

Covid-19 classification using ML and DL

Presented by:

Youssef Fadel

Engy Khaled

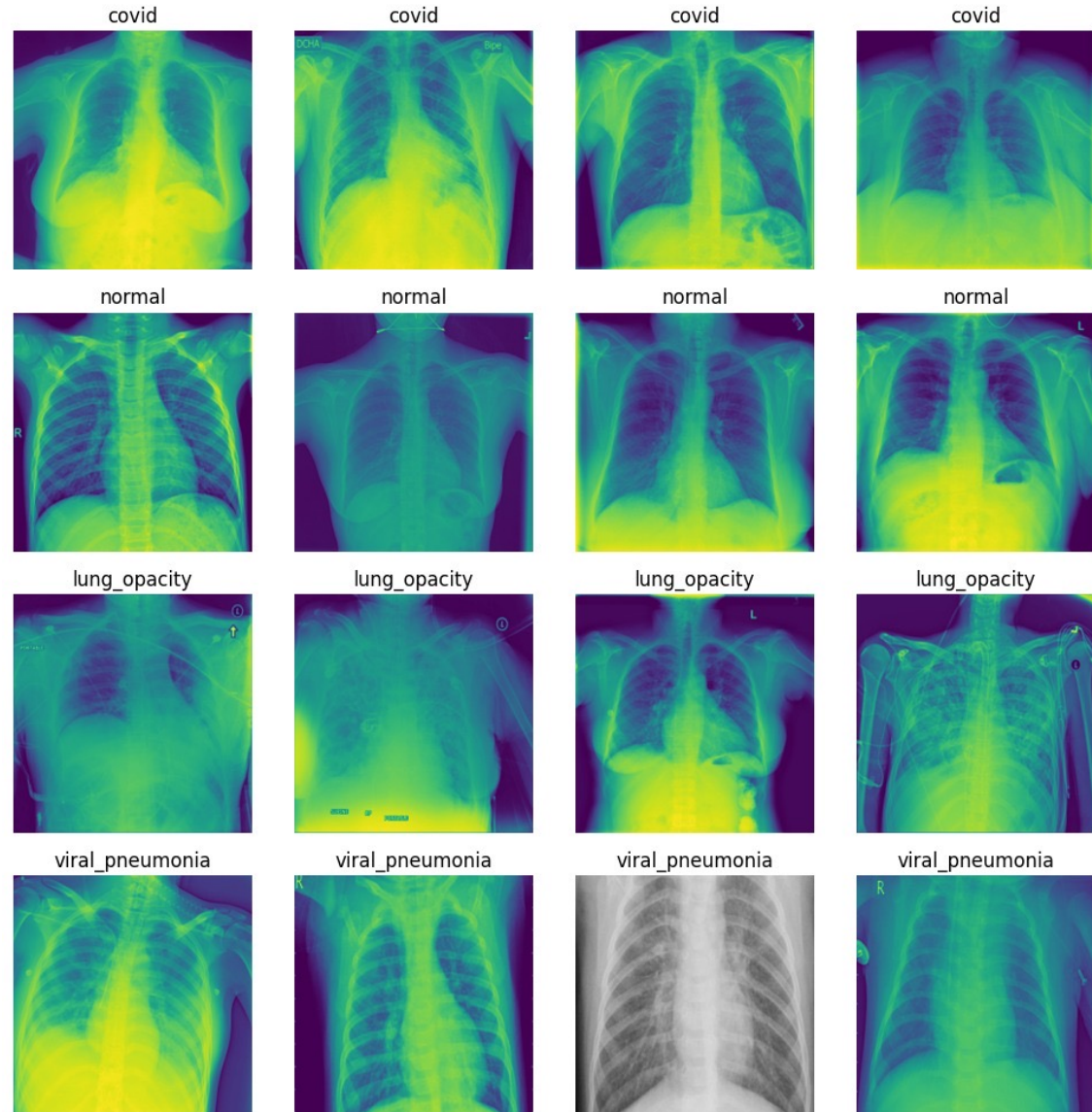
Hannah Hisham

Outlines

- Problem definition
- Methodology
 - Data collection
 - Data preprocessing
 - Classification Machine learning models
 - Classification using deep learning models
- Results
 - Train results
 - Test confusion matrix results
- Conclusion

Problem Definition:

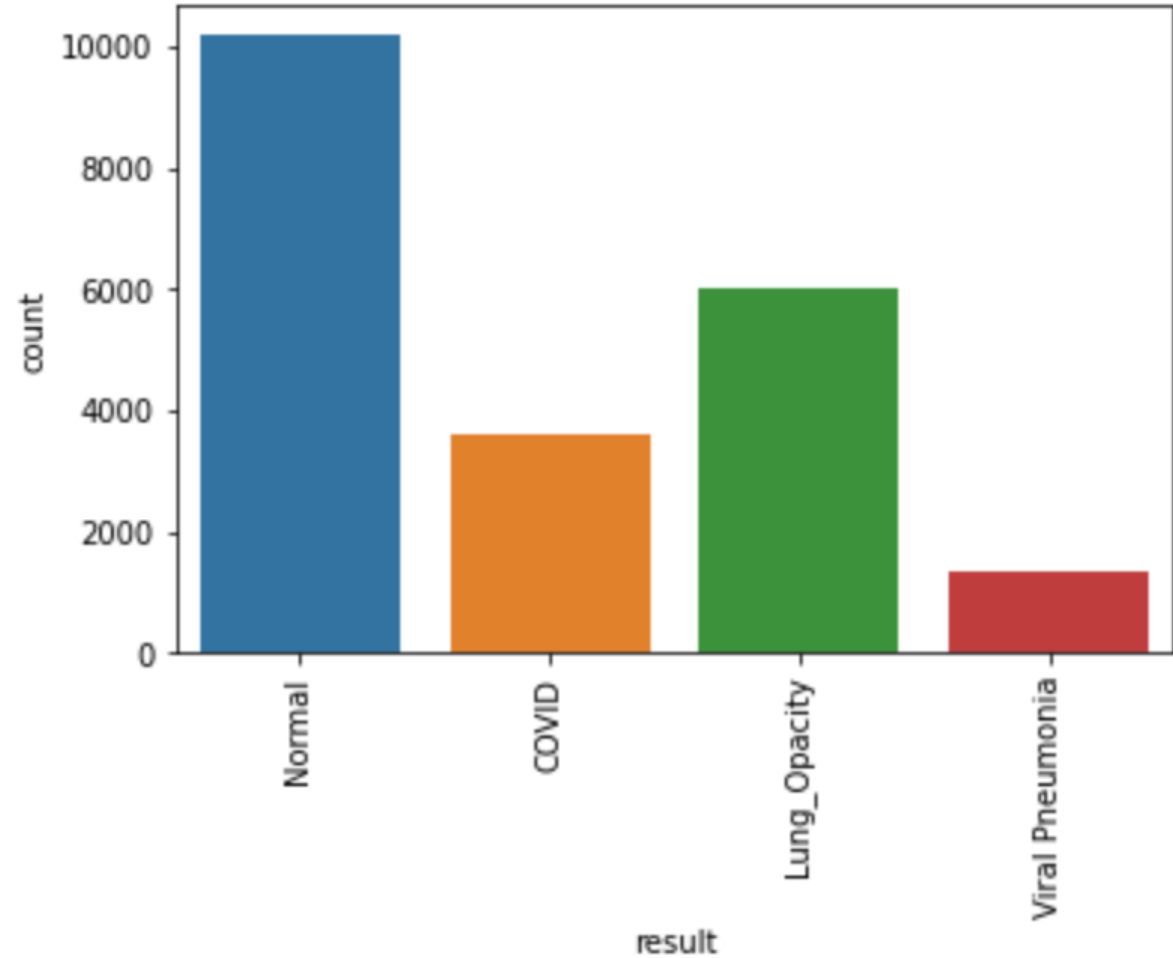
- The aim of this project is to classify Lung X-ray images into 4 classes: Lungs infected with COVID-19, lung opacity, viral pneumonia and normal lungs of healthy individuals.



Methodology

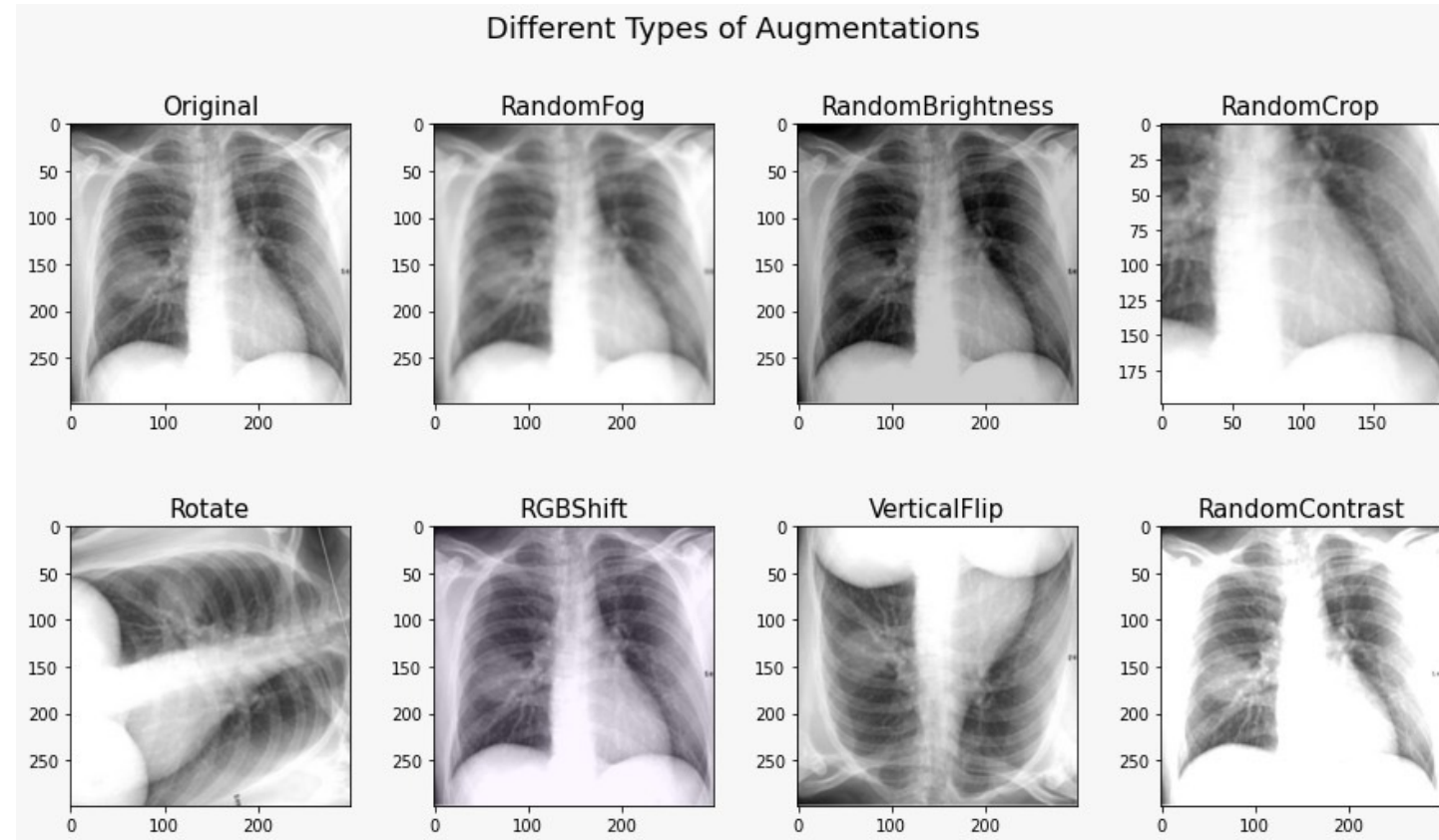
Data Collection:

- X-ray were collected from Kaggle under the link “<https://www.kaggle.com/datasets/taw-sifurrahman/covid19-radiography-database>”
- The methodology used for deep learning was from the paper titled “**Detection and analysis of COVID-19 in medical images using deep learning techniques**”.



Data Preprocessing:

- The data was subjected to resizing and normalization.
- The data was split into 3 parties: Train (70%), Val(15%) and Test(15%).
- Each party was split into 4 classes: Normal, Covid and lung opacity, viral pneumonia .
- Data imbalance was taken into account by adjusting sample weights based on the bias, data augmentation was performed.



Classification

ML

DL

Feature extraction

Classification

Classification

VGG16

resnet50

resnet152

desnet121

Pixels

HOG

lbp

Logistic regression

Decision tree

Random forest

svm

xgb

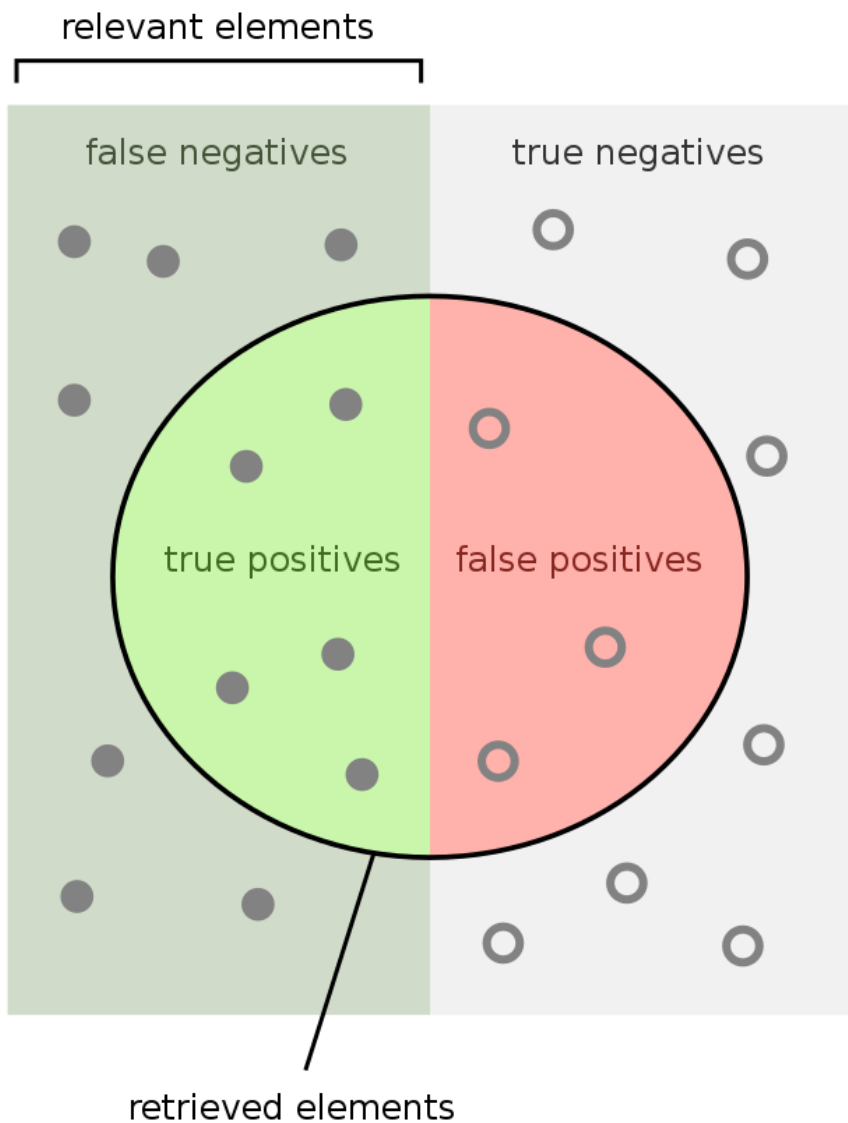
KNN

NB

DL Training:

- Based on the paper, 4 pretrained models were used which are VGG16, DENSENET121, RESNET50, and RESNET152.
- The optimizer used was ADAM.
- The Epochs were set to 10 and 13 for each model.
- The learning rate was set to 0.0001 based on the methodology of the paper.
- Loss function was included as well.

Results



How many retrieved items are relevant?

Precision = $\frac{\text{true positives}}{\text{true positives} + \text{false positives}}$

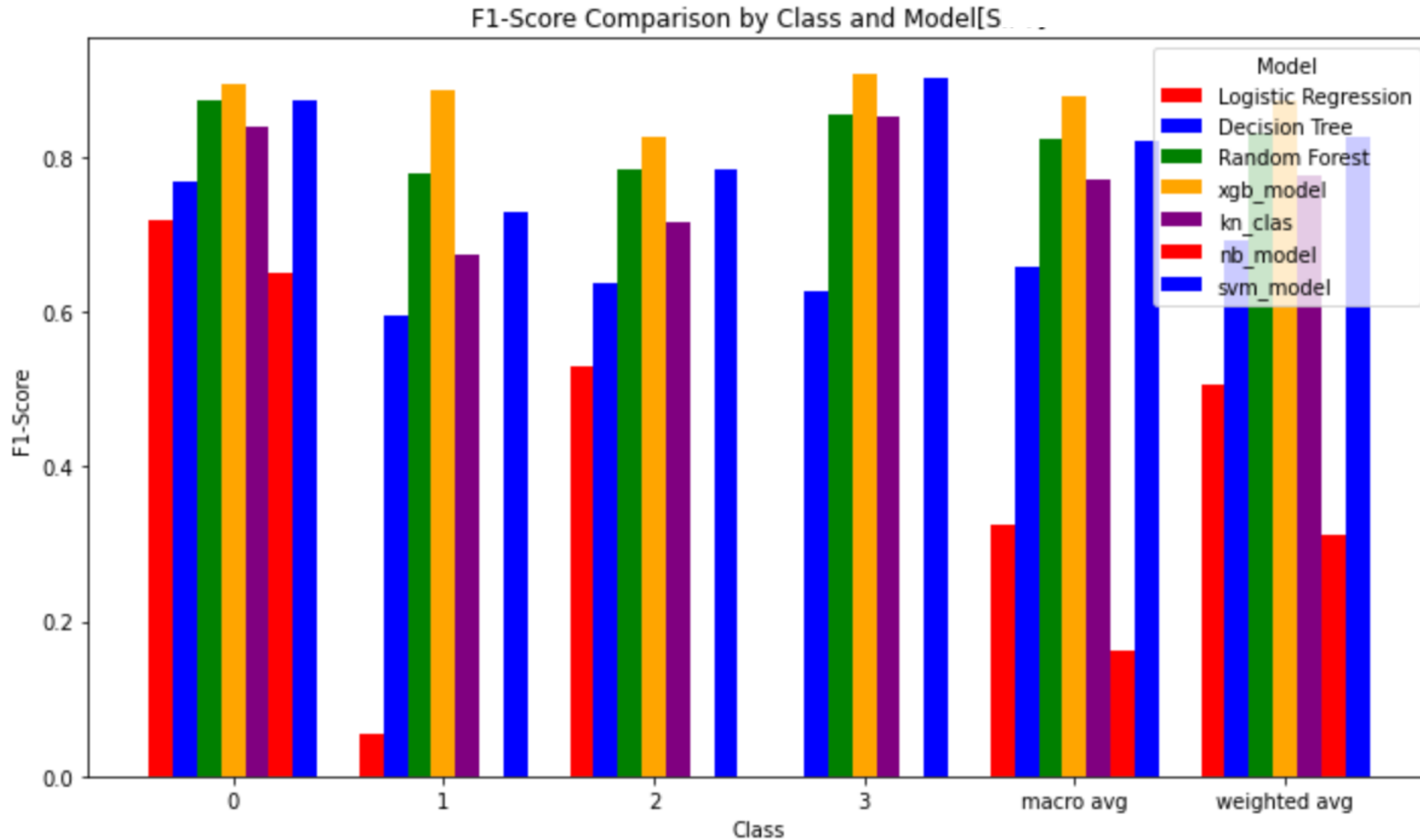
How many relevant items are retrieved?

Recall = $\frac{\text{true positives}}{\text{true positives} + \text{false negatives}}$

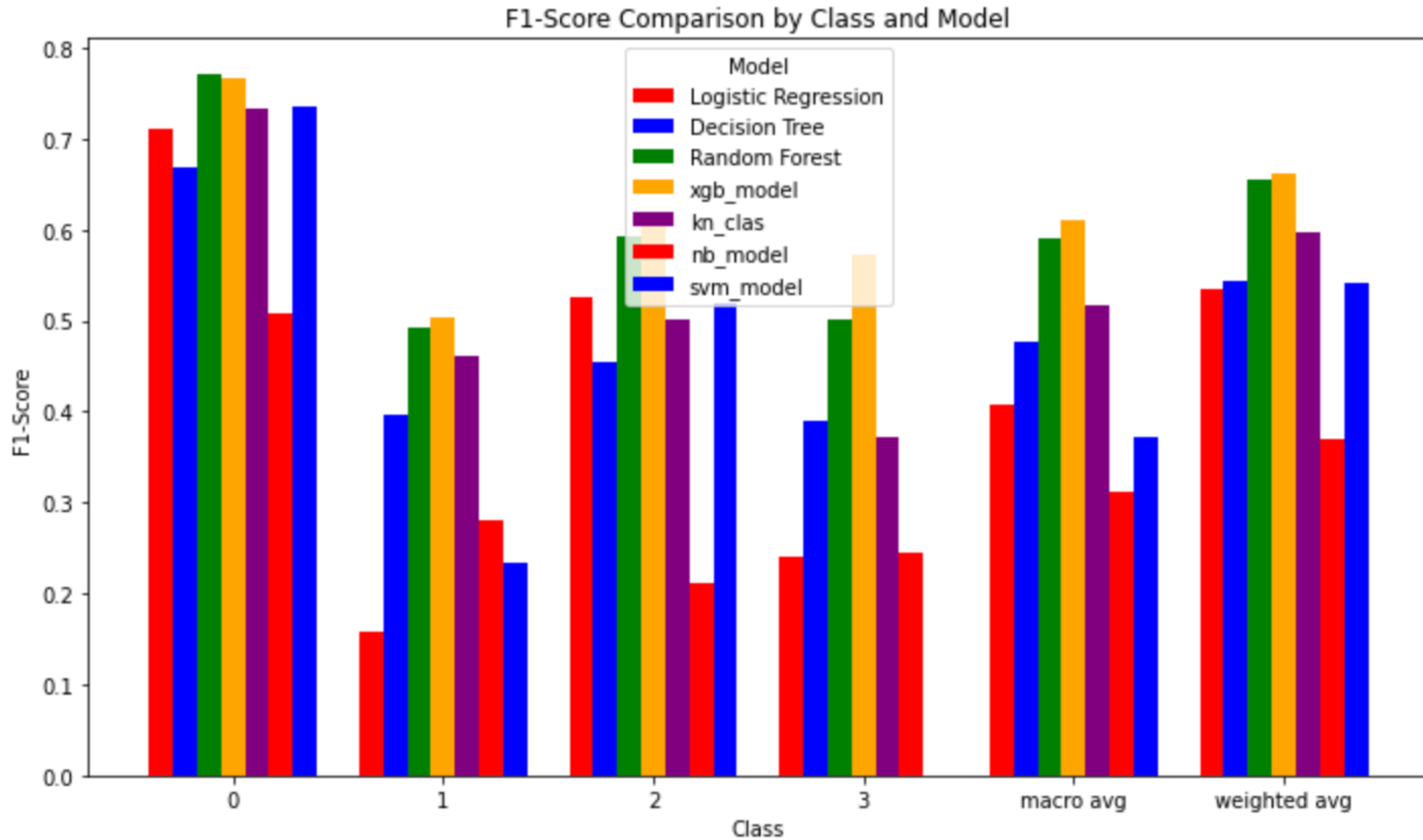
$$F_1 = \frac{2}{\frac{1}{\text{recall}} + \frac{1}{\text{precision}}} = 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}}$$

$$= \frac{2 \times \text{tp}}{2 \times \text{tp} + \text{fn} + \text{fp}}$$

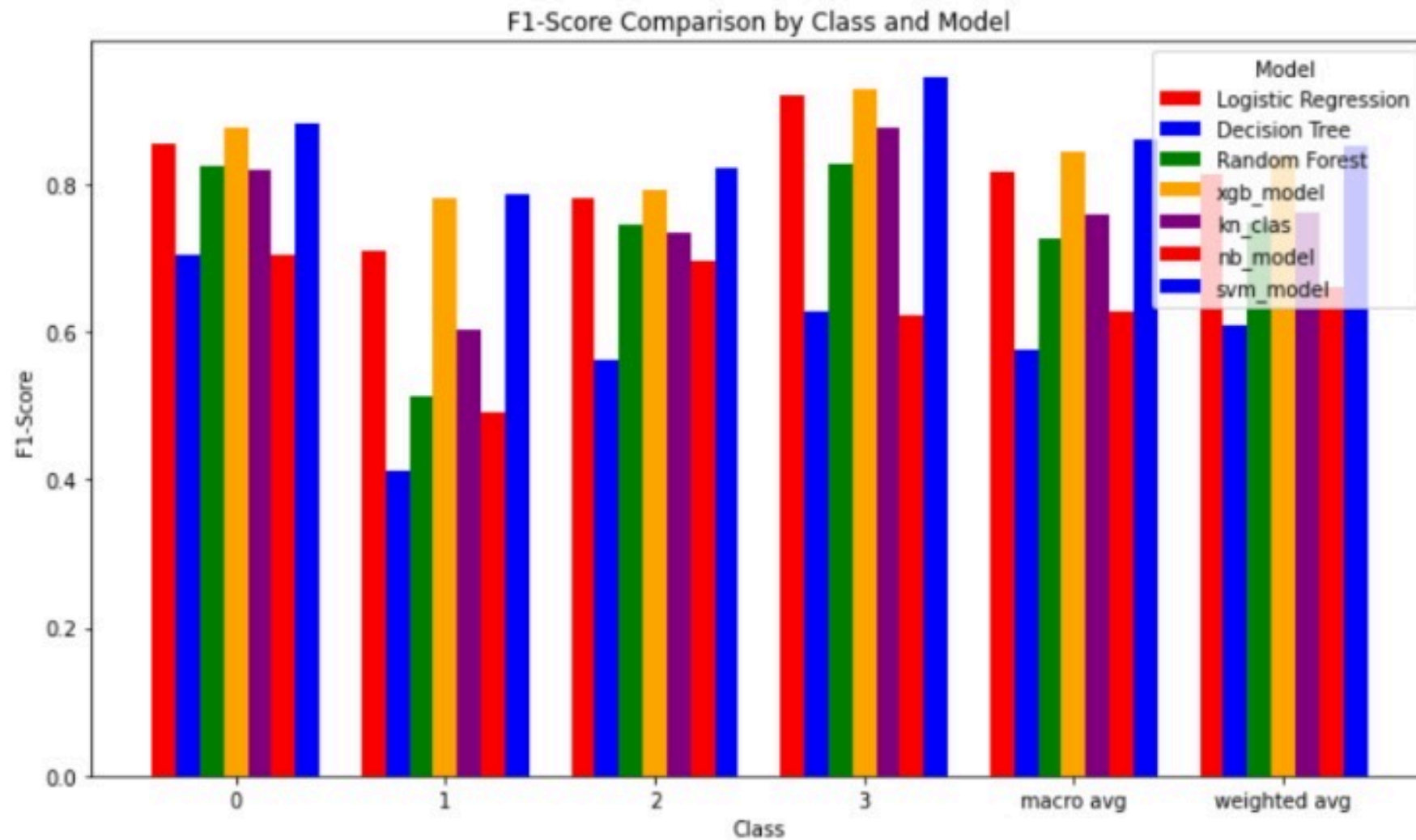
Classification according pixels



Classification according lbp

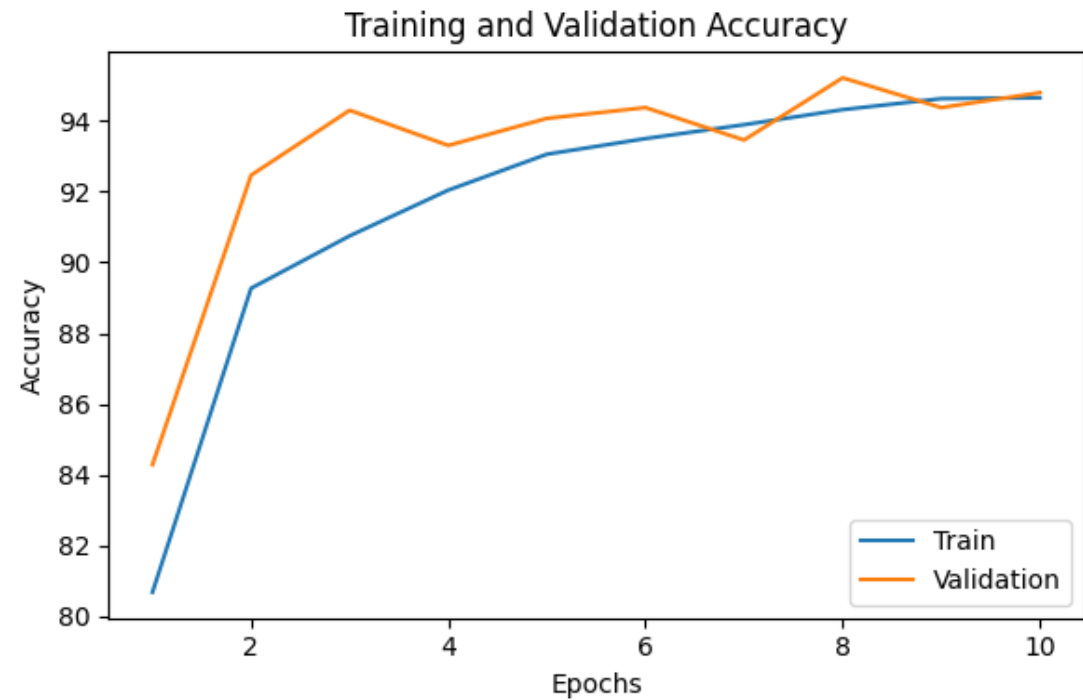
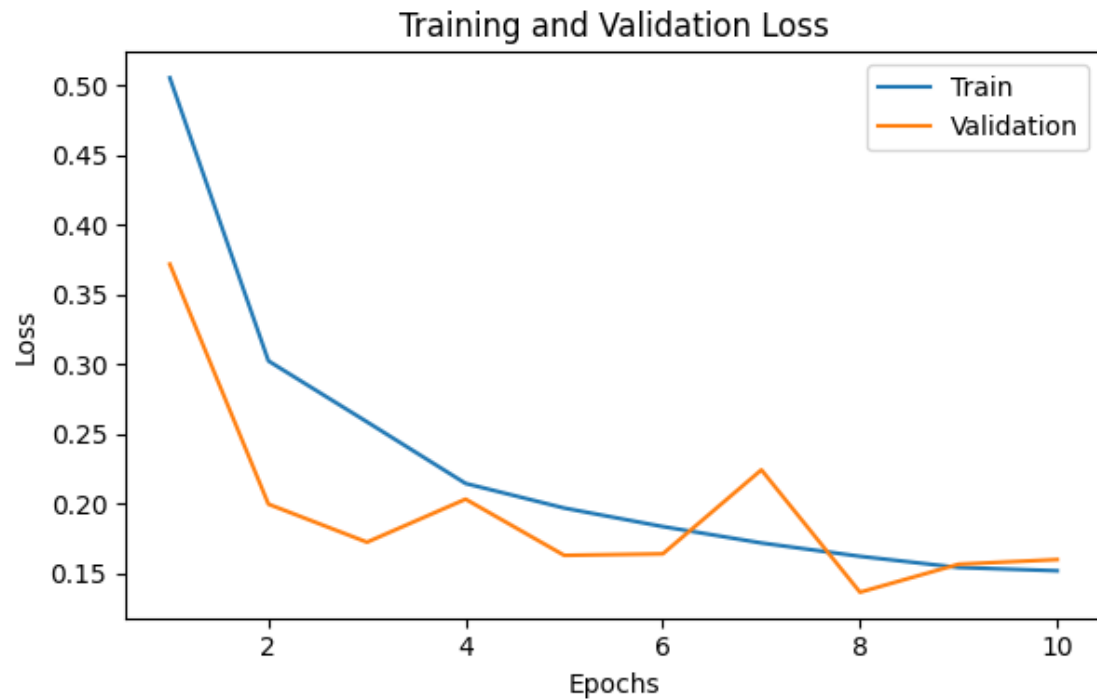


Classification according HOG

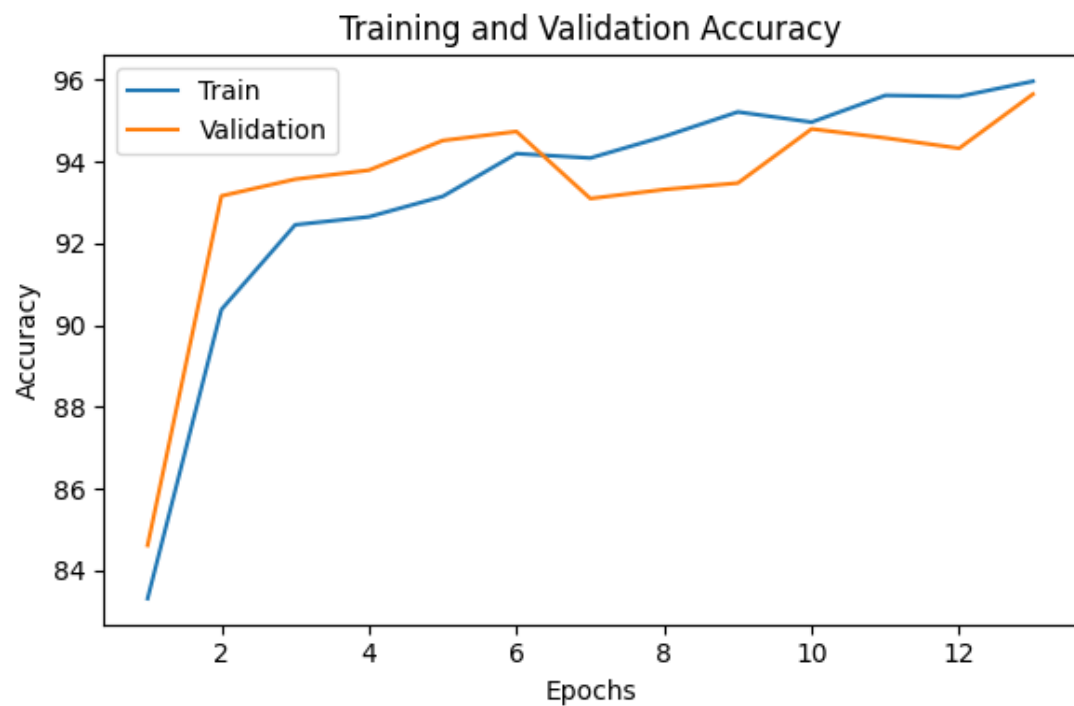
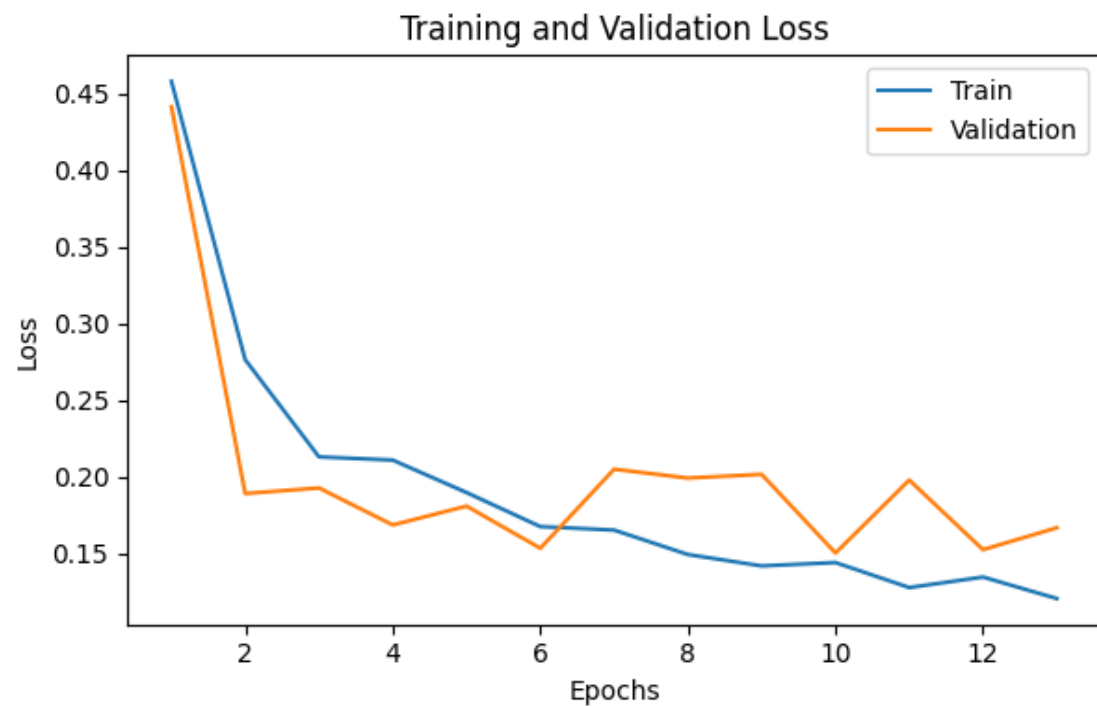


DL results

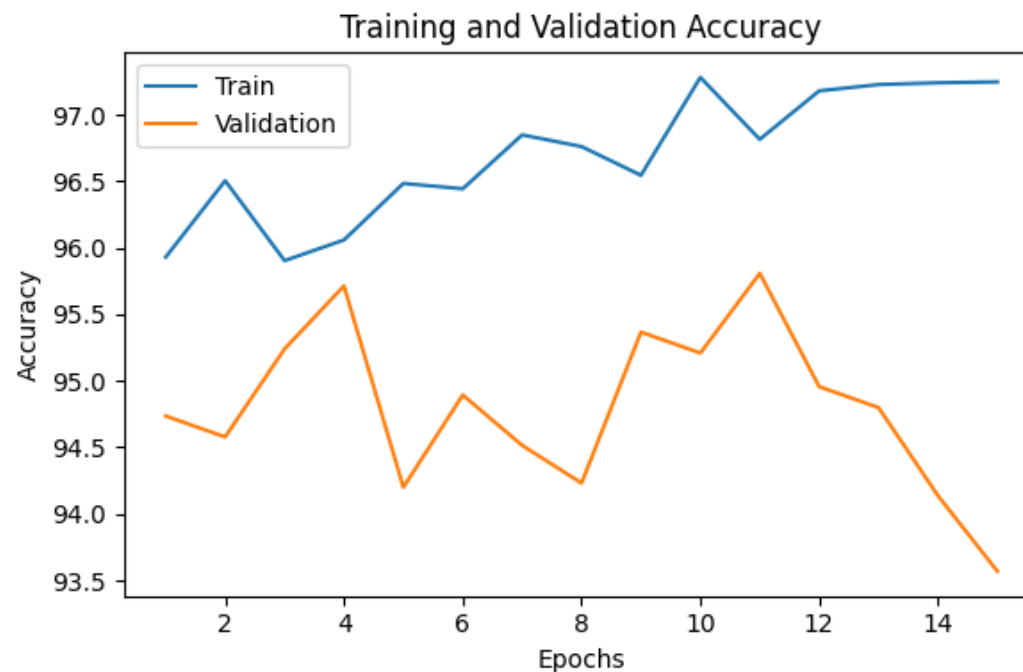
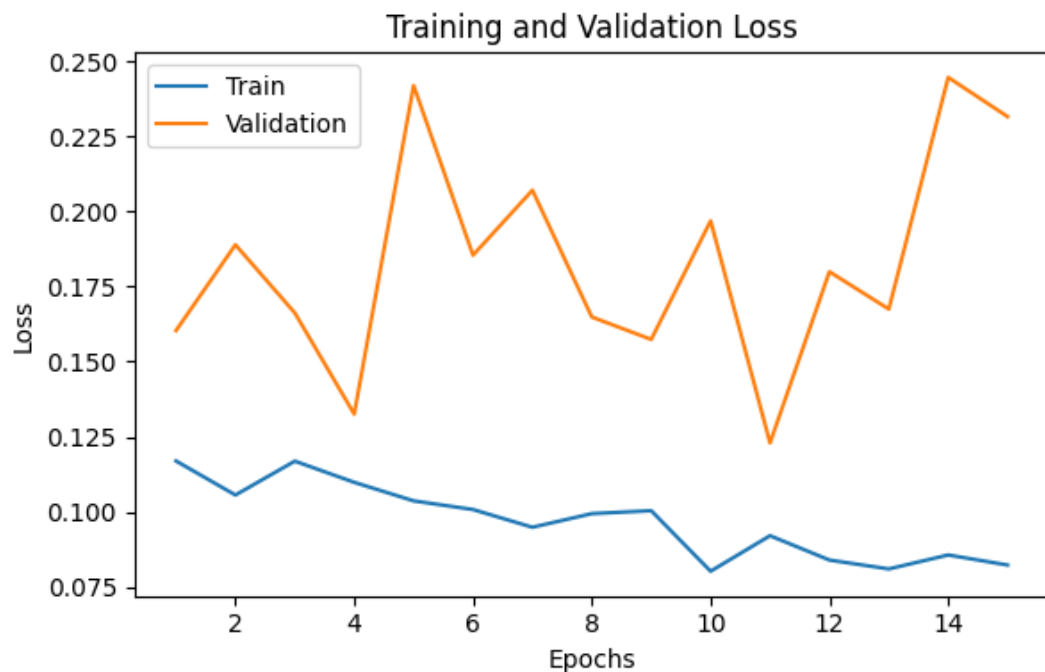
VGG16 train results: epochs=10



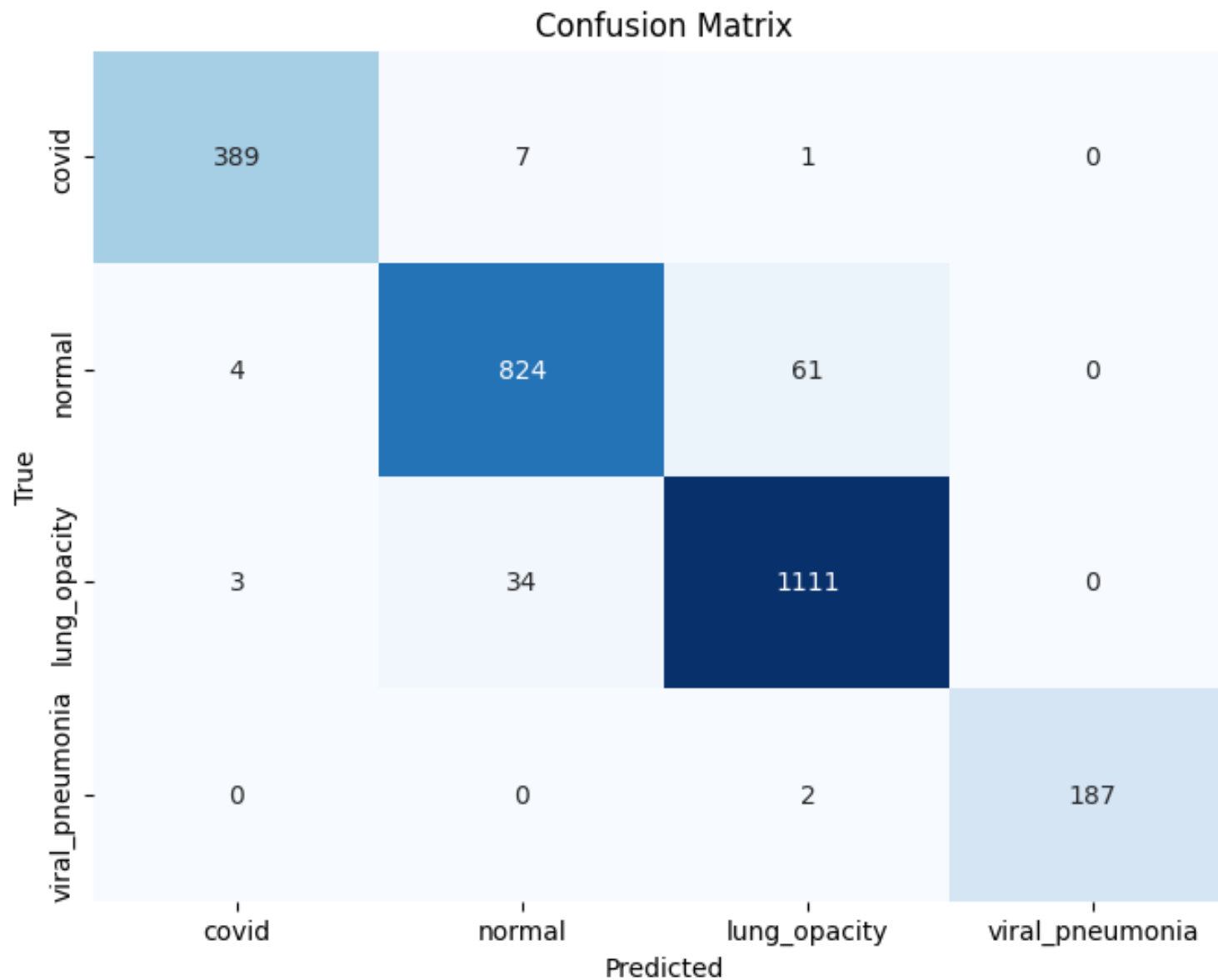
VGG16 train results: epochs=13



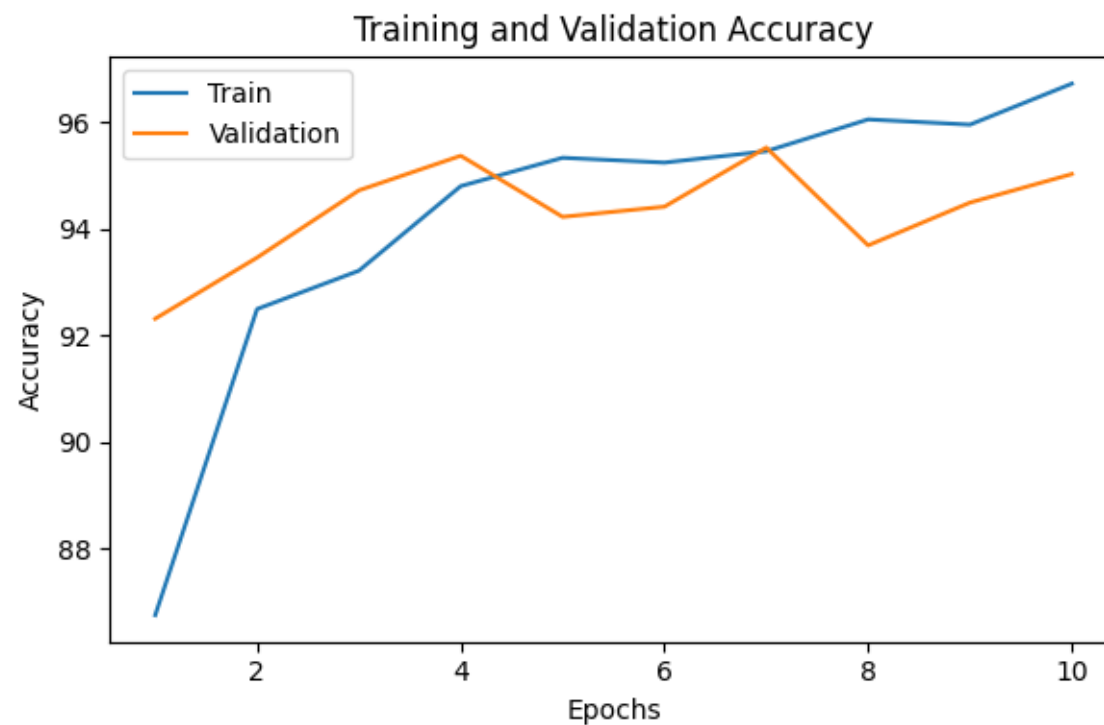
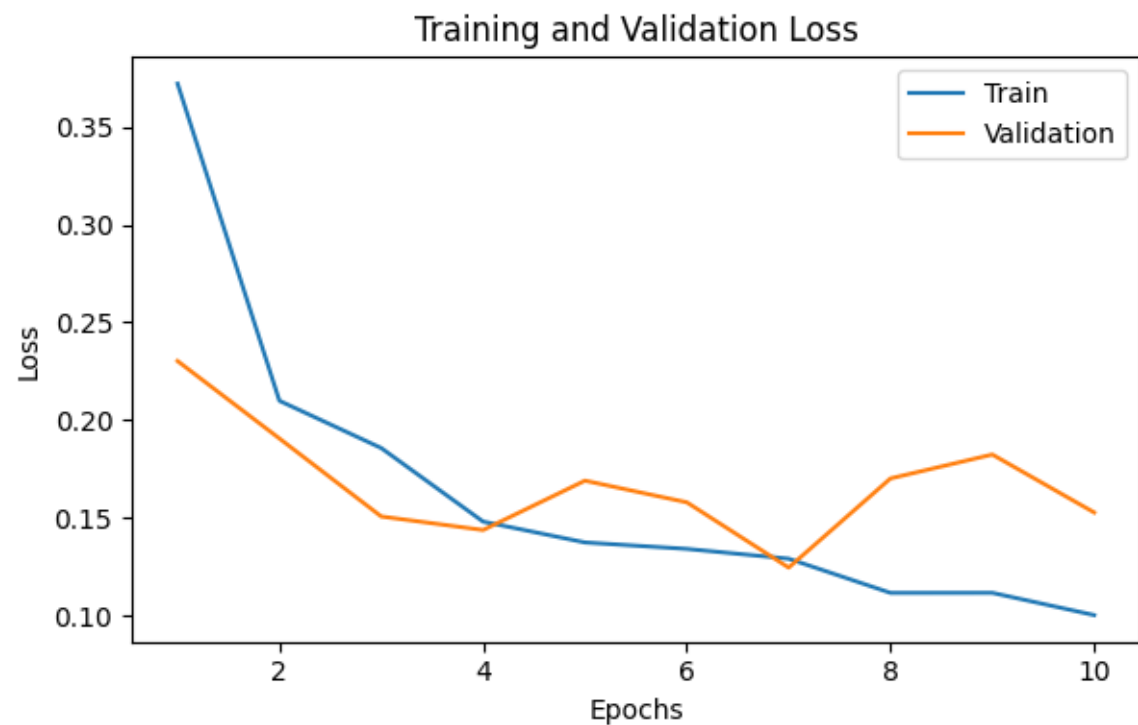
VGG16 train results: epochs=15



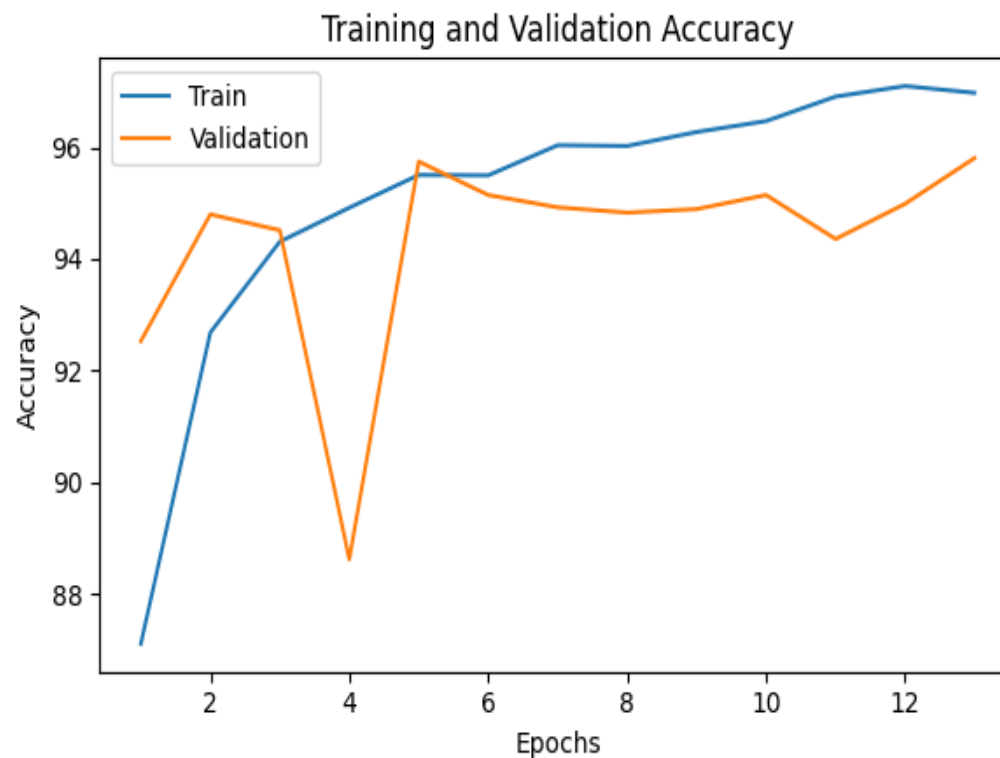
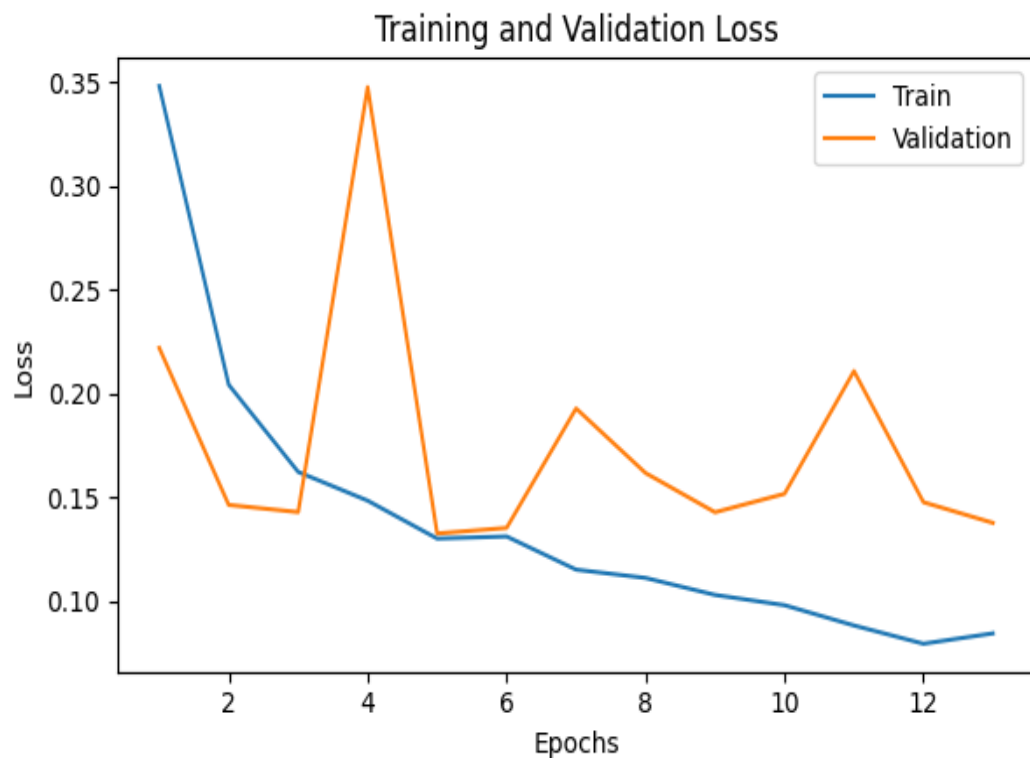
VGG16 TEST CONFUSION MATRIX



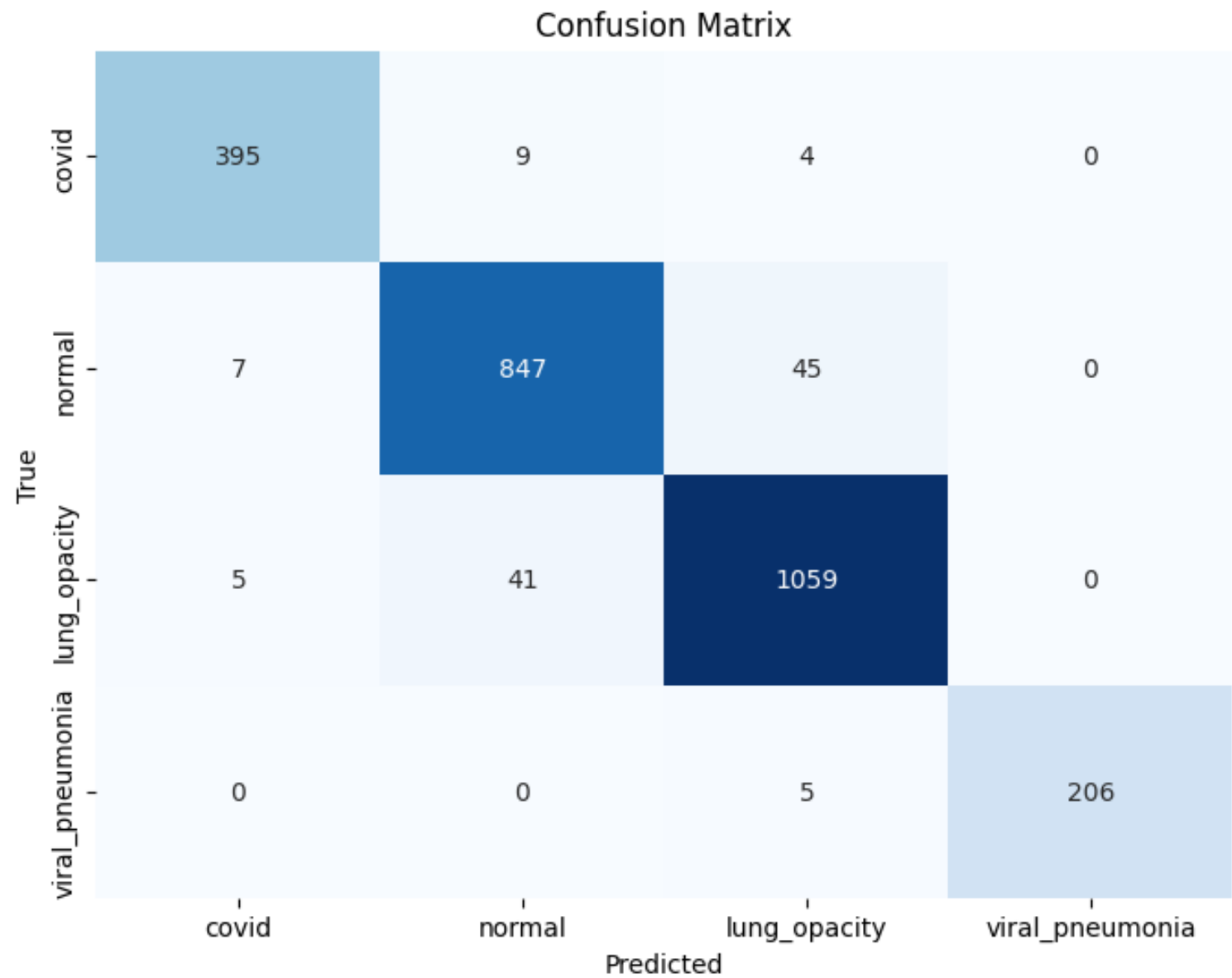
DENSENET121 TRAIN RESULTS: Epochs=10



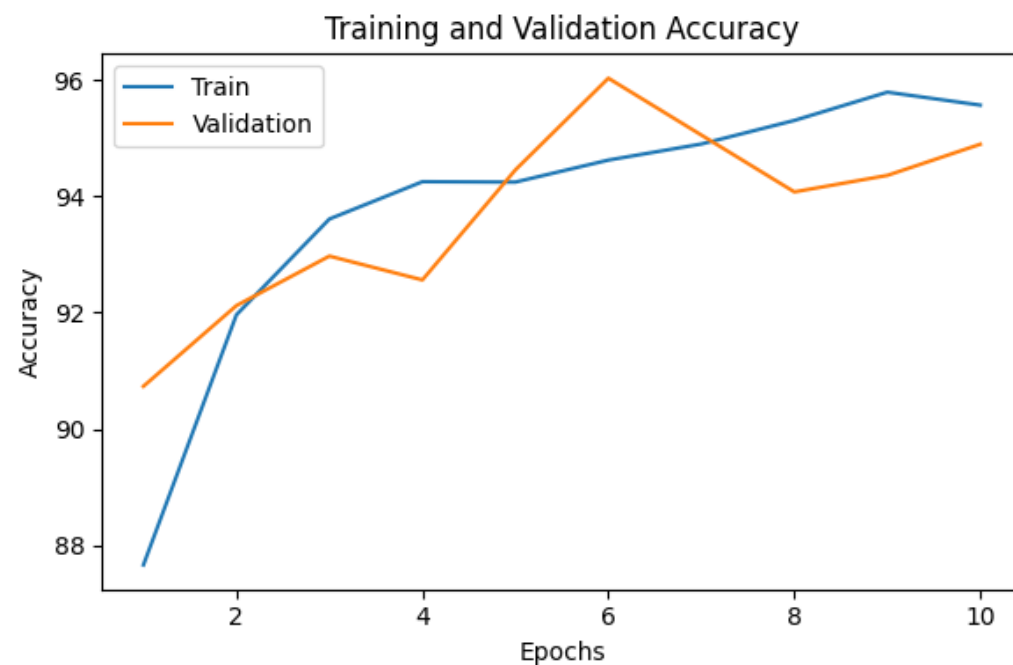
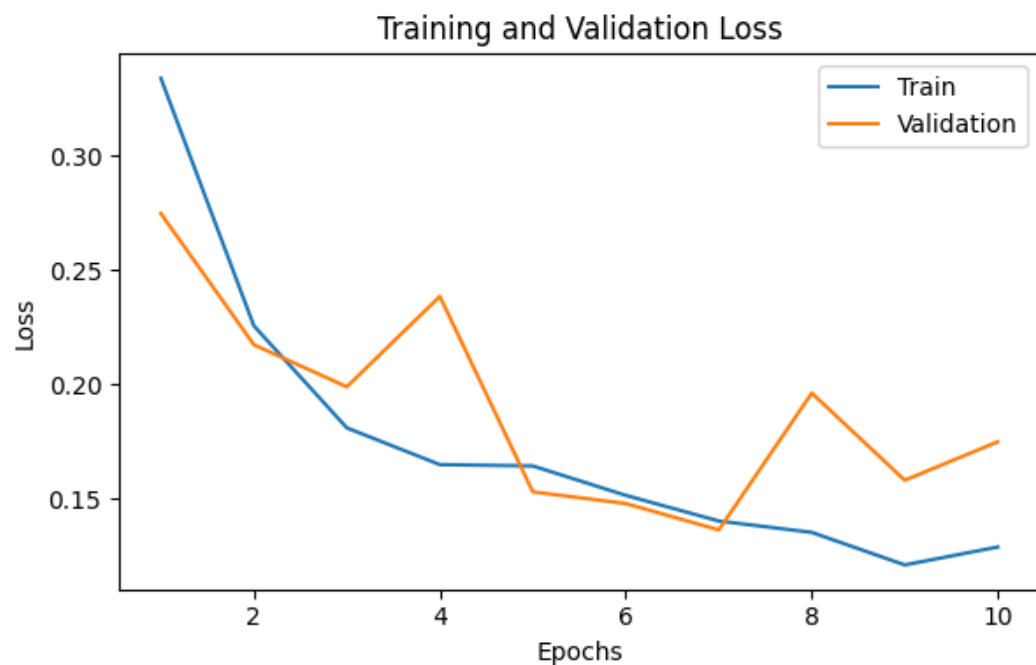
DENSENET121 TRAIN RESULTS: Epochs=13



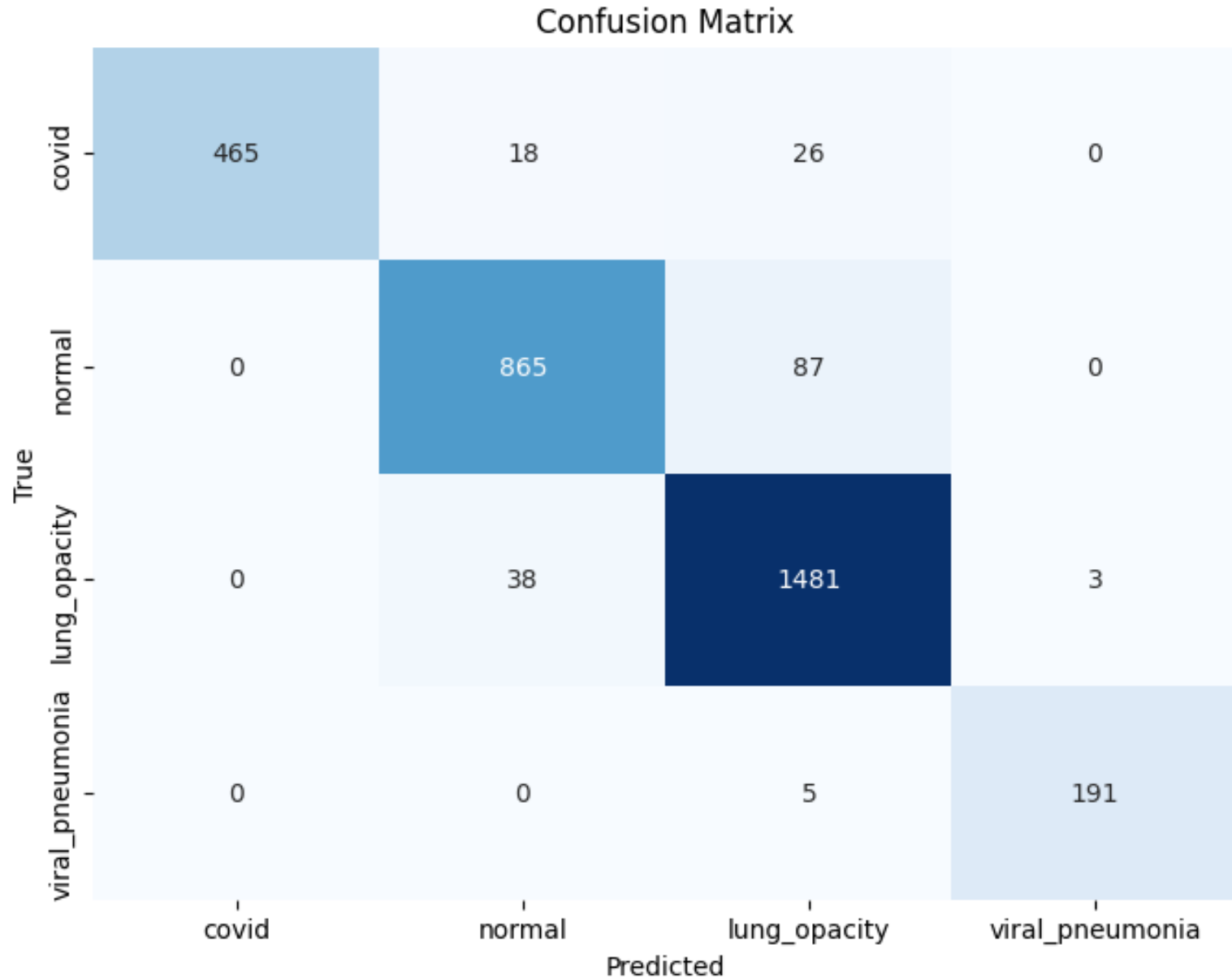
DENSENET121 confusion matrix



RESNET50 TRAIN RESULTS: Epochs=10

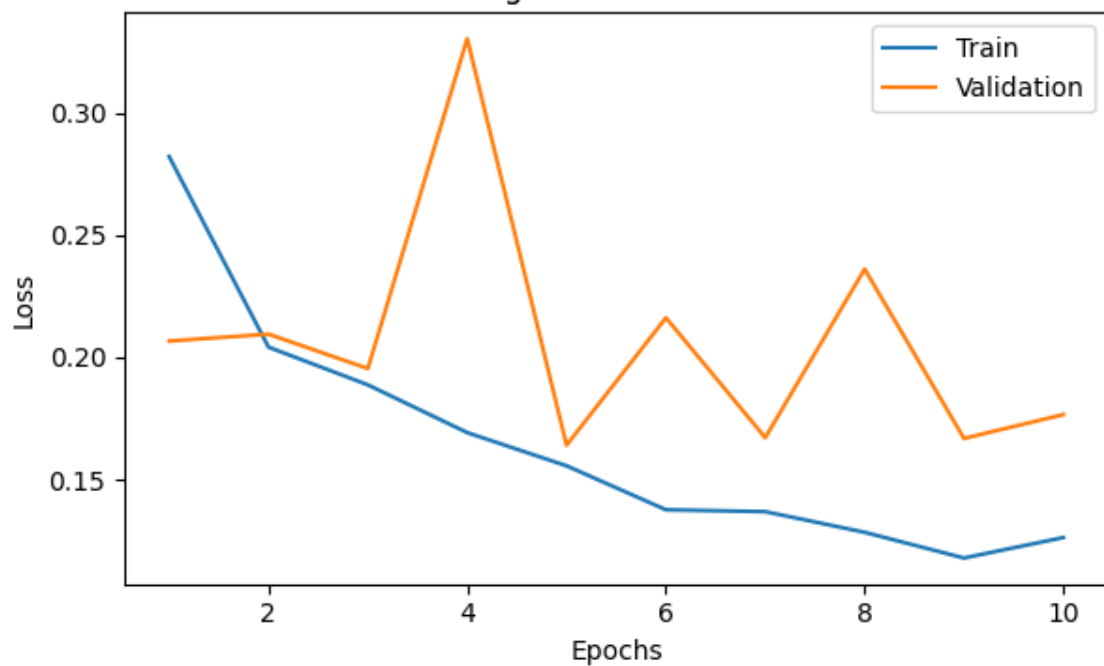


RESNET50 confusion ma

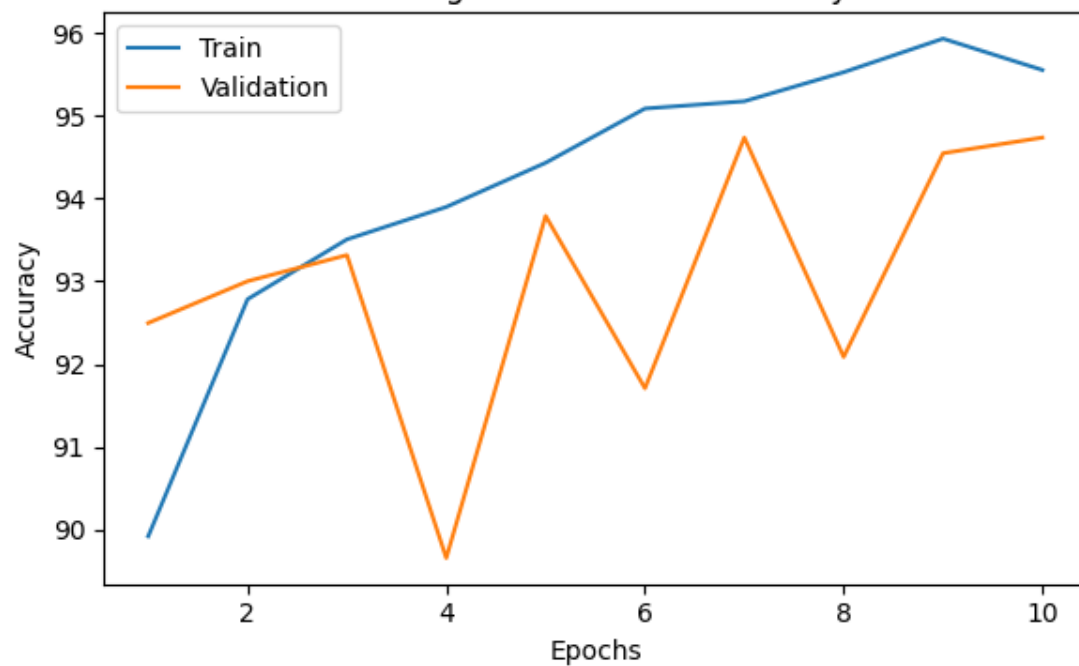


RESNET152 TRAIN RESULTS: Epochs10

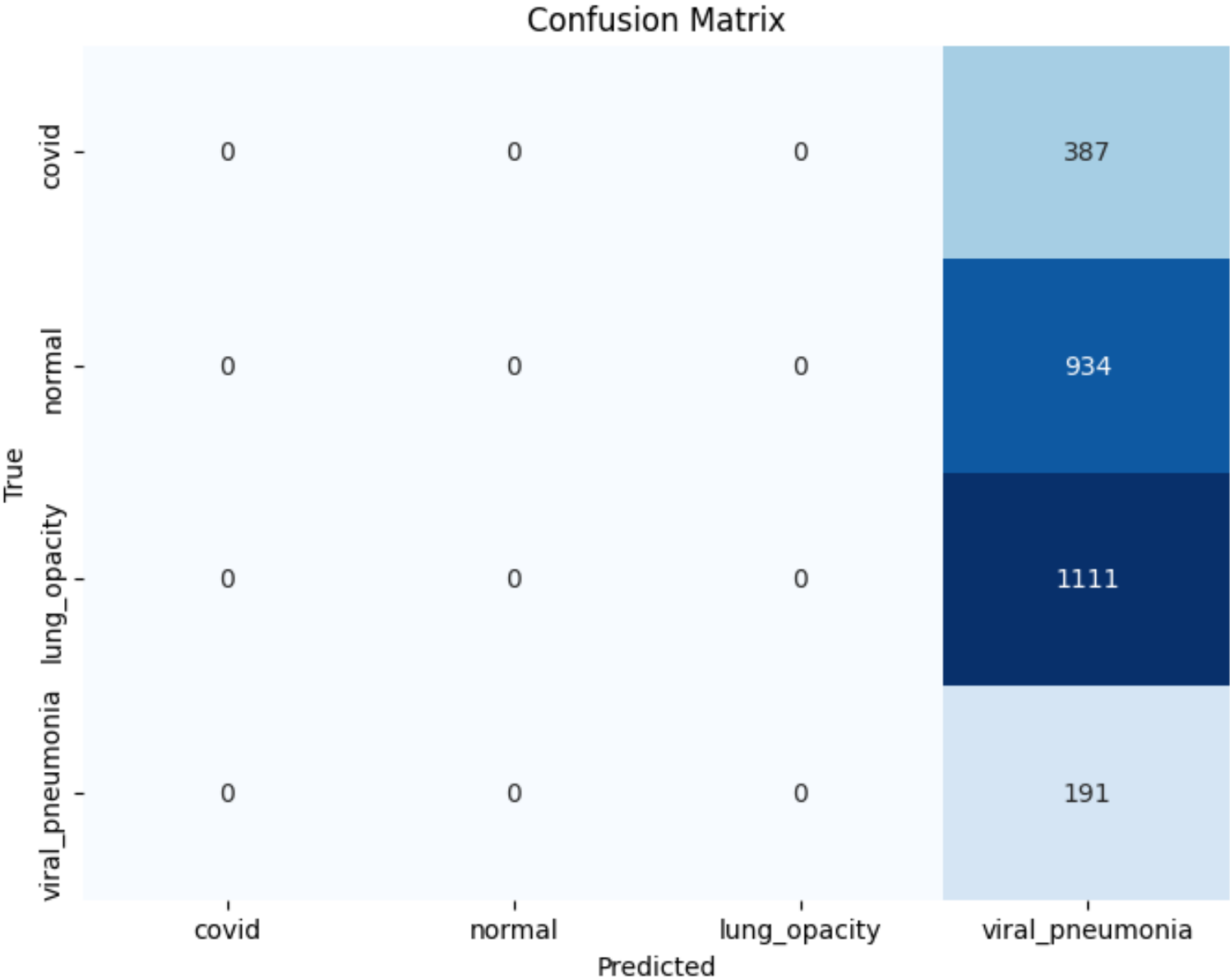
Training and Validation Loss



Training and Validation Accuracy



RESNET50 confusion matrix



Conclusion

Deep Learning Part:

- **By Increasing Epochs Number, Overfitting Occur due to :**

1- Imbalanced data , Limited training data

2- Data shift Phenomena

3- Model Complexity

4- Validation Error Phenomena as seen in DESNET 121

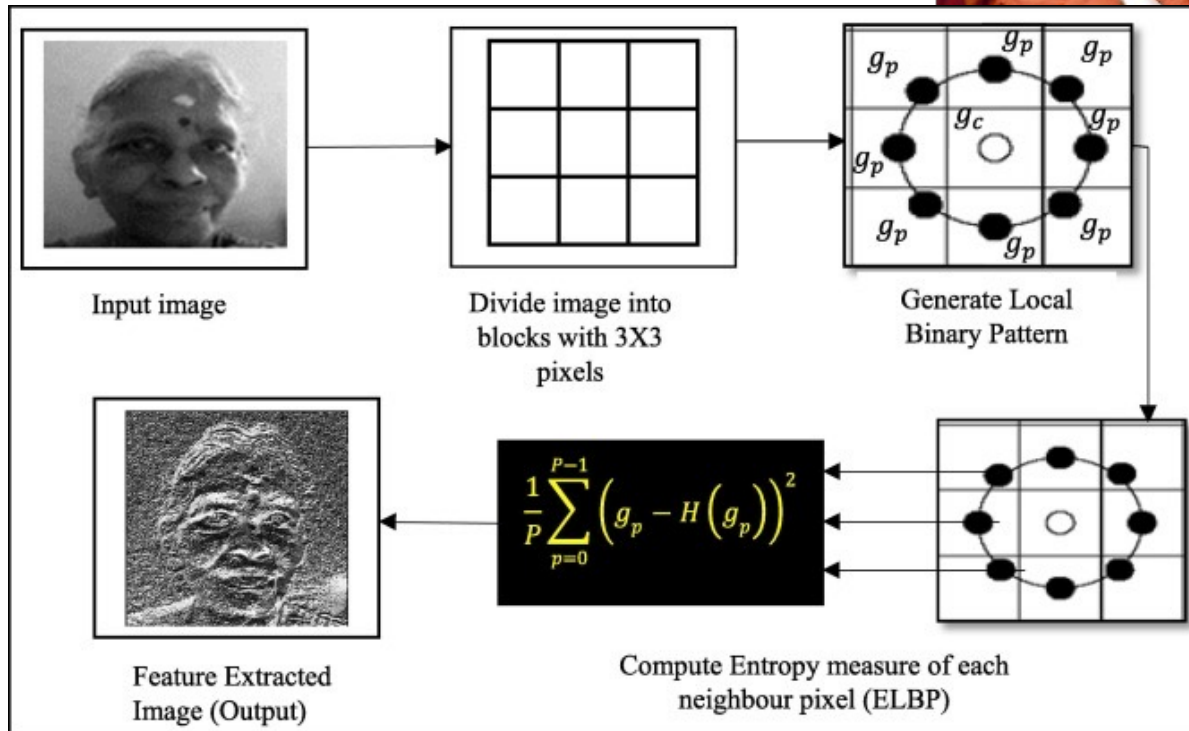
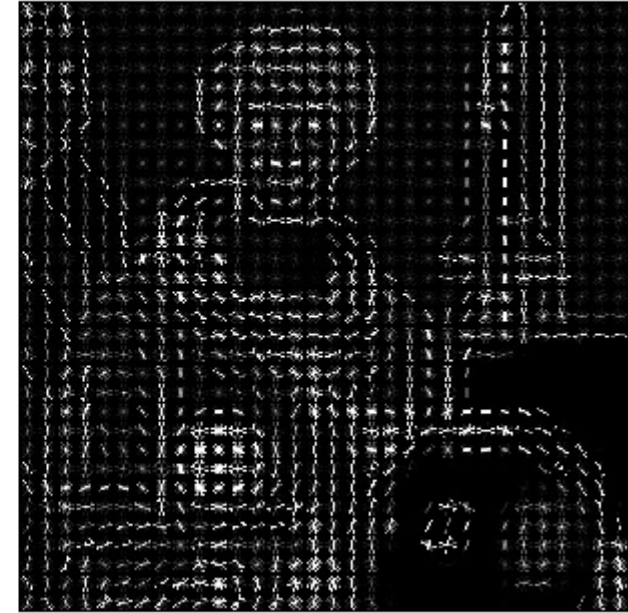
- VGG16 was best with Epochs=13 , DESNET121=10, RESNET50 = 10, RESNET=152 (Convolutional layer concept), Taking In consideration Sensitivity and Specificity

HOG Vs LBP

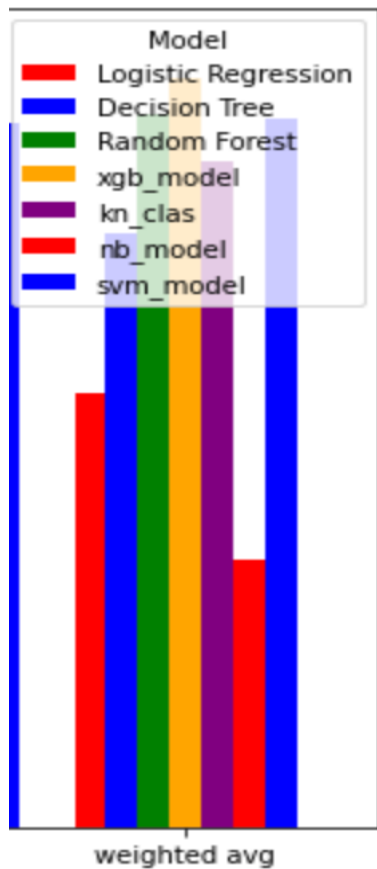
Input image



Histogram of Oriented Gradients



pixels



xgb
Random forest
SVM

lbp



xgb
Random forest
KNN

HOG



SVM
xgb
Logistic regression



Future work

- 1- Found another disease with a research gap on it for better novality
- 2- Balance between classes of the disease
- 3- Better and more accurate Pre-Processing steps on Images
- 4- Benchmarking of models and better model choose ,
In Deep learning : Optimizers, Epocs, learning rate
In Machine learning : SURF, SWIFT ,
give more try and tru Unsupervised ML