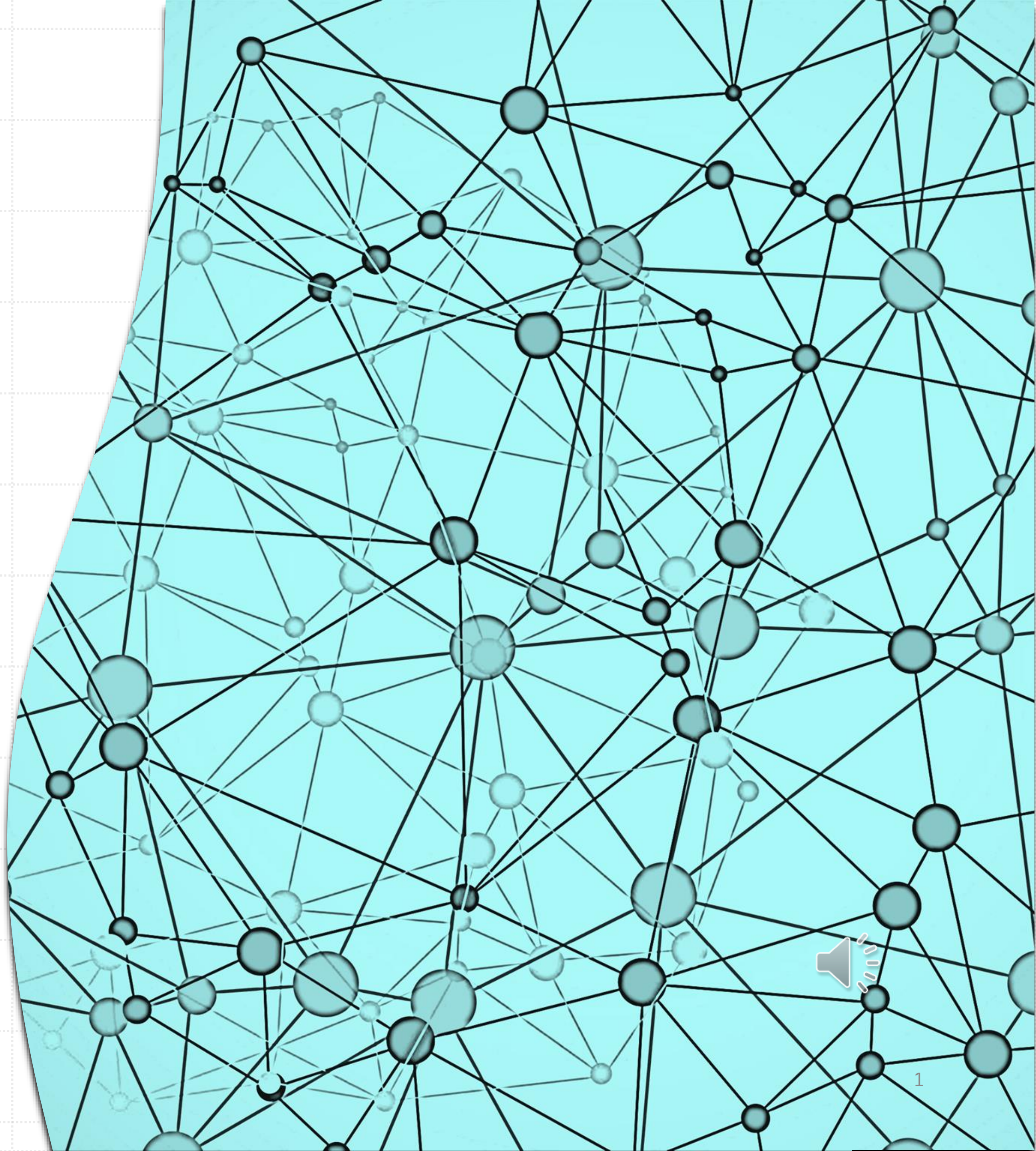
Binary Classification



List Of Content

- Literature Review
- Data loading and preprocessing
- Modeling
- Evaluation
- Ablation Study
- Conclusion

Literature Review

Literature Review

01 Reviewed 14 papers on image classification.

02 Image classification models comparison

03 Model selected: ResNet-50

04 Explore the latest advancements in image classification models.

05 Resnet-50 [2015], InceptionV3 [2016], InceptionResnetV2 [2017] MobileNetV2 [2018], YoloV8 [2023].

Brain tumors from MRI scans DataSet [1]

Model	Train Accuracy	Validation Accuracy	Test Accuracy
SVM Classifier	71.34%	52.56%	50.51%
Random Forest	72.78%	64.3%	64.23%
VGG16	96.3%	92.23%	90.54%
Inception V3	93.4%	64.8%	63.94%
ResNet	99.7%	82.12%	81.92%

K. Dong [5]ImageNet dataset

Model	Top-1 Accuracy	Multiply-Adds (M)
MobileNetV1	70.6%	569
MobileNetV2	72.0%	301
ResNet-101	75.2%	1550
VGG-16	74.5%	15300

S. Sharma [6] , C. Ma [7]

Model Name	Accuracy	Data
ResNet	99.3%	MNIST
GoogleNet	94.5%	
VGG16	98.4%	
VGG-16	93.4%	CIFAR-10
Inception v3	94.2%	
ResNet	95.1%	
DenseNet	95.8%	

Data loading and preprocessing

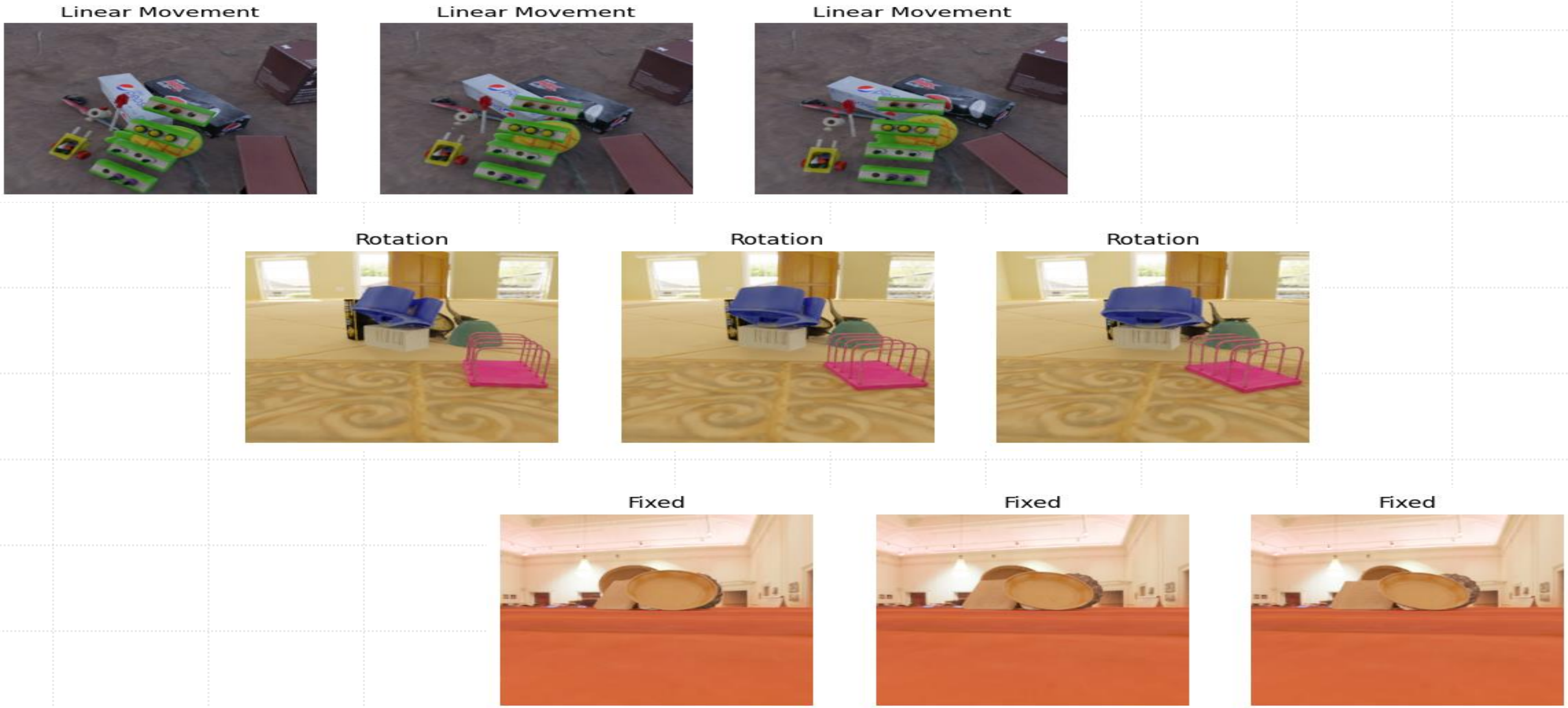
Data

Three Categories
Linear movement, rotation
, fixed random rotate

30 Video
each consisting of 24 frames

2160 Images
with 11 items

Samples of data, Kubric: a scalable dataset generator. In IEEE/CVF CVPR, 2022.



Data loading and preprocessing

01 Convert the data into dataframe

02 Find the Most suitable Item to classify.
14: 2208, 6: 2064, 13: 1752, 16: 888, 1: 216, 0: 216, 3: 192, 11: 72, 10: 72, 9: 72, 12: 48

03 Getting labels for each image

04 Resize all images to the same size.

05 Images enhancement

Modeling

Models

ResNet-50

InceptionResnet V2

MobileNetV2

InceptionV3

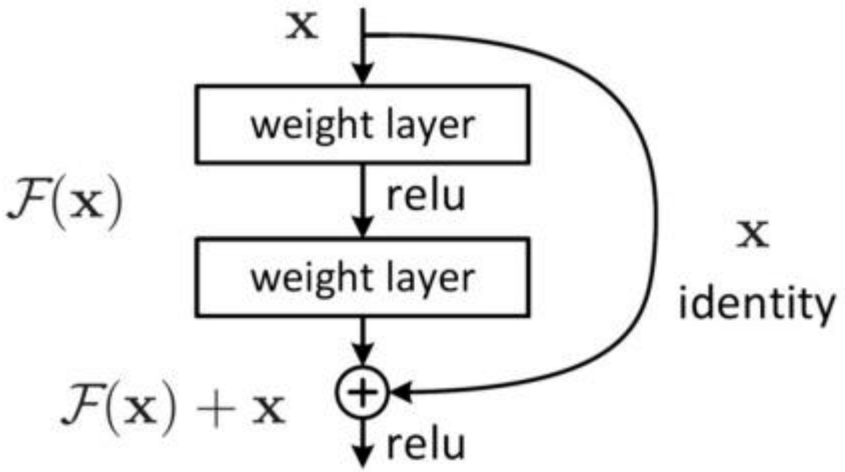
YoloV8n

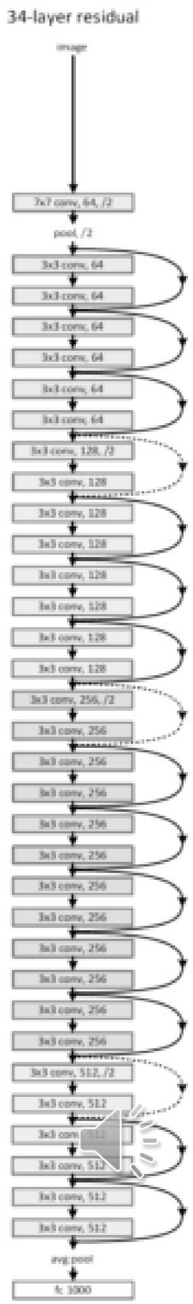
Models Parameters

<table>
<tr><th></th><th>ResNet50</th><th>InceptionV3</th><th>InceptionResNetV2</th><th>MobileNetV2</th><th>Yolov8n</th><th>Yolov8s</th></tr>
<tr><td>Total params (Million)</td><td>24.6 M (93.98 MB)</td><td colspan="3">22.852 M (87.17 MB)</td><td>2.7 M (6 MB)</td><td>11.2 M (24 MB)</td></tr>
<tr><td>Trainable params (Million)</td><td>1.049 M (4.00 MB)</td><td colspan="3">22.818 M (87.04 MB)</td><td>-</td><td>-</td></tr>
<tr><td>Non-trainable params (Million)</td><td>23.58 M (89.98 MB)</td><td colspan="3">34432 M (134.50 KB)</td><td>-</td><td>-</td></tr>
<tr><td>Training Time (Seconds)</td><td>40.83 s</td><td>66.65 s</td><td>78.8 s</td><td>79.6 s</td><td>26.7 s</td><td>40.2 s</td></tr>
<tr><td>Evaluation Time [Test Data] (Seconds)</td><td>0.37 s</td><td>0.385 s</td><td>0.55 s</td><td>0.55 s</td><td>0.4 s</td><td>0.42</td></tr>
</table>

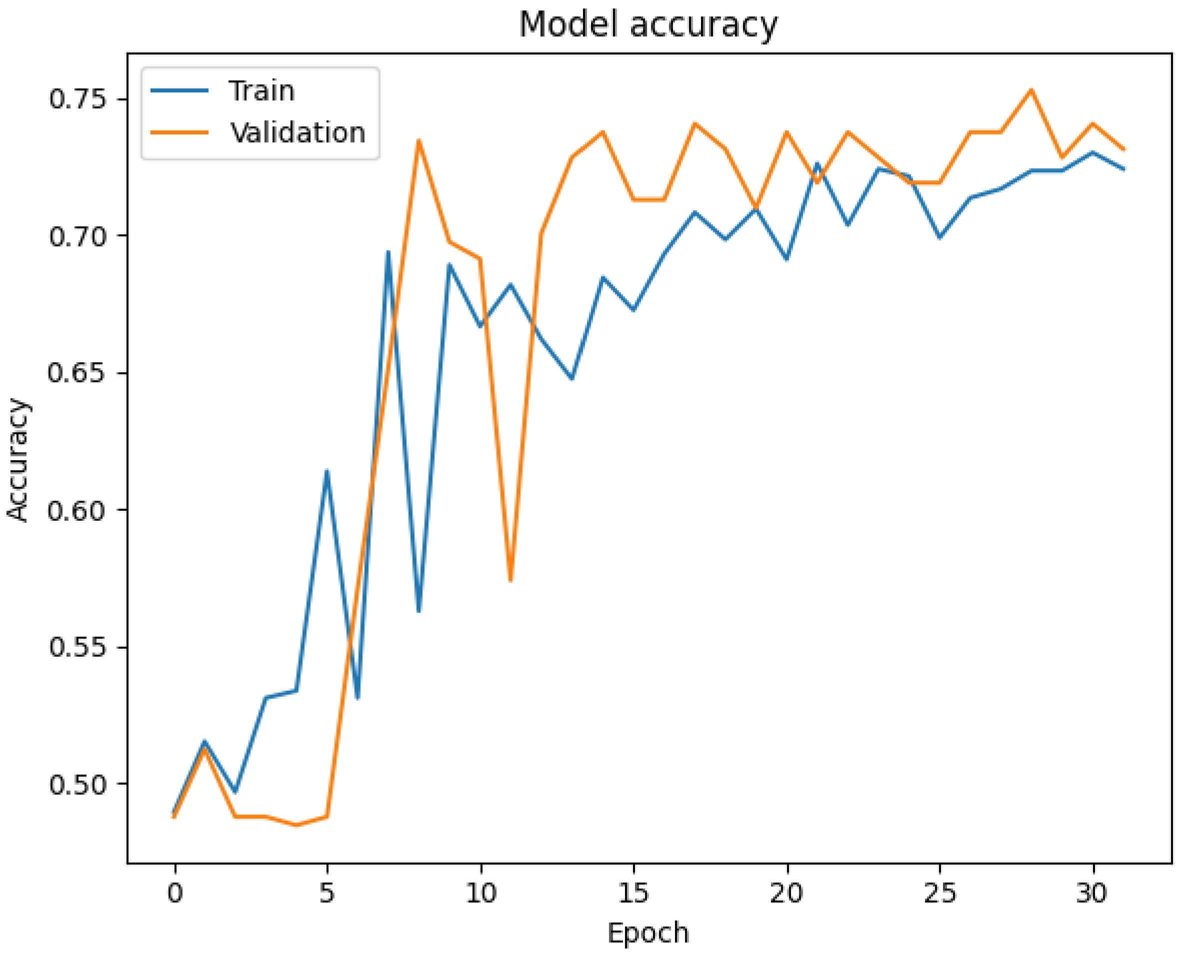
ResNet-50

- ResNet50 excels in binary classification tasks with its 50 convolutional layers.
- These layers extract intricate features from images.
- Skip connections combat the vanishing gradient problem, boosting training efficiency for accurate classification.

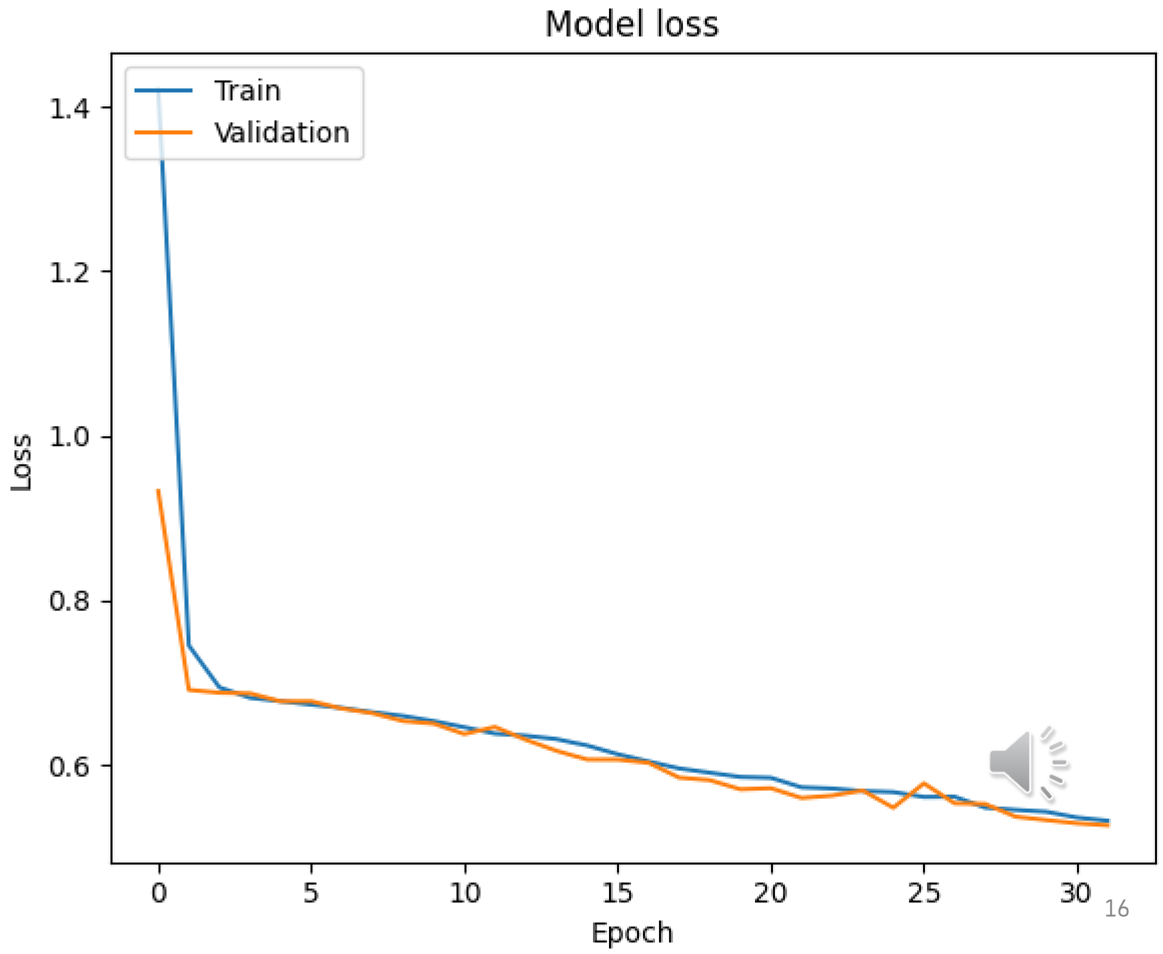




ResNet-50

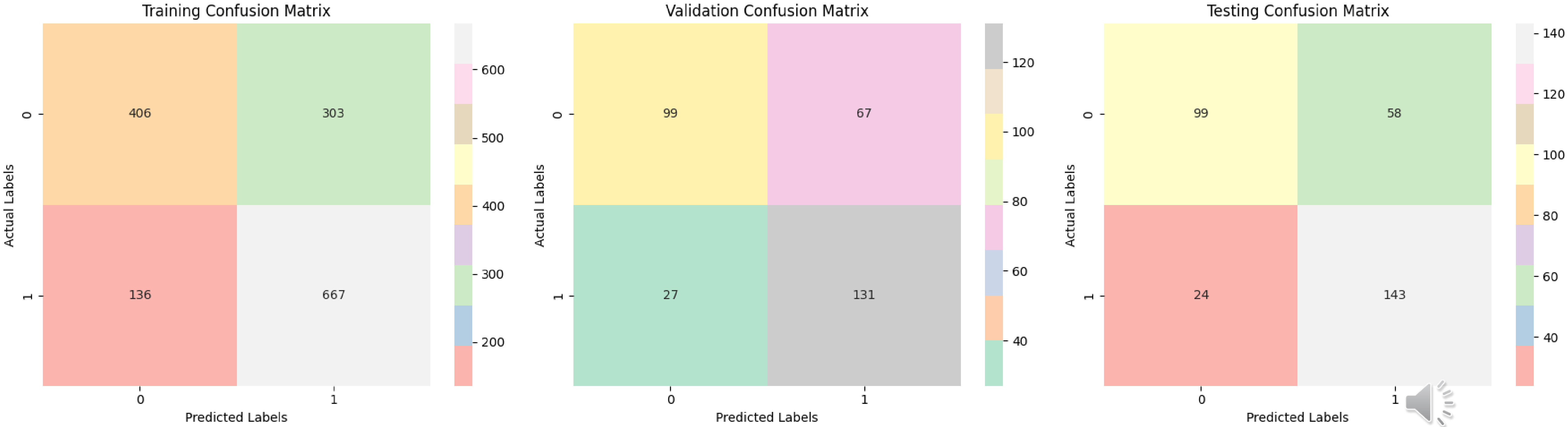


Model accuracy
Train
Validation
Accuracy
0.75
0.70
0.65
0.60
0.55
0.50
0
5
10
15
20
25
30
Epoch



Model loss
Train
Validation
Loss
1.4
1.2
1.0
0.8
0.6
0
5
10
15
20
25
30
Epoch

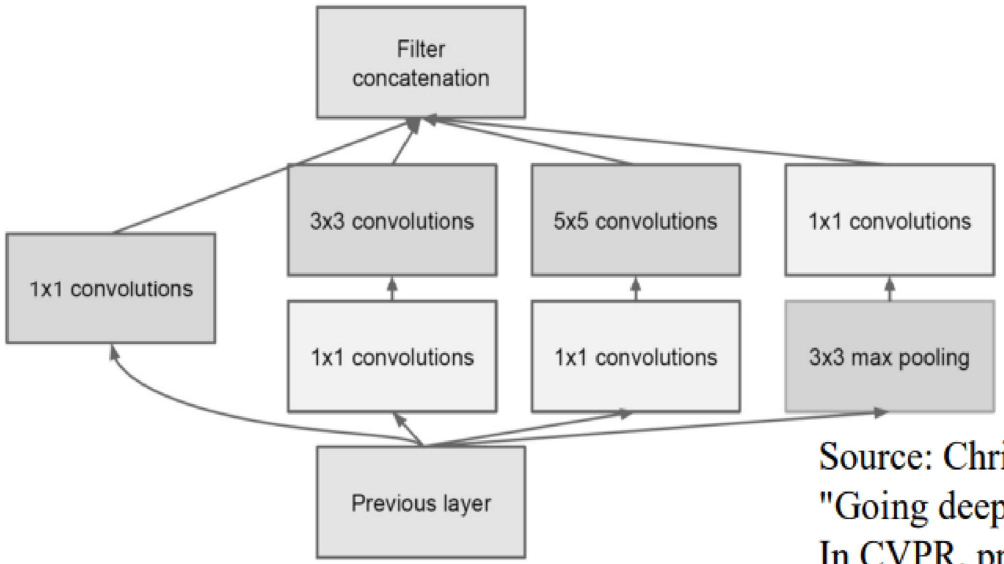
ResNet-50



Training Confusion Matrix
406
303
136
667
Actual Labels
Predicted Labels
Validation Confusion Matrix
99
67
27
131
Testing Confusion Matrix
99
58
24
143

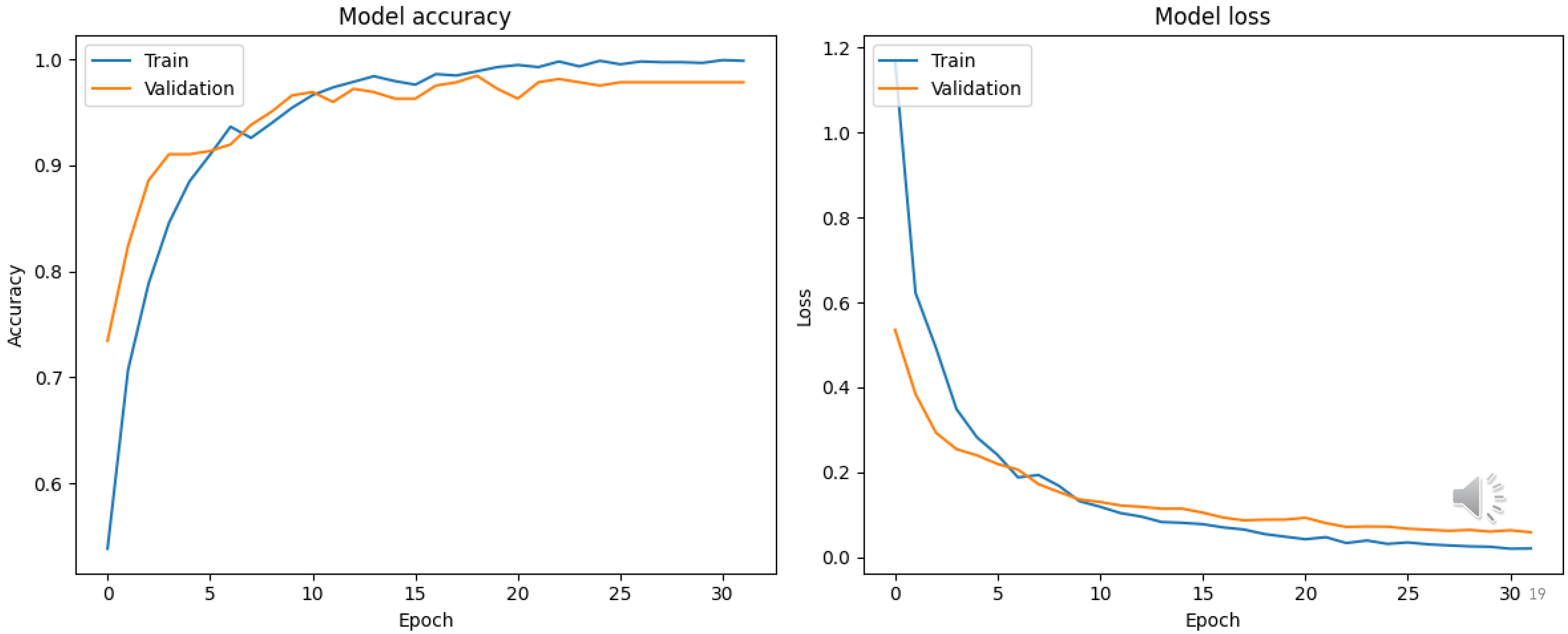
InceptionV3

- InceptionV3 powers the binary Inception model.
- Features extracted through convolutions and global pooling.
- prevents overfitting, dense layers with sigmoid for effective binary classification, especially in distinguishing two image classes.

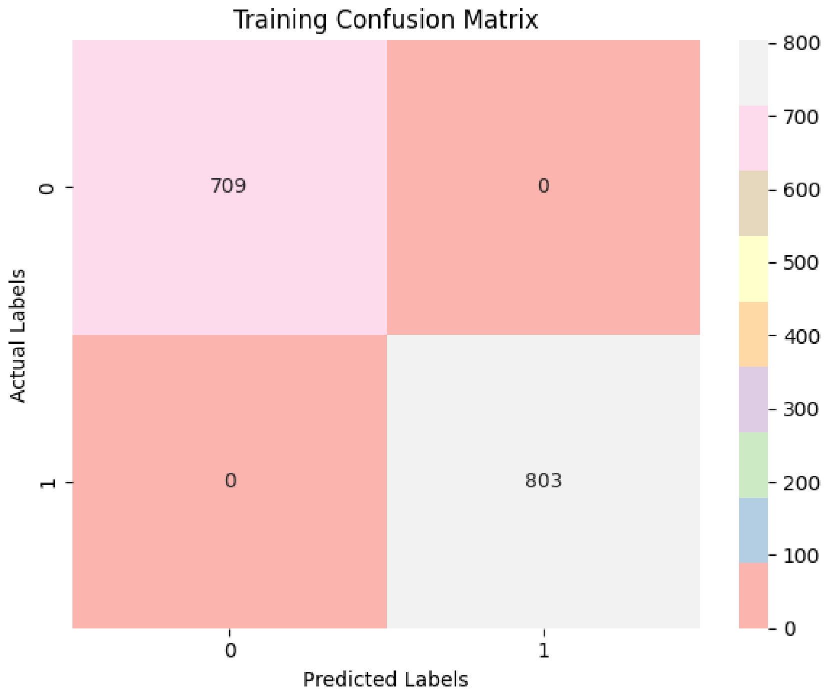


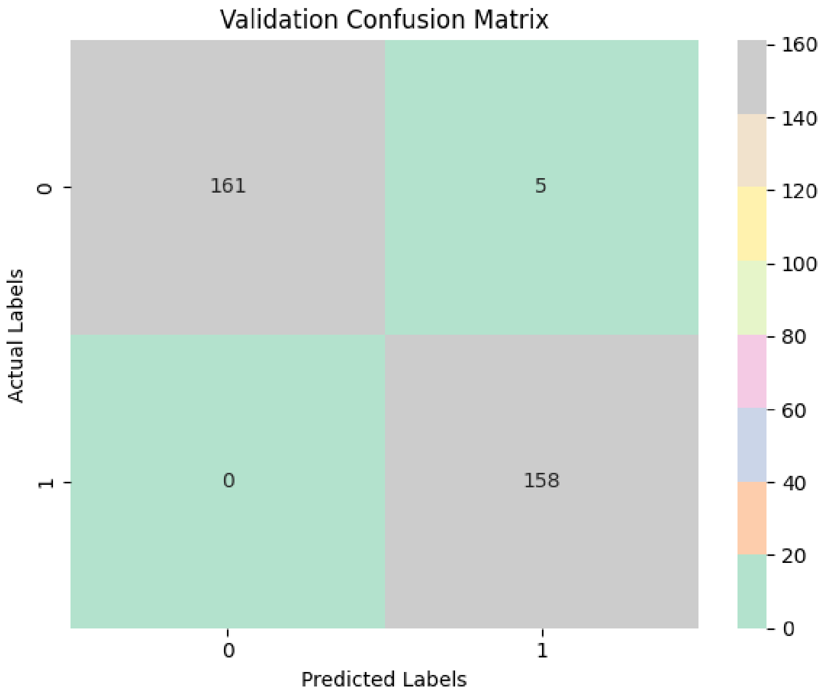
Source: Christian Szegedy et al., "Going deeper with convolutions." In CVPR, pp. 1-9. 2015.

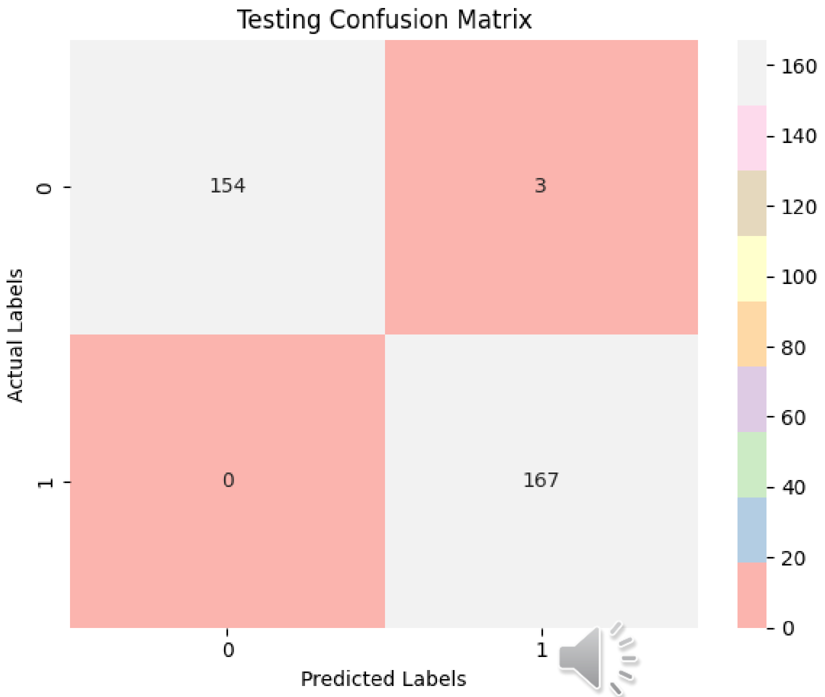
InceptionV3



InceptionV3

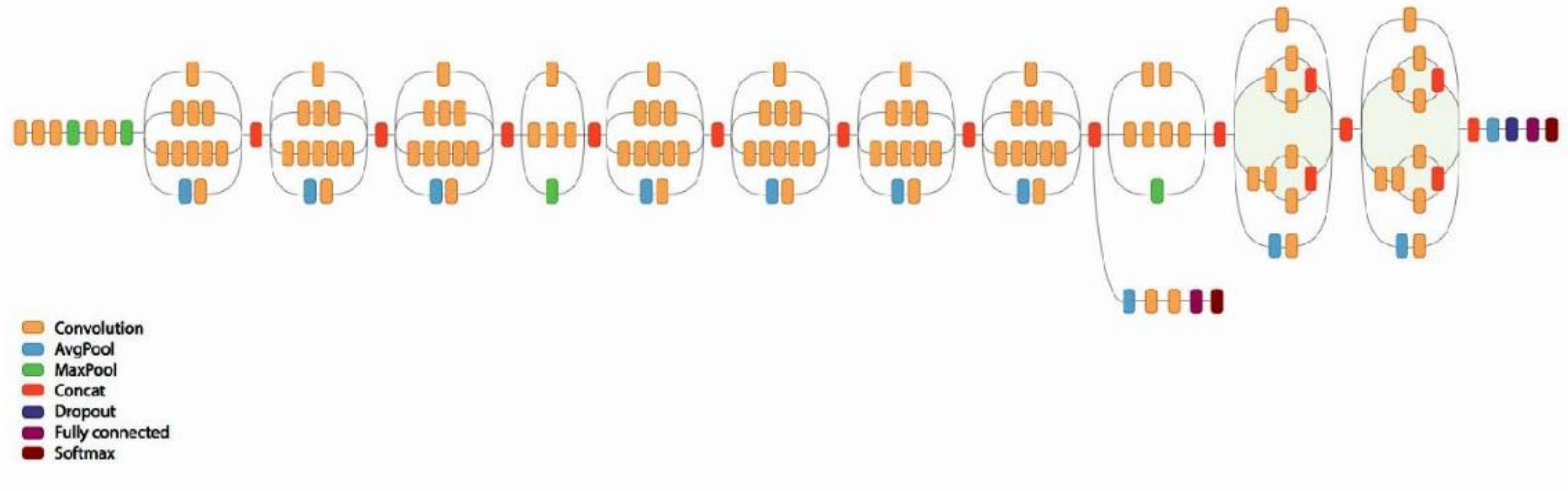




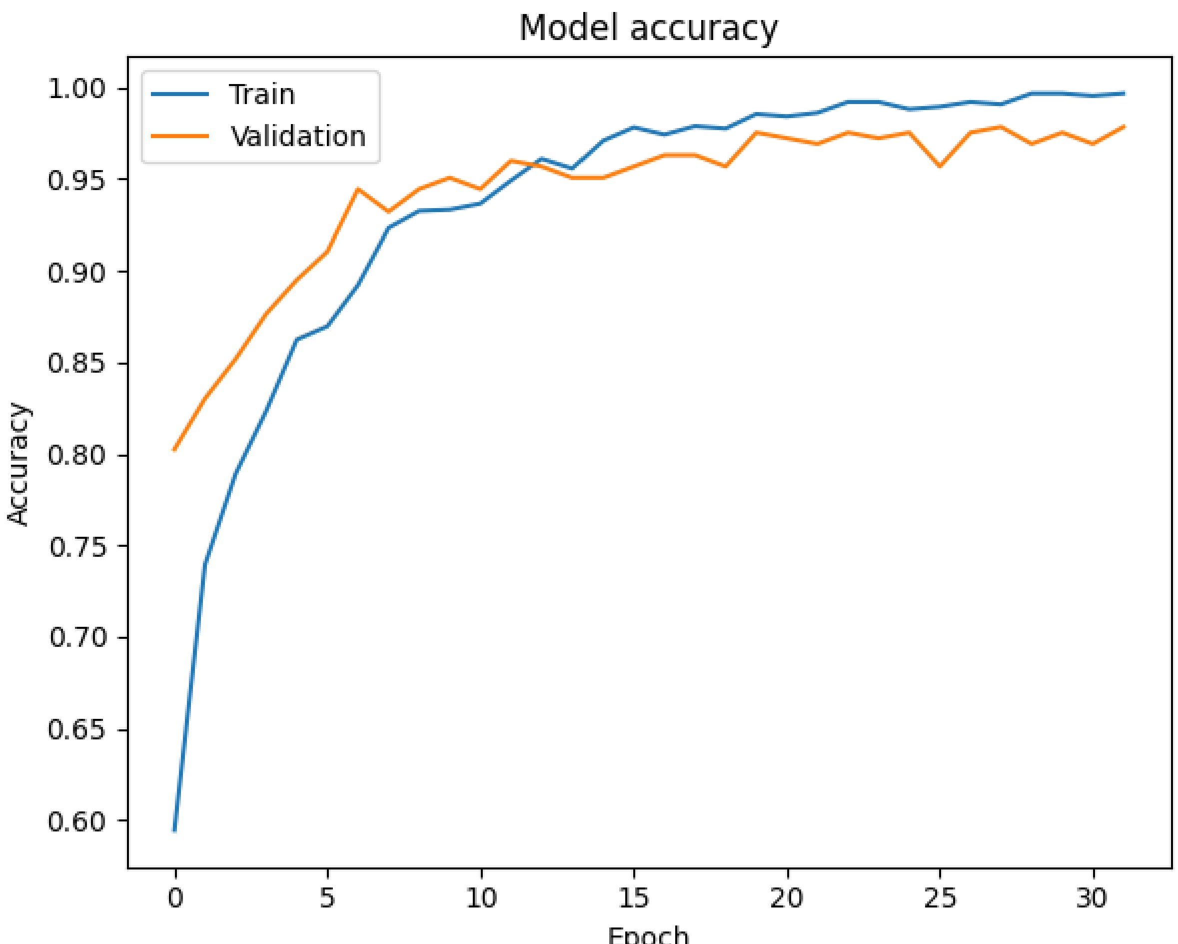


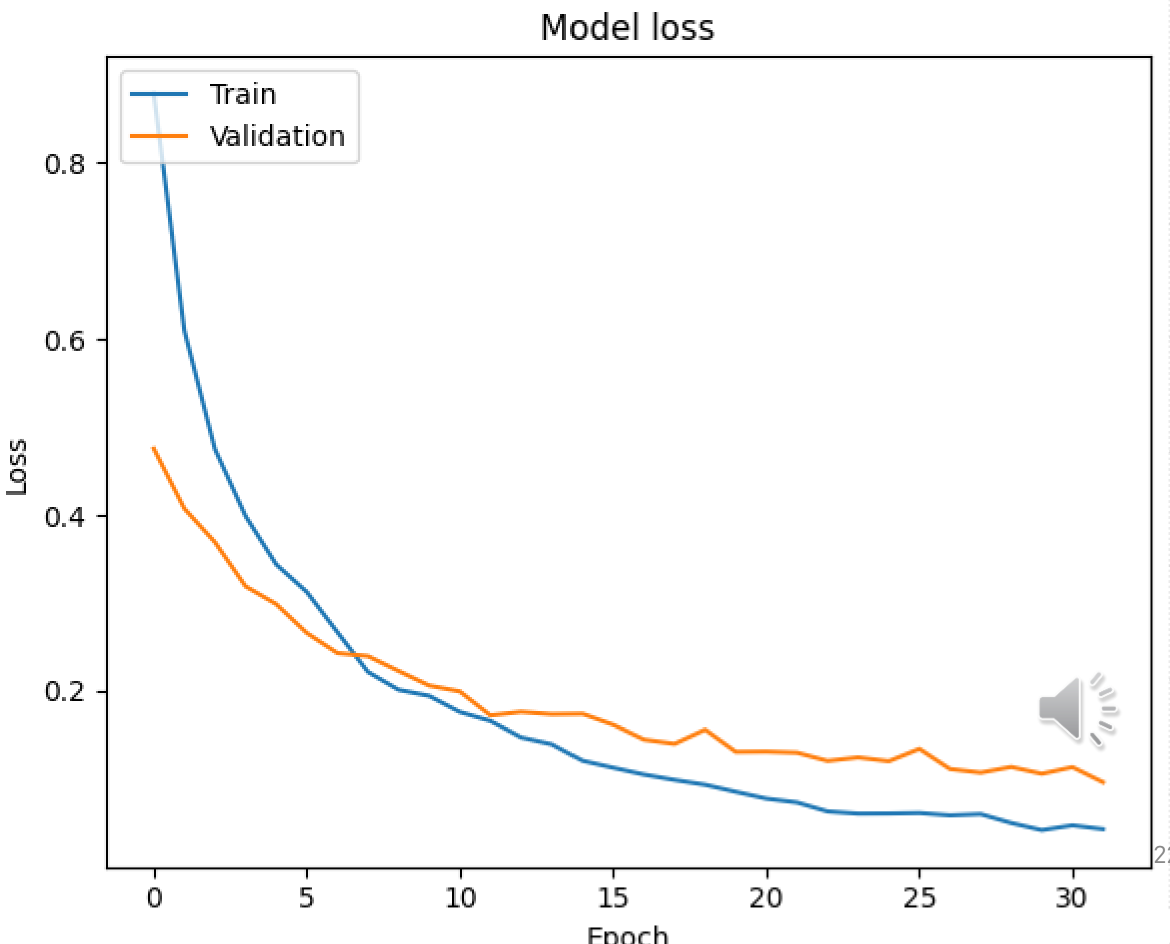
InceptionResnetV2

- It seamlessly integrates features from the Inception and ResNet models, leveraging advanced techniques to enhance accuracy and performance in complex visual recognition tasks.

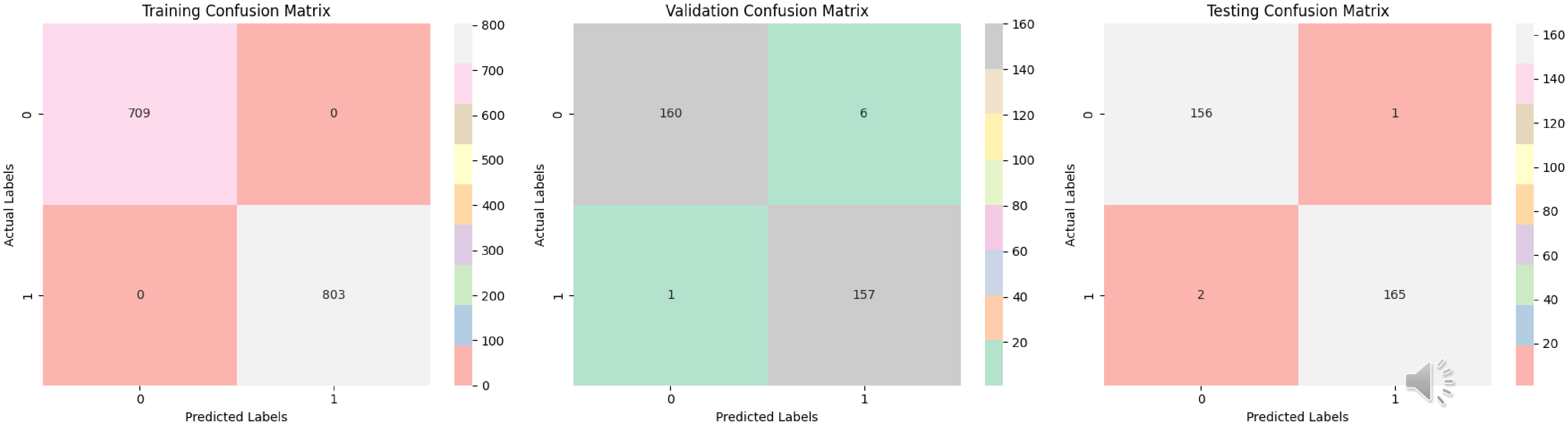


InceptionResnetV2



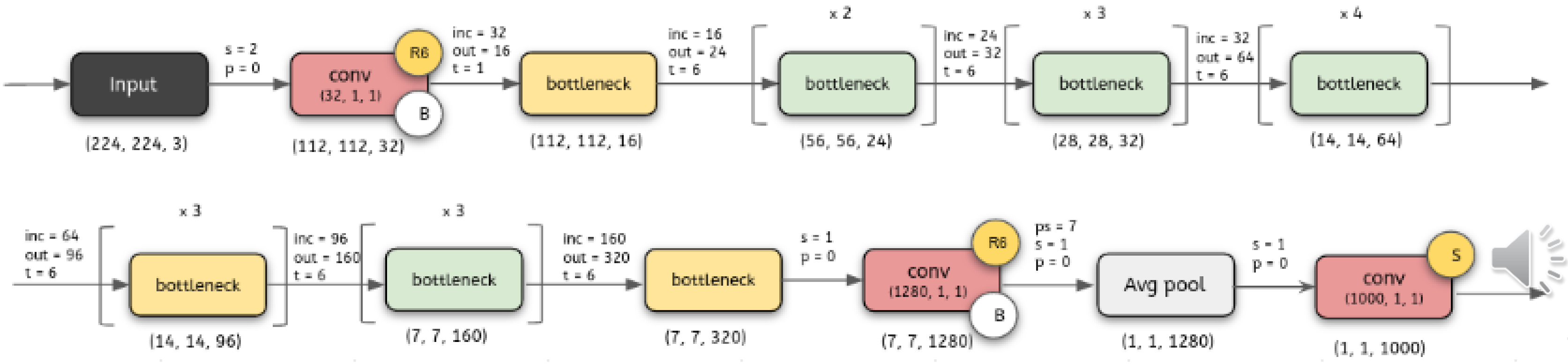


InceptionResnetV2

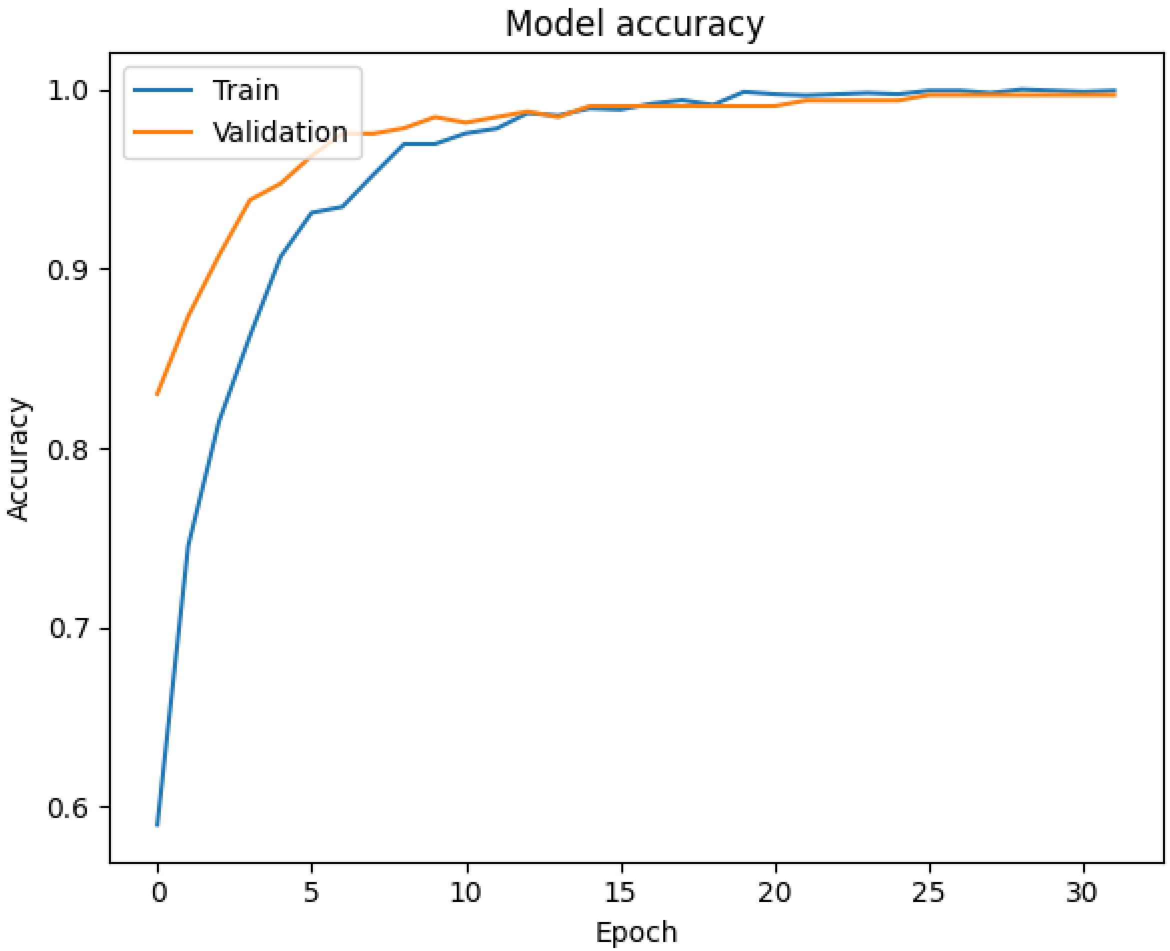


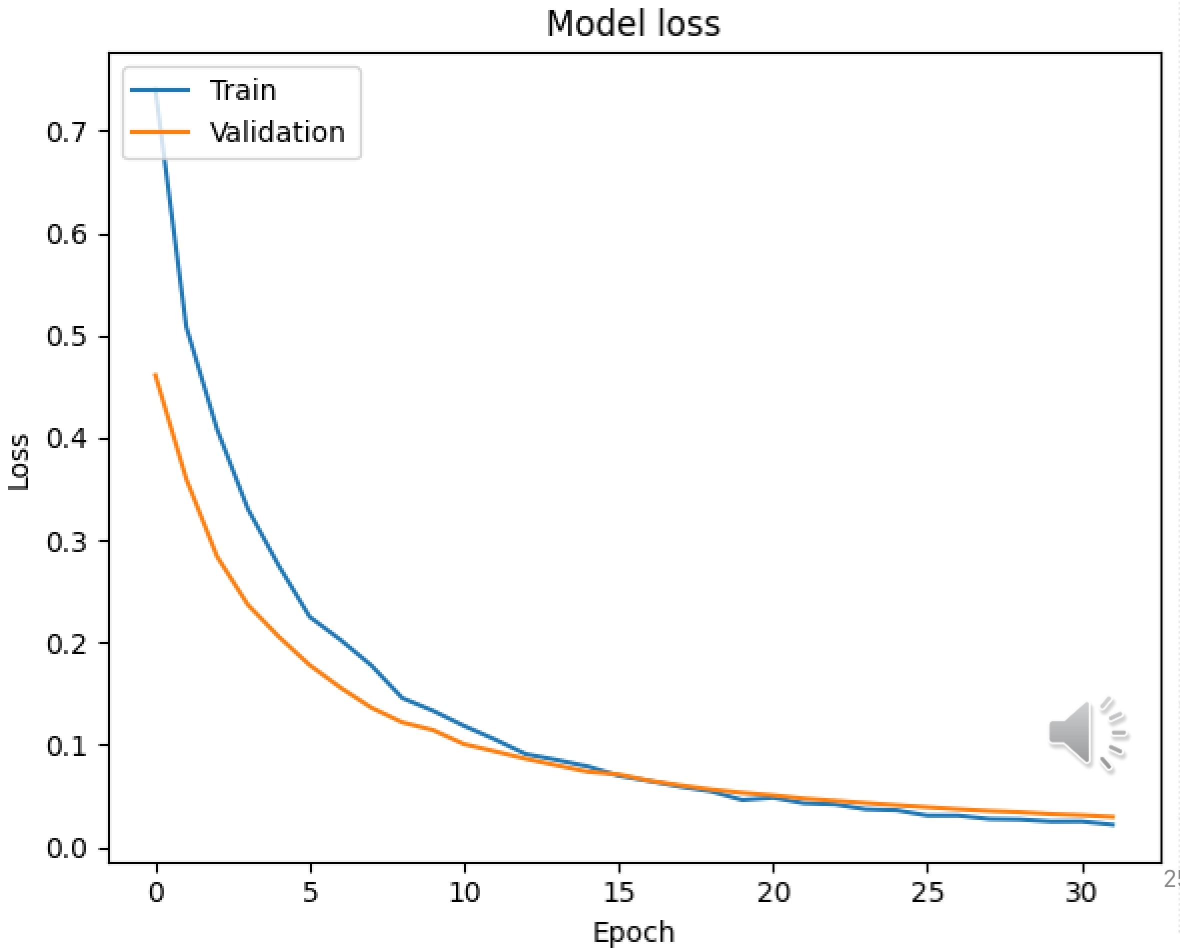
MobileNetV2

- MobileNetV2 is a lightweight model designed for mobile phones and embedded systems.
- It uses depth-wise separable convolutions for efficient yet accurate image recognition, making it a popular choice for real-time processing in resource-constrained devices.

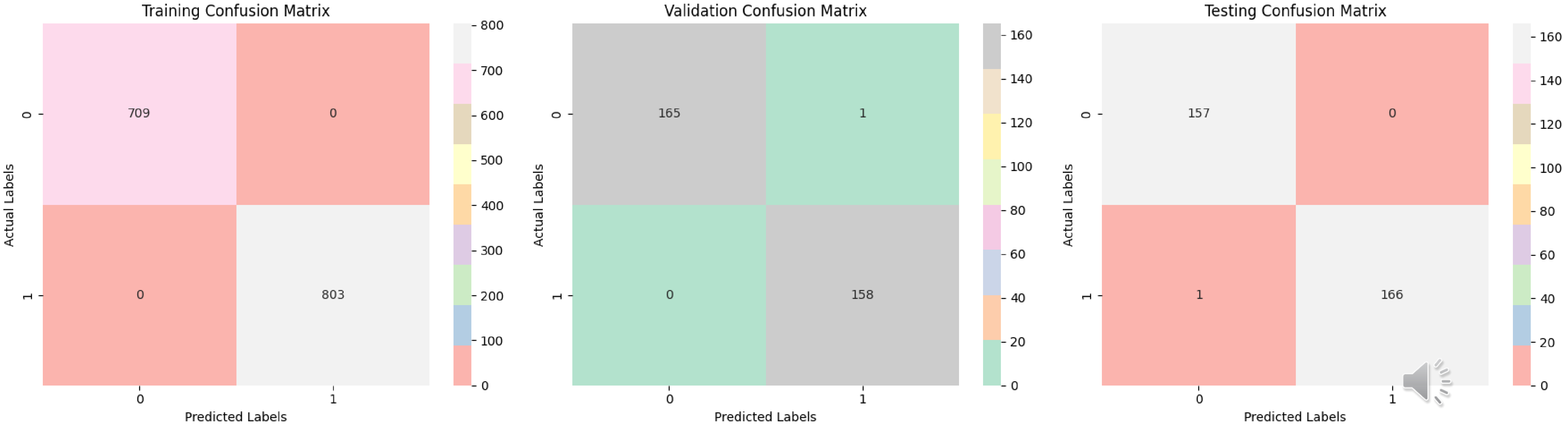


MobileNetV2





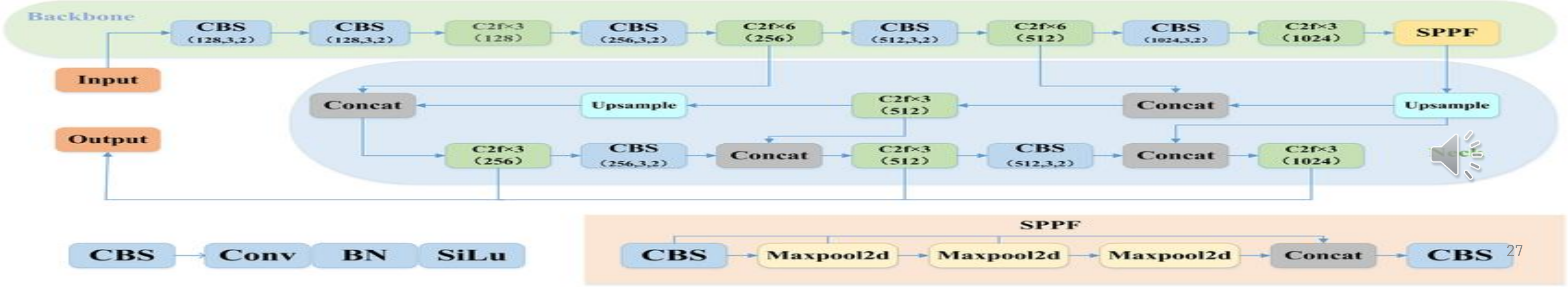
MobileNetV2



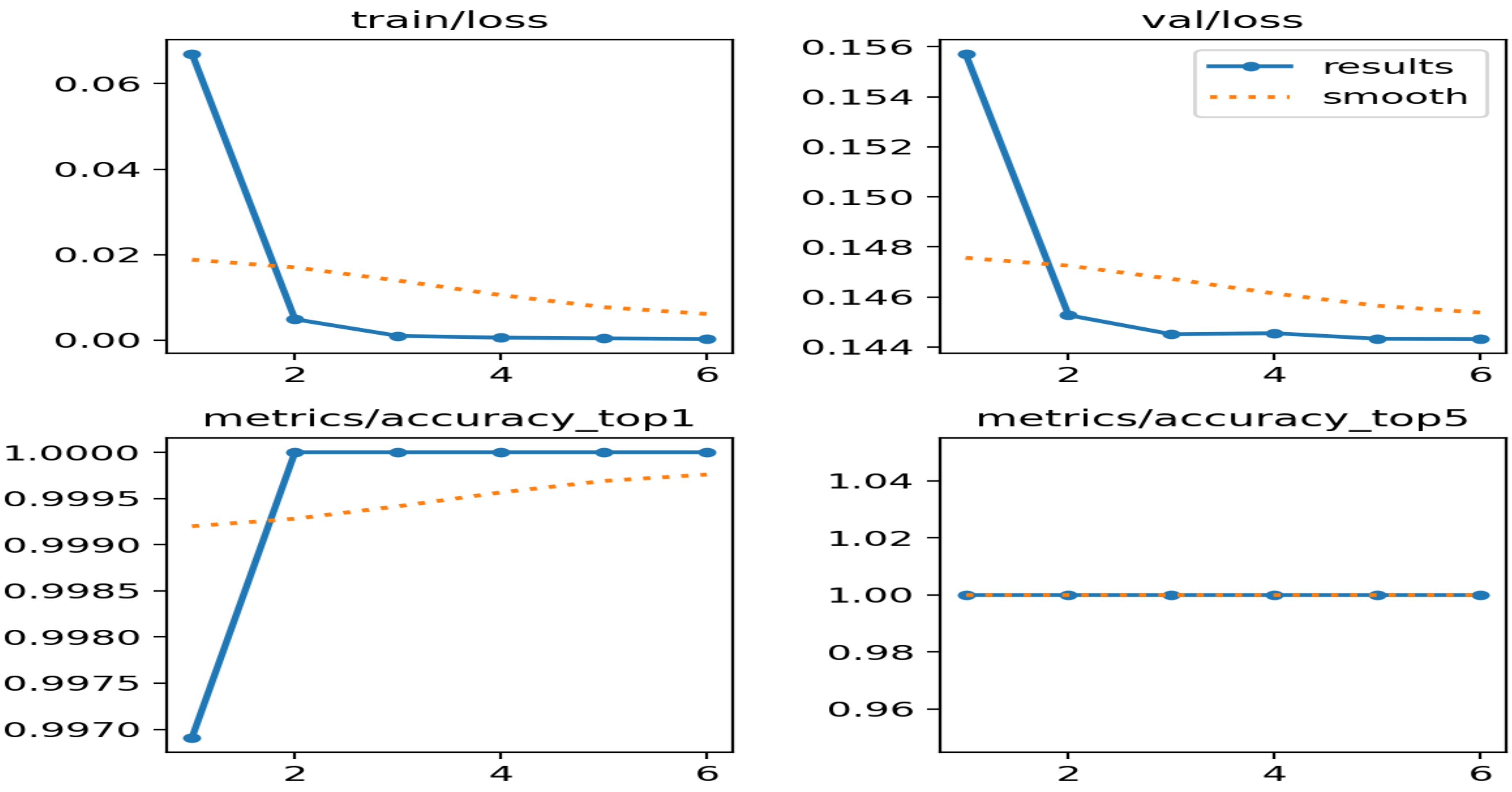
YoloV8n

- YOLO v8 is a cutting-edge binary classification model known for its speed and accuracy.

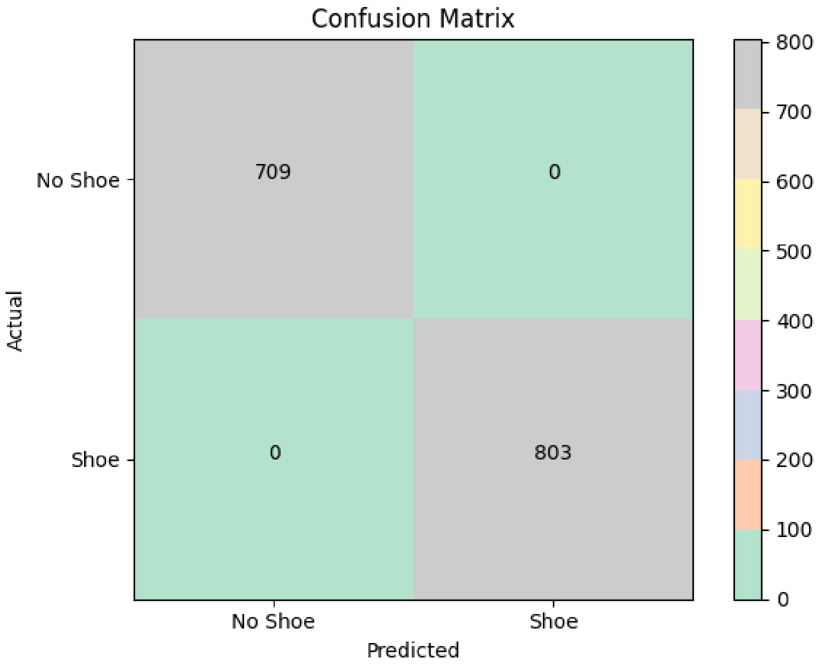
- It employs a single-pass network approach, dividing the image into a grid to simultaneously detect and classify objects, enabling real-time performance in applications like object recognition

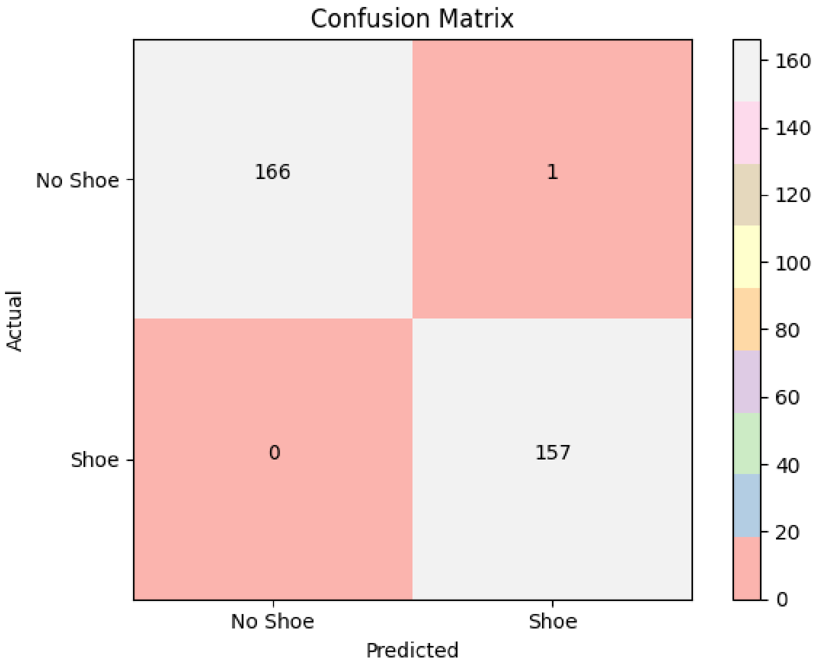


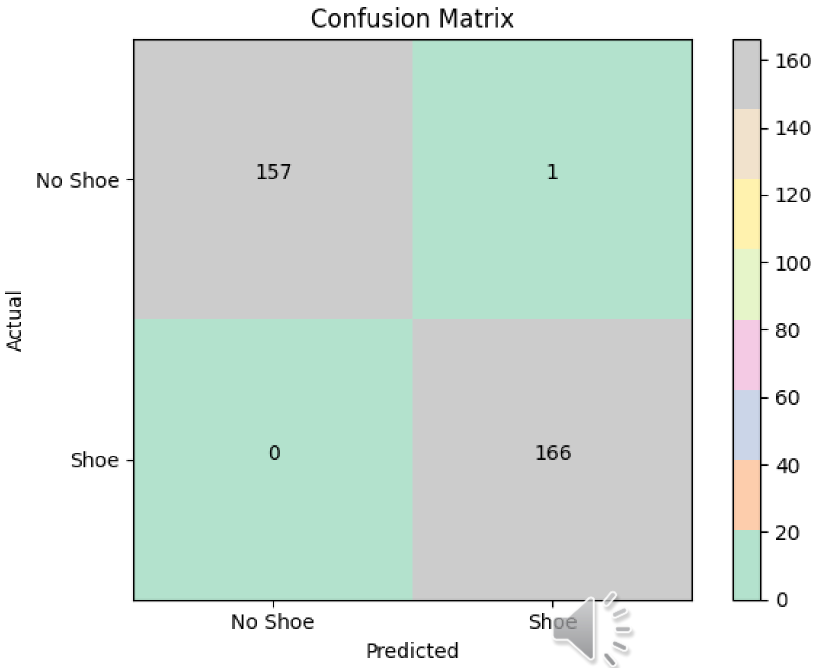
YoloV8n


train/loss
0.06
0.04
0.02
0.00
2
4
6
val/loss
results
smooth
0.156
0.154
0.152
0.150
0.148
0.146
0.144
2
4
6
metrics/accuracy_top1
1.0000
0.9995
0.9990
0.9985
0.9980
0.9975
0.9970
2
4
6
metrics/accuracy_top5
1.04
1.02
1.00
0.98
0.96
2
4
6

YoloV8n


Confusion Matrix
No Shoe
Shoe
709
0
0
803
Actual
Predicted
800
700
600
500
400
300
200
100
0


Confusion Matrix
No Shoe
Shoe
166
1
0
157
Actual
Predicted
160
140
120
100
80
60
40
20
0


Confusion Matrix
No Shoe
Shoe
157
1
0
166
Actual
Predicted
160
140
120
100
80
60
40
20
0

Evaluation

Models Performance Summary (Shoe [14])

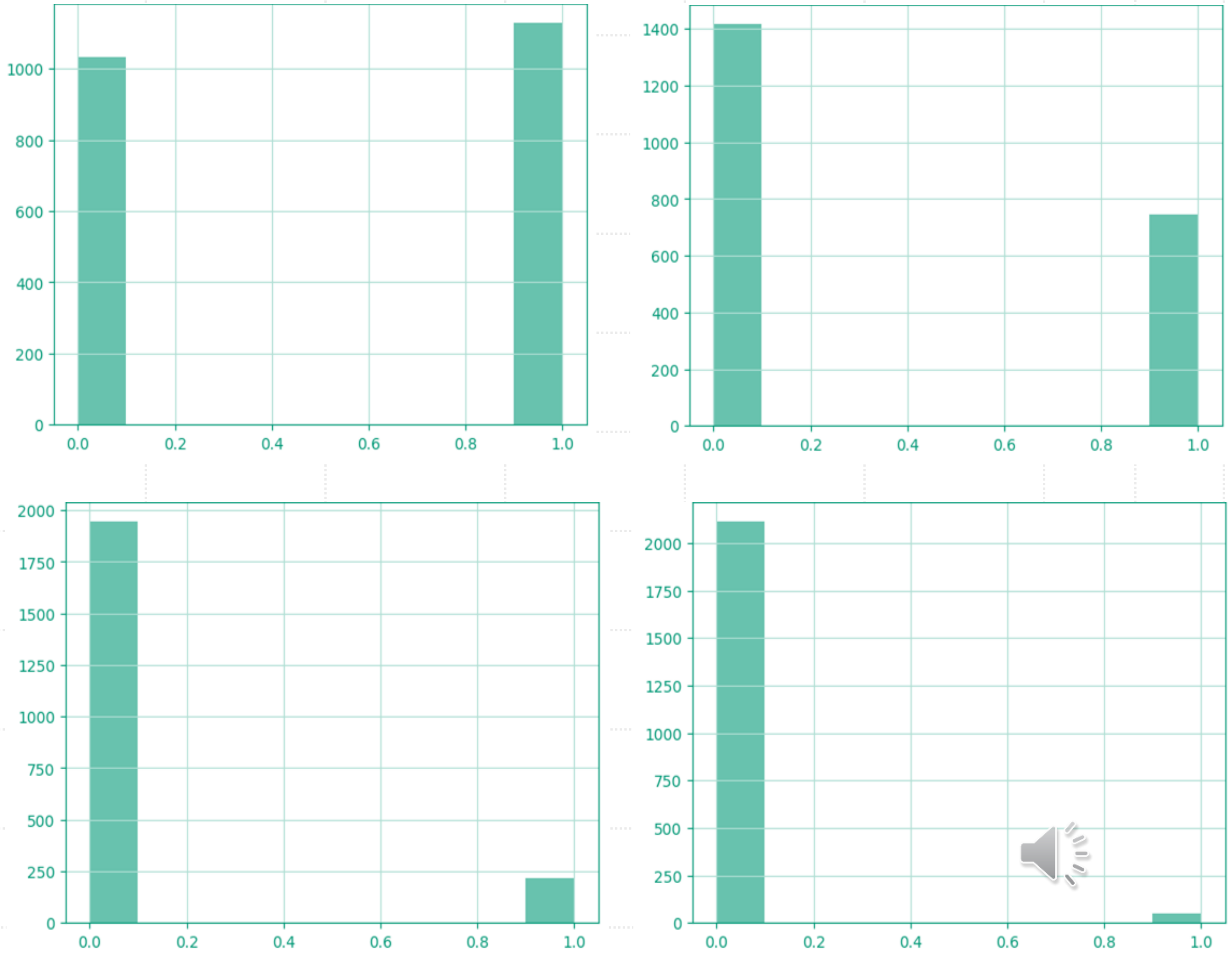
			ResNet50		Inception		InceptionResNet V2		MobileNet V2		Yolov8n		Yolov8s	
Testing	F1-Score		70.92%		99.39%		99.7%		99.7%		100%		100%	
	Accuracy		71.91%		99.38%		99.69%		99.69%		100%		100%	
	Loss		0.567		0.01528		0.0268		0.025		0.0023		0.0017	
	TN	FP	122	35	157	0	157	0	156	1	157	0	157	0
	FN	TP	56	111	2	165	1	166	0	167	0	167	0	167
	Epoch Till Early Stop (Out of 32)		32		20		32		32		6		6	

Testing On Different Items

Shoe 1160	Toy 760	Bag 250	Mouse 48

Out of 2160 Image in total

Various Selected Objects for Binary Classification



Models Performance Summary (Toy [16])

			ResNet50		Inception		InceptionResNetV2		MobileNetV2		Yolov8n	
Testing	F1-Score		23.07%		99.53%		97.2%		98.14%		100%	
	Accuracy		69.13%		99.69%		98.14%		97.24%		100%	
	Loss		0.57		0.005		0.042		0.064		0.01158	
	TN	FP	209	8	217	1	214	3	213	4	217	0
	FN	TP	92	18	1	106	3	104	3	104	0	107
	Epoch Till Early Stop (Out of 32)		32		29		32		32		6	

Models Performance Summary (Bag [1])

			ResNet50		Inception		InceptionResNetV2		MobileNetV2		Yolov8n	
Testing	F1-Score		11.42%		100%		95.23%		93.5%		100%	
	Accuracy		90.43%		100%		99.07%		98.7%		100%	
	Loss		0.234		0.003		0.0197		0.022		0.00028	
	TN	FP	219	0	291	0	291	0	291	0	291	0
	FN	TP	25	3	0	33	3	30	4	29	0	33
	Epoch Till Early Stop (Out of 32)		31		29		32		32		6	

Models Performance Summary (Mouse [12])

			ResNet50		Inception		InceptionResNetV2		MobileNetV2		Yolov8n	
Testing	F1-Score		0%		100%		100%		100%		100%	
	Accuracy		97.53%		100%		100%		100%		100%	
	Loss		0.1		0.00008		0.00016		0.002		0.00010	
	TN	FP	316	0	316	0	316	0	316	0	316	0
	FN	TP	8	0	0	8	0	8	0	8	0	8
	Epoch Till Early Stop (Out of 32)		21		32		32		32		6	

Ablation Study

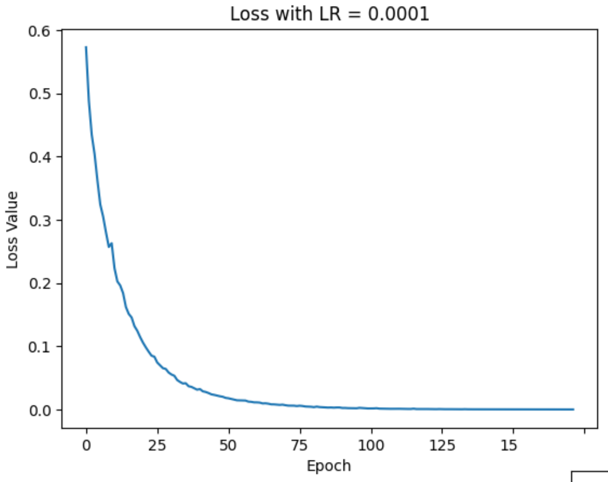
Ablation Study

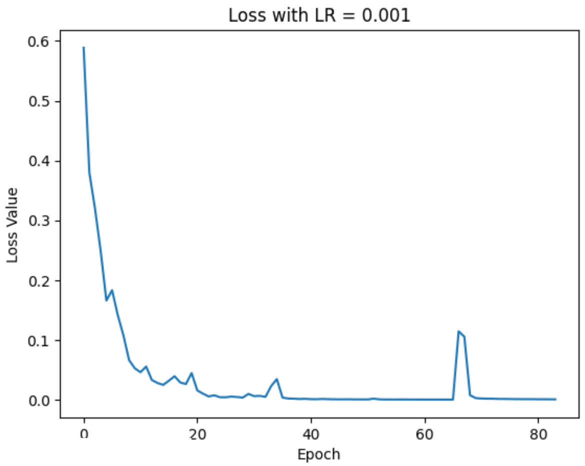
Trying a different learning rates for MobileNetV2 Model.

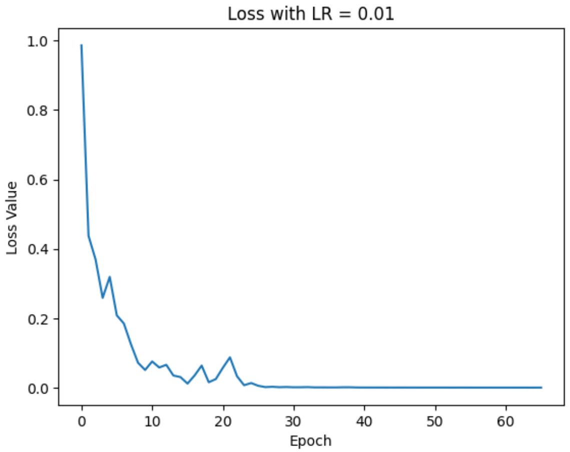
Keep all the conditions the same, but only change the learning rate (0.5, 0.1, 0.01, 0.001).

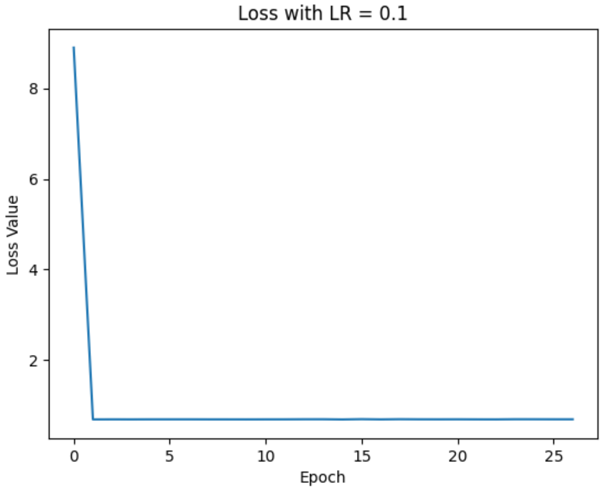
With use of early stop to save running time and considering it in our study.

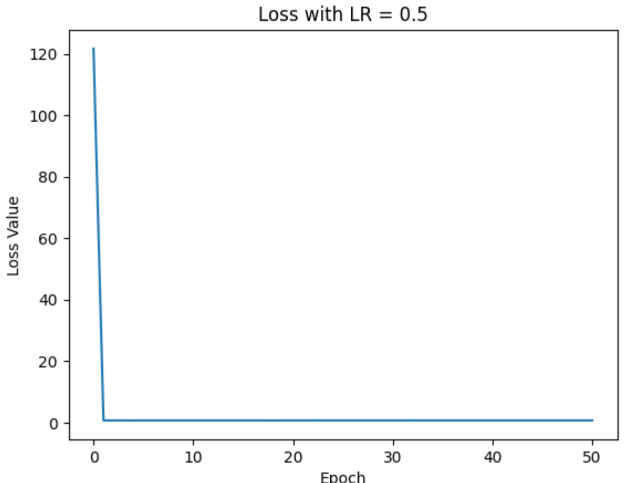
Ablation Study



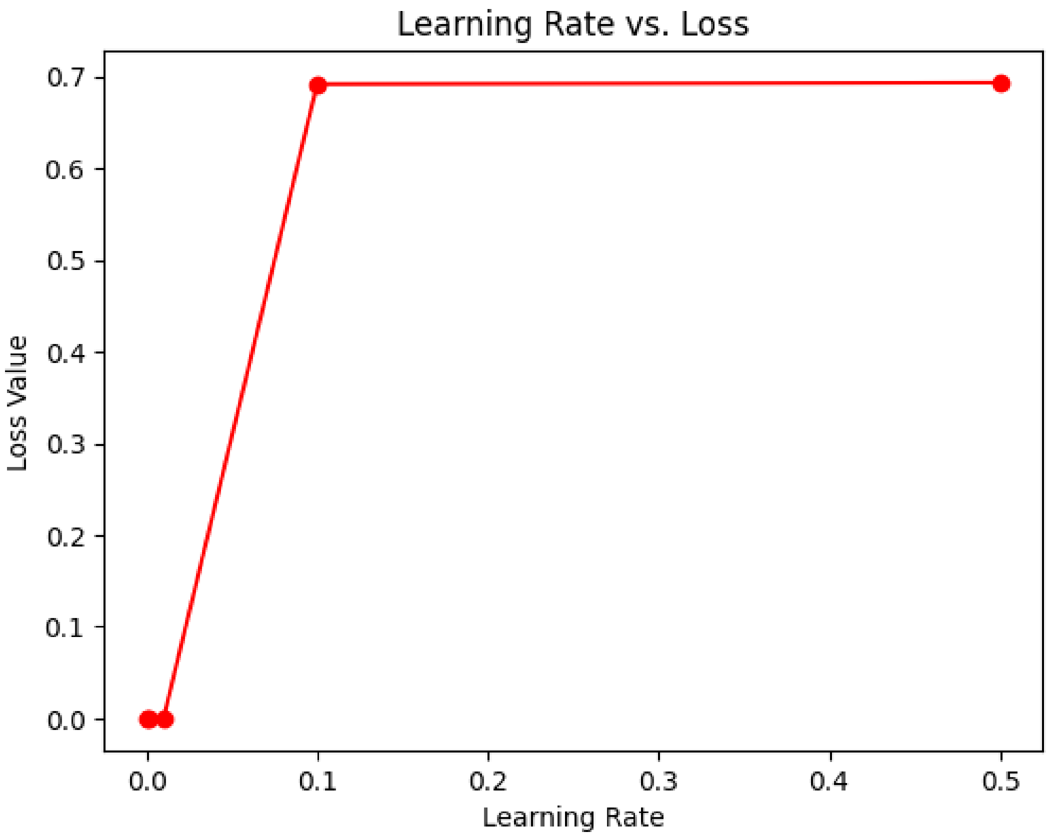


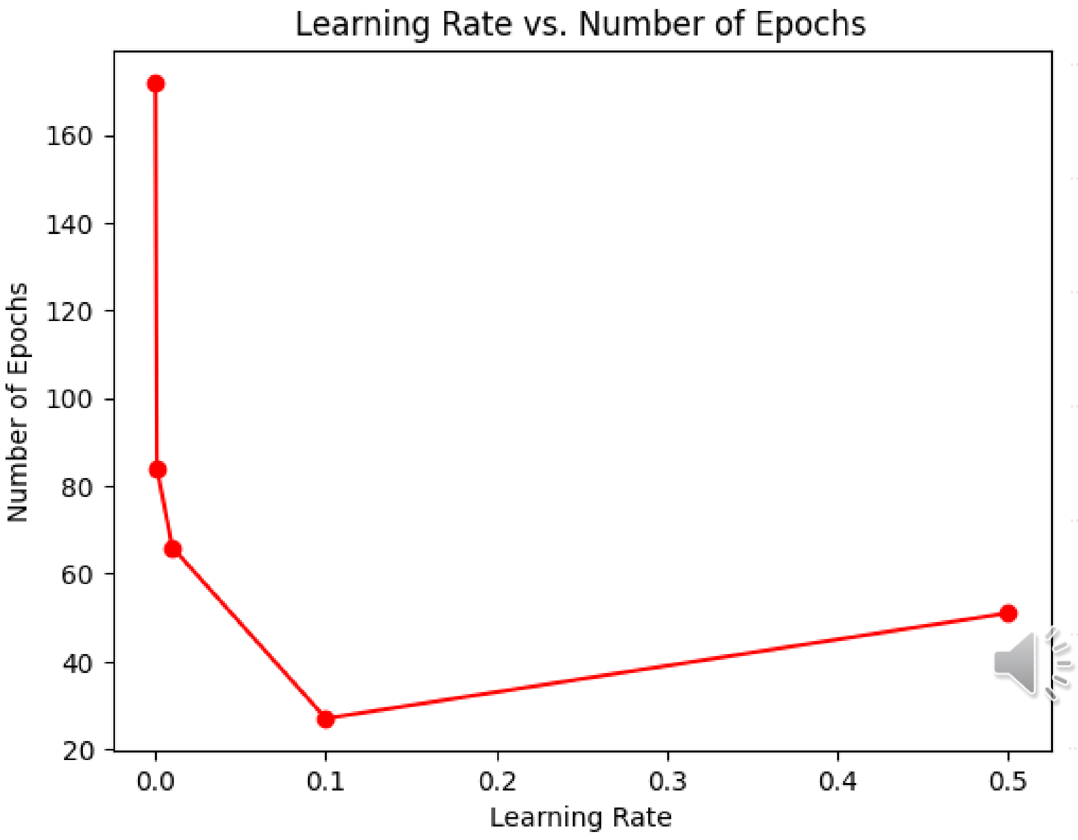






Ablation Study

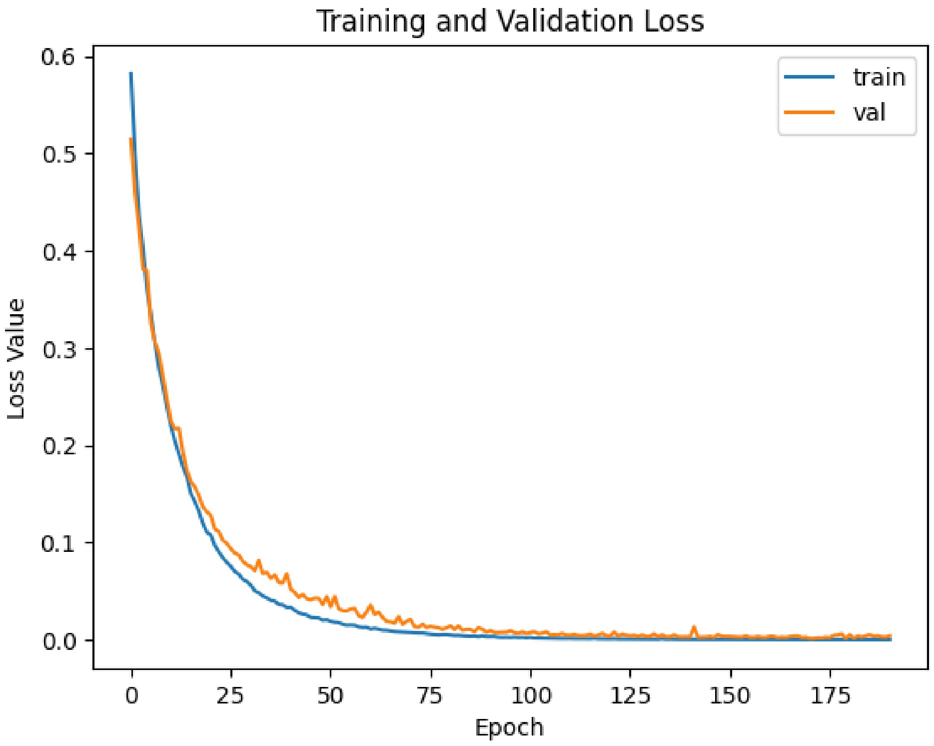


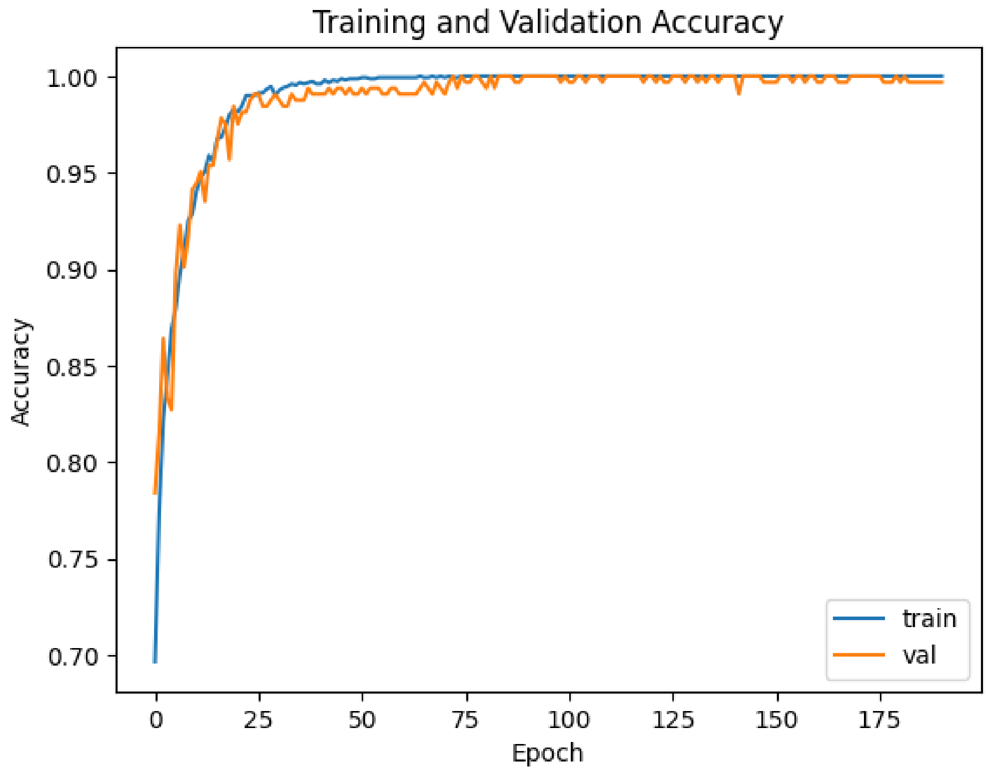


Ablation Study

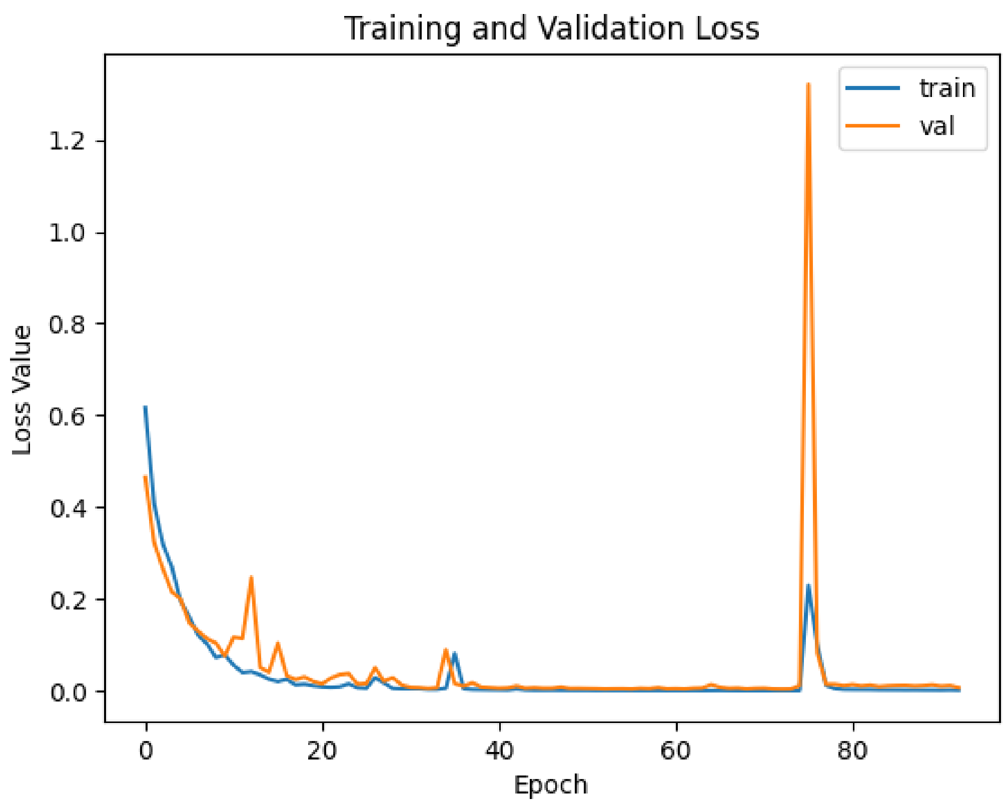
LR	Epochs Till Early Stop	Lowest loss
0.0001	172	6.67 e-05
0.001	84	25.1 e-0.5
0.01	66	9.5 e-05
0.1	27	69143 e-05
0.5	51	69318 e-05

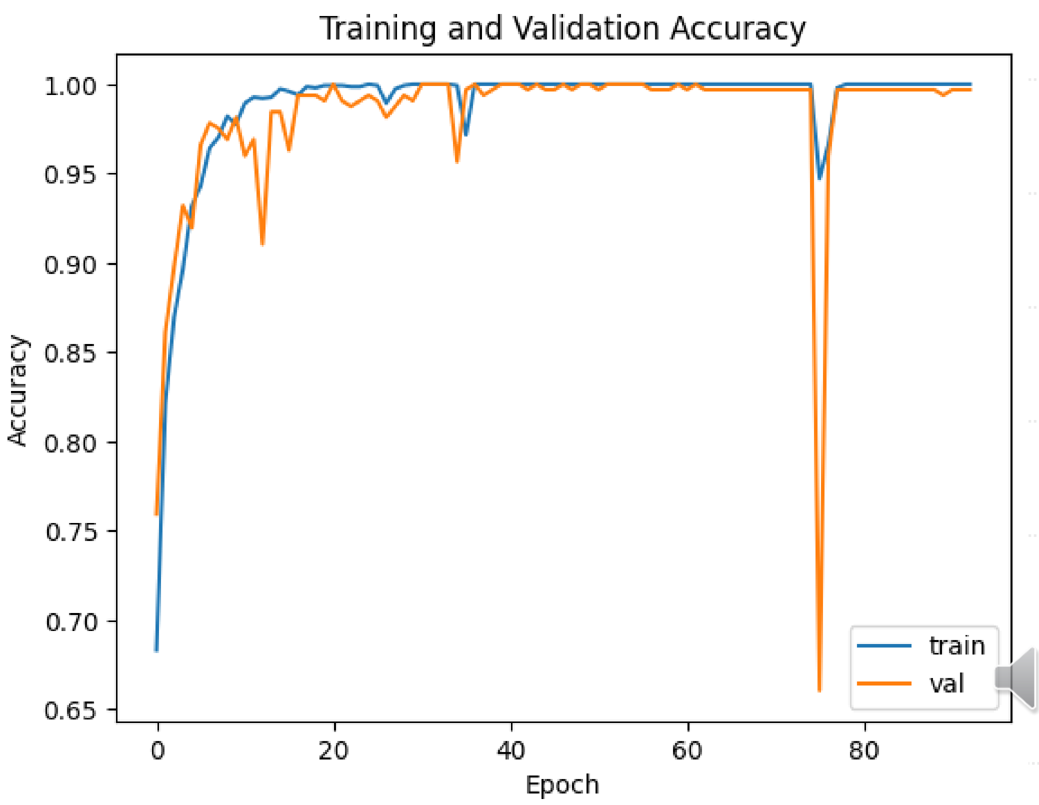
Learning Rate = 0.0001



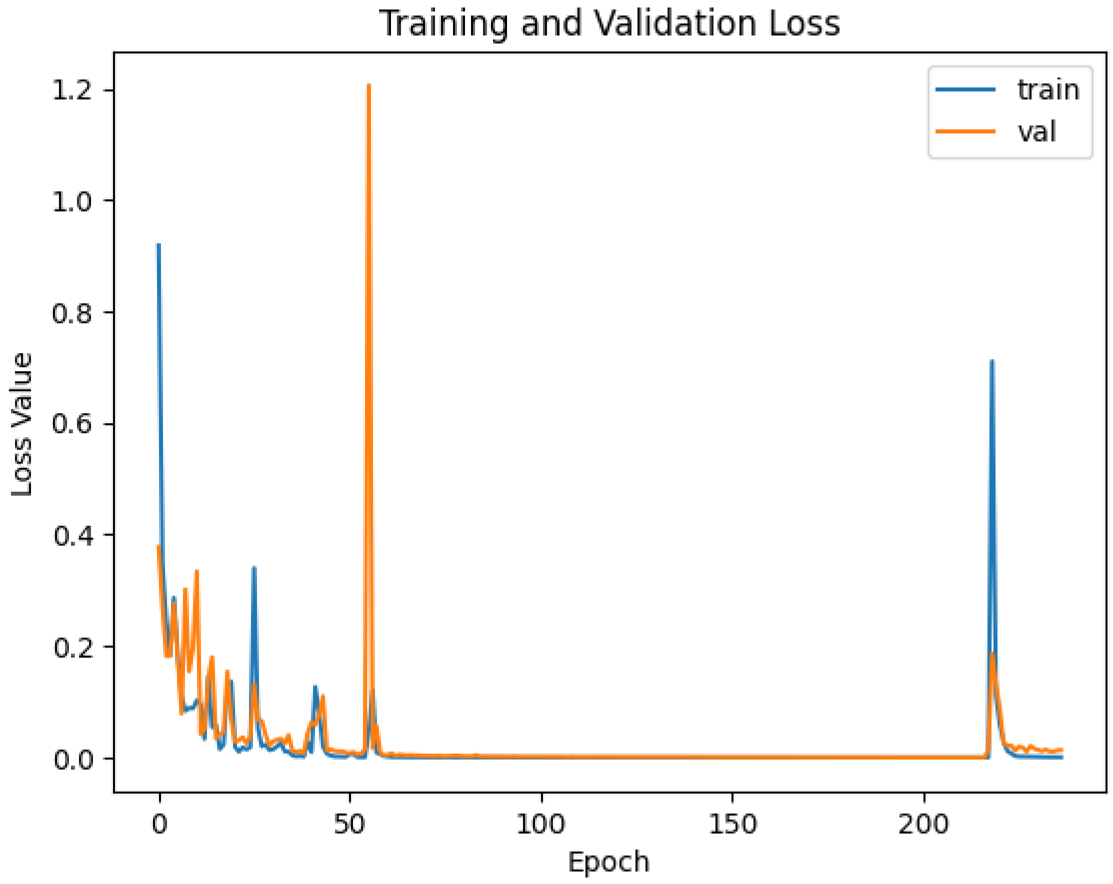


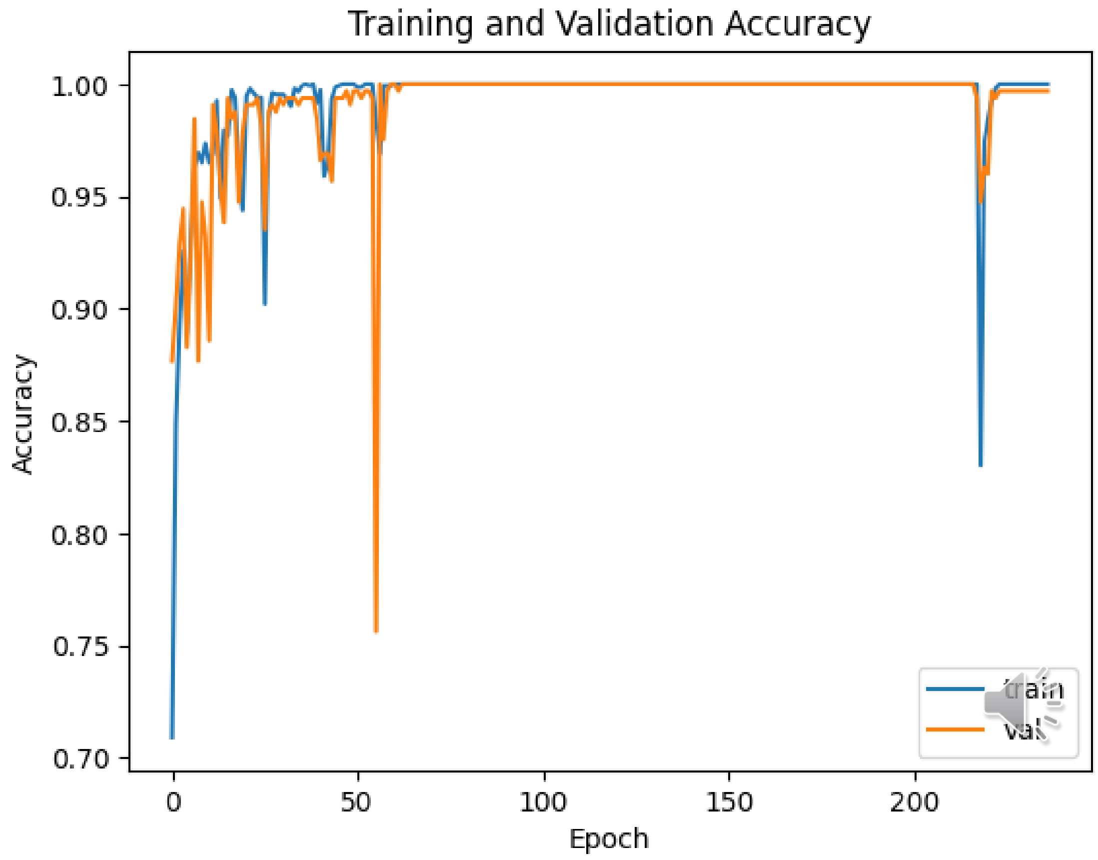
Learning Rate = 0.001



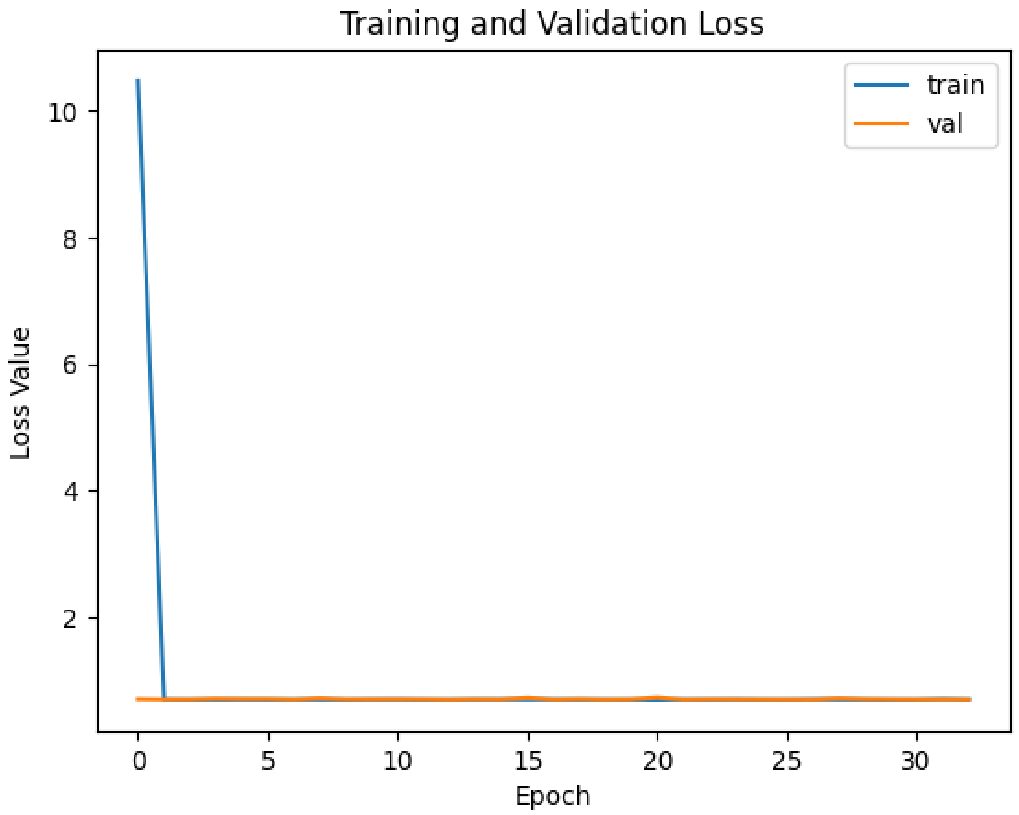


Learning Rate = 0.01

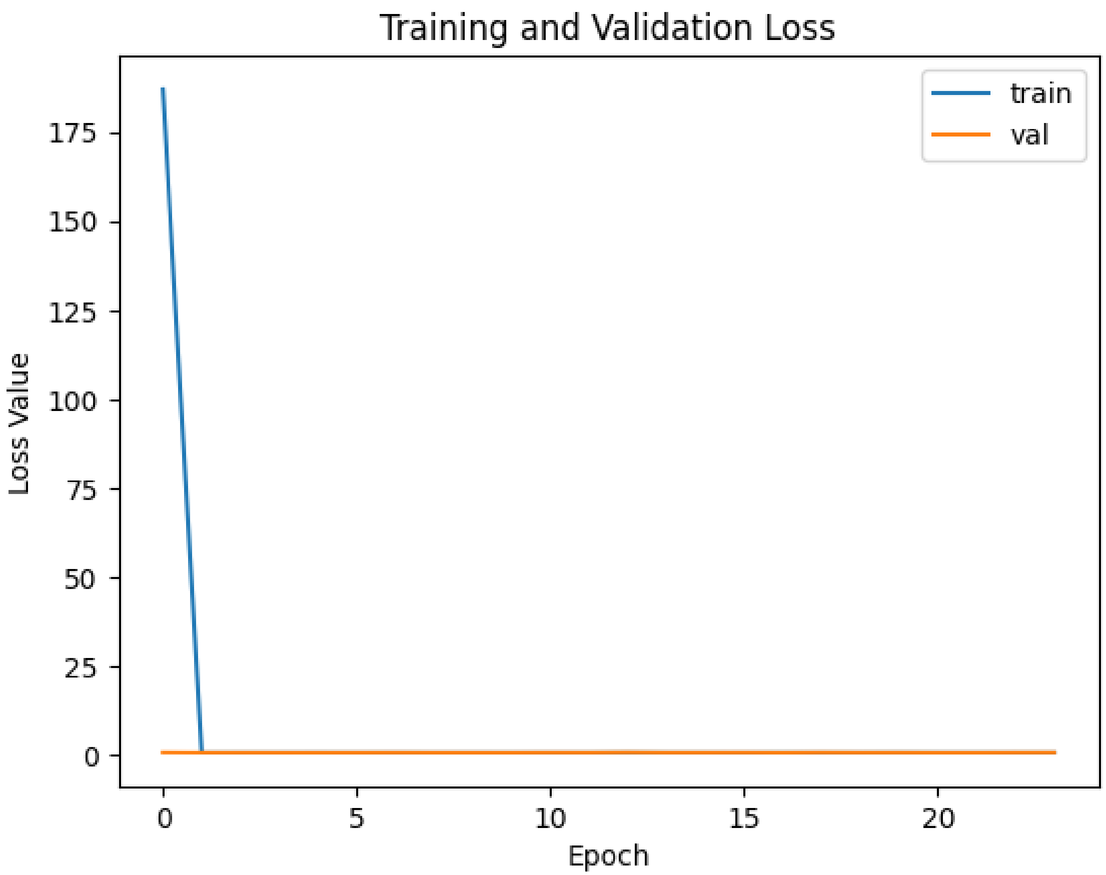


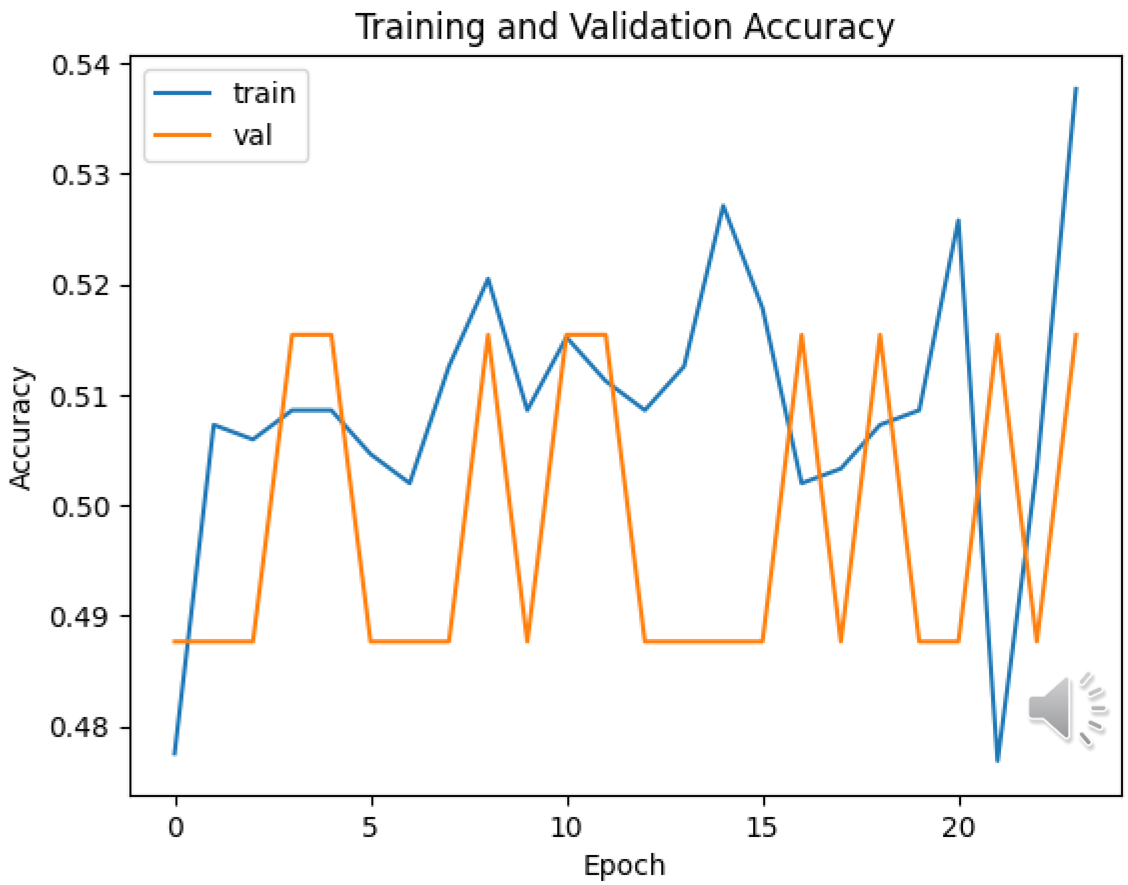


Learning Rate = 0.1

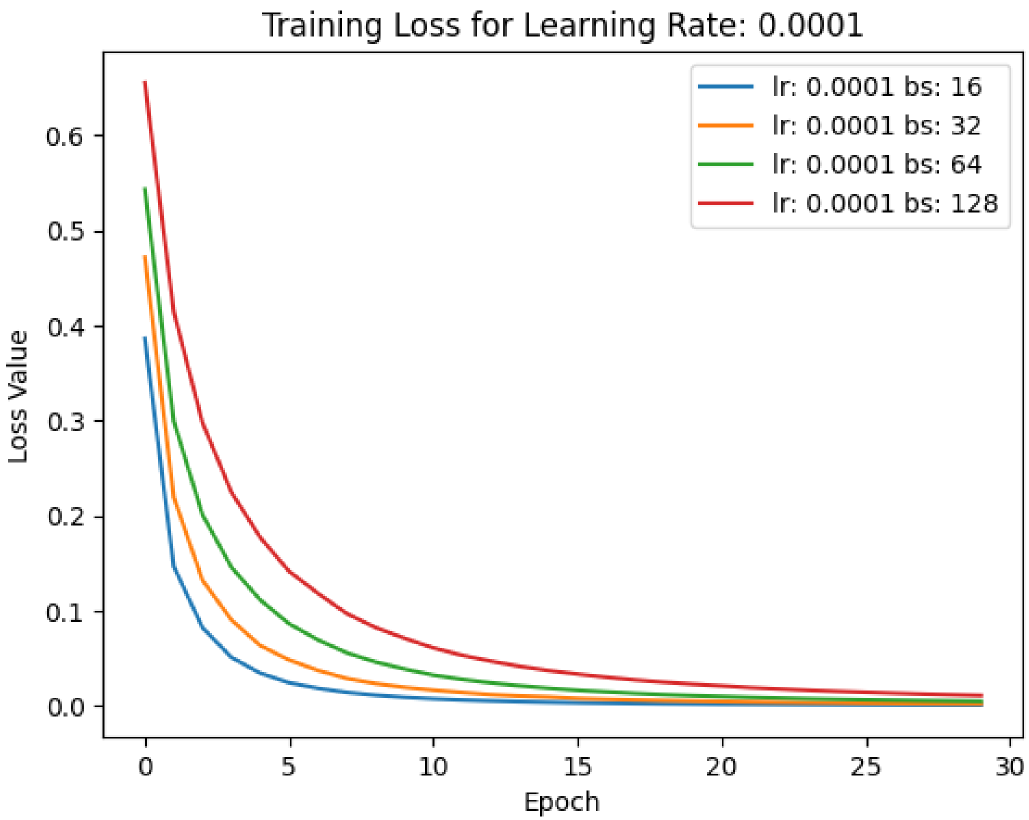


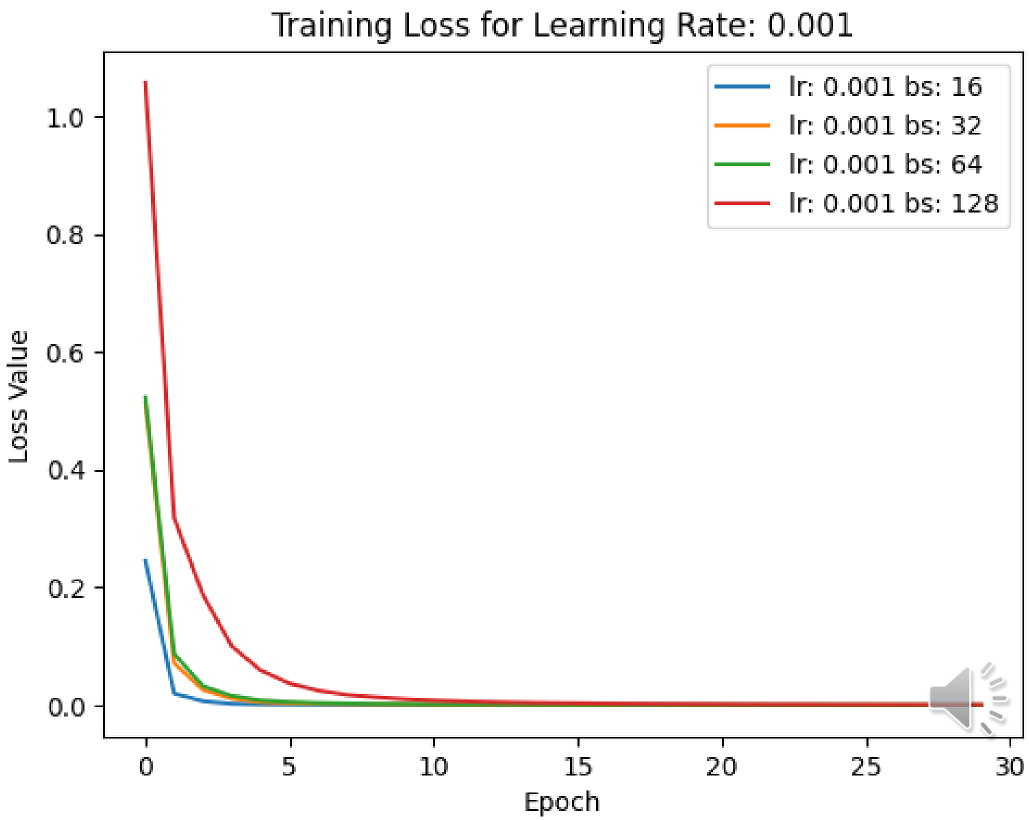
Learning Rate = 0.5



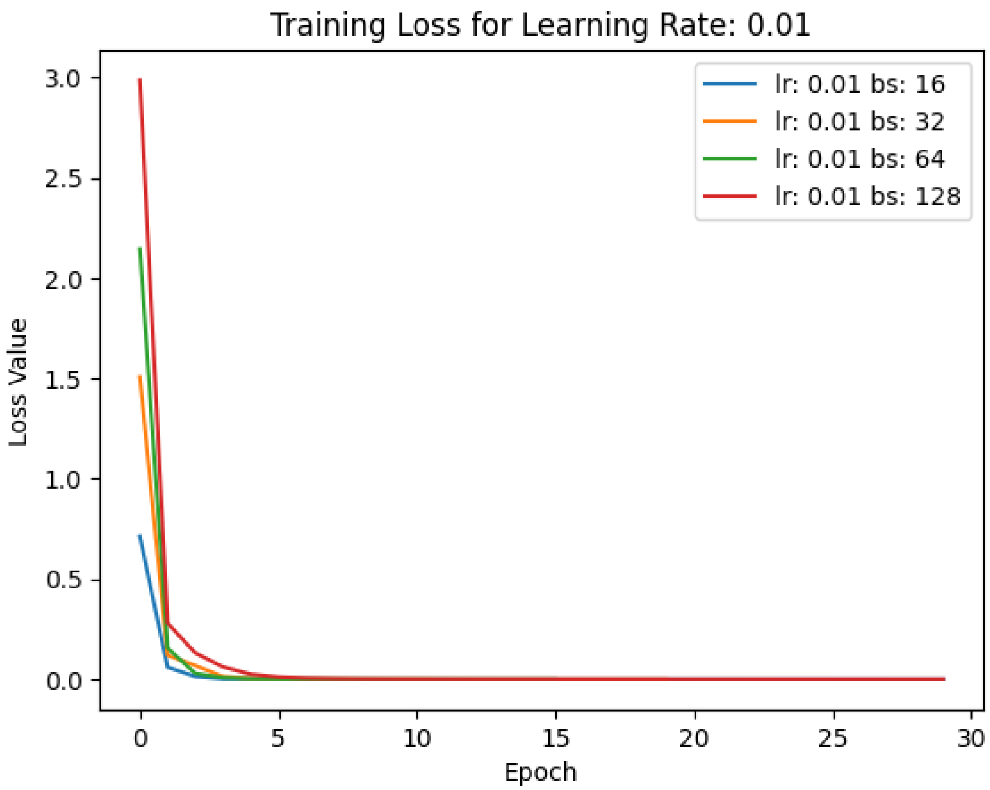


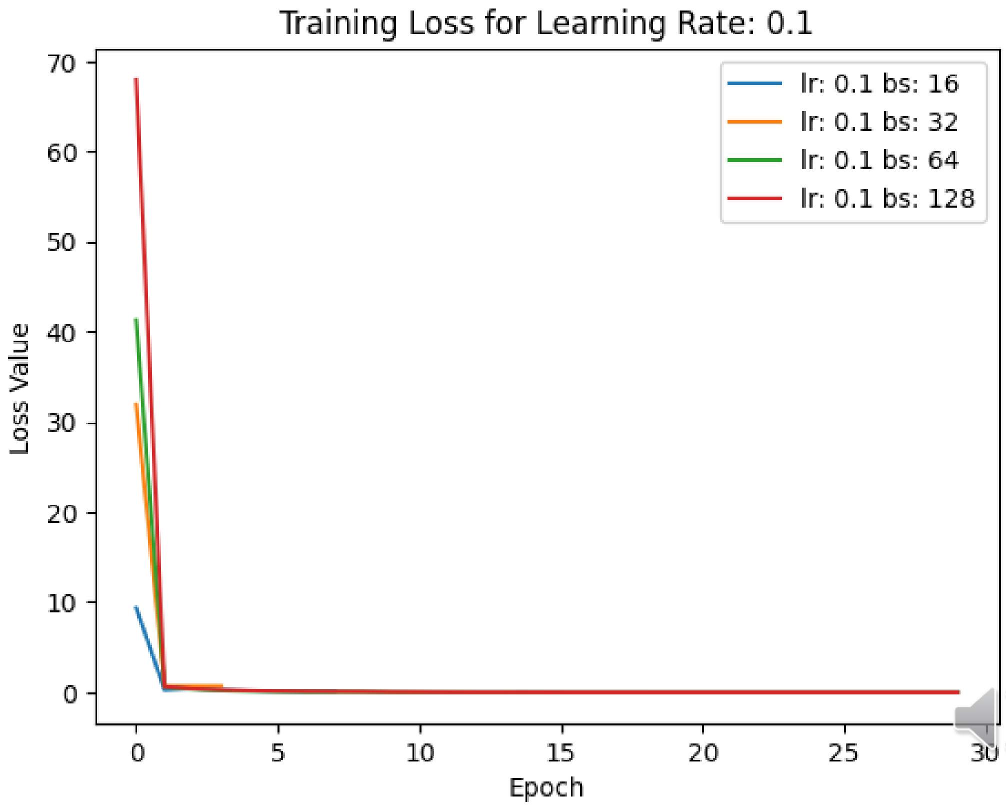
Inspecting the change of Batch Size with Different learning rates



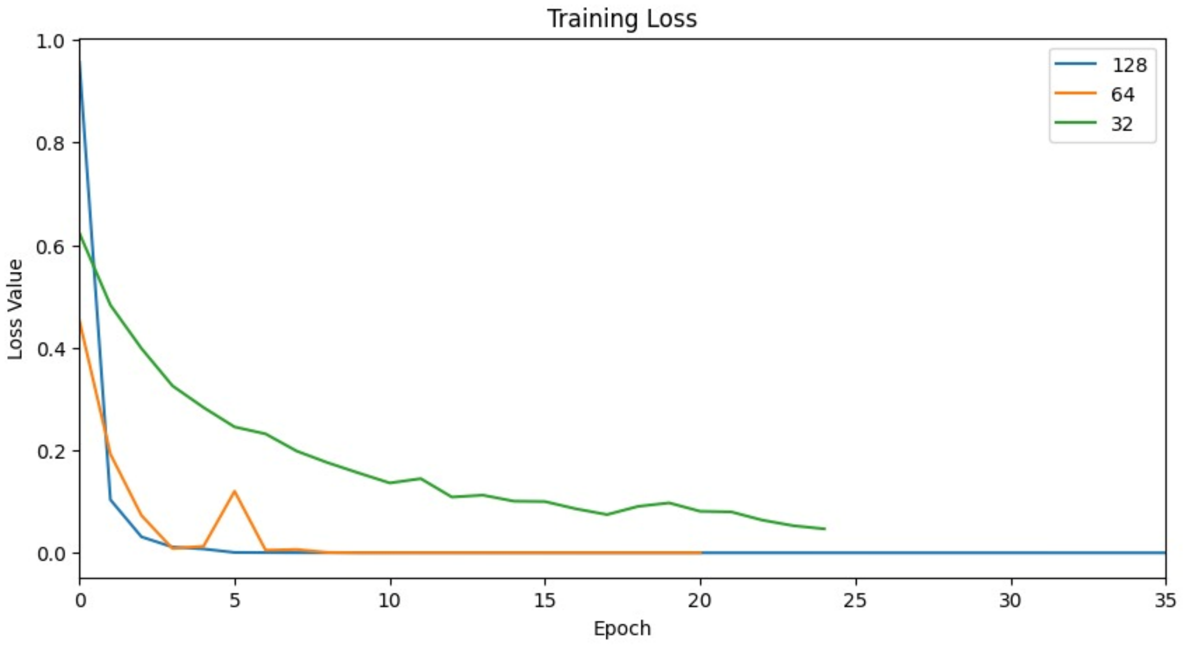


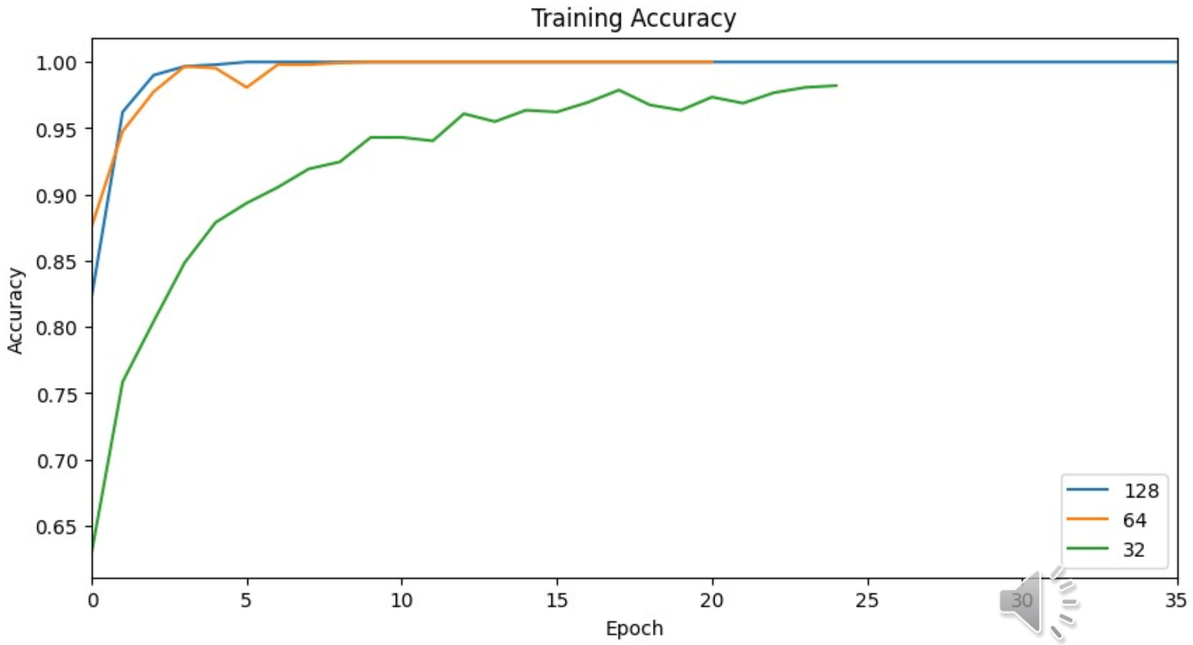
Inspecting the change of Batch Size with Different learning rates



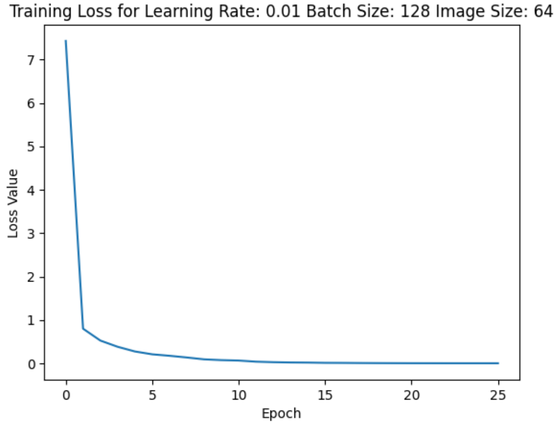


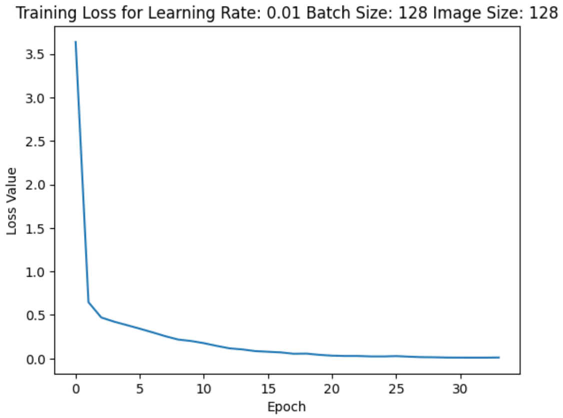
Inspecting the change of Image input size with learning rate=0.01 and batch size=16.

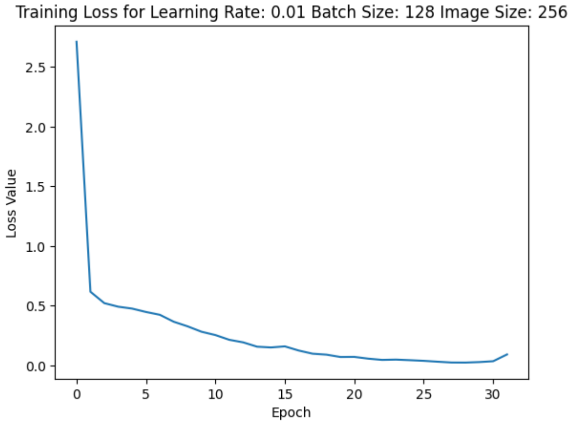


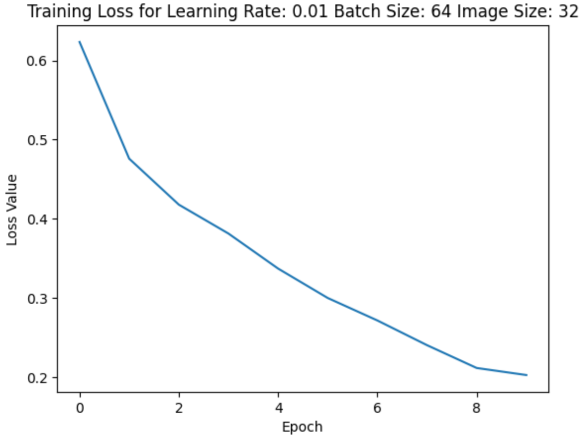


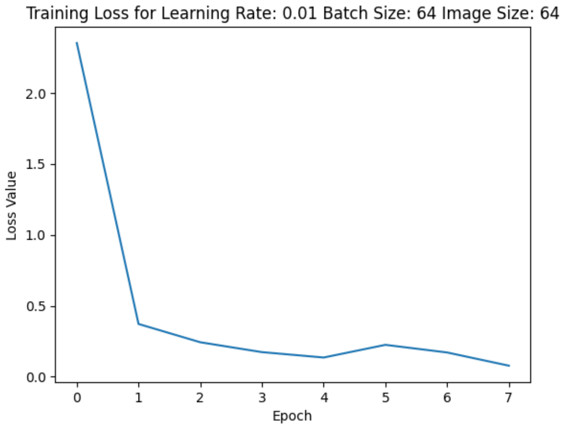
Further Studies on The Model Parameters
We Consider the change of the image size with learning rate and batch size.

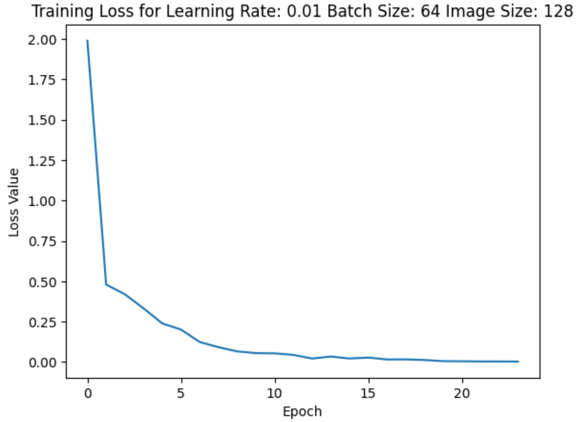


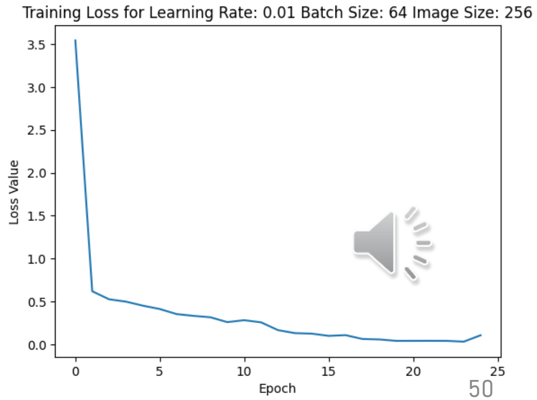




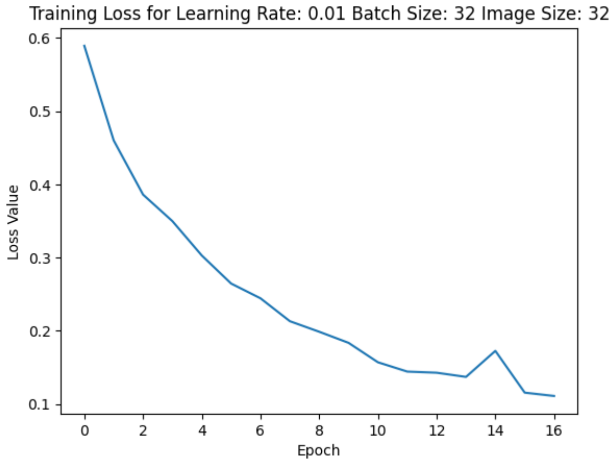


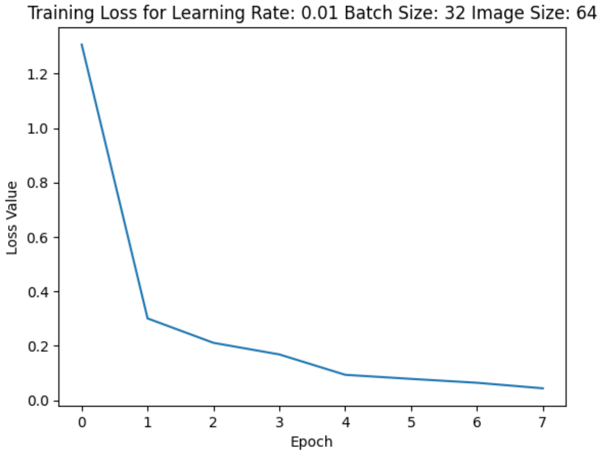


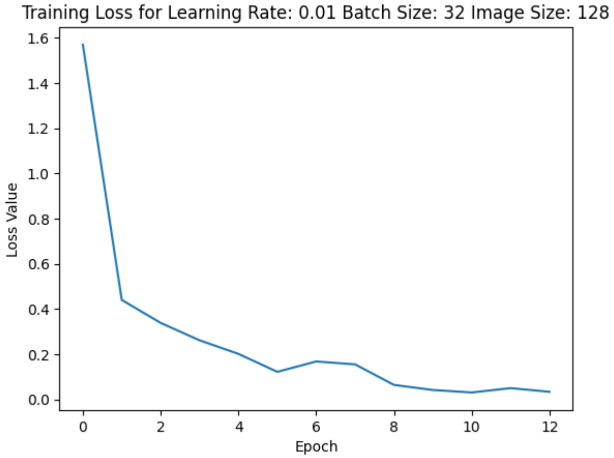


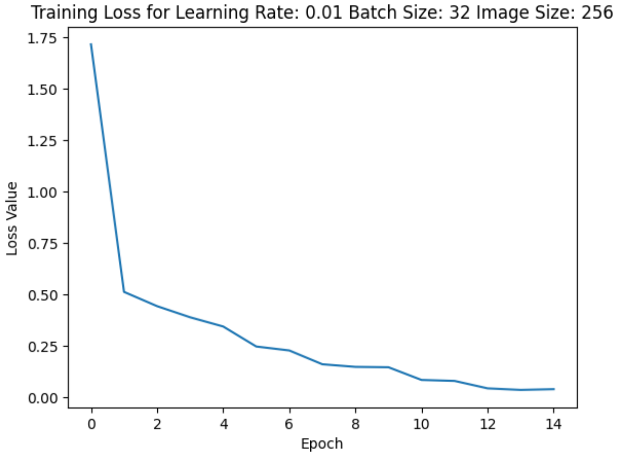


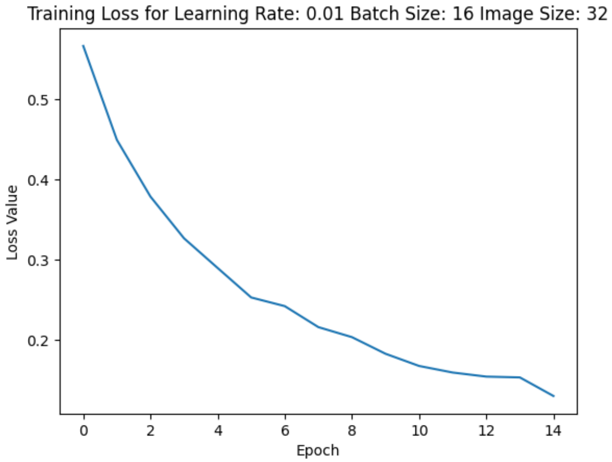
Further Studies on The Model Parameters
We Consider the change of the image size with learning rate and batch size.

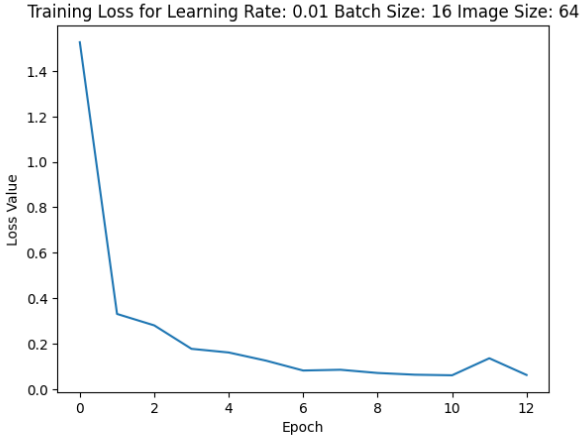


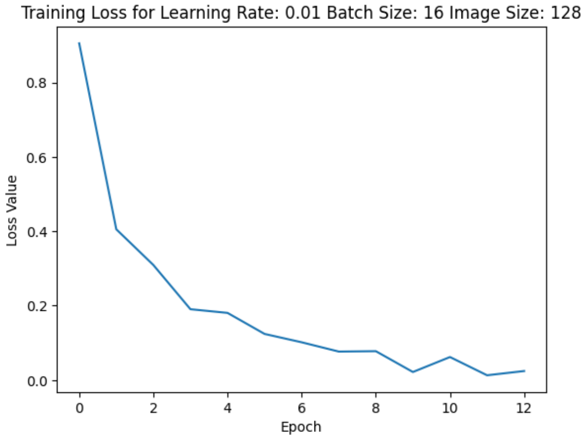




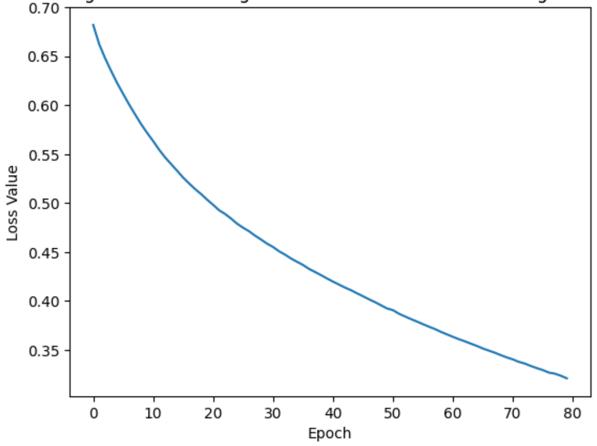




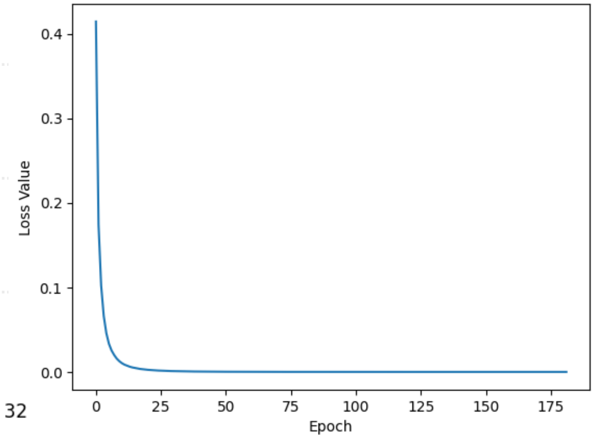




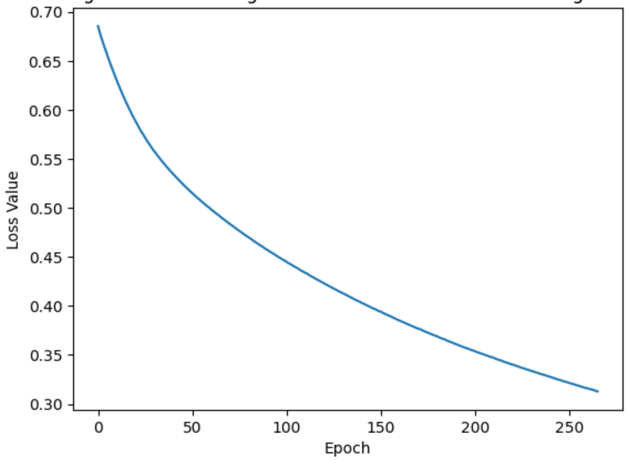
Further Studies on The Model Parameters
We Consider the change of the image size with learning rate and batch size.



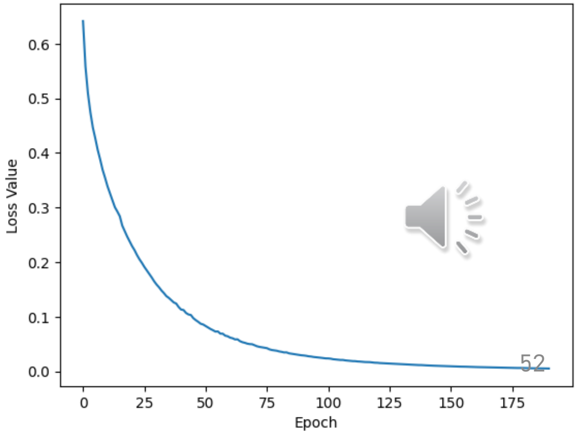
Training Loss for Learning Rate: 0.0001 Batch Size: 16 Image Size: 32



Training Loss for Learning Rate: 0.0001 Batch Size: 16 Image Size: 256



Training Loss for Learning Rate: 0.0001 Batch Size: 128 Image Size: 32



Training Loss for Learning Rate: 0.0001 Batch Size: 128 Image Size: 256

Conclusion

Insights (Literature review)

- Binary classification has many applications in real-world problems like medical imaging and anomaly detection.
- It is an important first step before more complex tasks like object detection and tracking.
- A variety of machine learning models have been developed for binary classification over the years.
- Newer models often achieve better performance through updated architectures and optimization techniques.
- The performance of classification models continues to improve with newer releases as the models become more accurate and efficient. However, newer is not always better - older papers still provide valuable insights.
- To keep up with progress, literature reviews need to focus on more recent papers to avoid recommending outdated models that have been since surpassed.

Insights (Preprocess, Model Select and Different Conditions)

- Preprocessing tasks like loading data into appropriate formats like TF records is straightforward and there is many forms for the data to be stored not always csv.
- Evaluating multiple model architectures is useful, but performance should only be compared when models use identical hyperparameters and training procedures.
- Choosing the right evaluation metrics allows for more meaningful analysis and comparison of model performance.
- Exposing models to challenging or outlier cases helps validate their true capabilities and limitations.
- Advanced models like MobileNet and YOLO demonstrate techniques like robustness to class imbalance that overcome limitations of earlier approaches.

Insights (Ablation study)

- The goal of an ablation study is not to find the best model configuration, but rather to systematically evaluate how different design choices and hyperparameters impact model performance.
- By carefully varying one parameter at a time, we can develop a deeper understanding of how each component contributes to the overall behavior and learnability of the network.
- Lower learning rates require more epochs to converge but lead to smoother optimization and potentially better local minima. Higher rates may prevent the model from adequately learning.
- Batch size impacts both training speed and performance, with smaller sizes tending to converge faster but being noisier. This effect is magnified at very low learning rates.
- Larger input image sizes provide more visual detail but slow down training. While validation metrics may appear better early on, smaller sizes require fewer computations to stabilize.

Presented By

Group: 5

Name:- Ahmed Shafik ,	**ID:-**300389391
Name:- Ibrahim Elshenhapy ,	**ID:-**300389386
Name:- Mohamed Hany ,	**ID:-**300389913
Name:- Mohamed Salem ,	**ID:-**300389921
Name:- Nada Zakaria ,	**ID:-**300389901