Computer Systems Engineering
CSE 263: Machine Learning and Pattern Recognition (Fall 2025)

Title: Medical Image Classification

Course: CSE263 (Machine Learning and Pattern Recognition)
Term: 1
Submission day: 16th of December 2025

S#	Student Name	Student ID
1	Yaseen Mohamed	247699
2	Mohamed Sameh	243557
3	Habiba Fekry	248773
4	Jana Mohamed	247035
5	Adham Hassan	242335

Marks		
Free Topic Description		**/40**
Machine learning solution	Problem formulation	**/40**
	Implementation/Evaluation /Sample output	**/60**
	Project interface & Poster Design	**/25**
	Report Style and Formatting	**/15**
	Presentation / response to questions	**/20**
	Creativity (**Bonus**)	**/10**
Total		**/200**

1.Problem Description

In this project, a system is developed to be able to categorize medical images in one of four categories:

- CT Lung Scan Covid Affected
- CT Lung Scan Normal
- X-ray Chest Covid Affected
- X-ray Chest Normal

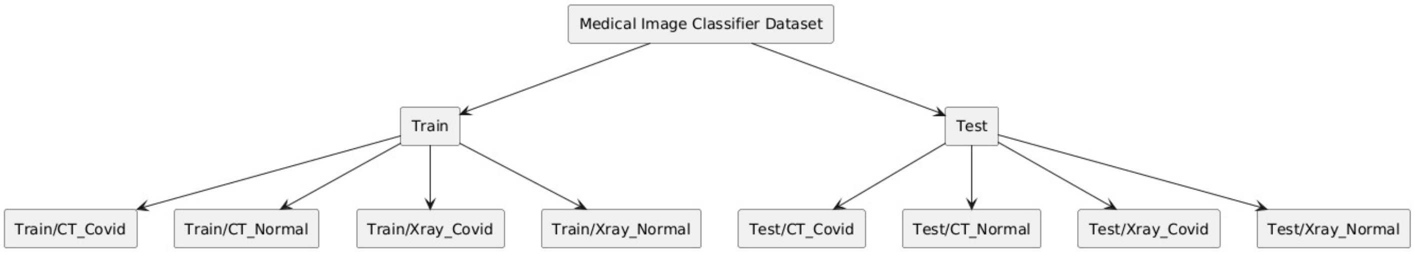
COVID-19 primarily attacks the lungs, causing inflammation, pneumonia, and potentially severe complications like ARDS (Acute Respiratory Distress Syndrome) by damaging tiny air sacs (alveoli), leading to fluid buildup, oxygen deprivation, scarring (fibrosis), and long-term issues. It might be revealed in CT scans and X-rays. An automatic pattern recognizer model of these images can be useful as an aid to analysis.

The main goal is to create a complete and fully working solution that works in the following Conceptual framework:

- Tests CT and X-ray images of a collection of data.
- Extracts features as numbers of each image.
- Trains with the training set.
- Evaluates the trained model on test images that have not been trained.
- Offers a straightforward application through which a user posts an image and receives a prediction.

2. Data Preparation

Data Hierarchy



- With this construction, the system automatically attaches the appropriate label depending on the folder where the image is kept.
- Image loading and validation will be done here.
- Path names of all image files are obtained by scanning the dataset folders. The only image formats which are used are common image formats: .png, .jpg, .jpeg

Some images in the dataset may be unreadable or corrupted. To prevent training failures, each image is validated during the loading process. If an invalid image is detected, it is safely skipped using error-handling mechanisms, ensuring that the training process continues without interruption.

That is one of the main reasons the following preprocessing data steps were used:

- **Resize:** All images are resized to 96 x 96 pixels to minimize the time consumed in computation and to ensure all the inputs are similar.

- **Channel processing:** In case an image is saved in an alpha format (RGBA), the alpha value is removed and the image is converted to RGB.
- **Grayscale conversion:** Before feature extraction, images are converted to grayscale, as feature extraction depends on patterns of intensity, and color is not necessary (HOG).
- **Feature Extraction (HOG):** Histogram of Oriented Gradients (HOG) is used to convert every image into a feature-length numerical value after preprocessing. HOG characterizes the image by edge direction and local gradient patterns that are useful in extracting significant texture of medical images
- **Batch Processing:** Because the dataset is large, features cannot be extracted from all images at the same time. Instead, the images are processed in groups of 200. This reduces memory usage and allows the program to run smoothly on standard hardware.

- **Output of data preparation:**
  By the conclusion of the stage, the data is transformed into:

  X_train, y_train for training

  X_test, y_test for evaluation

Where:

X is a HOG feature vectors extraction. y is a representation of the class labels.

3. Methods and Models

The Project Methodology is based on classical machine learning pipeline for the classification of medical images. The images are first converted into HOG feature vectors, which then undergo several steps of processing before training and testing the models.

Label Encoding has been used to convert the class labels from textual form into a numerical one for feeding into the model. Later on, features extracted are normalized using the StandardScaler that allows it to set the mean as 0 and standard deviation as 1. This is particularly important for SVM and PCA.

To decrease feature dimensionality and computational time, PCA is applied after scaling. PCA removes the redundant information from the data while preserving the most important features, selecting no more than 120 components automatically on the basis of the information contained in the data

Three different models have been trained: **Support Vector Machine** (RBF kernel), **KNearest Neighbors** (k = 5), and **Random Forest**. Where applicable, class weighting is used in an attempt to handle class imbalance

Actually, the ensemble classifier does an excellent job of combining three models into one by hard voting to improve stability and lift overall performance. At the end, it will pick up all the trained components: the scaler, PCA transformer, label encoder, and voting model into one single file that later could be used within the GUI application.

4. Results and Evaluation

4.1 Training Data Distribution

The training data is distributed among the different classes. It can be seen that the dataset is not evenly distributed:

- CT_Covid and Xray_Normal contain the largest number of samples
- CT_Normal has the smallest number of images

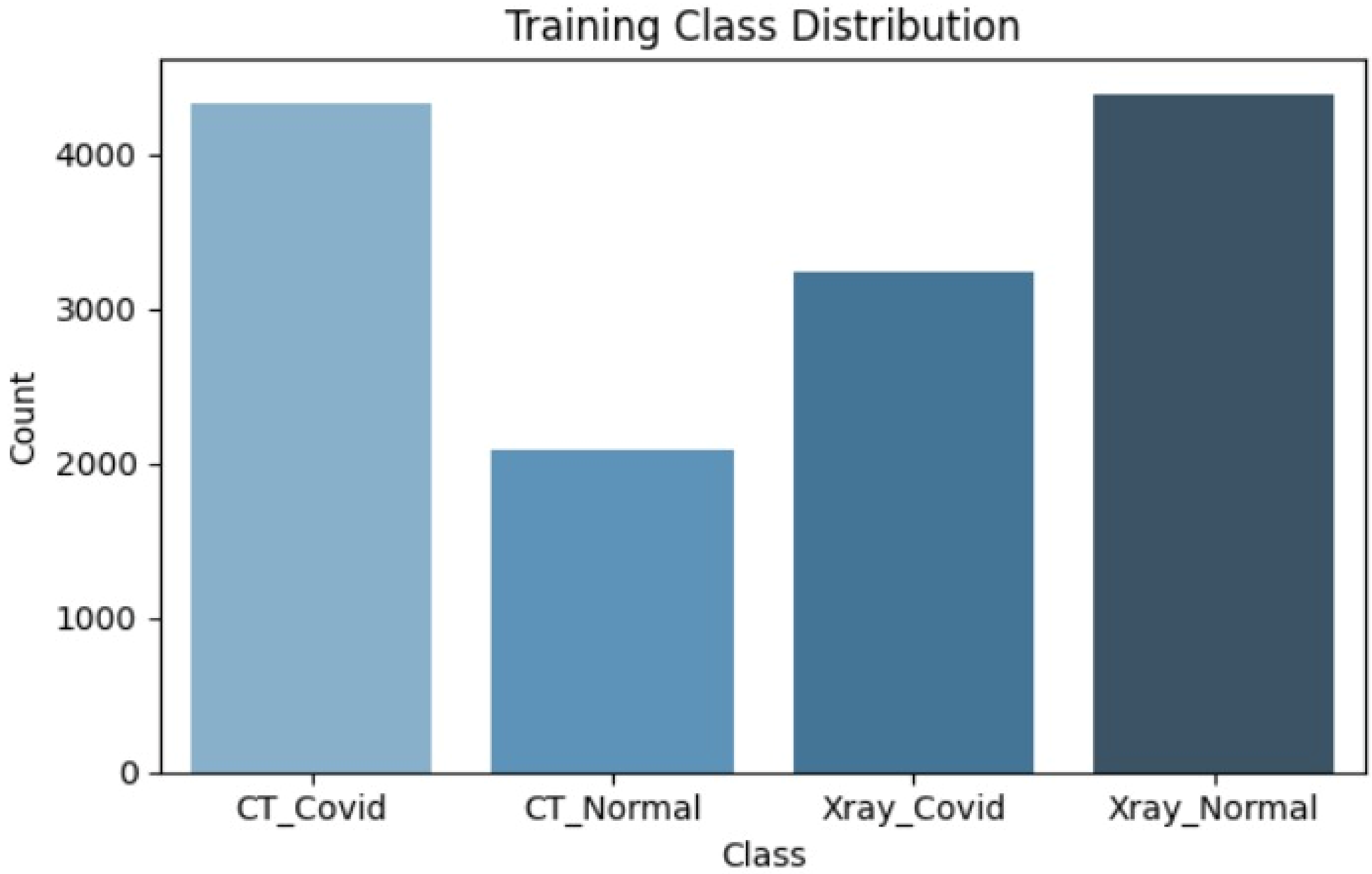
Because of this imbalance, class weights were used during model training. This also explains why the recall values differ between classes in the final results.

4.2 Dataset Size Summary

After loading and validating the dataset, the following numbers of images were used:

Training Set

- CT_Covid: 4320 images
- CT_Normal: 2079 images
- Xray_Covid: 3234 images
- Xray_Normal: 4389 images



Test Set

- CT_Covid: 1080 images
- CT_Normal: 525 images
- Xray_Covid: 810 images
- Xray_Normal: 1104 images

These values show that a large and diverse test set was used, which makes the evaluation results more reliable.

4.3 Overall Performance (Accuracy)

Accuracy: 0.71

Macro Average:

Precision: 0.77   Recall: 0.76   F1-score: 0.71

Weighted Average:

Precision: 0.82   Recall: 0.71   F1-score: 0.71

Total Samples: 3519

- The model achieved an overall accuracy of 71.01% on the test dataset.
- This means that the system correctly classified around 7 out of every 10 images.

4.4 Classification Report Analysis

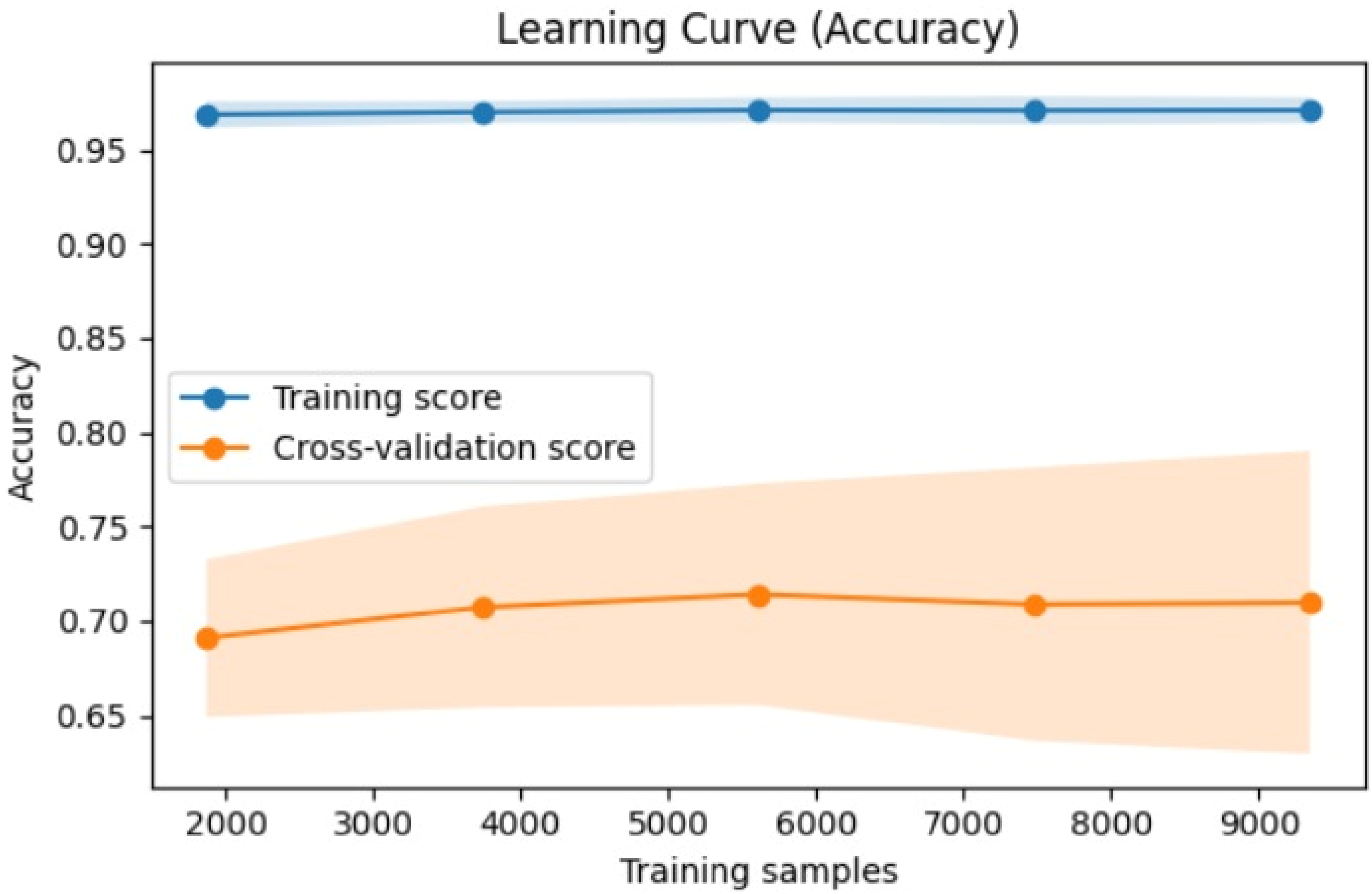
	precision	recall	f1-score	support
CT_Covid	0.96	0.61	0.74	1080
CT_Normal	0.54	0.95	0.69	525
Xray_Covid	0.59	0.98	0.74	810
Xray_Normal	0.98	0.50	0.66	1104

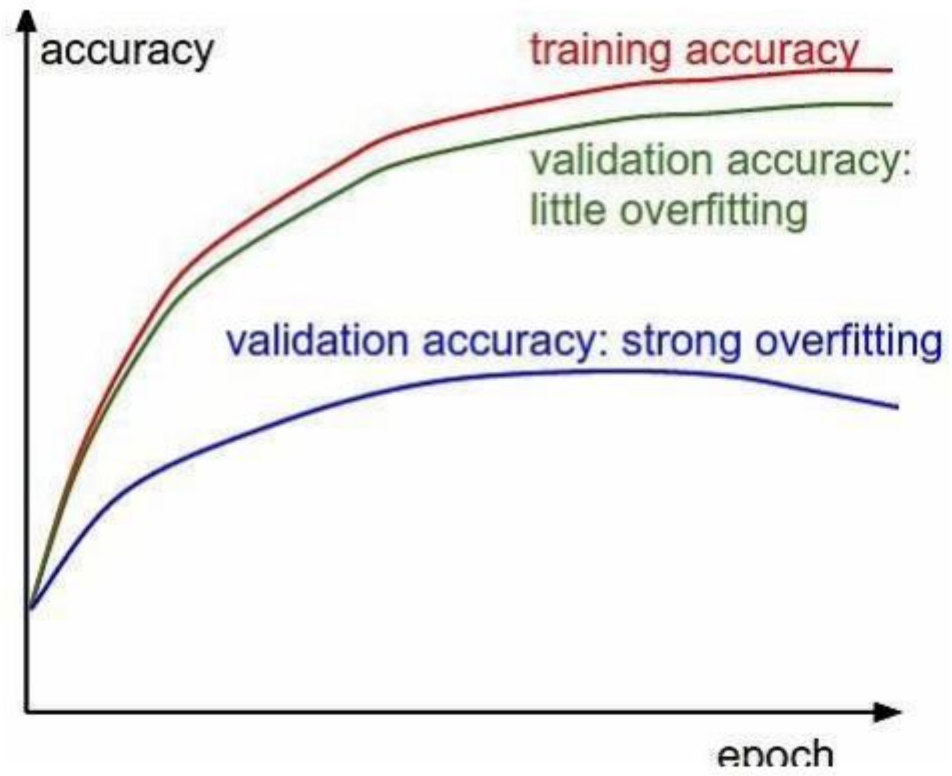
- For CT_Covid, the model achieves very high precision (0.96), which means that most images predicted as COVID CT scans are correct. However, the recall is 0.61, indicating that some CT COVID cases were missed.
- For CT_Normal, the recall is high (0.95), showing that most normal CT images were correctly identified. The precision is lower (0.54), which means that some normal cases were confused with other classes.
- For Xray_Covid, the model shows very high recall (0.98), indicating strong detection of COVID cases in X-ray images. The precision is moderate (0.59), suggesting the presence of some false positives.
- For Xray_Normal, the precision is very high (0.98), but the recall is lower (0.50). This means that while predicted normal X-ray images are usually correct, the model fails to detect all normal cases.
- Overall, the weighted F1-score of 0.71 indicates a reasonable level of performance, especially when considering the class imbalance present in the dataset.

4.5 Learning Curve Analysis

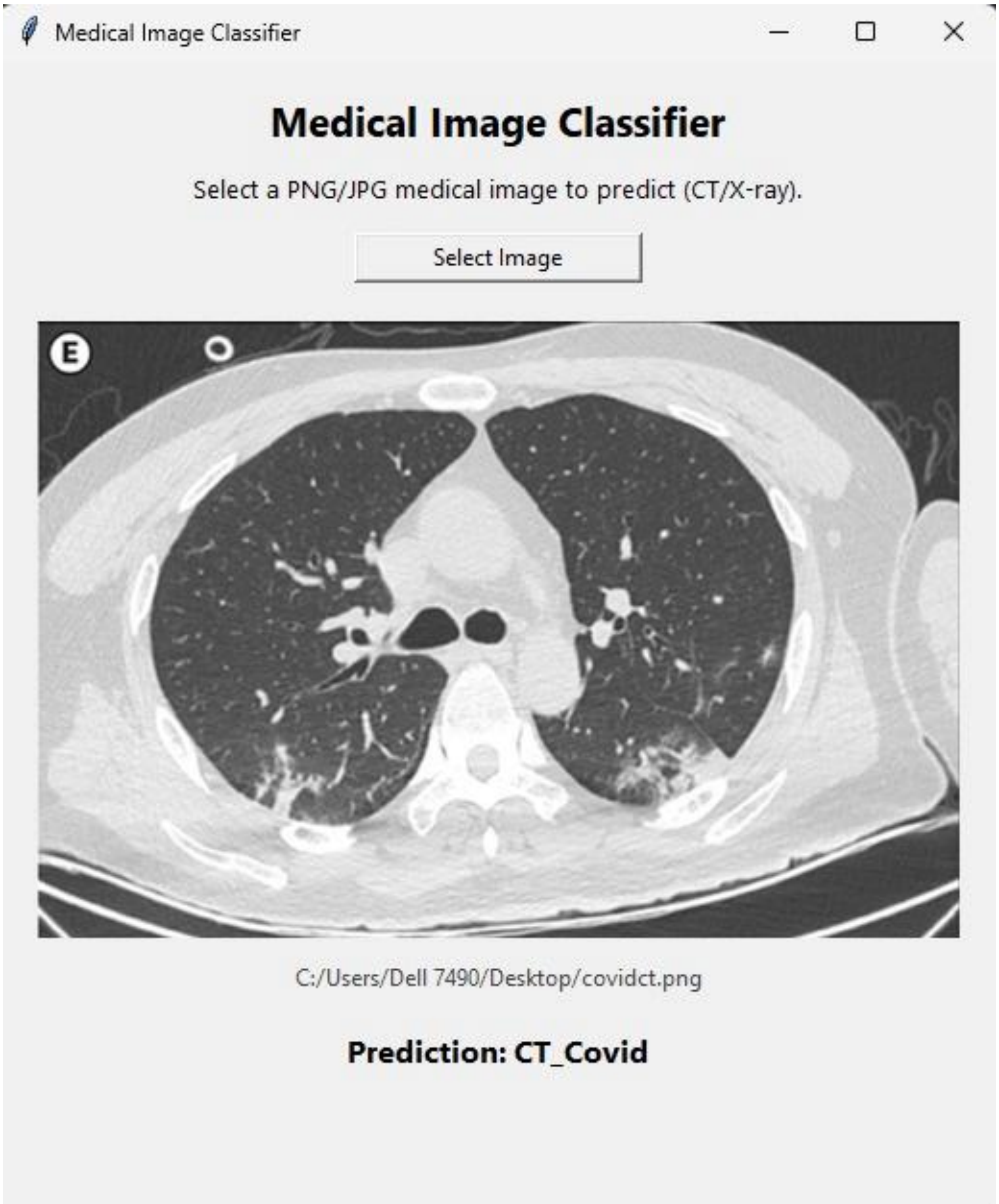
The learning curve shows how the model's performance changes as more training data is added:

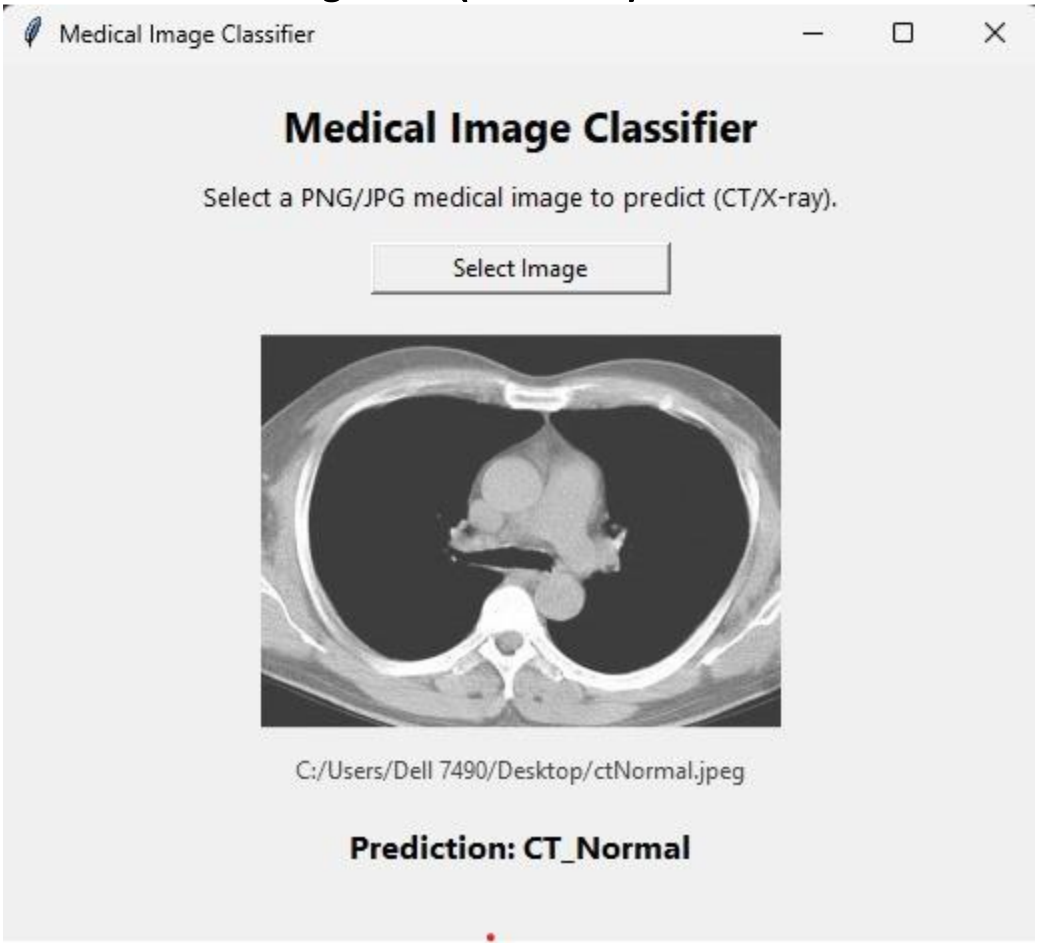
- Training accuracy remains consistently high
- Validation accuracy increases gradually and then stabilizes around 71%
- The gap between training and validation accuracy suggests mild overfitting, which is common when using classical machine learning with handcrafted features
- This suggests that adding more data may help slightly, but better feature representations would likely lead to bigger improvements.


Learning Curve (Accuracy)
Accuracy
0.95
0.90
0.85
0.80
0.75
0.70
0.65
Training score
Cross-validation score
2000
3000
4000
5000
6000
7000
8000
9000
Training samples

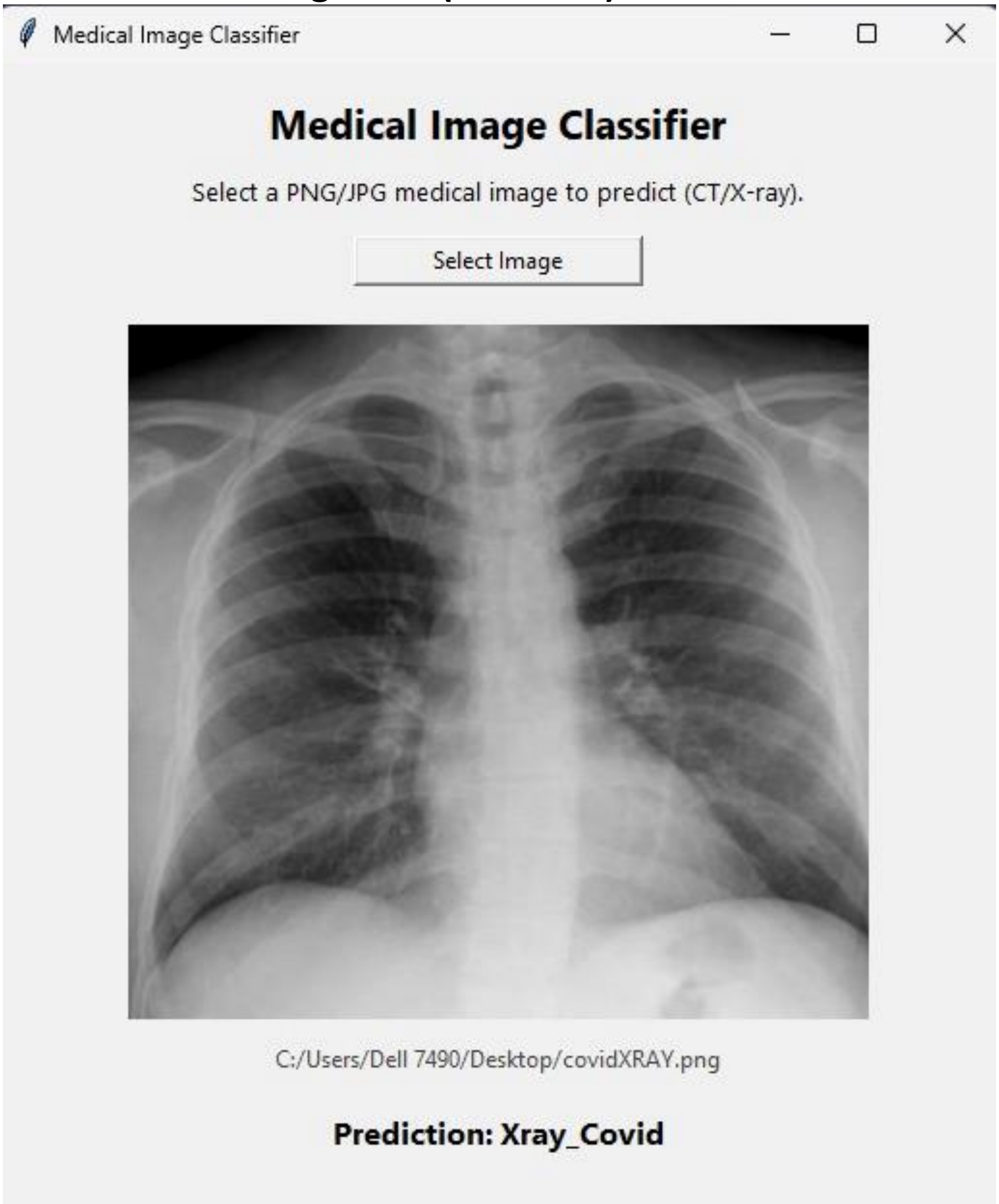

accuracy
training accuracy
validation accuracy:
little overfitting
validation accuracy: strong overfitting
epoch

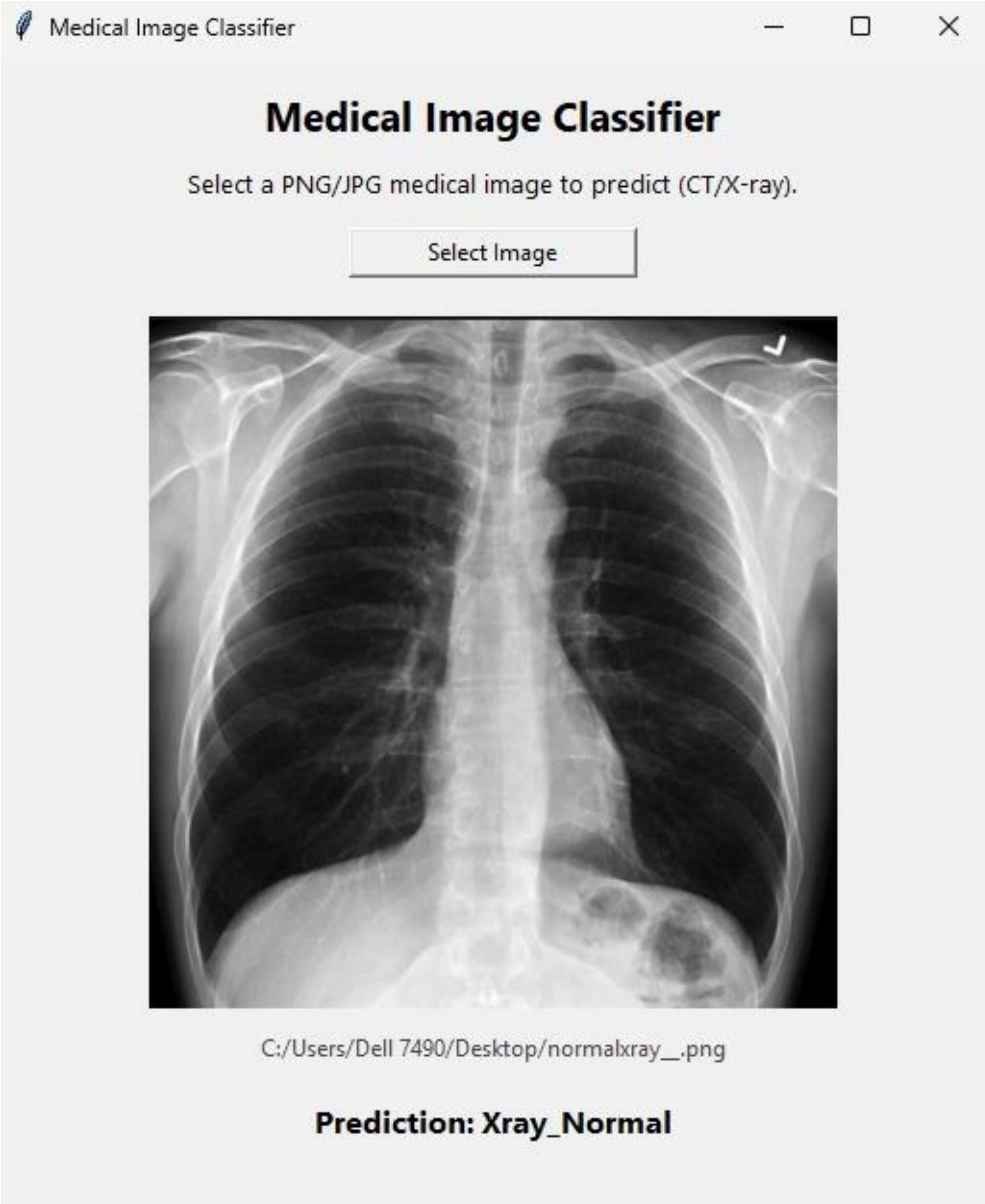
4.6 Sample Output




Medical Image Classifier
Medical Image Classifier
Select a PNG/JPG medical image to predict (CT/X-ray).
Select Image
C:/Users/Dell 7490/Desktop/ctNormal.jpeg
Prediction: CT_Normal

Computer Systems Engineering
CSE 263: Machine Learning and Pattern Recognition (Fall 2025)




Medical Image Classifier
Medical Image Classifier
Select a PNG/JPG medical image to predict (CT/X-ray).
Select Image
C:/Users/Dell 7490/Desktop/normalxray__.png
Prediction: Xray_Normal

5. Discussion

5.1 Results

The results show that the model performs reasonably well using classical machine learning methods on a mixed dataset of CT scans and X-ray images. The lower recall in some classes can mainly be explained by:

- **Class imbalance** in the dataset
- **Subtle visual difference**s in X-ray images
- **Limitations of HOG features** in capturing complex medical patterns

Despite these challenges, the system produces consistent and interpretable results and demonstrates a complete and functional machine learning pipeline.

5.2 Project

This project demonstrates a classical machine learning technique that can reach a fair level of performance in medical image classification of chests. The accuracy level attained in this project is approximately 71%, which is quite good in such a challenging task performed using a classical approach.

The analysis shows that performance differs from class to class. Some classes with a large amount of training data tend to perform better, but some with less data tend to be impacted by misclassification. The model can classify CT images easily since they have clearer visual information. X-ray images have more similarities and differences in a subtle way between COVID and Normal cases, hence lower recall for Xray_Normal.

5.3 Limitations

Even though class weights were used, the model is still affected by unequal data, especially for classes with fewer images. The model learns the training data well, but it does not improve much on new data, which shows some overfitting. Adding more data might help a little, but bigger improvements would likely come from using better feature extraction methods or more advanced models. (Deep Learning would also help for example CNN)

6. Appendix

6.1 Feature Extraction

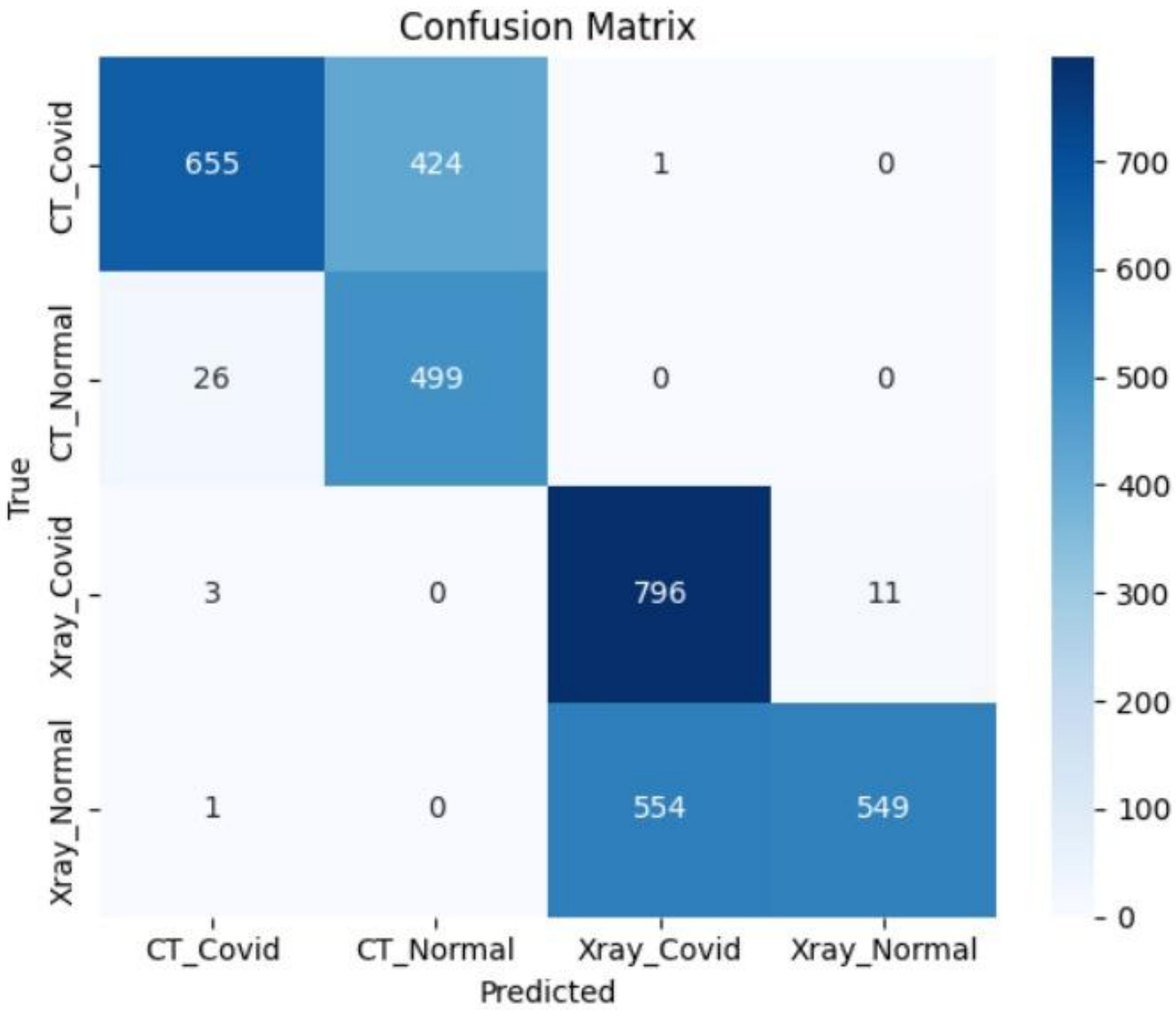
All images are resized to 96 × 96, converted to grayscale, and represented using HOG features with fixed parameters.

6.2 Model Components Saved

The trained system saves the following components in one file for later use:

- StandardScaler
- PCA transformer
- Label Encoder
- Voting Classifier (SVM, KNN, Random Forest).

6.3 Confusion Matrix



Computer Systems Engineering
CSE 263: Machine Learning and Pattern Recognition (Fall 2025)

Task Assignment Page

Model Development & System Design (UI): Yaseen Mohamed, Habiba Fekry

- ✓ Implemented the main machine learning pipeline used in the project.
- ✓ Performed image preprocessing and extracted features using HOG.
- ✓ Trained and tuned classical machine learning models including SVM, KNN, and Random Forest.
- ✓ Implemented the voting ensemble classifier to improve model stability.
- ✓ Evaluated the trained models and analyzed their performance Integrated the final trained models into the application.
- ✓ Designed the overall system workflow and project structure.
- ✓ Developed the conceptual design of the classification pipeline.
- ✓ Designed and tested the user interface for image-based prediction.
- ✓ Ensured that the application output is clear and easy to use

Dataset Handling, Experimental Setup & Documentation: Adham Hassan, Mohamed Sameh, Jana Badry

- ✓ Collected and organized the CT scan and chest X-ray image datasets.
- ✓ Prepared the training and testing splits used in experiments.
- ✓ Checked dataset integrity and handled corrupted or unreadable images.
- ✓ Managed dataset loading and batch processing during experiments.
- ✓ Wrote and edited different sections of the project report.
- ✓ Prepared the free topic report and poster content.
- ✓ Analyzed and interpreted experimental results.
- ✓ Organized figures, tables, and evaluation summaries