Universitat Pompeu Fabra Barcelona

Deep Learning 2024/2025

Spectrogram-Based Bird Sound Classification

Team G

Adrià García García 266036

Daniel Ortega Barberán 241163

Dídac Raya Rodríguez 242597

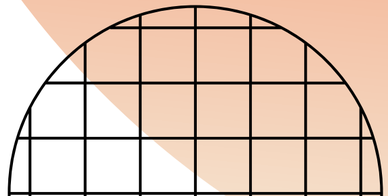
Motivation

Why birds?

- BirdCLEF → LifeCLEF
- Annual competition for 10+ years
- Real-world recordings
- Biodiversity monitoring & conservation

State of the art

- BirdCLEF contestants usually
  - Obtain short spectrograms from long soundscapes
  - Every single sound has a spectrogram instead of spectrograms for full songs
- CNNs are among the top models used (*EfficientNet*, *ResNet*, *MobileNetV2*)
- Accuracy is not disclosed in the official leaderboards



Background

- Extensive bibliography related to automatic sound recognition
  - **Zhang, B., Leitner, J., & Thornton, S.** (2019). *Audio Recognition using Mel Spectrograms and Convolutional Neural Networks*. Department of Electrical and Computer Engineering, University of California, San Diego.

- *MobileNetV2* architecture
  - **Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., & Chen, L.-C.** (2018). *MobileNetV2: Inverted Residuals and Linear Bottlenecks*

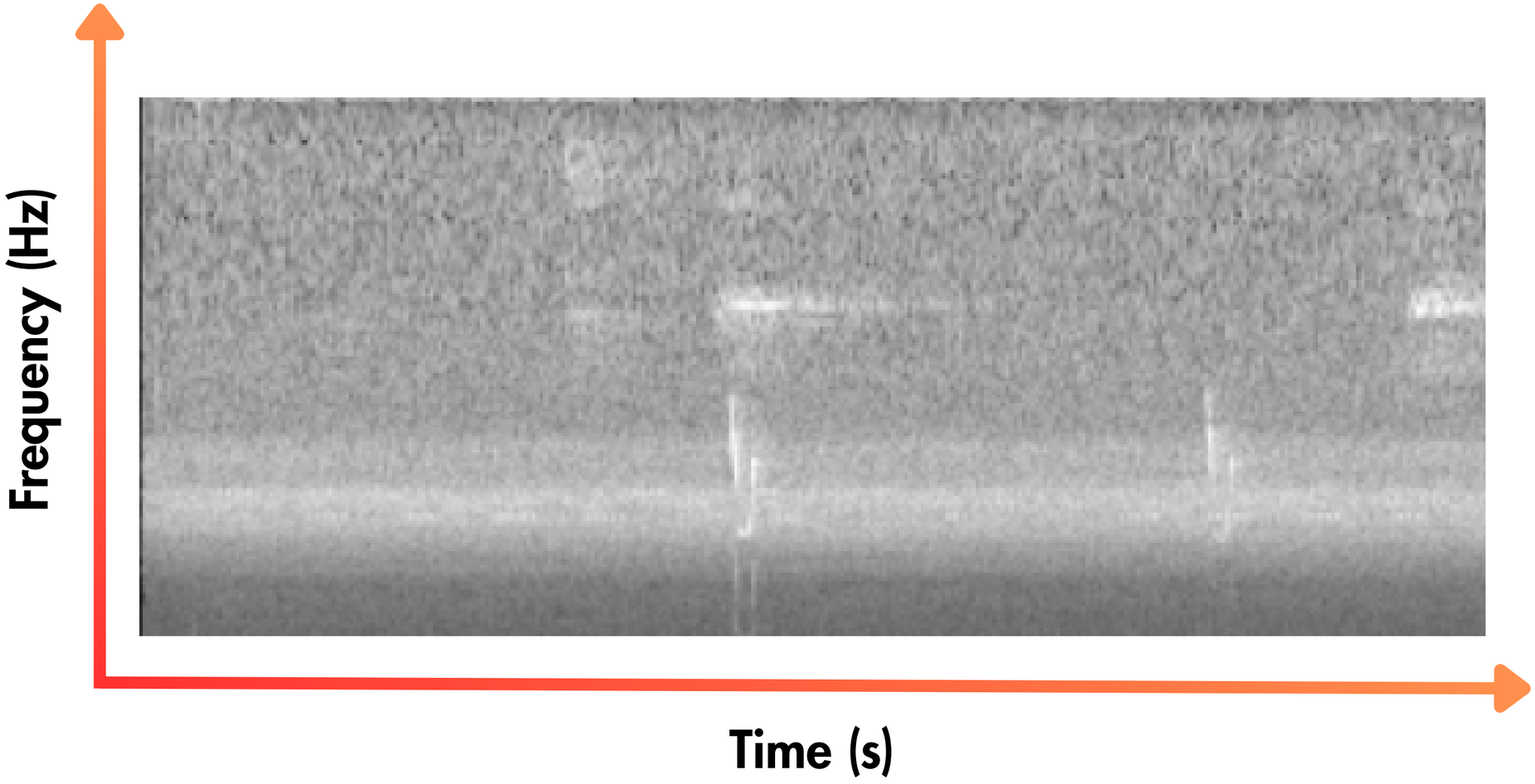
- *Microsoft Resnet18*
  - **He, K., Zhang, X., Ren, S., & Sun, J.** (2016). *Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770-778).

- Past BirdCLEF participants (Kaggle)

Mel Spectrograms

- Visualize audio frequencies over time
- Use the **Mel scale**, which aligns with human hearing
- Obtained from the **Fourier transform** of the audio signal
- Often log-scaled for better loudness representation
- Commonly used in speech and audio processing tasks

Mel Spectrograms



Objectives

Train a model for a simplified version of BirdCLEF competition:

- Using spectrograms of bird calls for the data of our model
- Using a variation of efficient CNN (MobileNetV2)
- Different approach to the use of spectrograms in audio recognition

Methodology

1. Extract a group of birds to classify. (397 → 6)
2. Clean the data to fit the model.
3. Create our model based on MobileNetv2
4. Create a train and validation batch.
5. Find optimal hyperparameters.
6. Try pre-trained ResNet18 on our dataset

Dataset

- BirdCLEF2021
- Large Dataset (62.900 samples, 397 classes)
- **`train_short_audio`**
  - Short Audio
  - Known species
- **`train_soundscapes`**
  - Longer audio
  - Some not including birds
  - Unknown species
- **`test_soundscapes`**
  - Examples for the test set
  - Audios not published

train_short_audio

397 Total Classes → 6 Used Classes

Melospiza melodia

Corvus corax

Henicorhina leucophrys

Passer domesticus

Cardinalis cardinalis

Loxia curvirostra

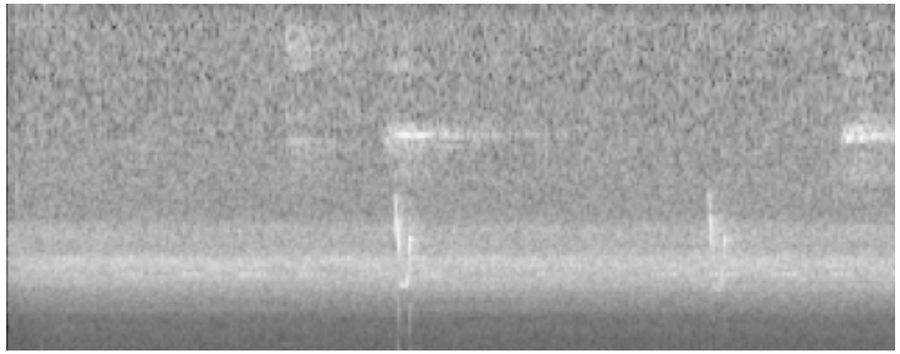
train_short_audio

14 Total Features → 3 Used Features

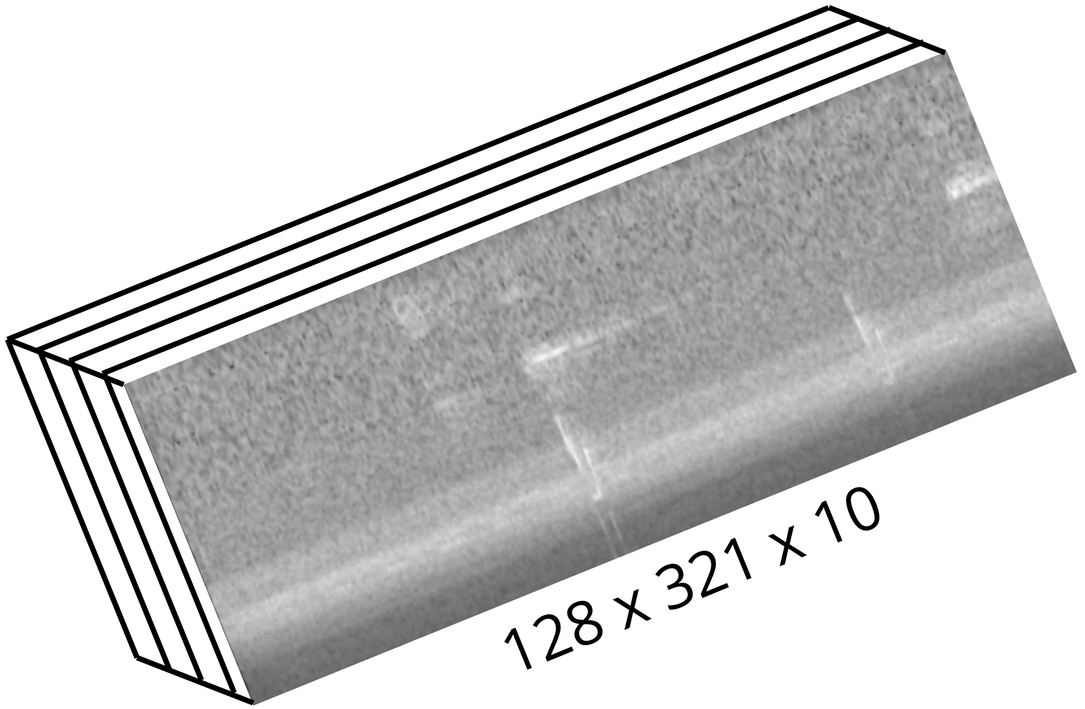
- file_name
- primary_label
- secondary_label

} label_id

Data Shape



128 x 321



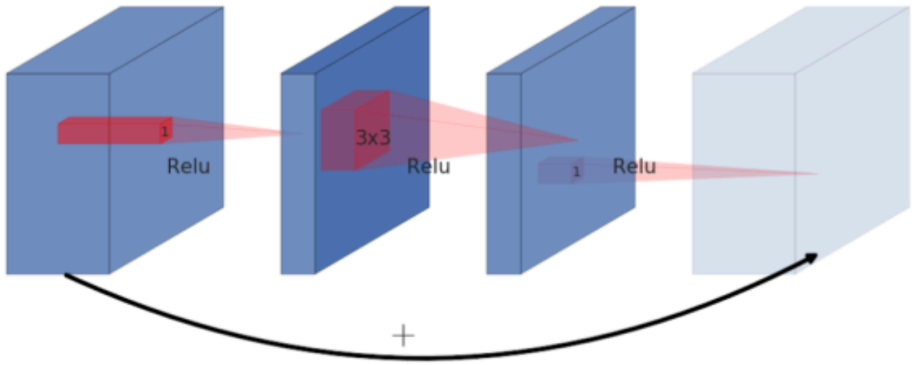
Total Samples left: 609

30% used to test

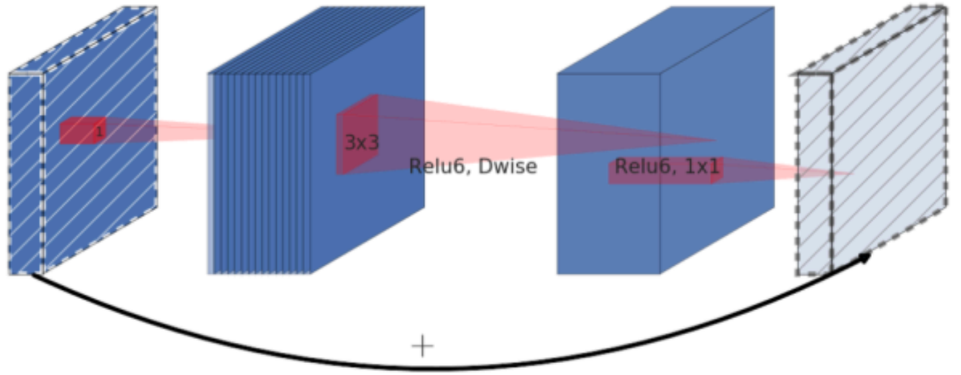
MobileNetV2

- Lightweight
- Scalable and Customizable Architecture
- Great Accuracy-Speed Tradeoff

(a) Residual block



(b) Inverted residual block

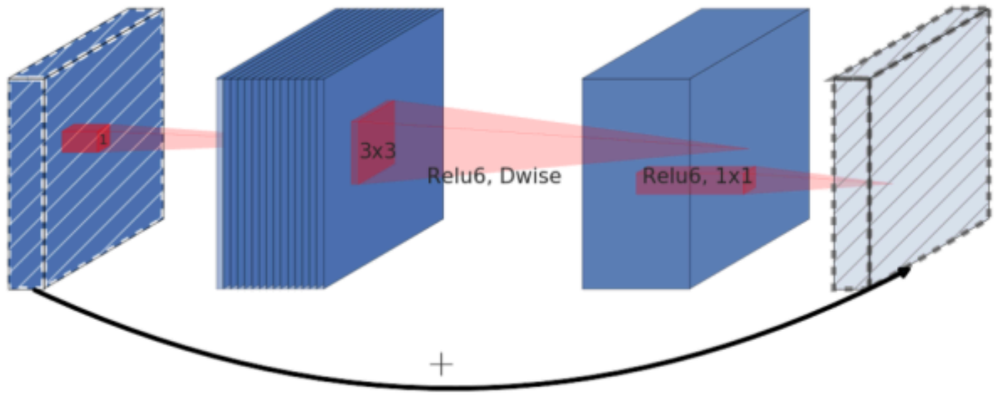


Layers

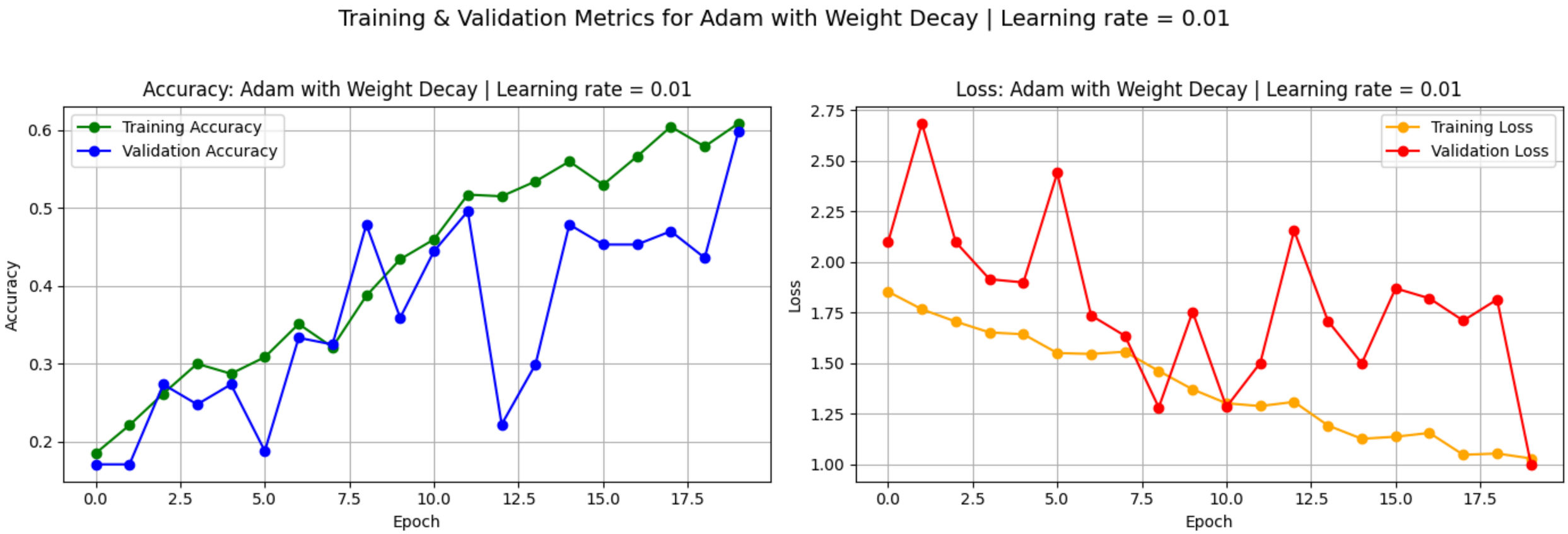
base settings
([1, 16, 1, 1],
[6, 24, 2, 2],
[6, 32, 3, 2],
[6, 64, 4, 2],
[6, 96, 3, 1],
[6, 160, 3, 2],
[6, 320, 1, 1])

[t, c, n, s]
t = expansion factor
c = output channels
n = number of blocks
s = stride of the first block

custom settings
([1, 16, 1, 1],
[6, 24, 2, 2],
[6, 32, 3, 2],
[6, 64, 2, 2],
[6, 64, 1, 1])

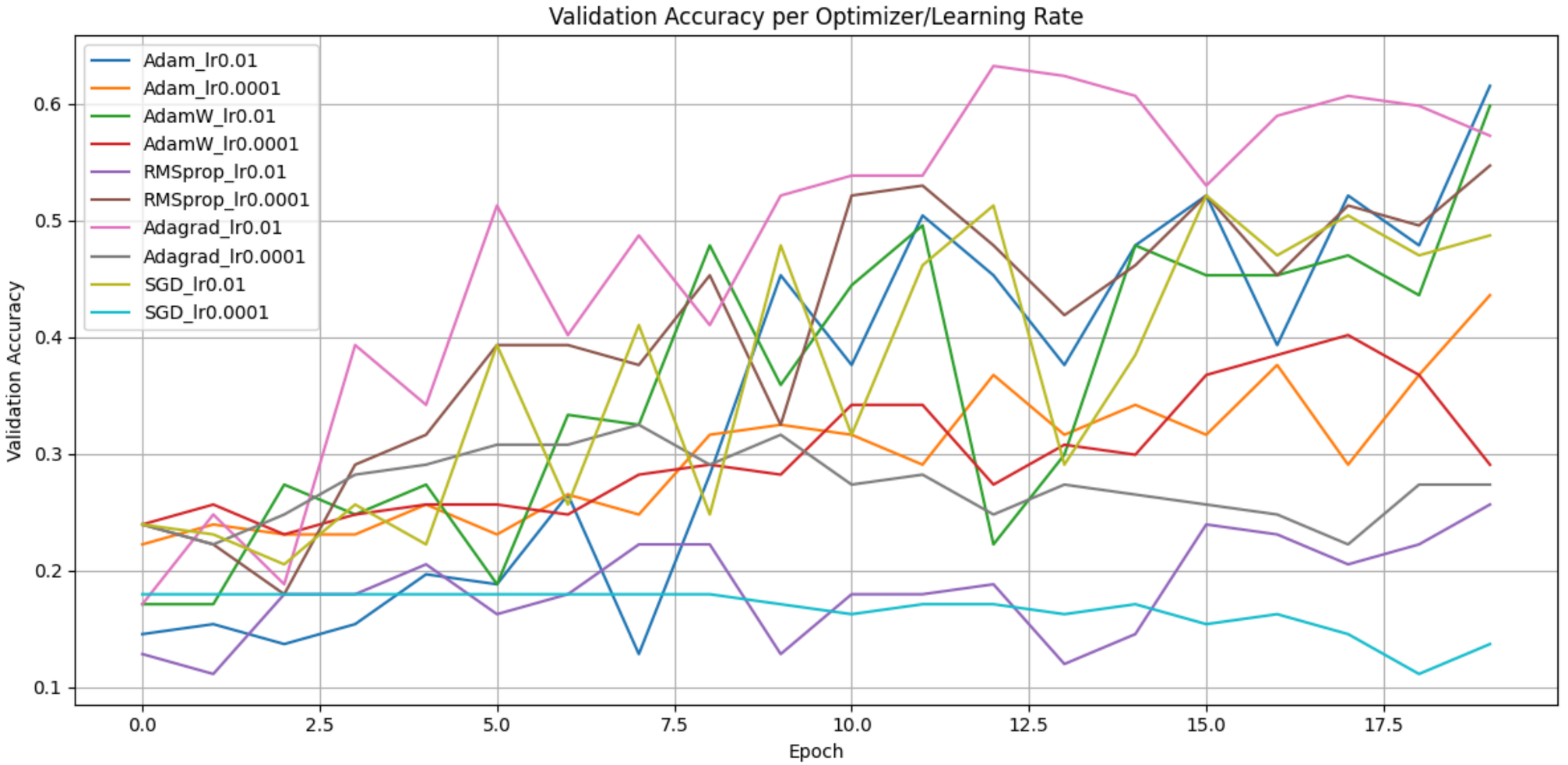


Results

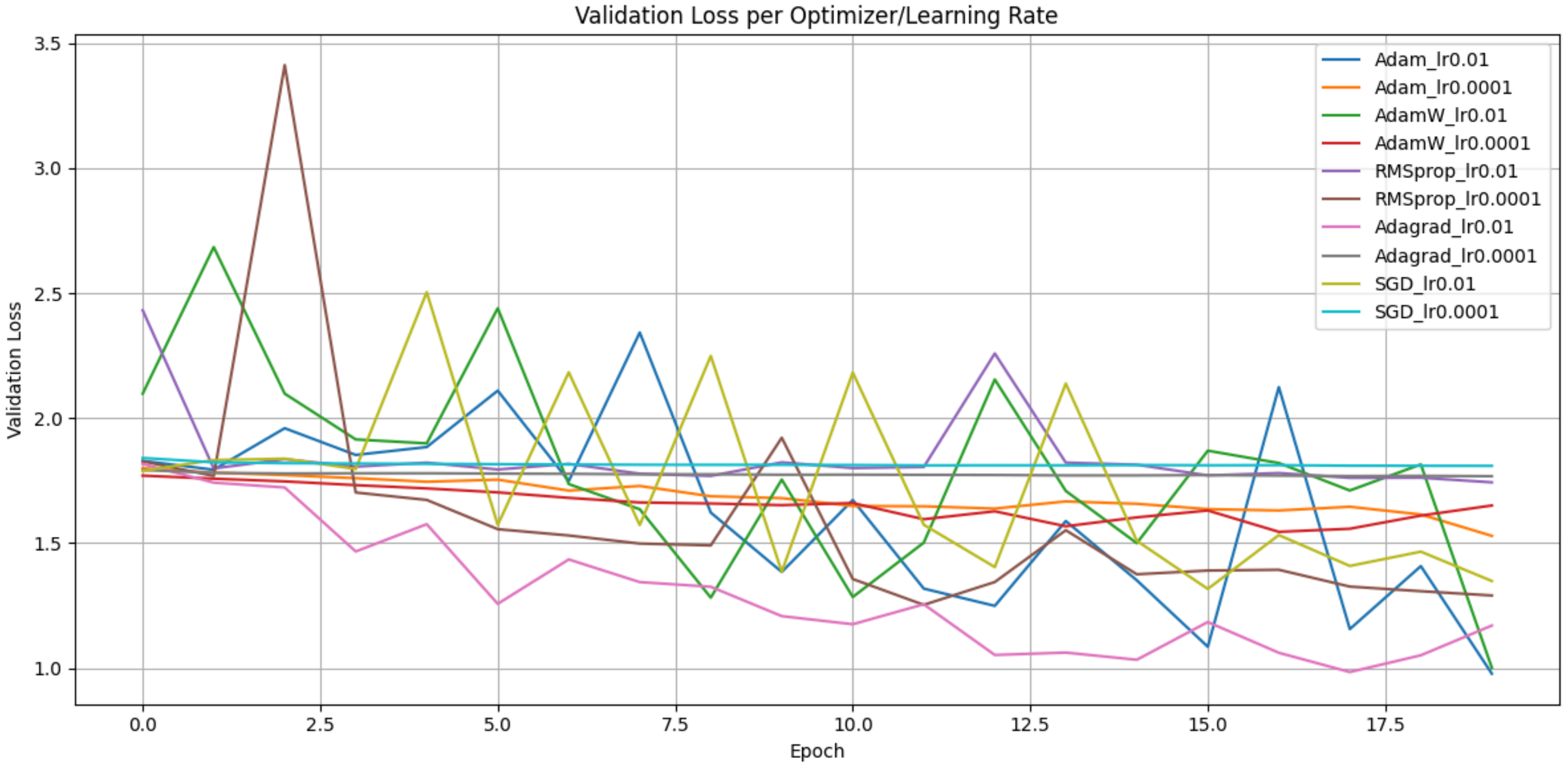


Batch_size = 16 Optimizer = Adam with weight decay Learning rate = 0.01 Loss Function = Cross Entropy

Results


Validation Accuracy per Optimizer/Learning Rate
Adam_lr0.01
Adam_lr0.0001
AdamW_lr0.01
AdamW_lr0.0001
RMSprop_lr0.01
RMSprop_lr0.0001
Adagrad_lr0.01
Adagrad_lr0.0001
SGD_lr0.01
SGD_lr0.0001
Validation Accuracy
0.1
0.2
0.3
0.4
0.5
0.6
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
Epoch

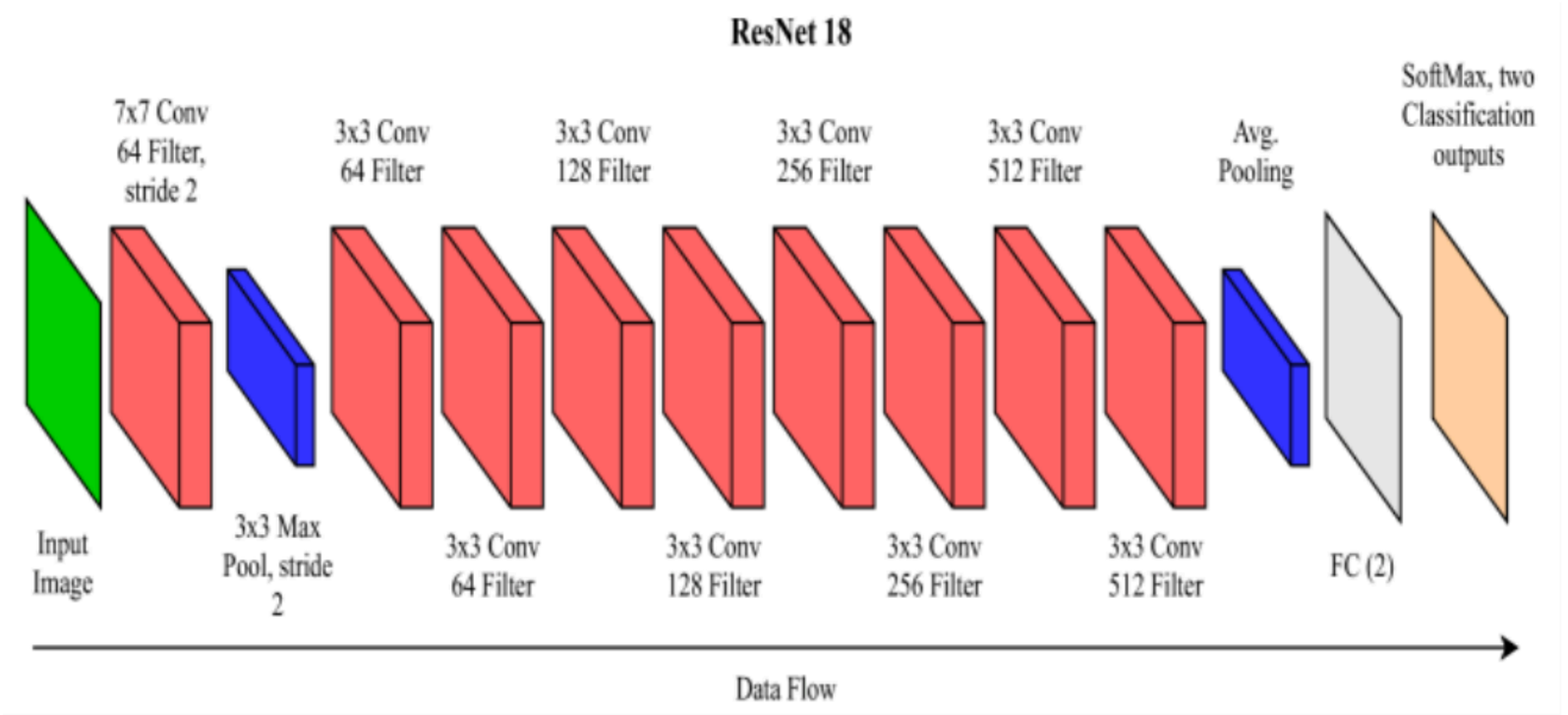
Results


Validation Loss per Optimizer/Learning Rate
Validation Loss
Epoch
3.5
3.0
2.5
2.0
1.5
1.0
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
Adam_lr0.01
Adam_lr0.0001
AdamW_lr0.01
AdamW_lr0.0001
RMSprop_lr0.01
RMSprop_lr0.0001
Adagrad_lr0.01
Adagrad_lr0.0001
SGD_lr0.01
SGD_lr0.0001

ResNet18

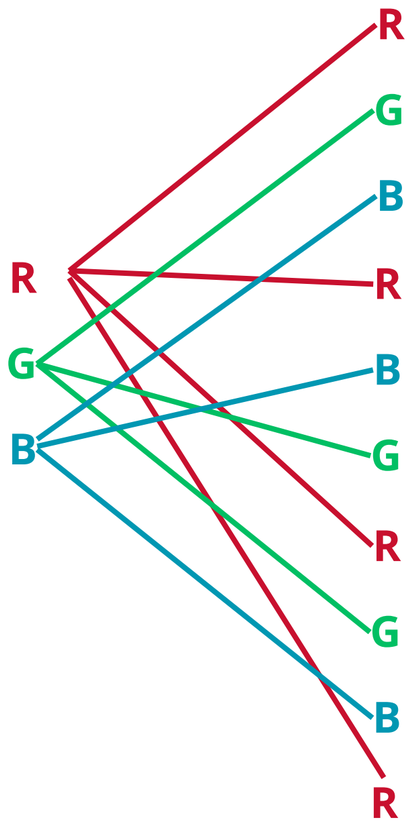
First Layer
RGB input
↓
10 input

Last Layer (Linear)
1000 output
↓
6 output

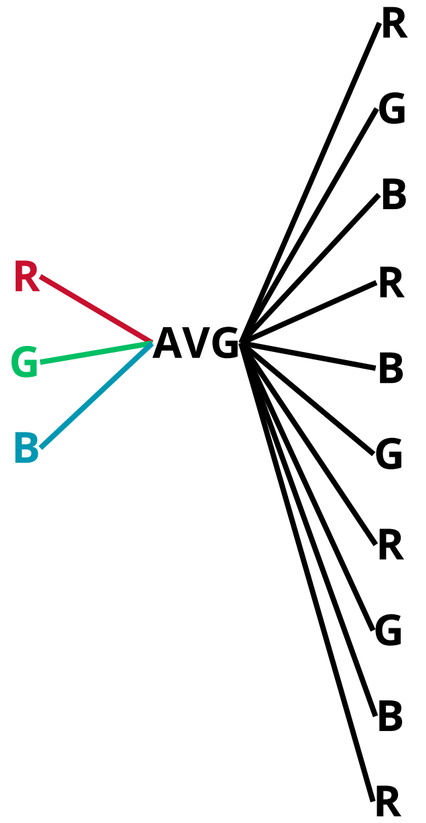


ResNet18

Repeated First layer weights

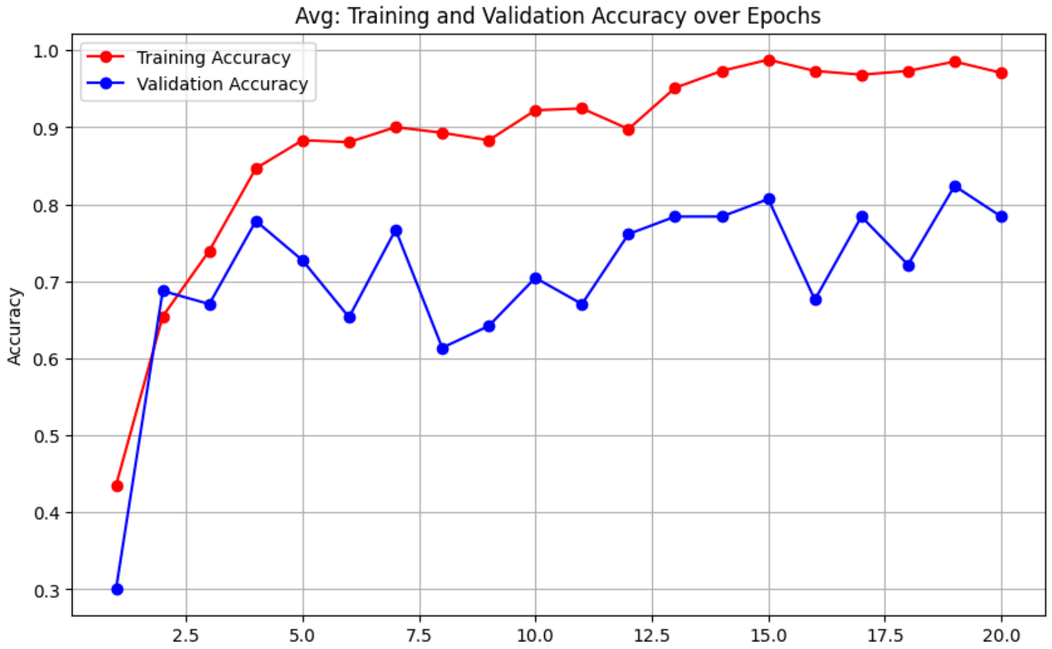


Avg First layer weights

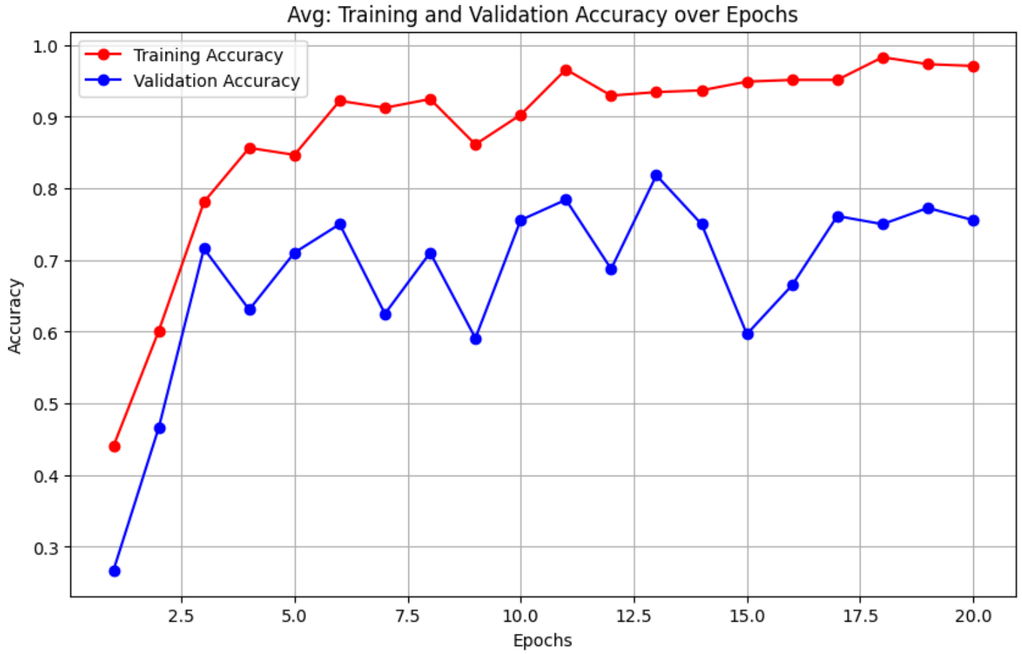


ResNet18

Repeated first layer weights



Avg First layer weights

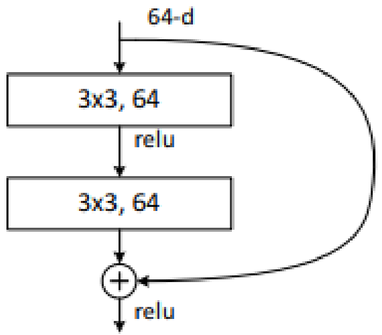


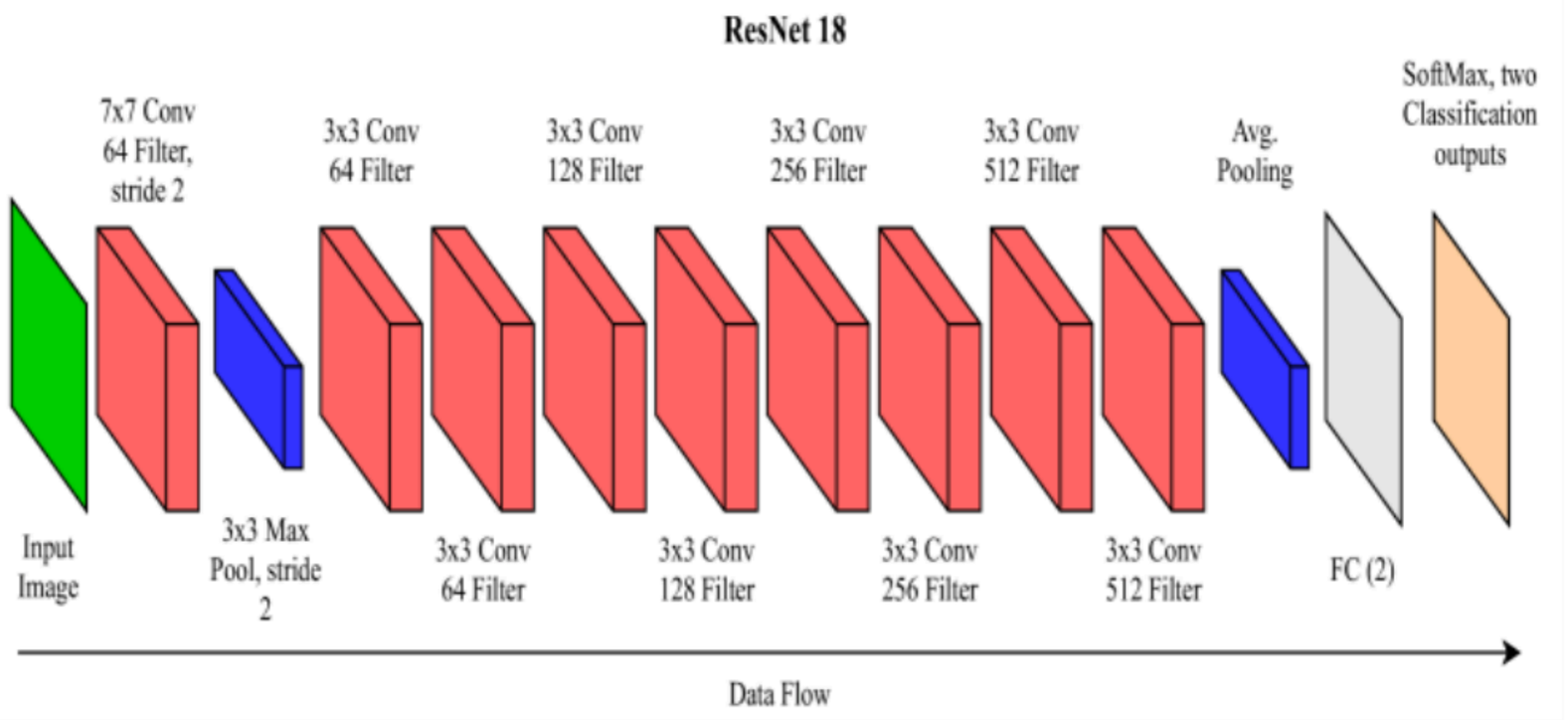
ResNet18

Pre-trained on ImageNet Dataset

1000 classes
1.4 million images

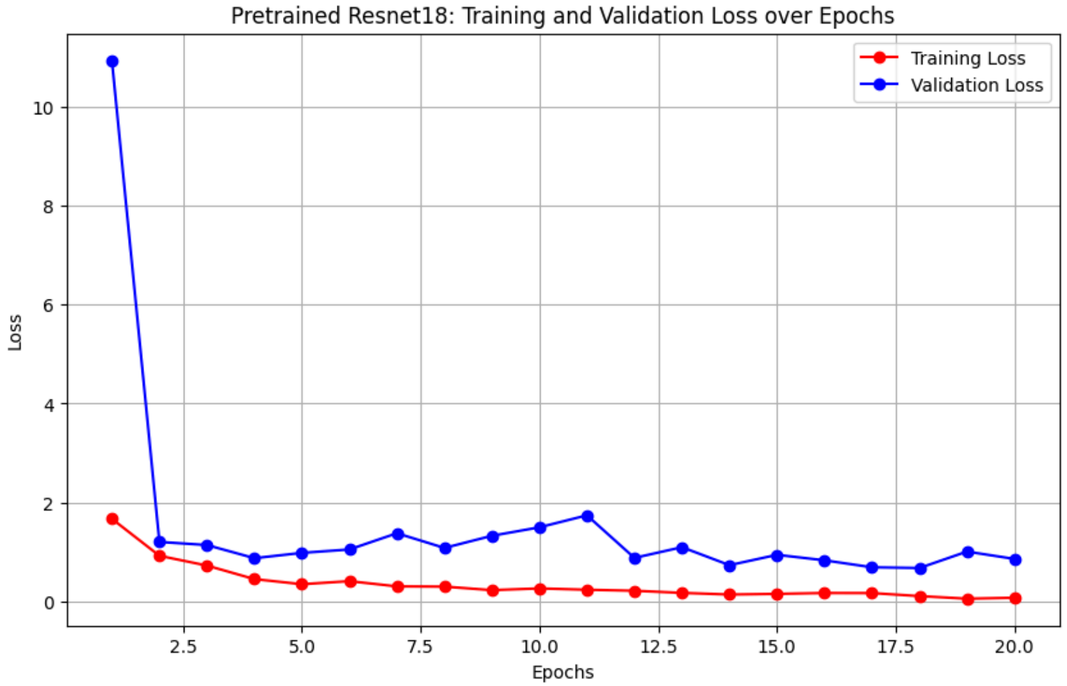
Basic Residual Block

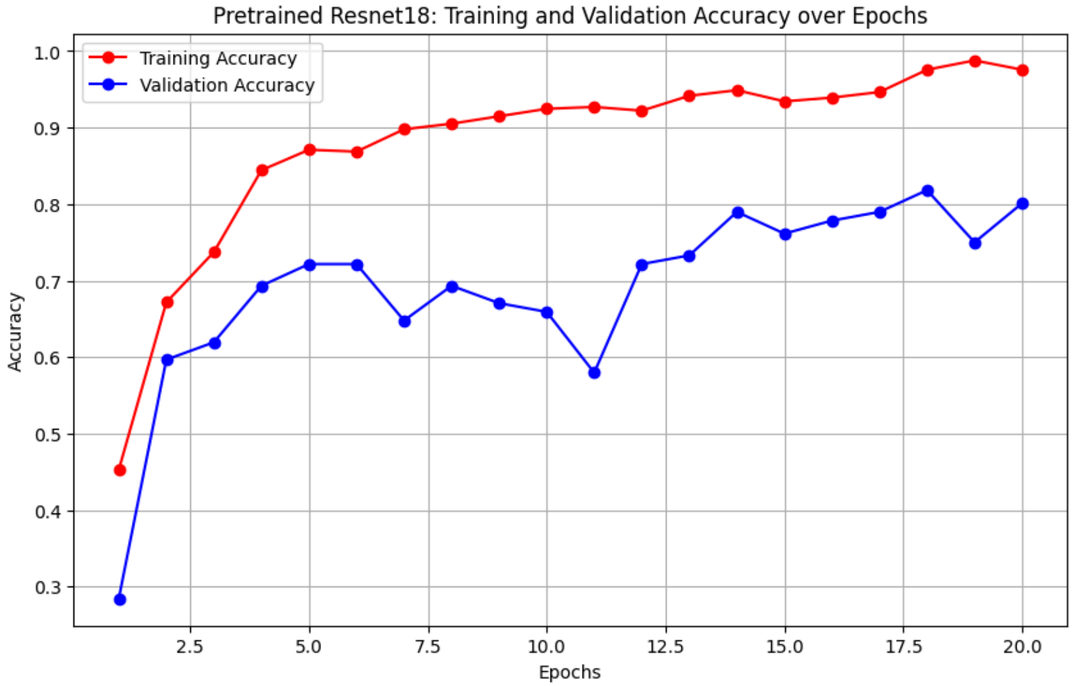




8 Residual Blocks total

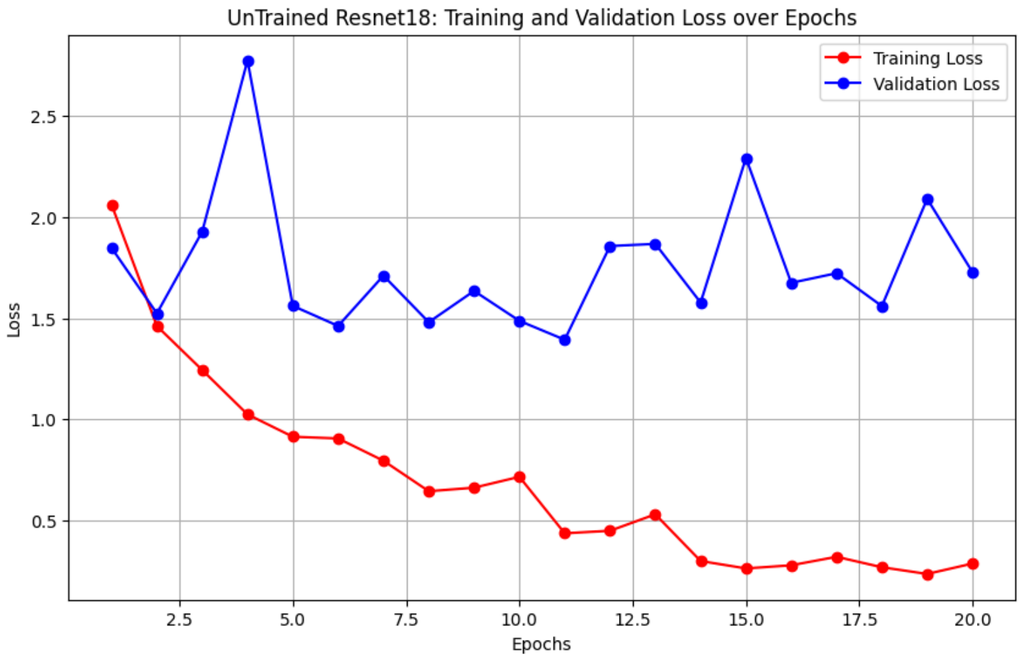
Results

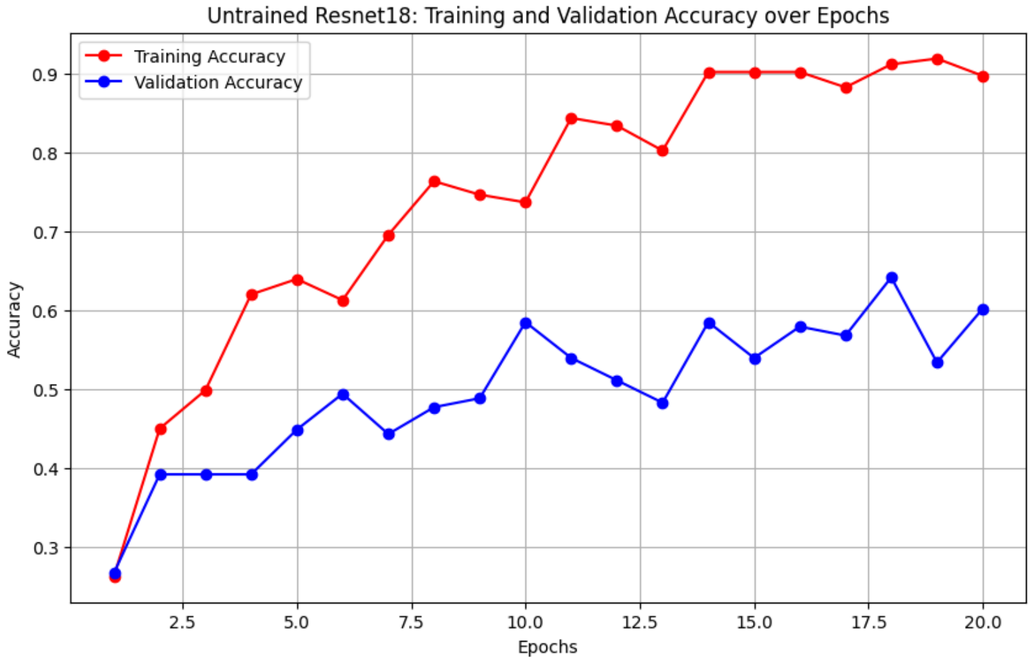




Batch_size = 16 Learning rate = 0.001 Optimizer = Adam with weight decay (0.0001) Loss Function = Cross Entropy

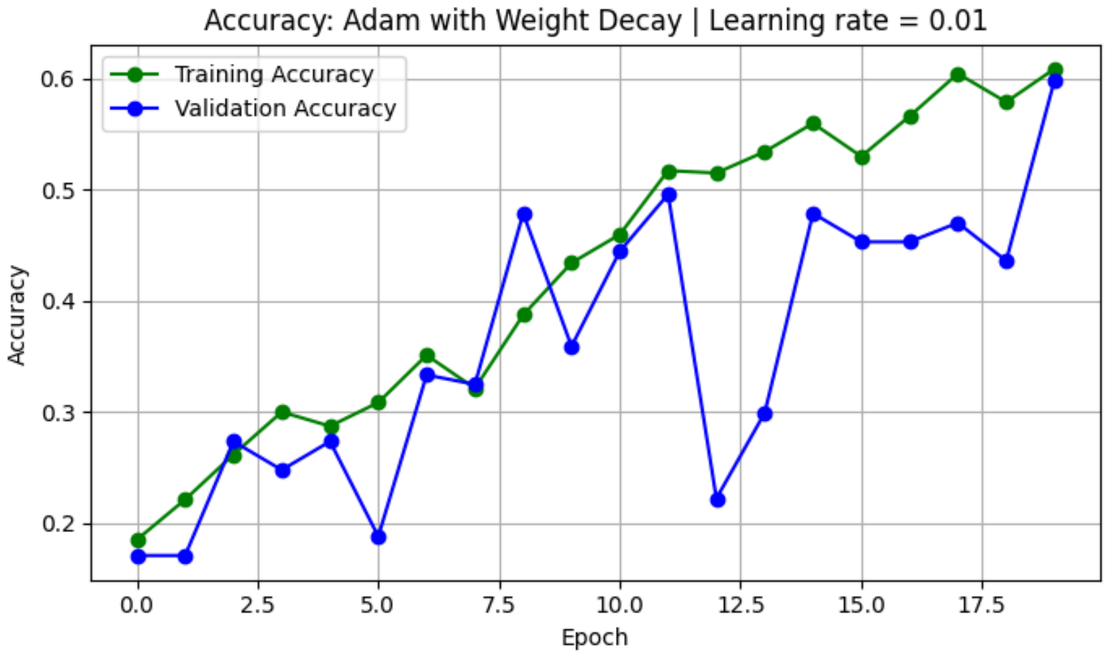
Results



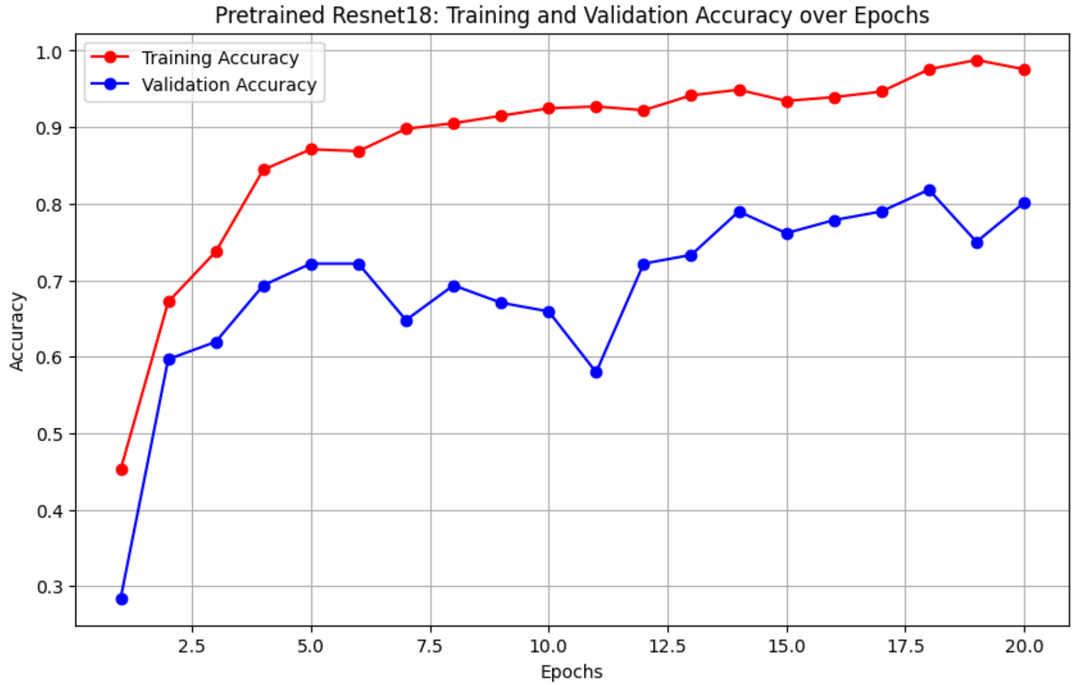


Batch_size = 16 Learning rate = 0.001 Optimizer = Adam with weight decay (0.0001) Loss Function = Cross Entropy

Comparison of Results



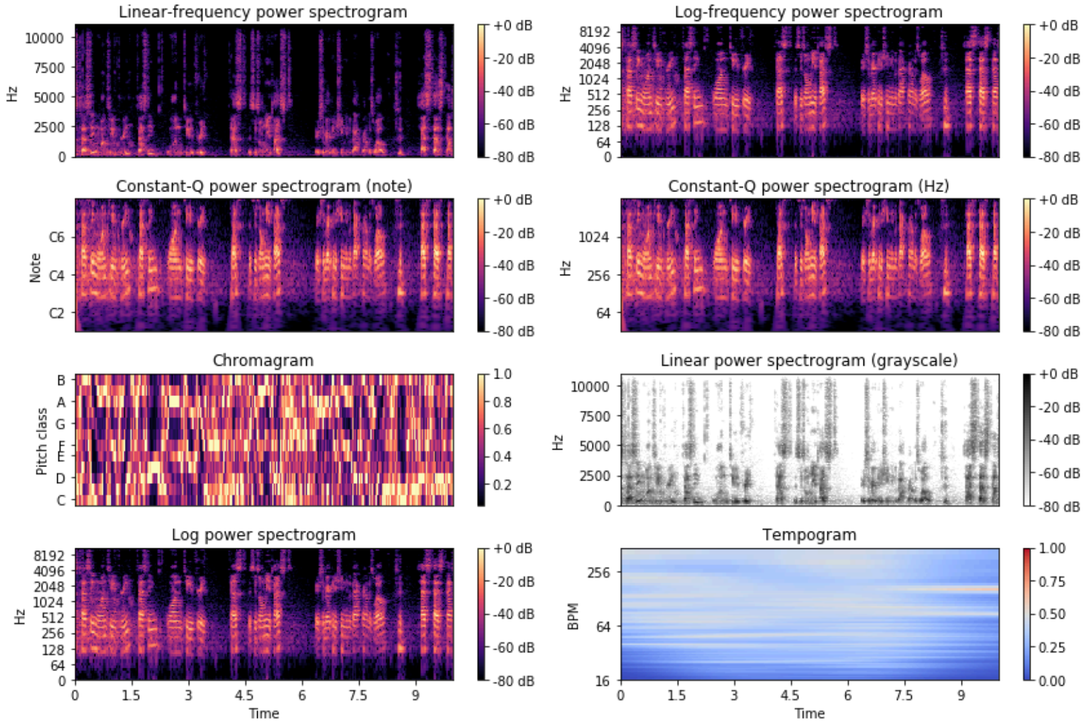
Custom MobilenetV2



ResNet18

Future improvements

- Try other dataset not focused on multi-classification (song/melody datasets) using transfer learning
- Try other types of spectograms and experiment more with the models
- Try our methodology in the real competition and for a whole dataset



Conclusions

- The way we encode the spectrograms has potential
  - It has a drawback, samples must be the same size
- Notable performance in our *MobileNetV2*.
- As expected the pre-trained *Resnet18* is more stable
- Time was limited, not all our initial objectives were met

Thank you for your attention

We are now open to answering your questions