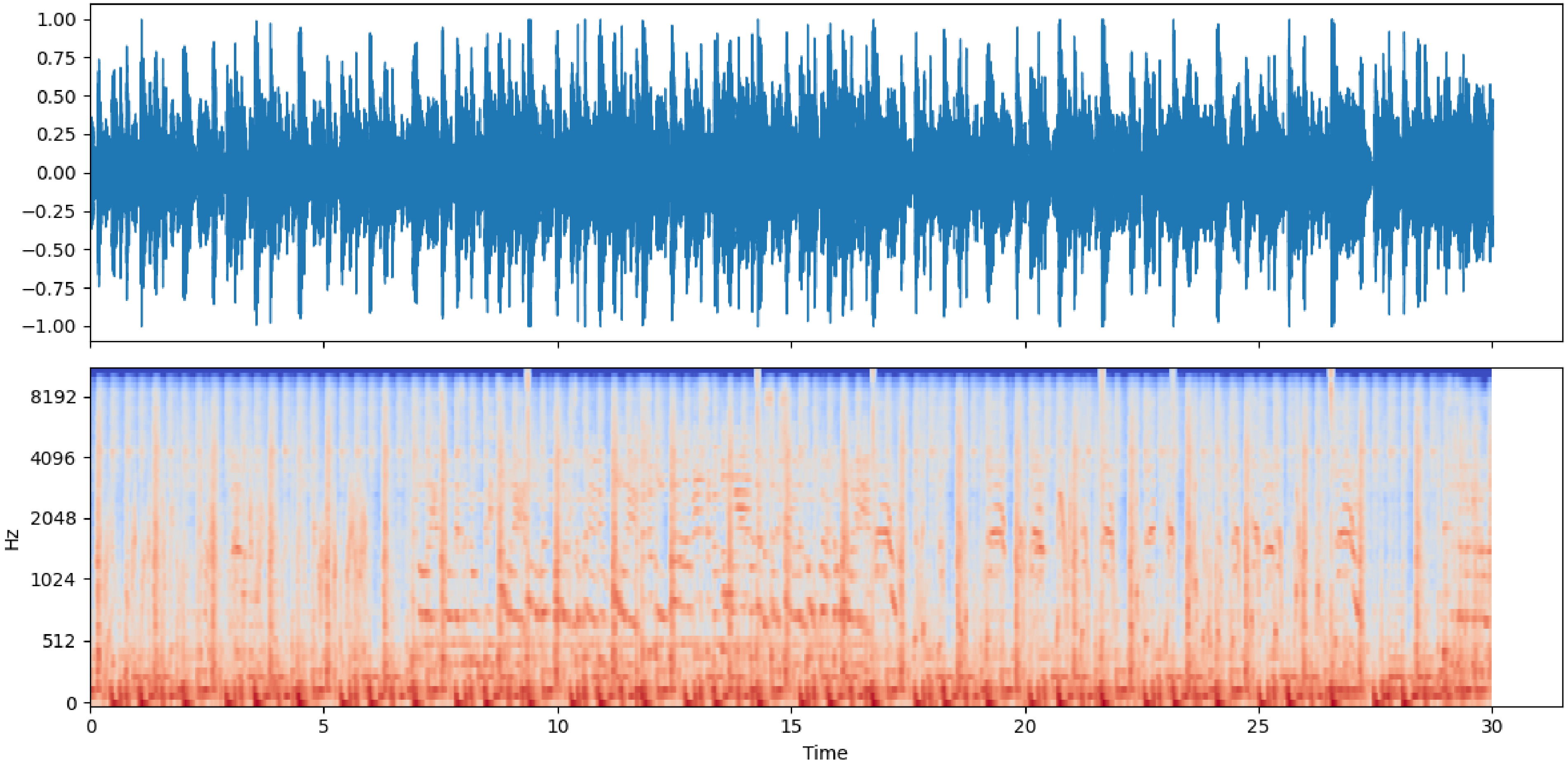
Sample input - **rock**

- Sound Wave
- Calculated spectrogram



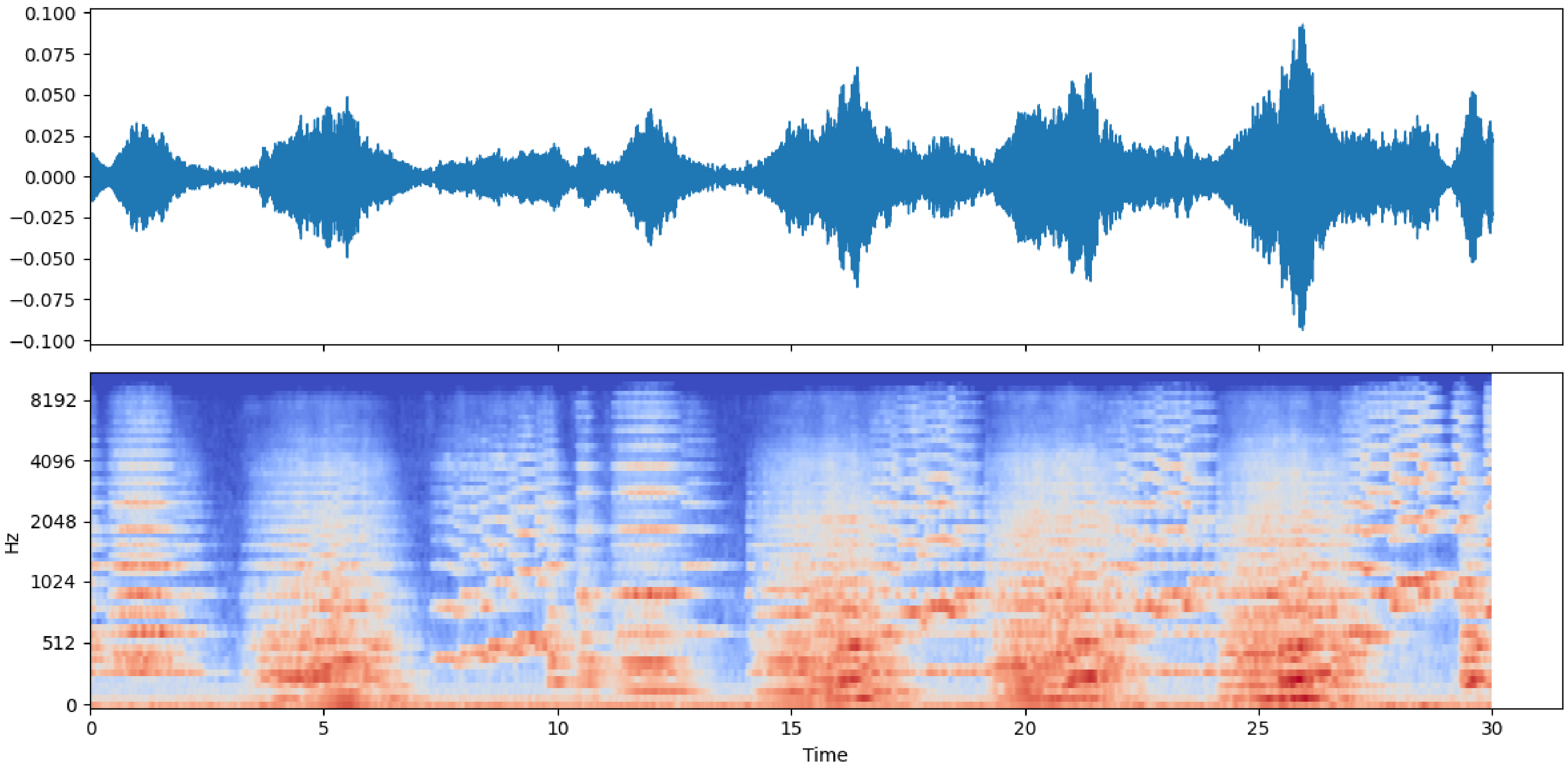
Sample input – **hip-hop**

- Sound Wave
- Calculated spectrogram



Sample input – **classical**

- Sound Wave
- Calculated spectrogram



Reasons of calculating MEL spectrograms

- Dimensionality reduction: [1, 661 500] -> [64, 1024] - **10 times less**
- More suitable representation of audio signals for human perception because of **frequency representation**
- Easier **pattern extraction**: from easy edges and corners to combining complex patterns in audio
- **Robustness to noise** and variations
- Ability to work with **sound as with images**

Tools used

Main tools

jupyter

Calculations

NumPy

pandas

Visualization

librosa

matplotlib

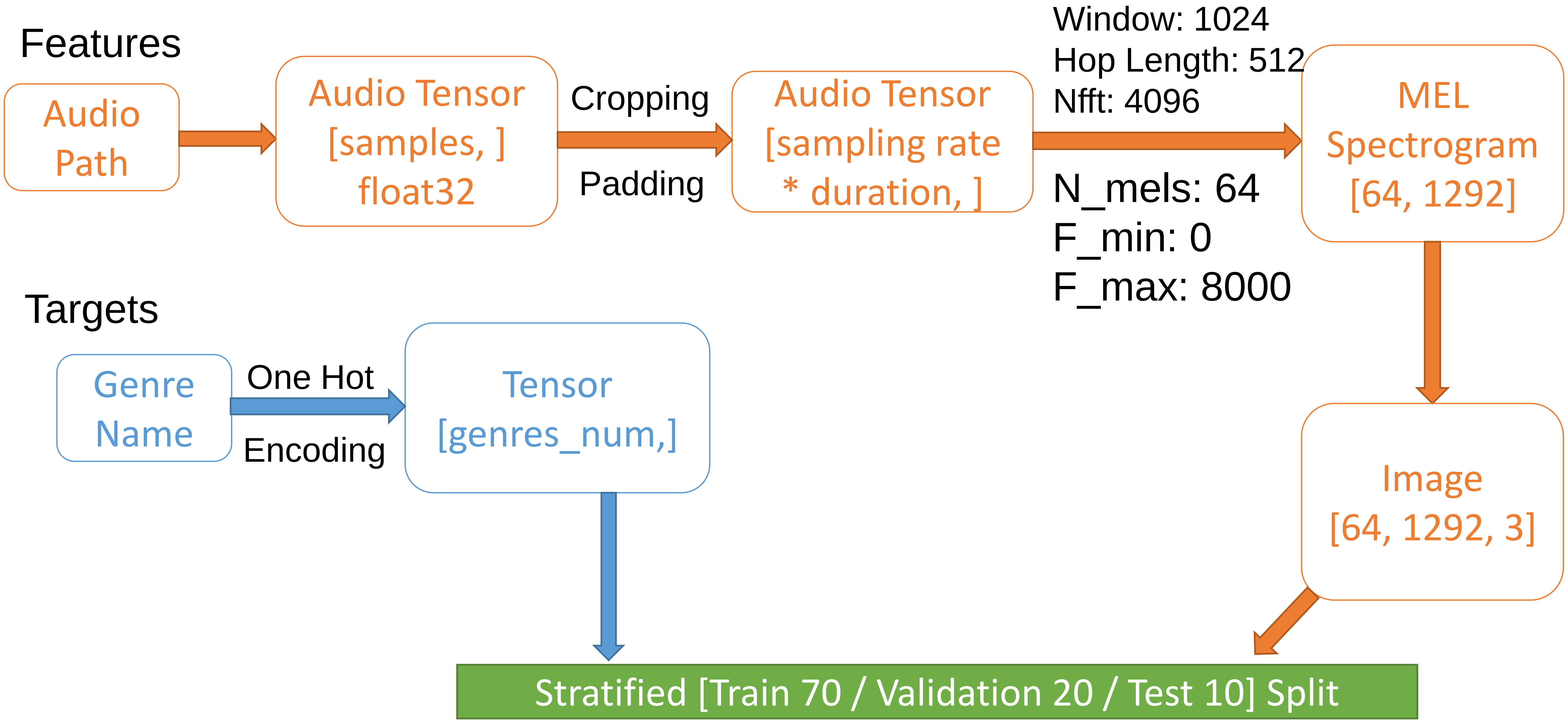
seaborn

Machine Learning

TensorFlow

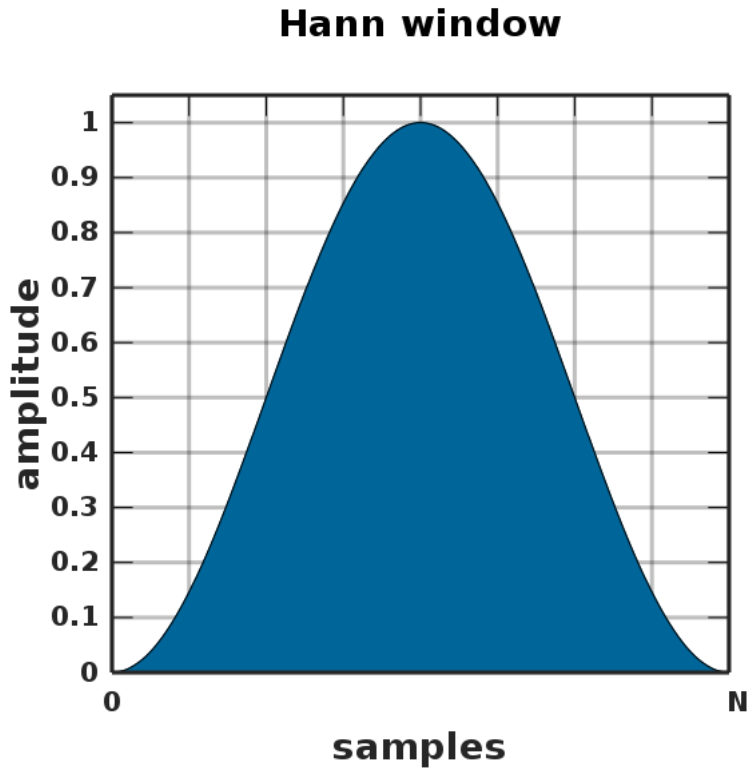
scikit learn

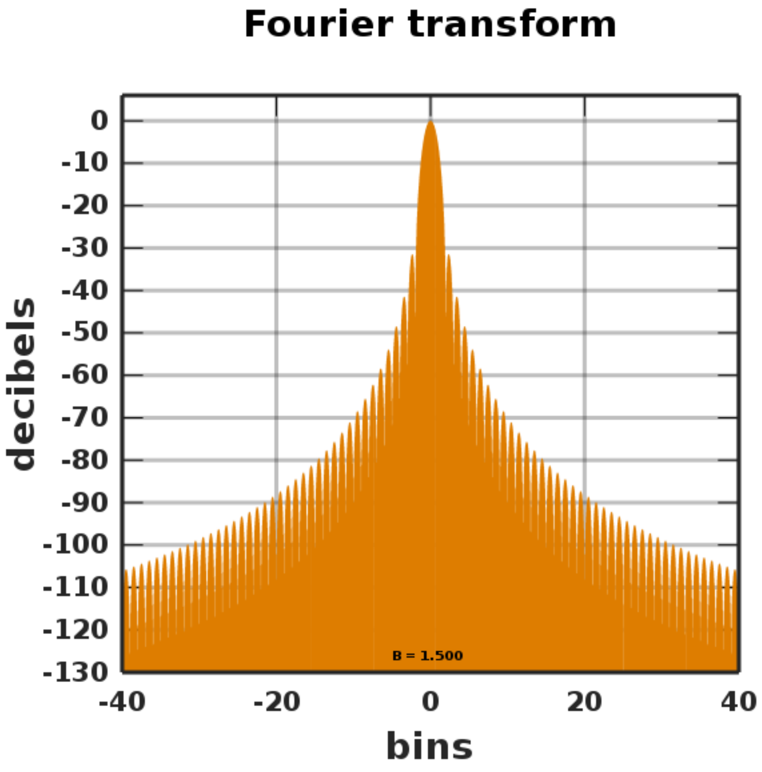
Data Processing



MEL Spectrograms – Parameters Meaning

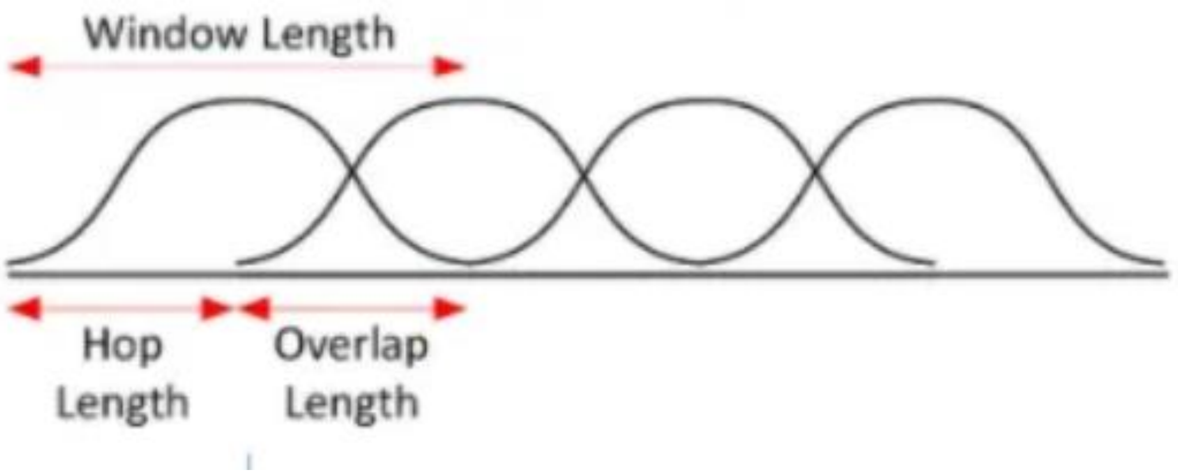
- **Window**: Length of the window function, default type: Hanning
- Longer window provides better frequency resolution but sacrifices temporal resolution





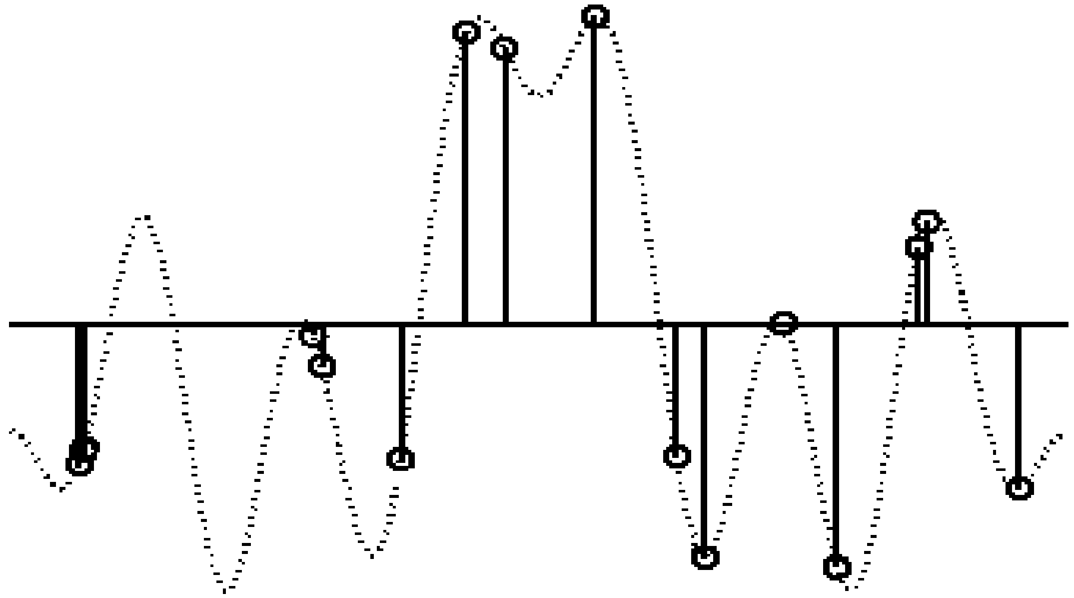
MEL Spectrograms – Parameters Meaning

- **Hop length**: the length of the non-intersecting portion of window length.
- A smaller hop length results in a higher temporal resolution because it means more frequent updates of the spectrogram



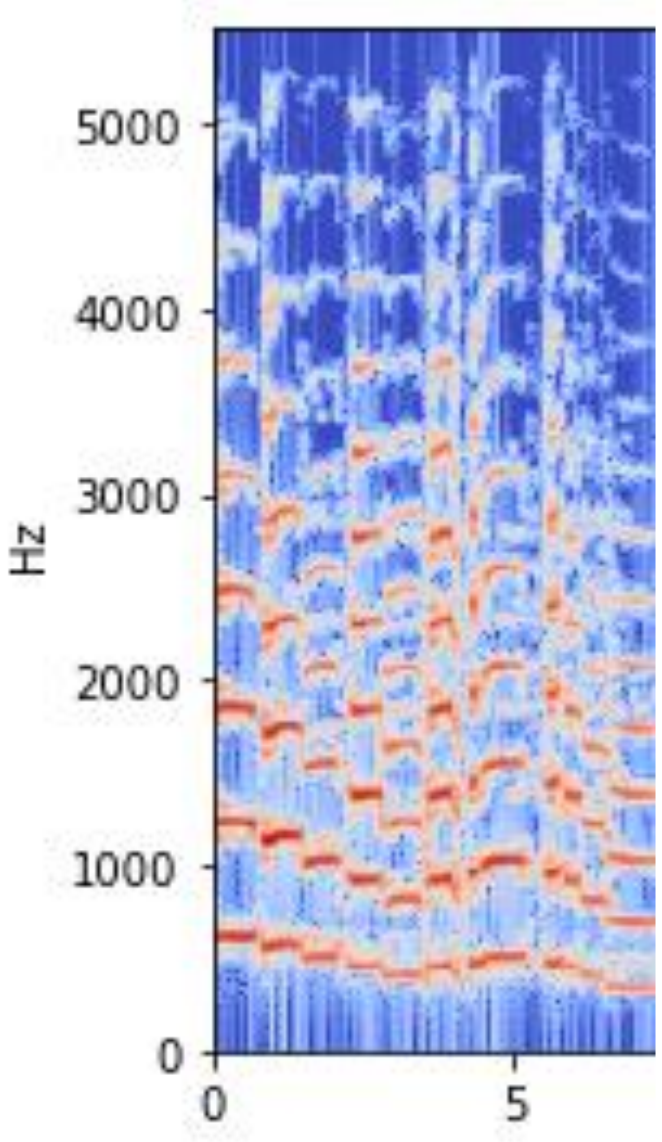
MEL Spectrograms – Parameters Meaning

- **NFFT** (Number of Fast Fourier Transform points)
- In spectrogram analysis, the audio signal is divided into small frames, and the Fourier Transform is applied to each frame to obtain its frequency content. The NFFT parameter specifies the number of points used in the Fast Fourier Transform (FFT) computation for each frame.

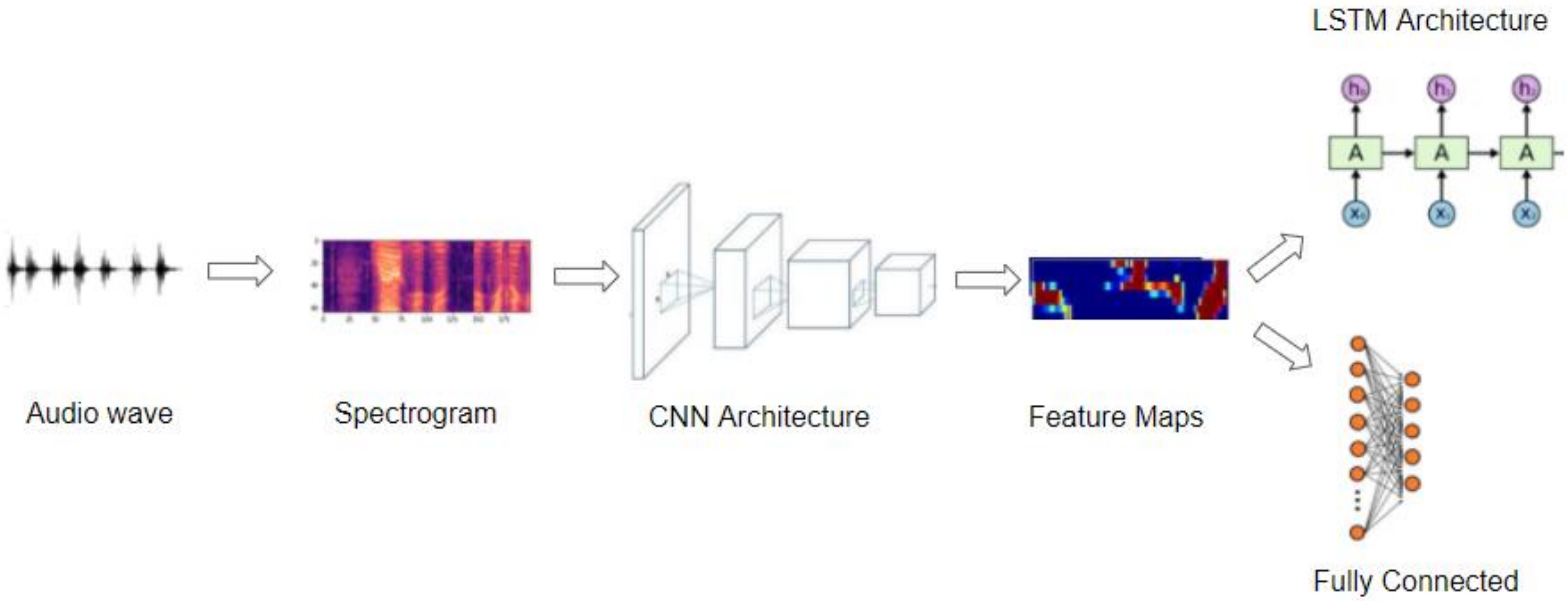


MEL Spectrograms – Parameters Meaning

- **N_Mels, f_min_ f_max**: parameters controlling the frequency (y) axis of MEL Spectograms setting minimum / maximum frequency and number of bins in this range



General approach



First Results – huge overfitting

```
Epoch 1/50
21/21 [==============================] - 7s 217ms/step - loss: 11.2553 - accuracy: 0.1769 - val_loss: 7.6273 - val_accuracy: 0.2292
Epoch 2/50
21/21 [==============================] - 4s 175ms/step - loss: 3.3925 - accuracy: 0.3298 - val_loss: 6.2294 - val_accuracy: 0.2500
Epoch 3/50
21/21 [==============================] - 4s 184ms/step - loss: 2.9664 - accuracy: 0.3898 - val_loss: 4.6011 - val_accuracy: 0.1042
Epoch 4/50
21/21 [==============================] - 4s 177ms/step - loss: 1.9550 - accuracy: 0.4558 - val_loss: 1.9168 - val_accuracy: 0.4219
Epoch 5/50
21/21 [==============================] - 4s 174ms/step - loss: 1.5035 - accuracy: 0.5637 - val_loss: 2.0455 - val_accuracy: 0.3594
Epoch 6/50
21/21 [==============================] - 4s 196ms/step - loss: 1.2772 - accuracy: 0.6492 - val_loss: 1.9323 - val_accuracy: 0.4167
Epoch 7/50
21/21 [==============================] - 4s 173ms/step - loss: 0.8874 - accuracy: 0.7526 - val_loss: 2.8685 - val_accuracy: 0.3802
Epoch 8/50
21/21 [==============================] - 4s 175ms/step - loss: 0.6582 - accuracy: 0.8066 - val_loss: 3.1151 - val_accuracy: 0.4010
Epoch 9/50
21/21 [==============================] - 4s 197ms/step - loss: 0.4924 - accuracy: 0.8606 - val_loss: 2.9091 - val_accuracy: 0.3802
Epoch 10/50
21/21 [==============================] - 4s 178ms/step - loss: 0.3715 - accuracy: 0.8981 - val_loss: 5.0246 - val_accuracy: 0.2969
Epoch 11/50
21/21 [==============================] - 4s 176ms/step - loss: 0.2945 - accuracy: 0.9340 - val_loss: 3.6722 - val_accuracy: 0.3177
Epoch 12/50
21/21 [==============================] - 4s 181ms/step - loss: 0.1988 - accuracy: 0.9460 - val_loss: 4.9887 - val_accuracy: 0.3281
4/4 [==============================] - 0s 16ms/step - loss: 2.0992 - accuracy: 0.3030
Test accuracy: 0.3030303120613098
```

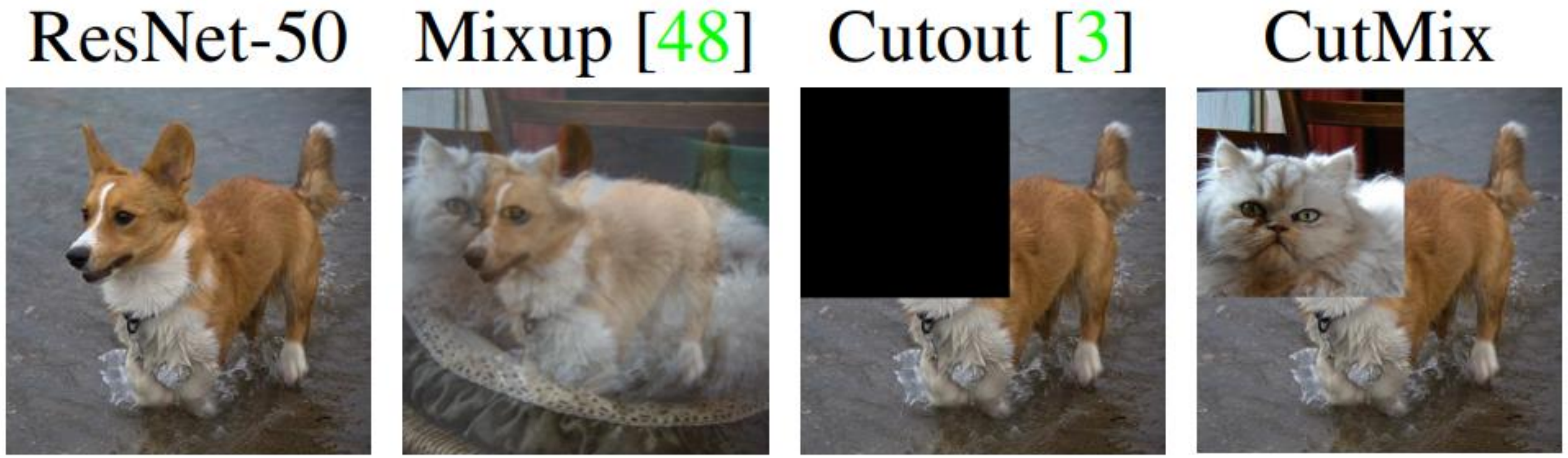
Methods used:

- Dropout
- Batch Normalization
- Different Model Complexities
- Simple **spectrogram-based** augmentation (frequency / time masking)

New Data Augmentation

Audio:

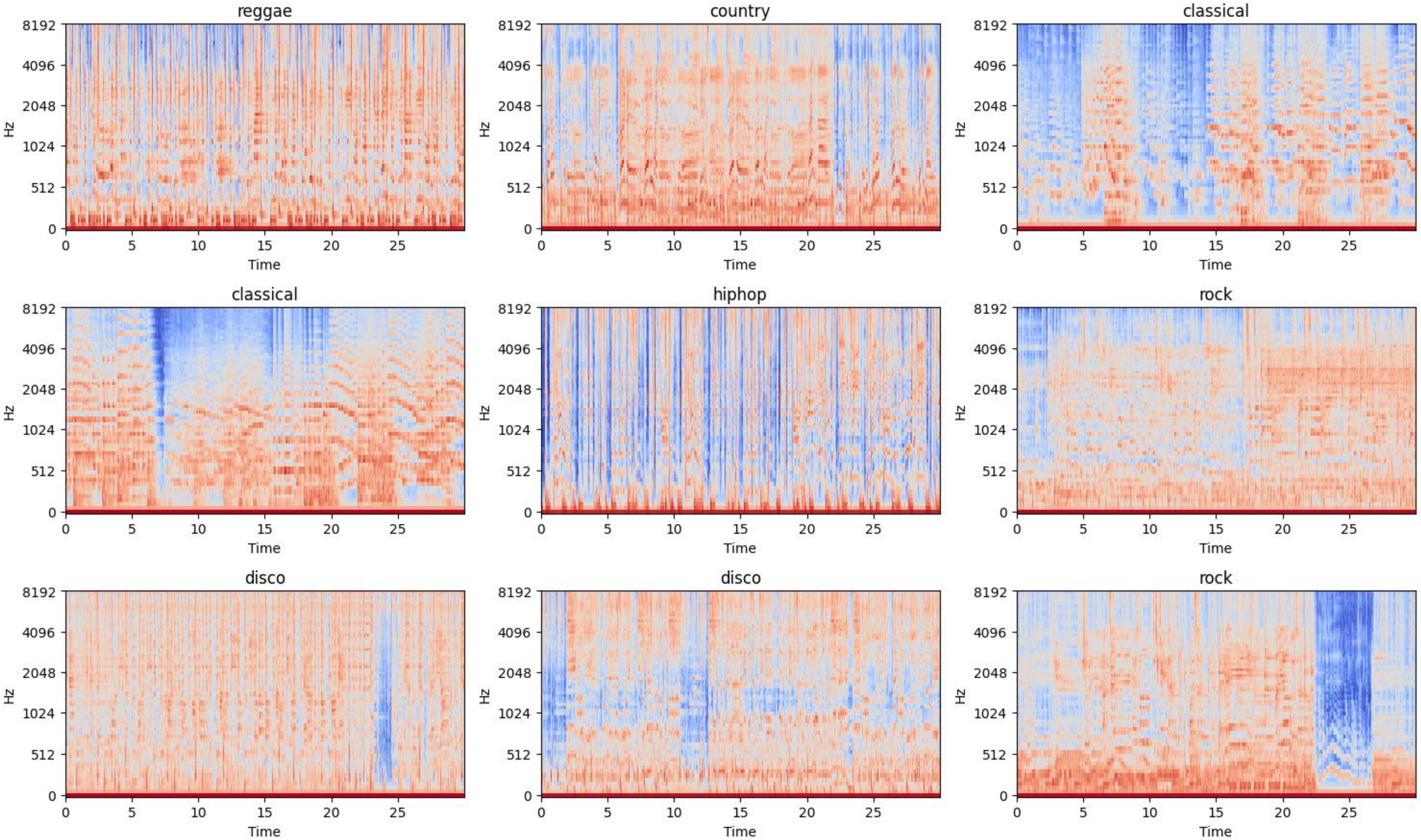
- Time Shifting
- Adding Gaussian Noise



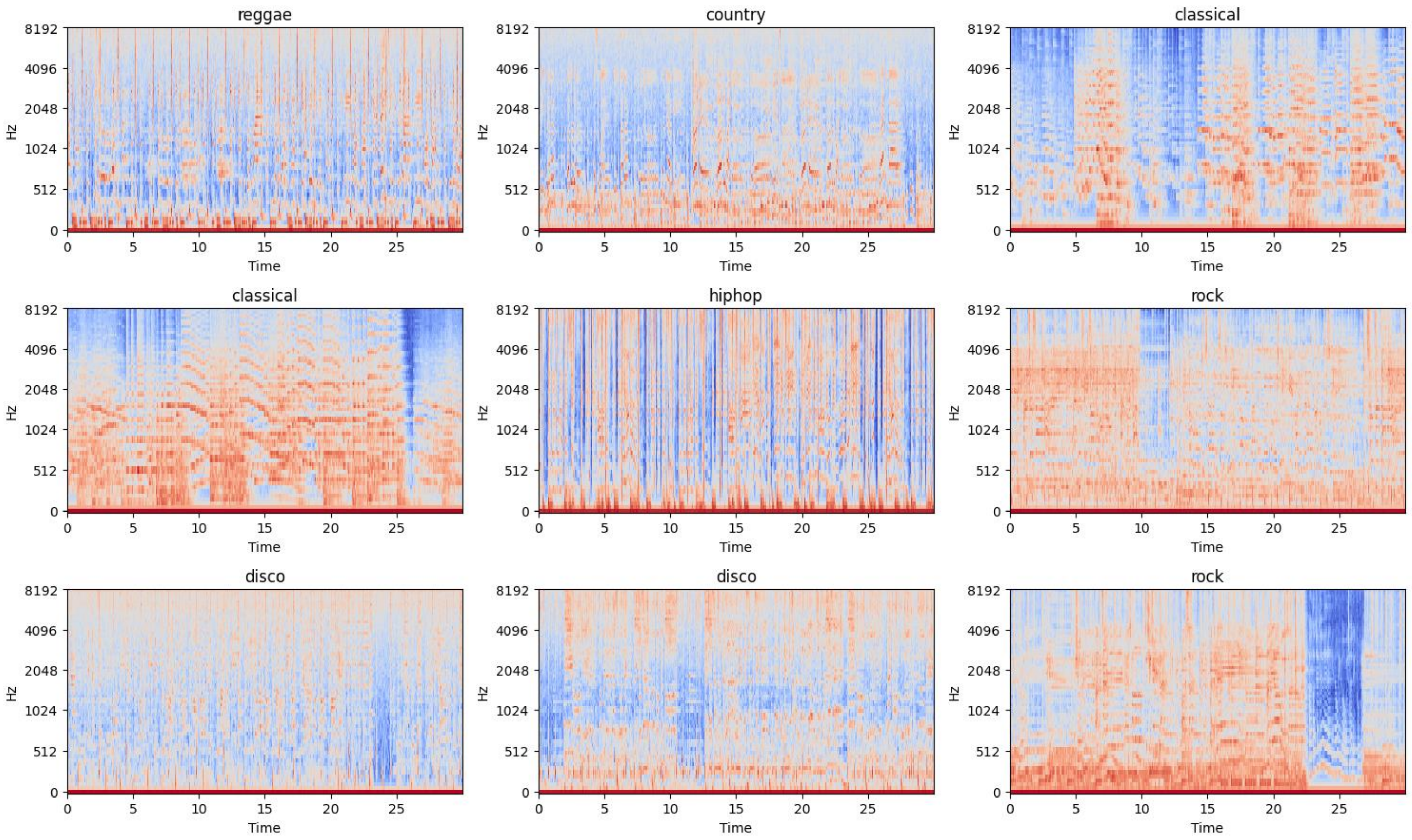
Spectrogram:

- Time Masking
- Frequency Masking
- Mix-Up: Take two images and their labels from the batch and create a new image and label as a mix of them
- Cut-Mix: Cut a part of the image and insert it into another image

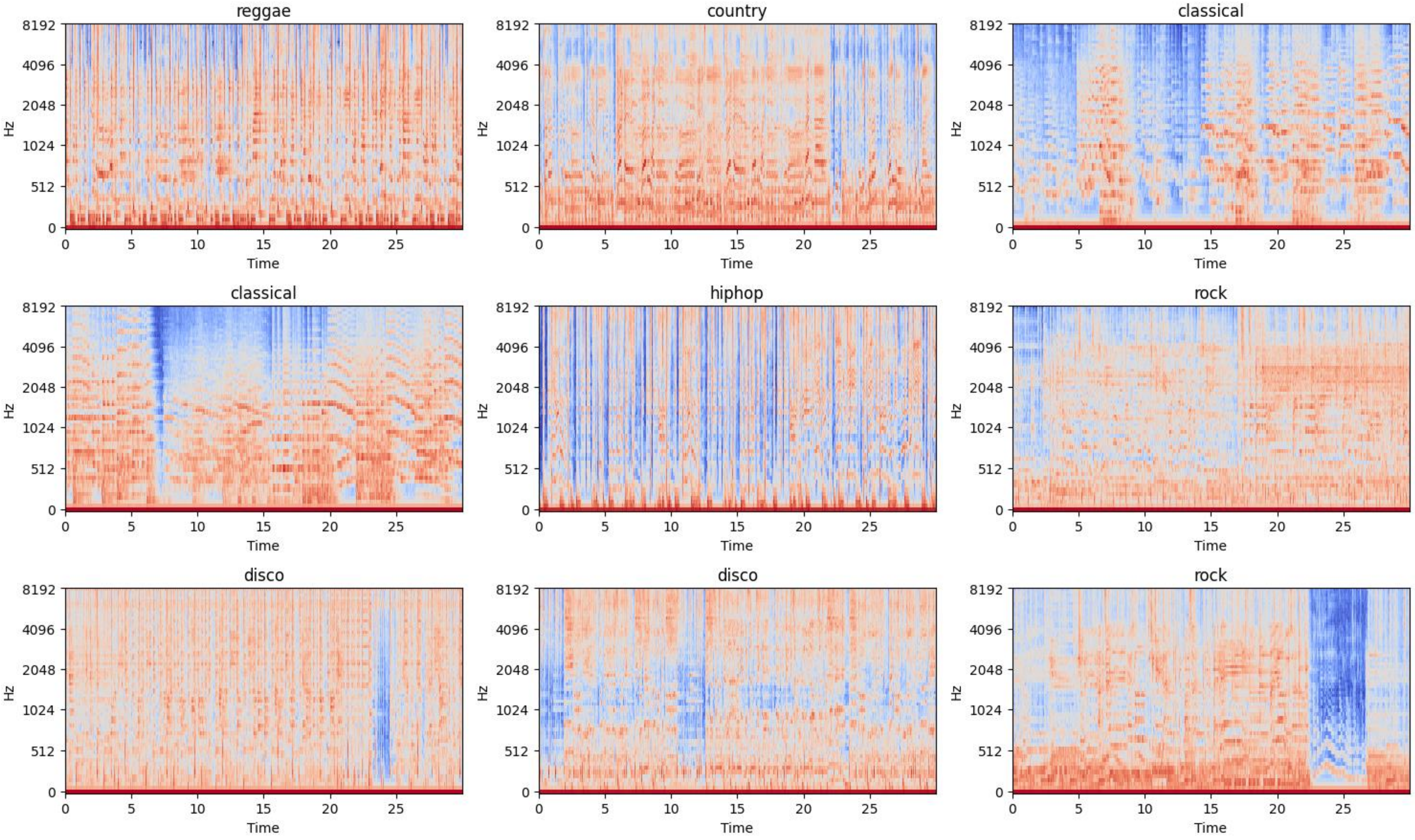
Sample data before augmentation



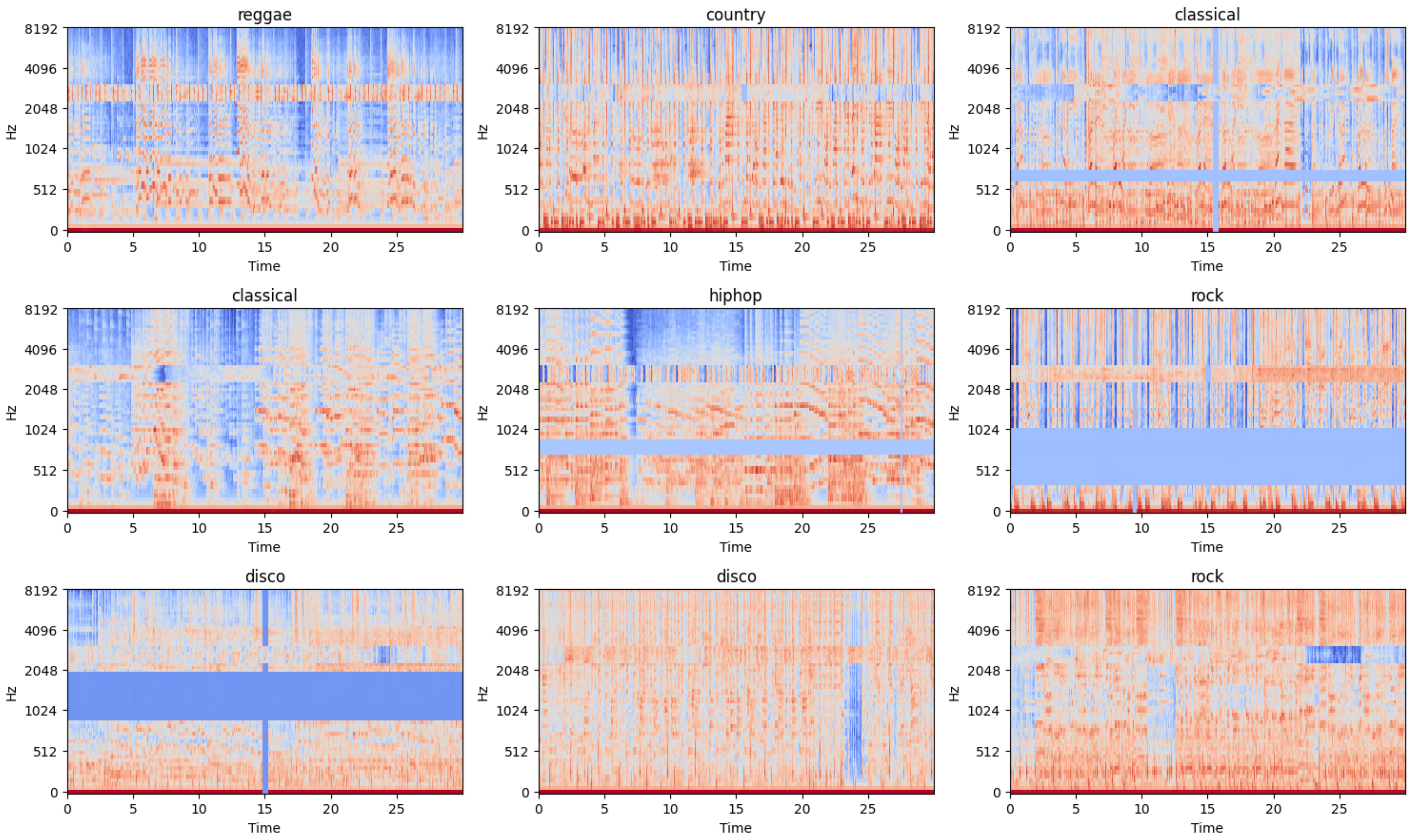
Samples after **audio** augmentation



Sample data before augmentation



Samples after **spec** augmentation

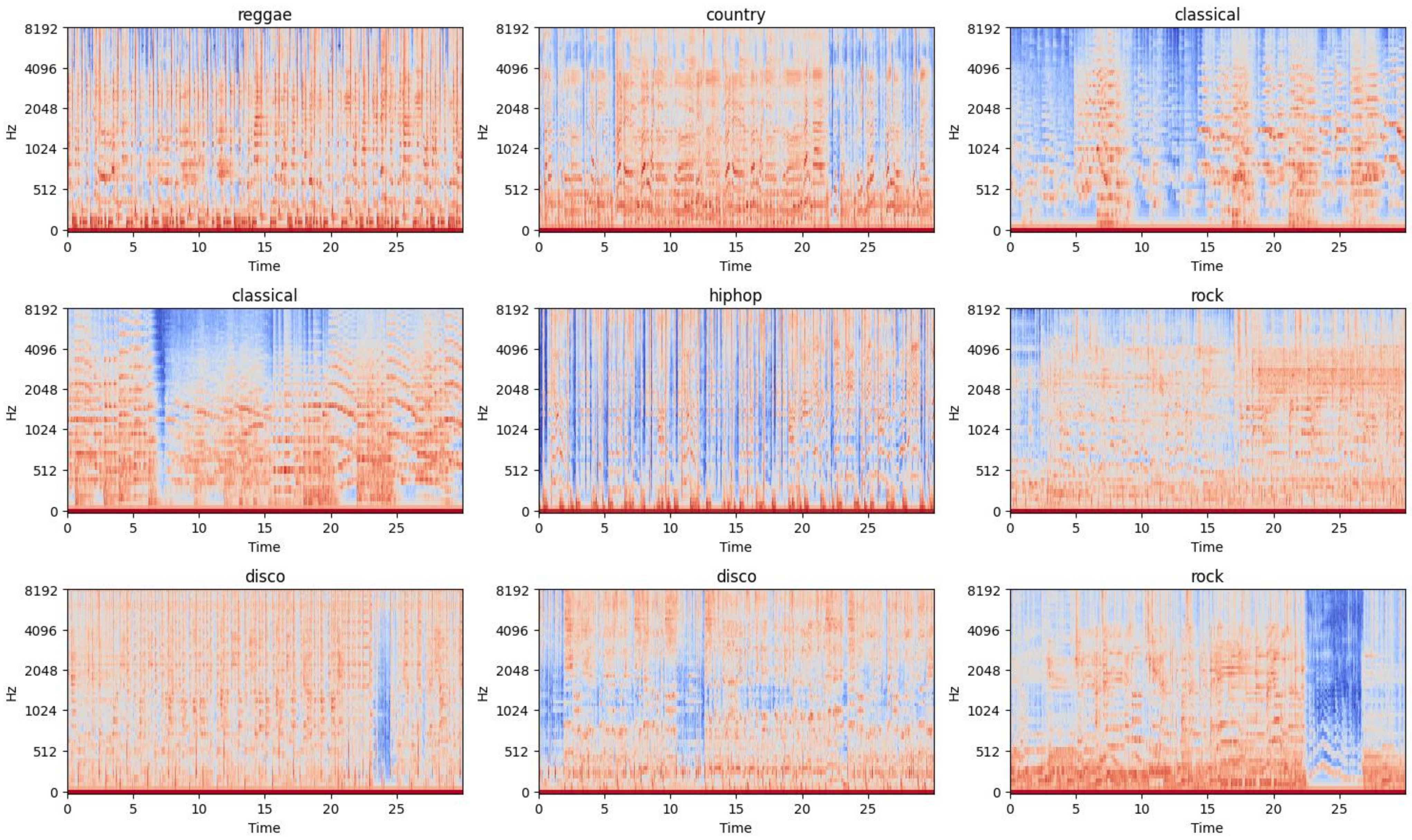


Augmentation Parameters

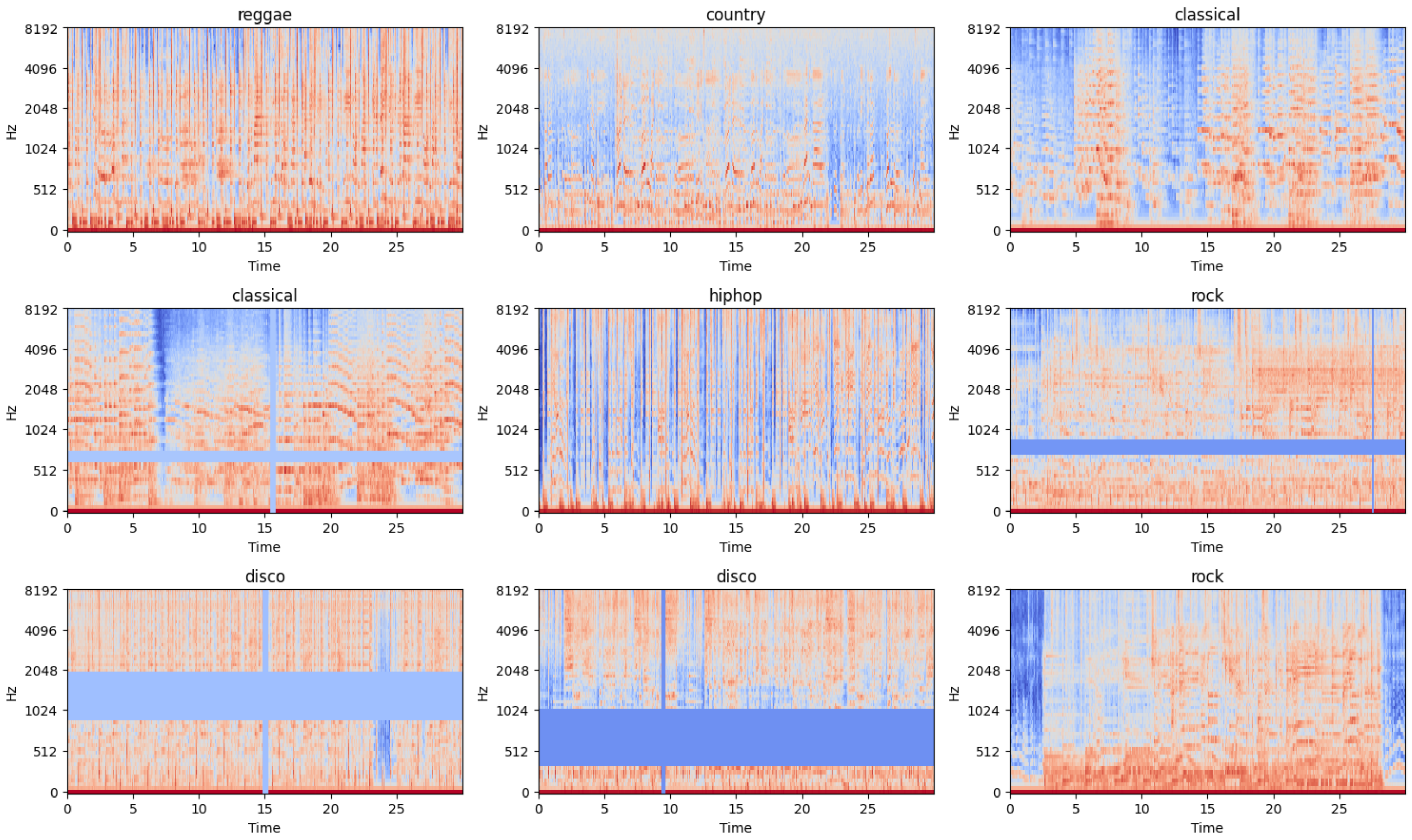
- Audio Augmentation Probability: 0.5
  - Time Shift Probability: 0.3
  - Gaussian Noise Probability: 0.35
- Spectrogram Augmentation Probability: 0.8
  - MixUp Probability: 0.65 with alpha: 0.5
  - CutMix Probability: 0.0 with alpha: 0.5
  - Masking Probability: 0.65 with:
    - Frequency mask max width: 20%
    - Time mask max width: 30%

```
# Spec augment
spec_augment_prob = 0.8

mixup_prob = 0.65
mixup_alpha = 0.5

cutmix_prob = 0.0
cutmix_alpha = 0.5

mask_prob = 0.65
freq_mask = 20
time_mask = 30


# Audio Augmentation Settings
audio_augment_prob = 0.5

timeshift_prob = 0.3

gn_prob = 0.35
```

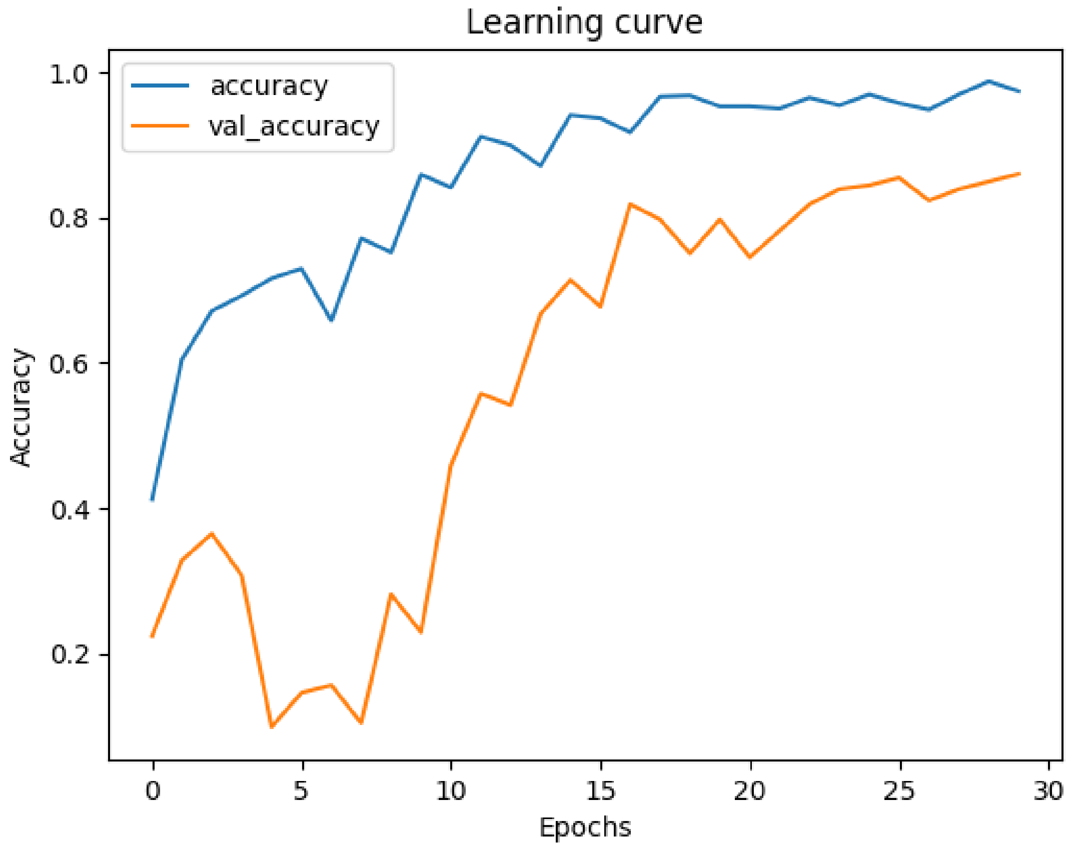
Sample data before augmentation

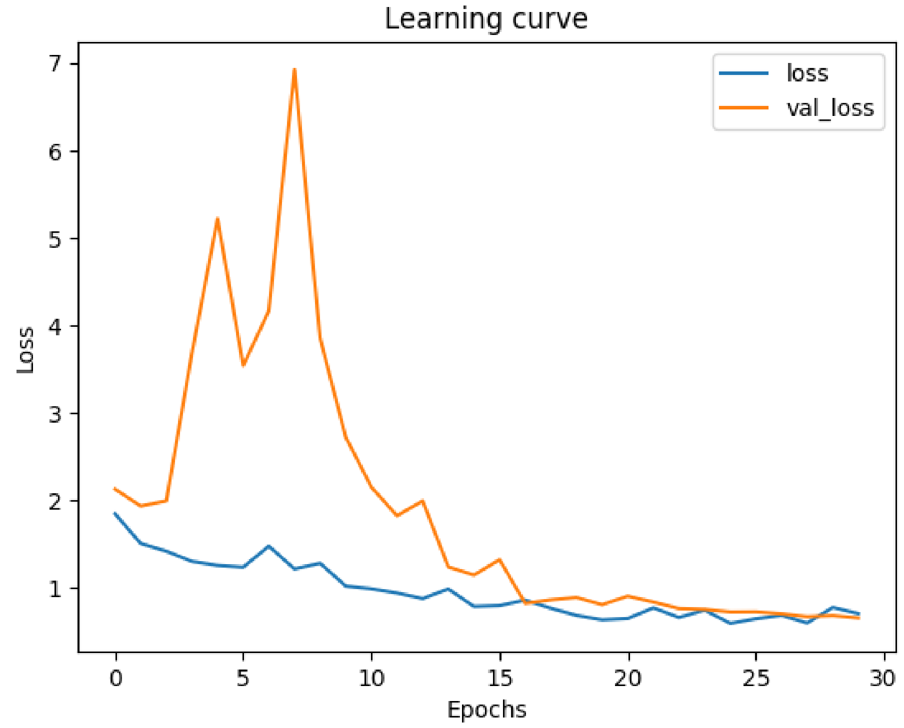


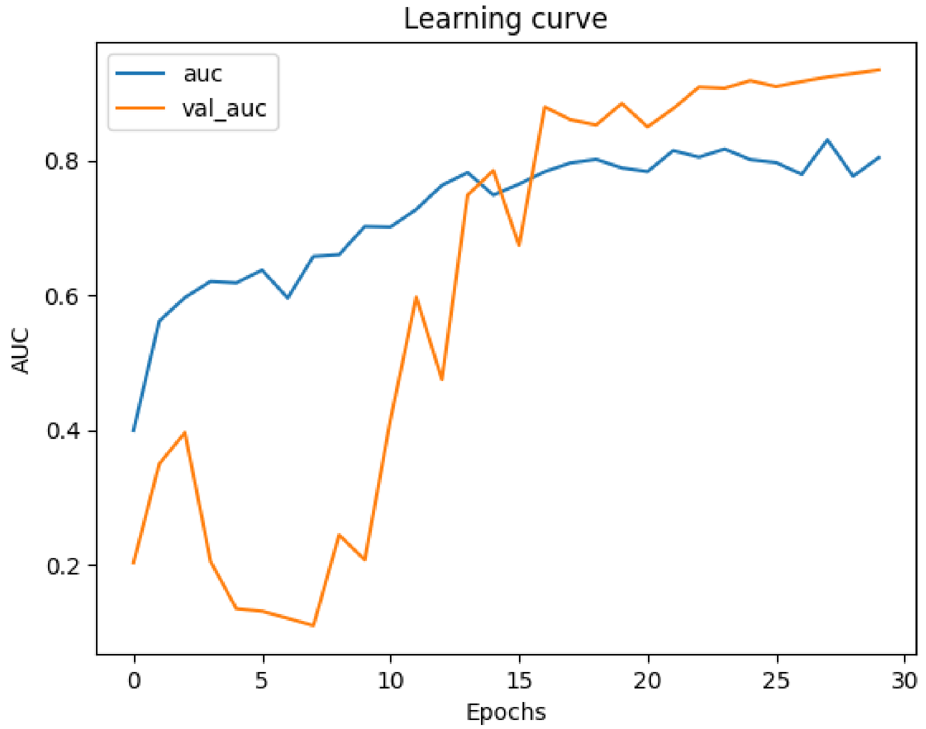
Samples after **chosen** augmentation



Efficient Net

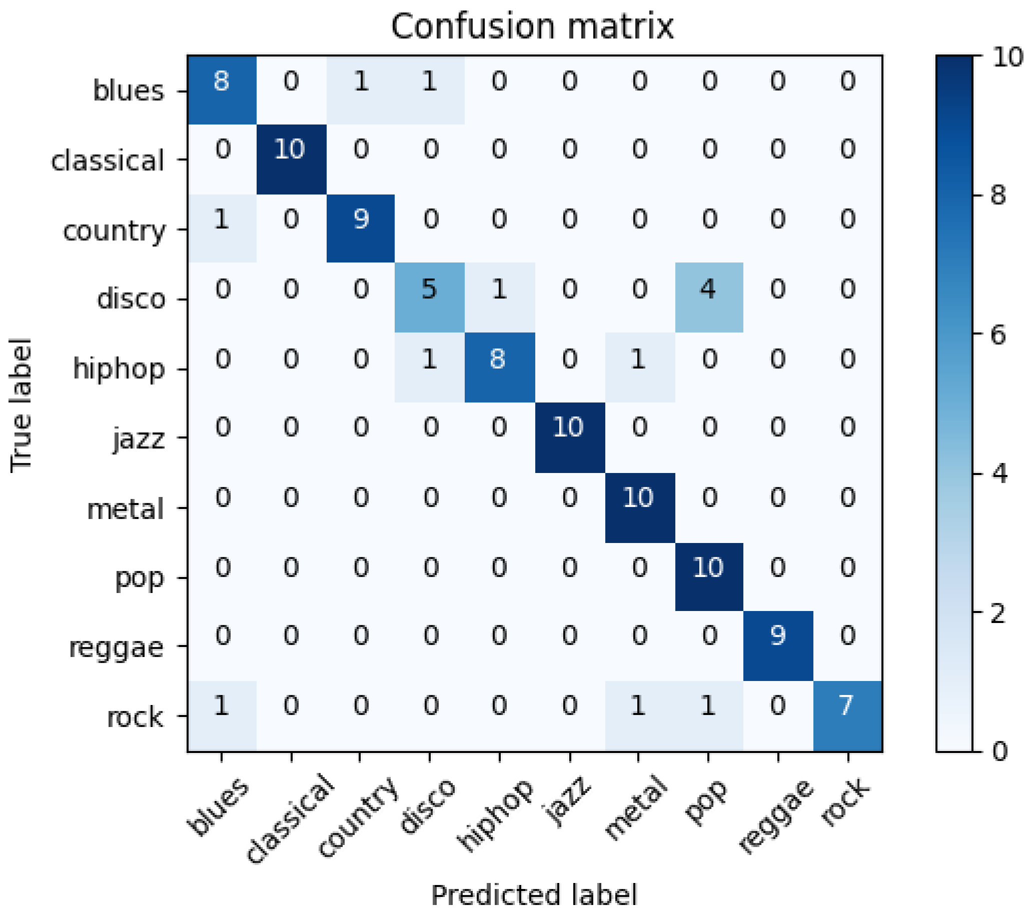




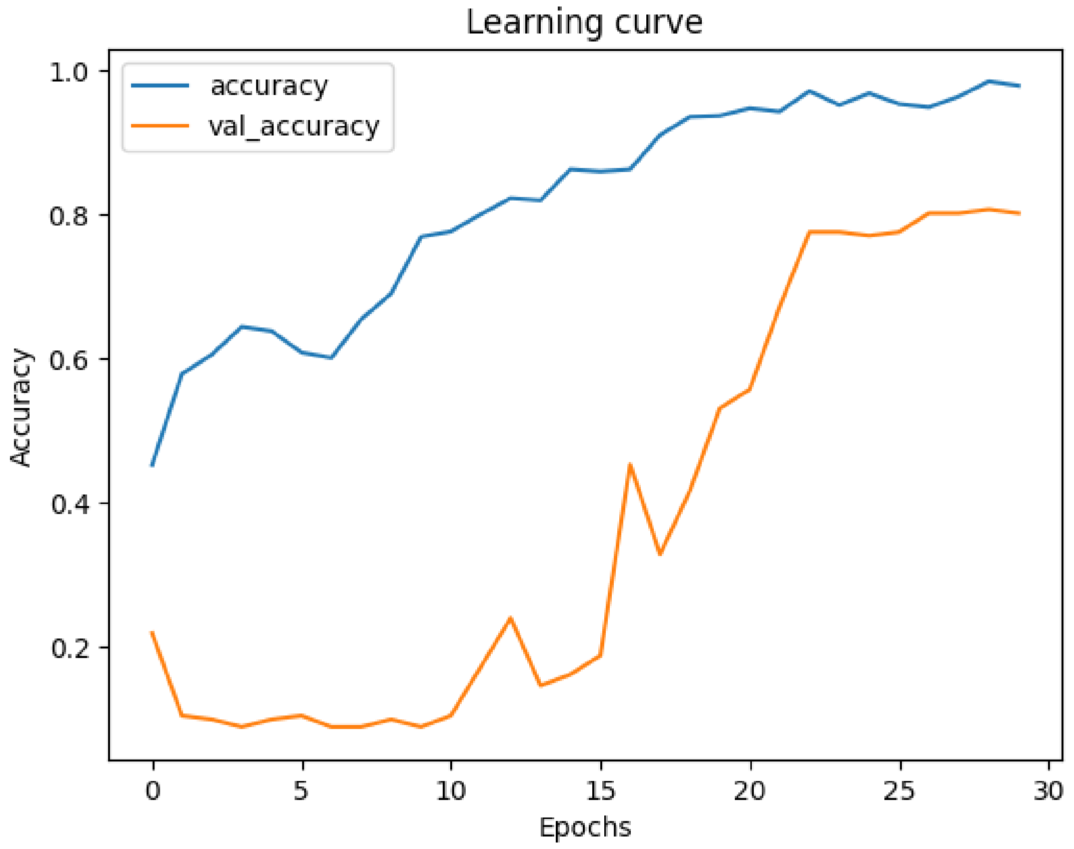


Efficient Net

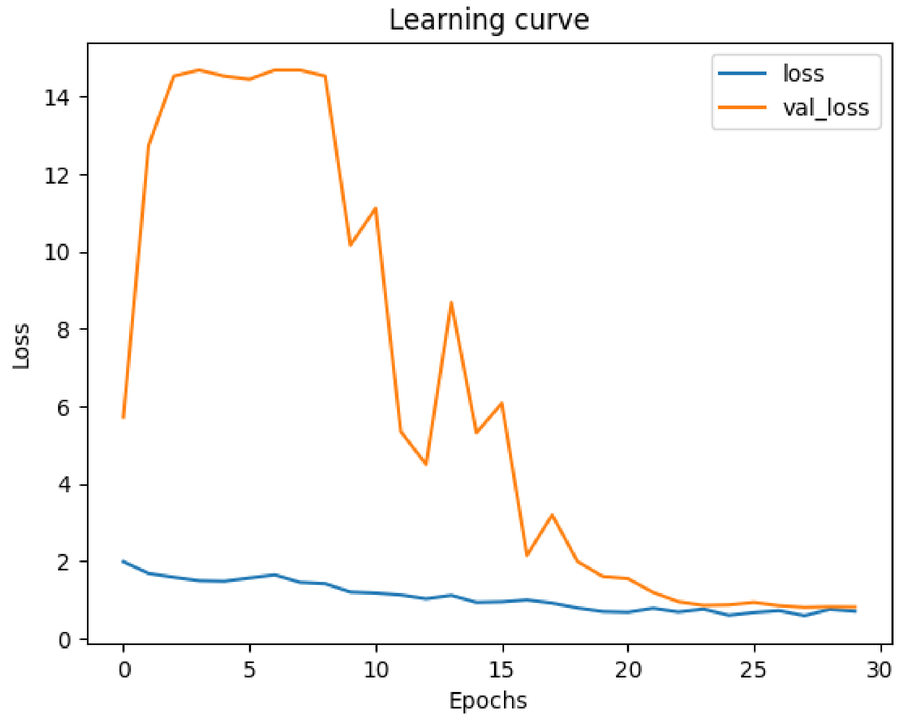
	Train	Validation	Test
Accuracy	97,3%	85,9%	86,9%
AUC	80,4%	93,4%	
Loss	0.70	0.65	



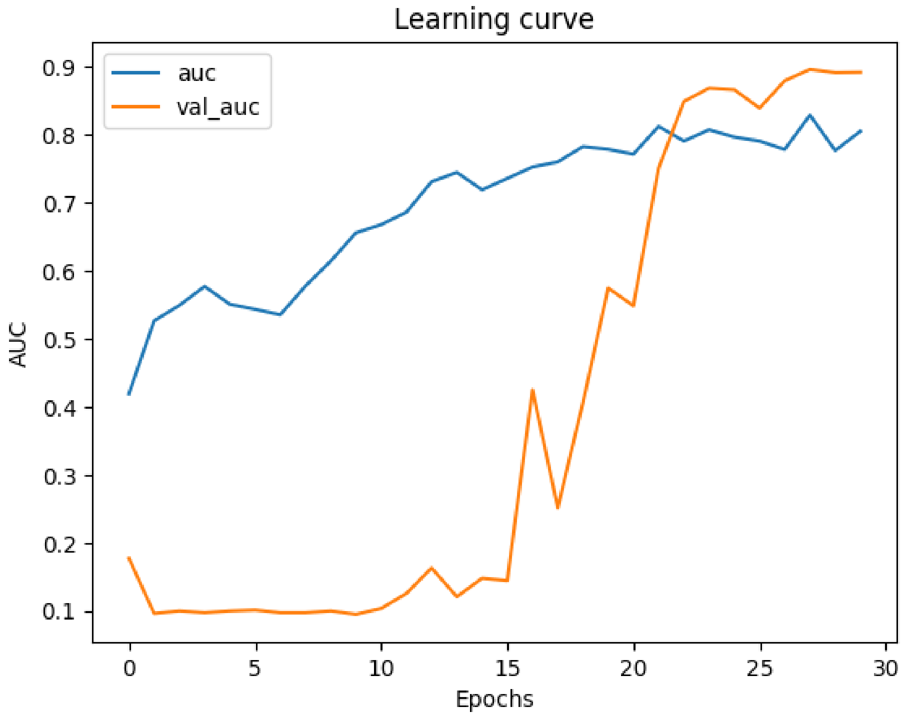
ResNet



Learning curve
accuracy
val_accuracy
Accuracy
Epochs
0
5
10
15
20
25
30
0.2
0.4
0.6
0.8
1.0



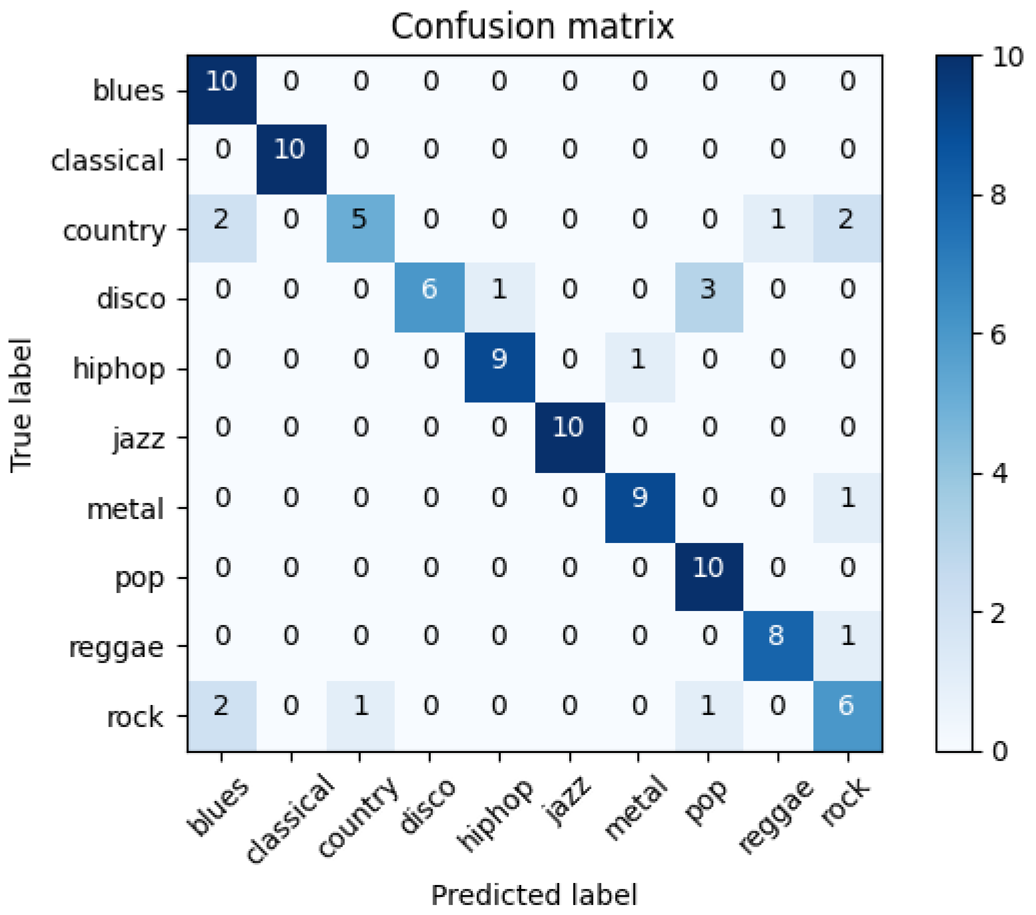
Learning curve
loss
val_loss
Loss
Epochs
0
5
10
15
20
25
30
0
2
4
6
8
10
12
14



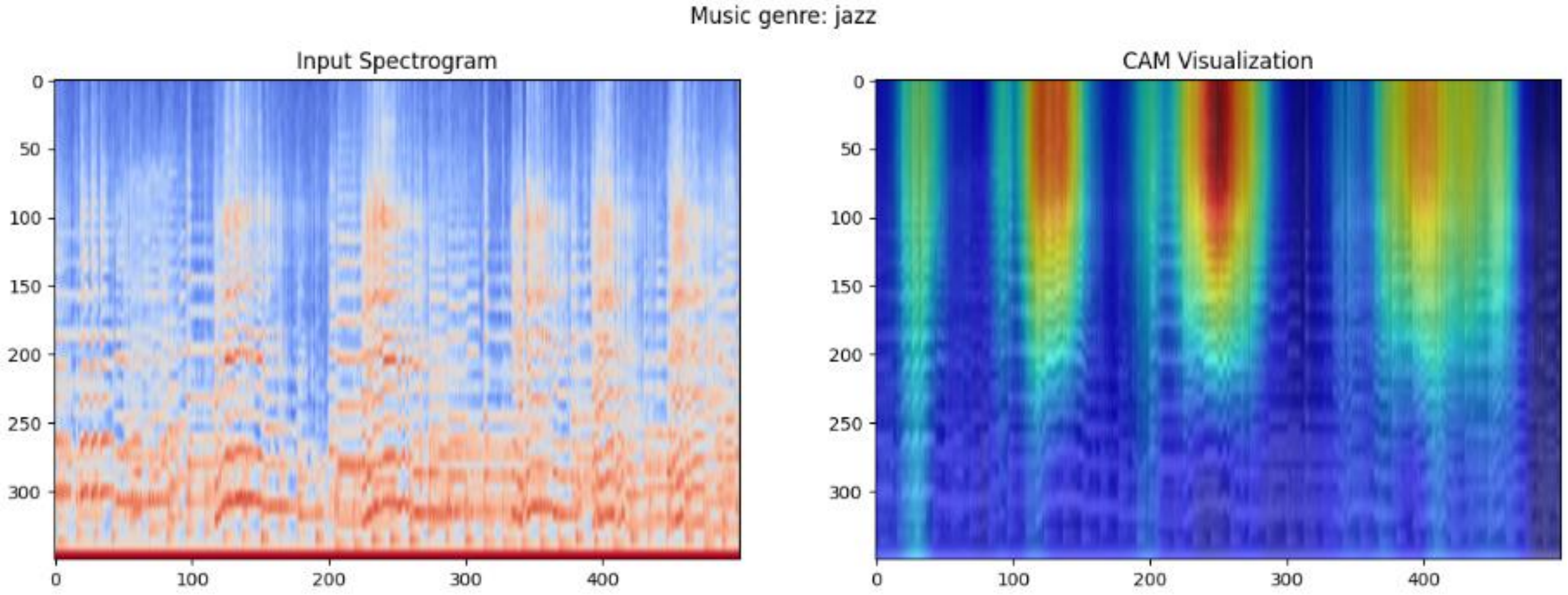
Learning curve
auc
val_auc
AUC
Epochs
0
5
10
15
20
25
30
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9

ResNet

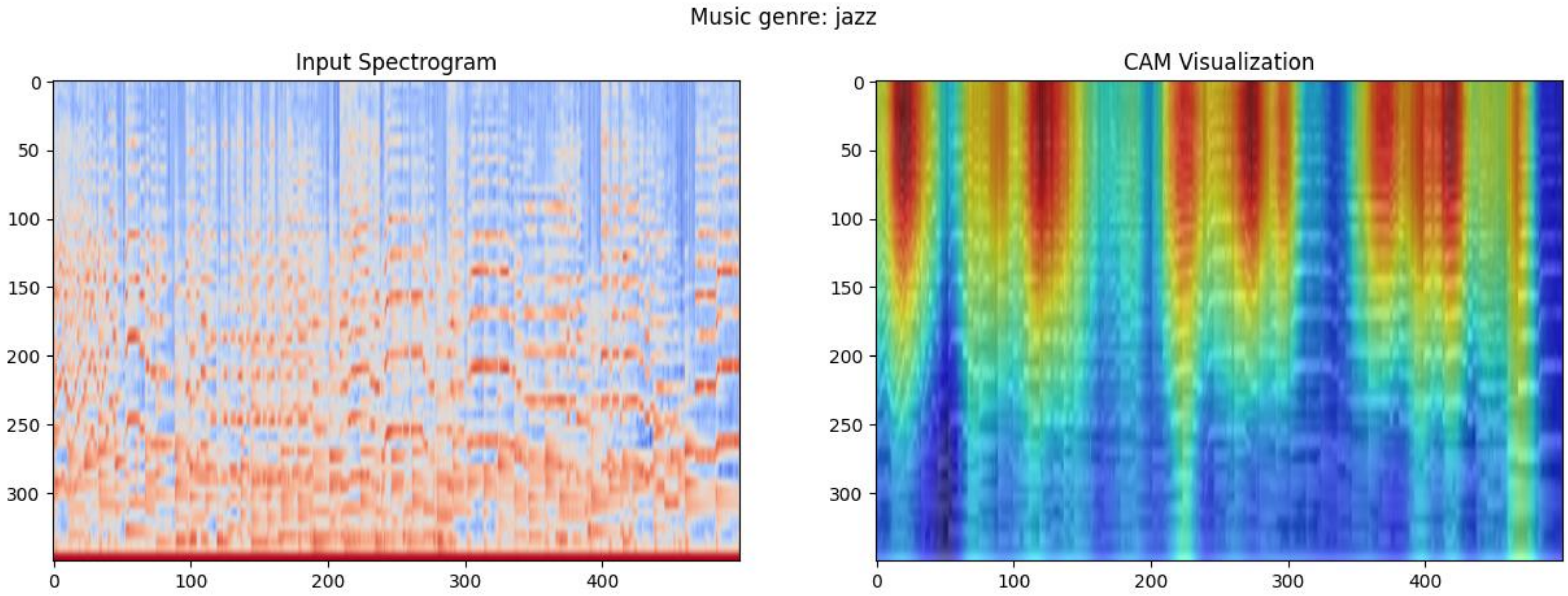
	Train	Validation	Test
Accuracy	98,5%	80,7%	83,8%
AUC	77,7%	89,1%	
Loss	0.76	0.83	



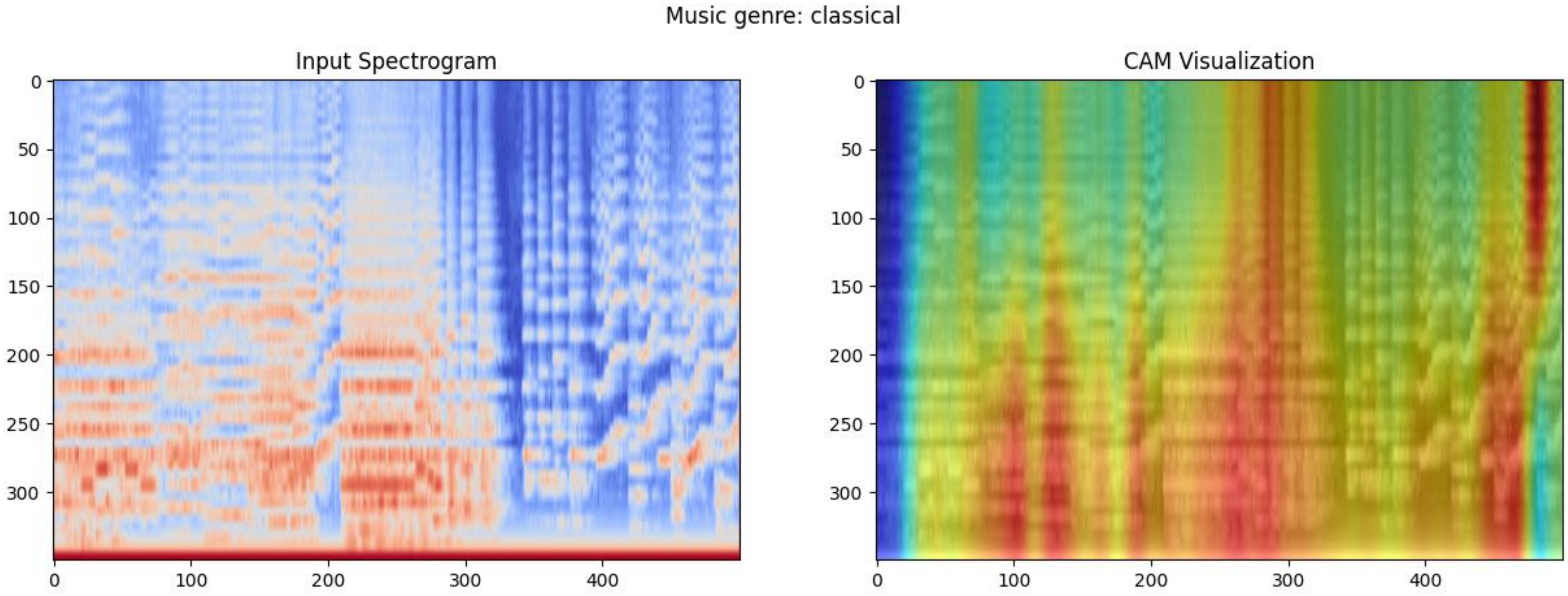
CAM Visualization



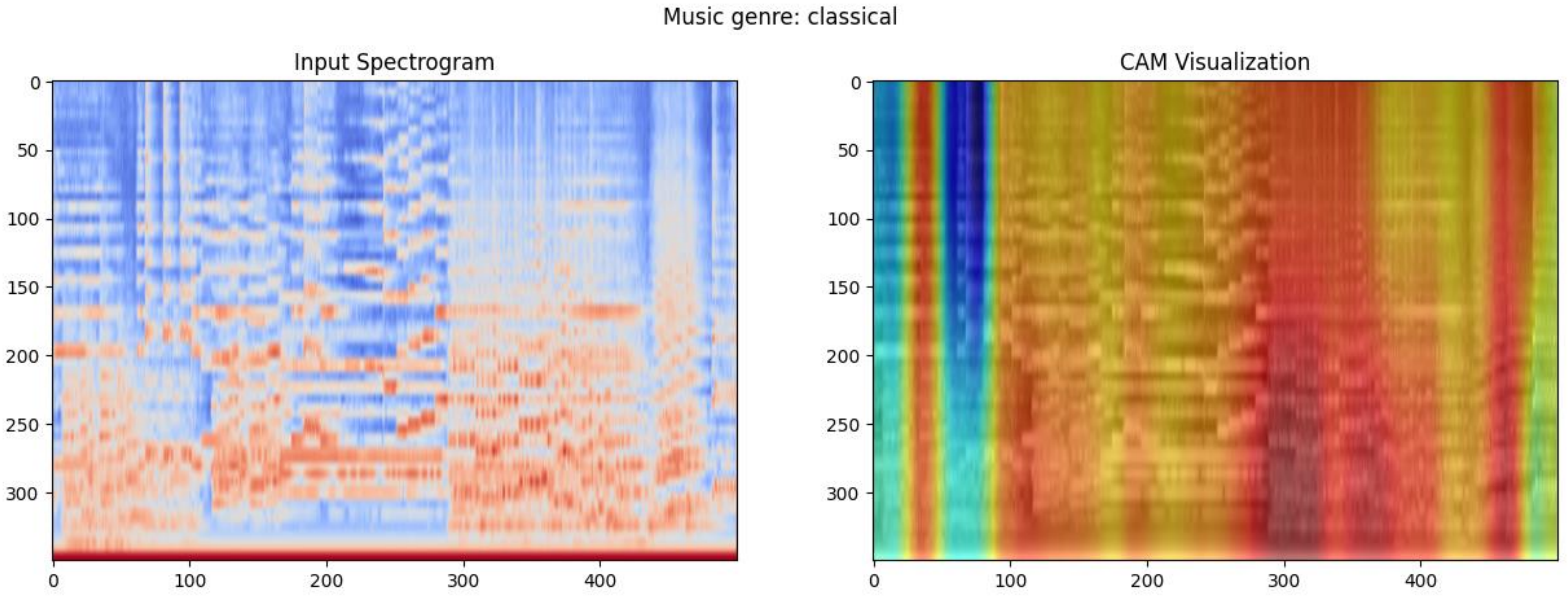
CAM Visualization



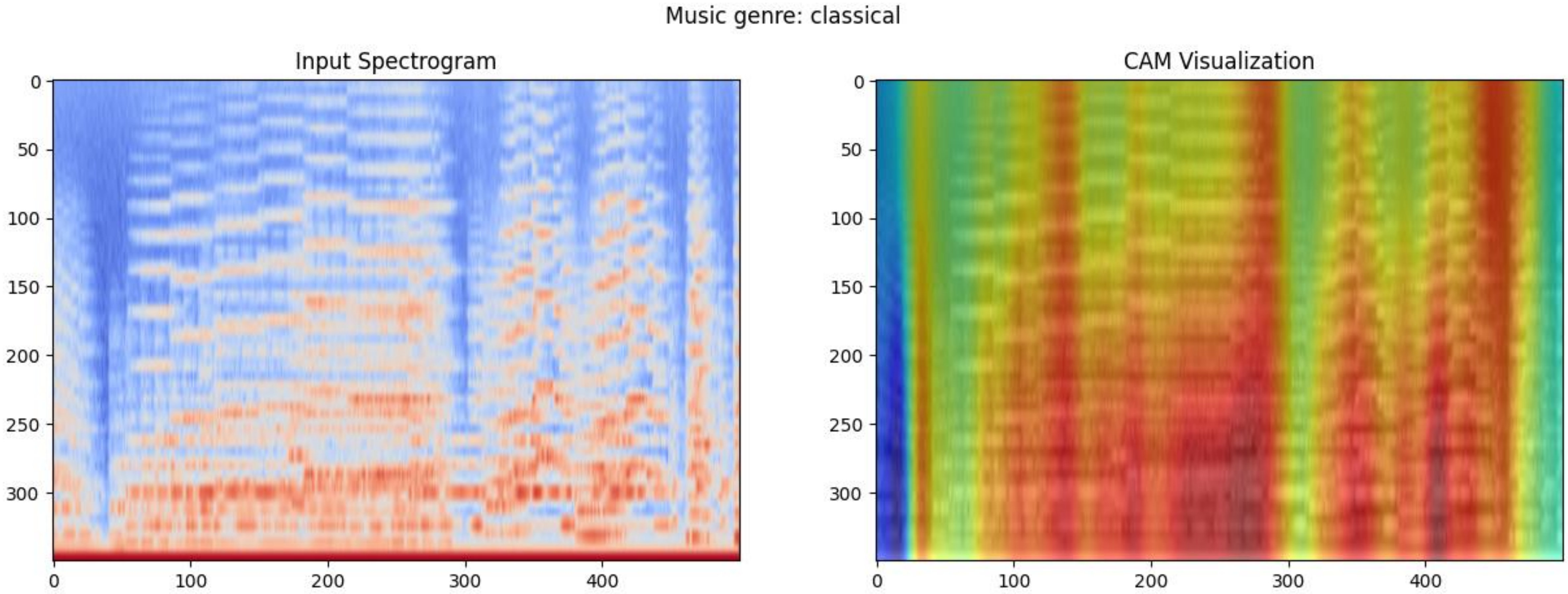
CAM Visualization



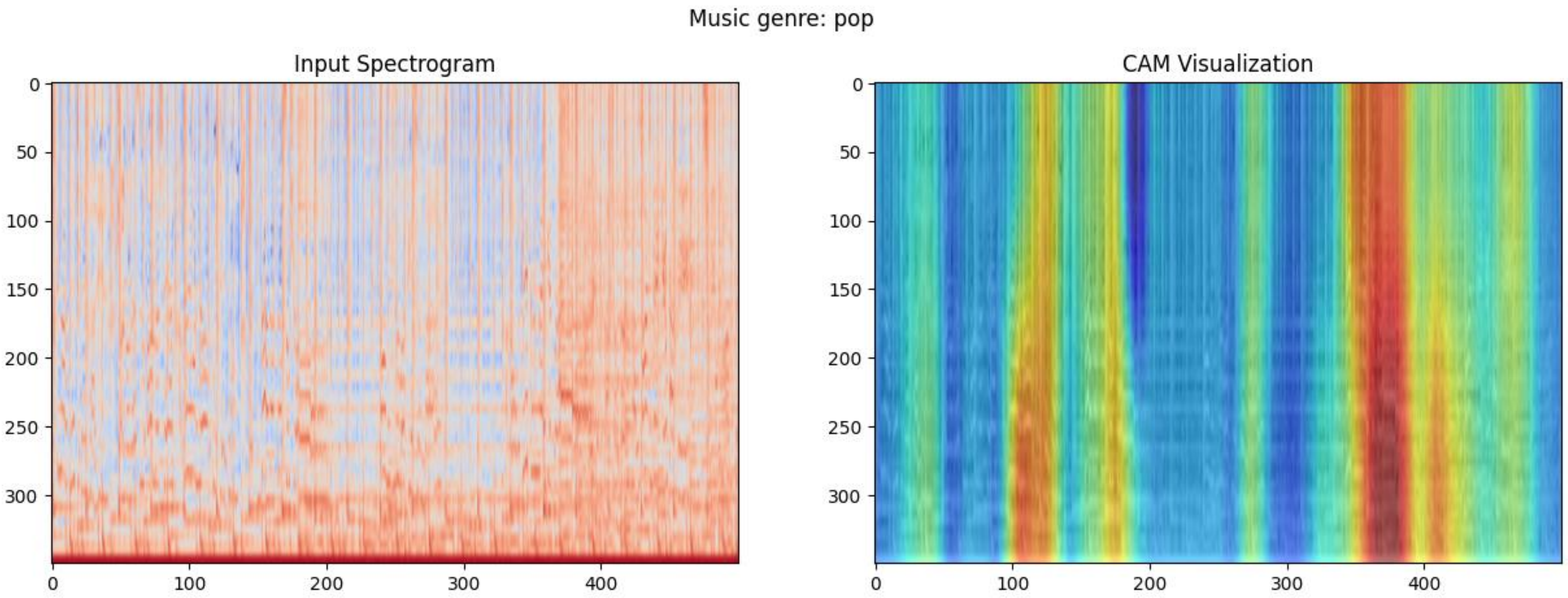
CAM Visualization



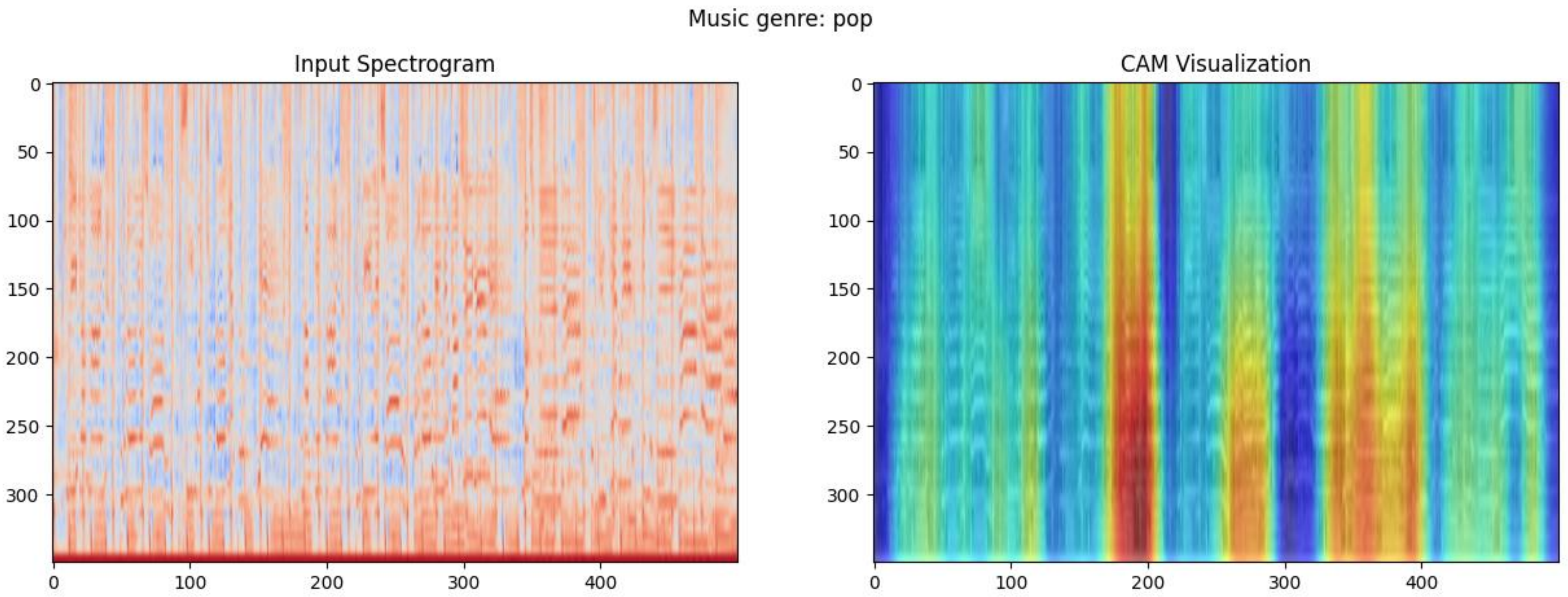
CAM Visualization



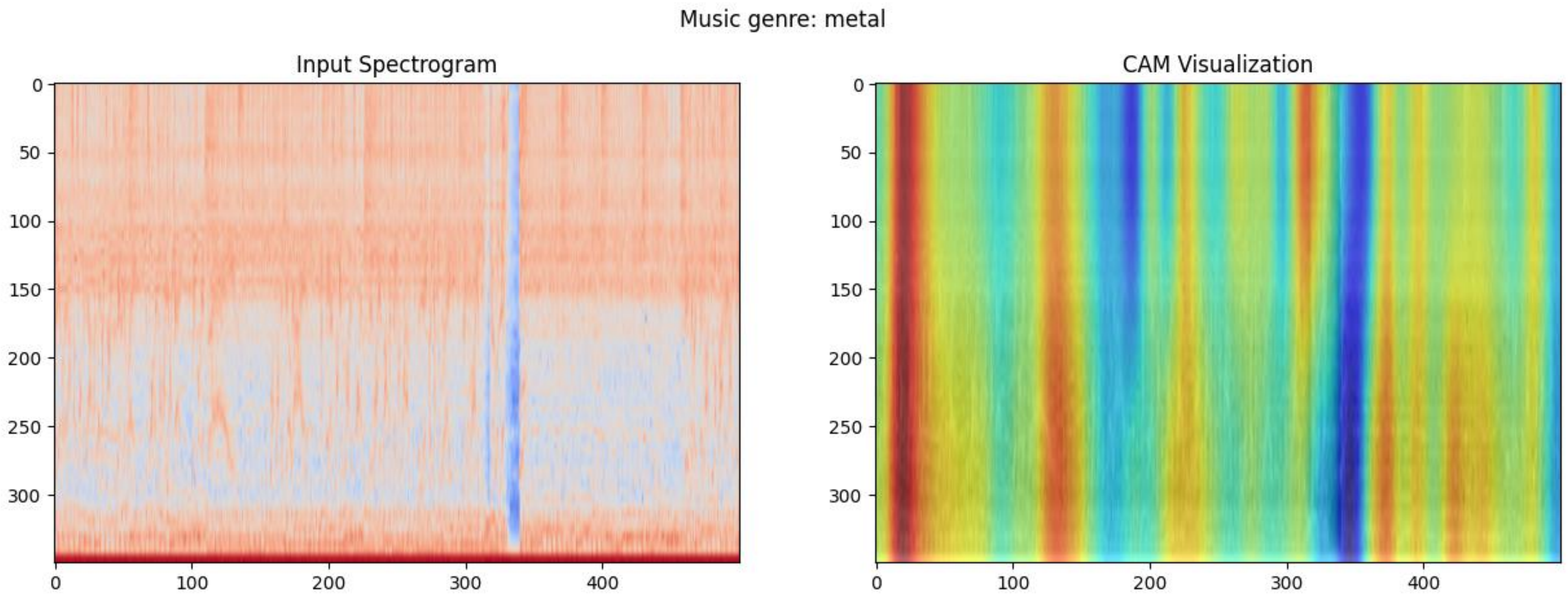
CAM Visualization



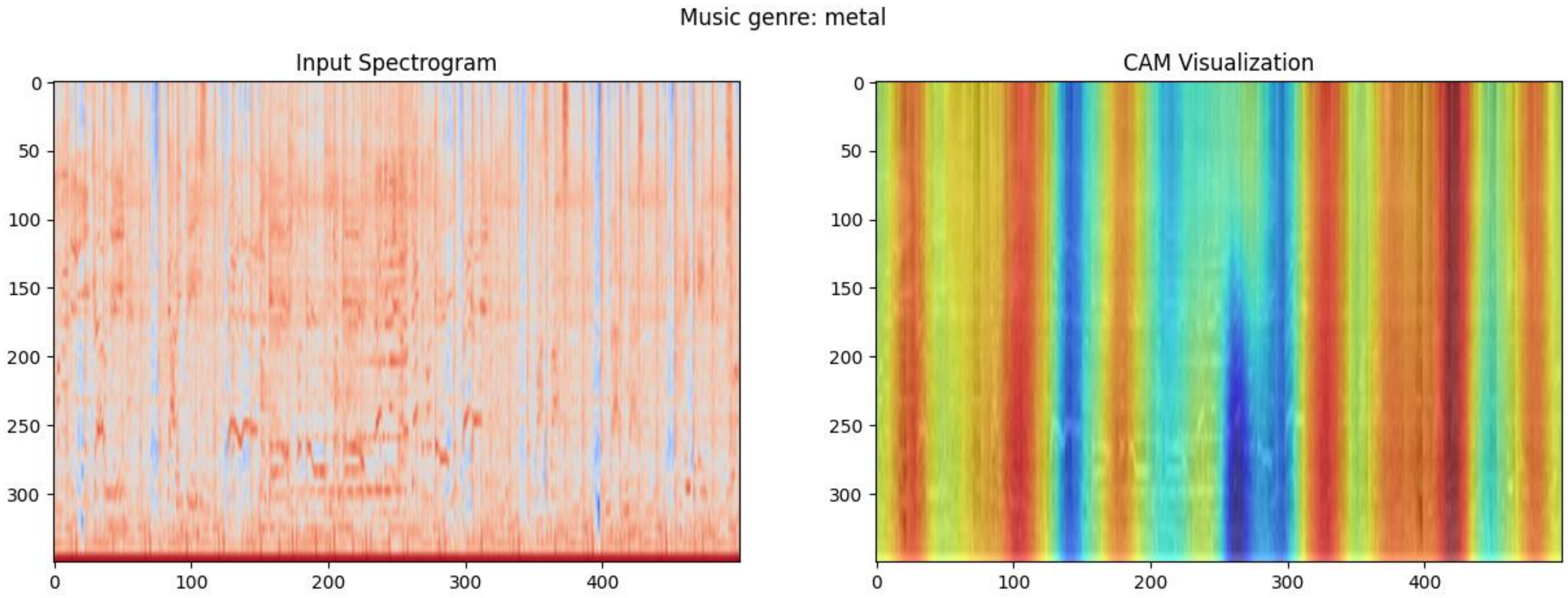
CAM Visualization



CAM Visualization



CAM Visualization



What have I learned?

- Audio file representation
- Dealing with audio files to create useful input into the Machine Learning models
- MEL Spectrograms
- Audio files augmentation
- Classification of the audio
- Comparison of a few models' performance
- Visualization of Saliency Maps

Any questions?

Thank you!