LUNG CANCER DETECTION

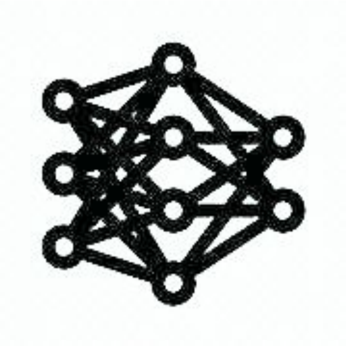
Using CNN

Shubham Kukadiya - 202411066
Jaydeep Darji - 202411029
Khelan Bhatt - 202411025

IT582-FOML
Prof. MV Joshi

Lung Cancer Detection

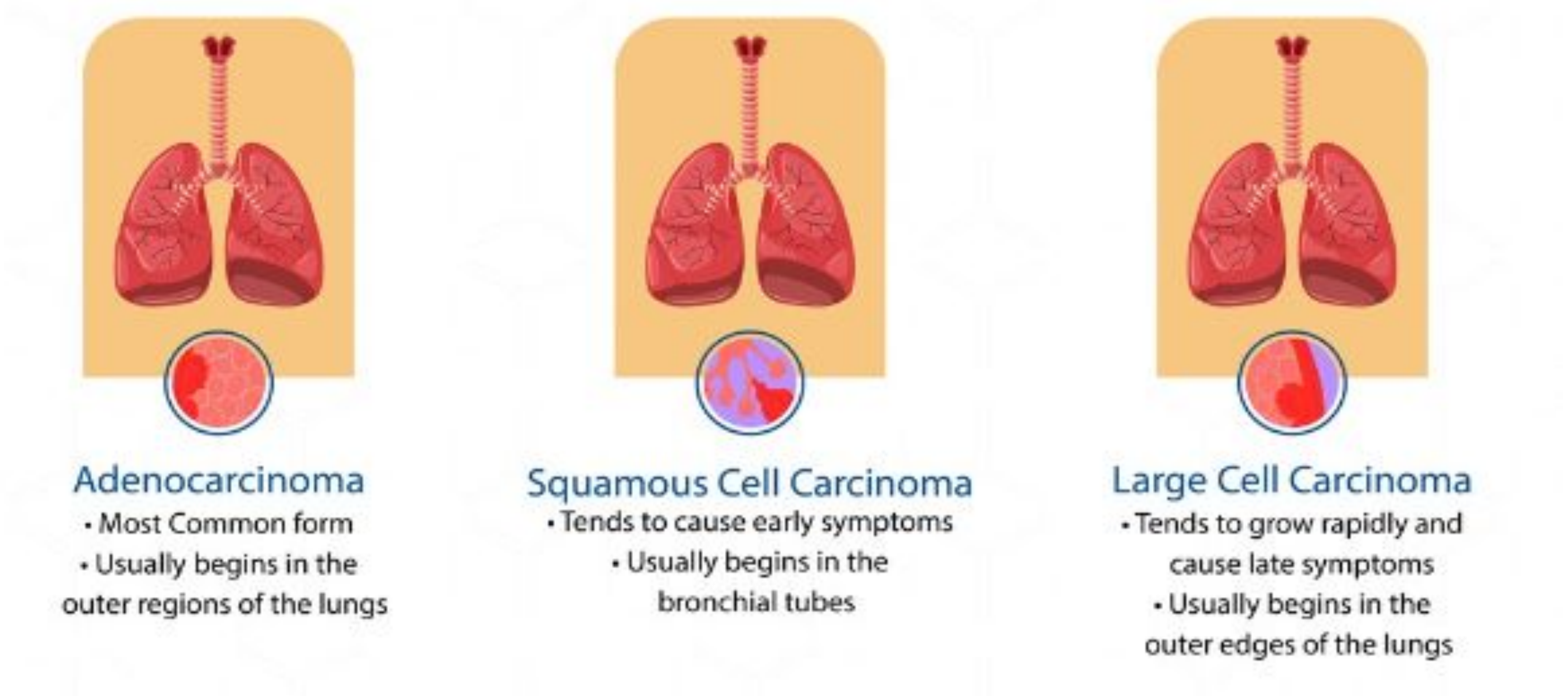
Chest CT-Scan images Dataset



Model

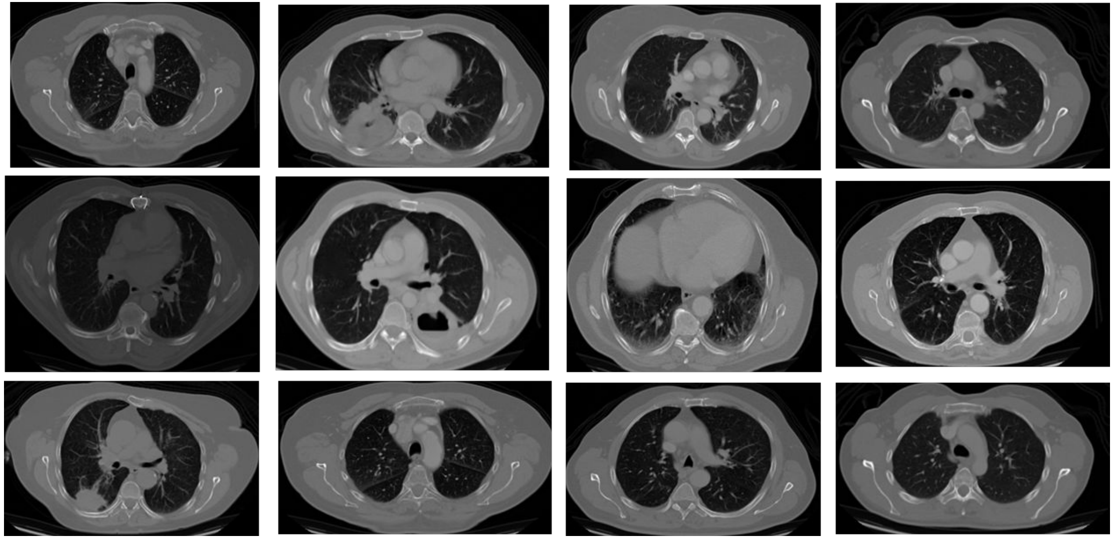
Determine Which type of Lung cancer patient has.

Three Main Types Of Lung Cancer



https://www.lalpathlabs.com/blog/types-of-lung-cancer/

DATA



The dataset[1] contains chest CT scan images.

It is sourced from Kaggle and includes images suitable for training, validation, and testing.

0

[1] https://www.kaggle.com/datasets/mohamedhanyyy/chest-ctscan-images/data

Introduction

❖ **Overview :**

- Lung cancer is the leading cause of cancer-related deaths, with late diagnosis driving its high mortality rate. CNN-based image analysis shows potential for early detection by automating diagnostics [2]

❖ **Key Highlights :**

- A VGG16 Model of CNN [3] is used for lung cancer detection. It enhances diagnostic speed and improves accuracy and analyzes CT scans to identify malignancies.

[2] Ren, S., He, K., Girshick, R., & Sun, J. (2015). *Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks*. Proceedings of Advances in Neural Information Processing Systems (NeurIPS 2015), 91-99

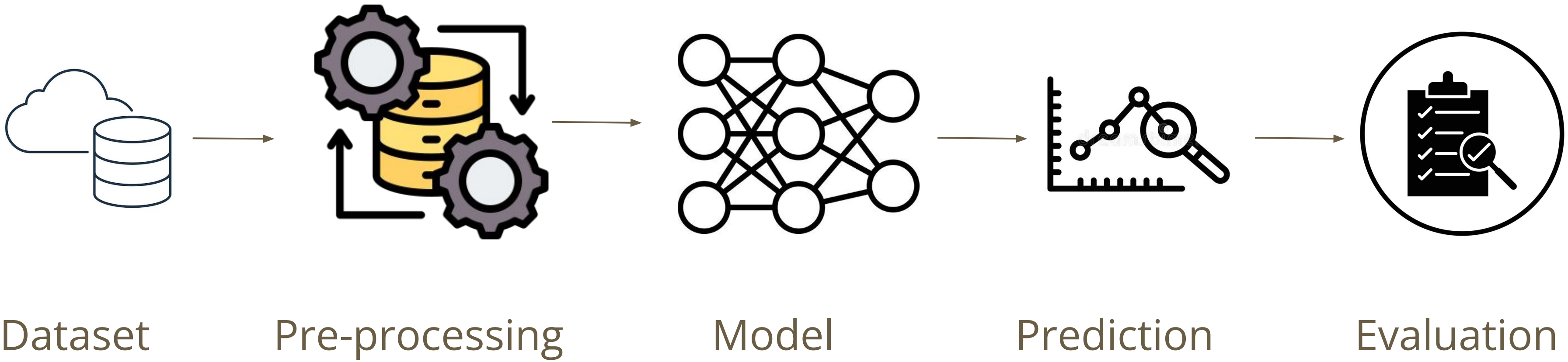
[3] Fang, A., Kornblith, S., & Schmidt, L. (n.d.). *Does progress on ImageNet transfer to real-world datasets?* University of Washington & Google Research, Brain Team.

Motivation

- Early detection significantly improves survival rates, reduces treatment costs, and enhances the quality of life. CNN-based models, particularly VGG16, help detect lung cancer faster and more accurately by analyzing CT scans. This scalable approach ensures reliable and timely diagnoses, which is essential for improving patient outcomes [4]

[4] Zhang, L., Li, H., and Zhao, X. "SimAM: A Parameter-Free Attention Module for CNNs in Medical Imaging." ICML, 2021.

Methodology



Methodology

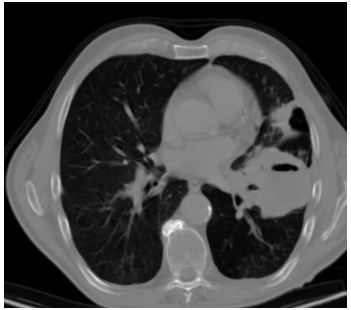
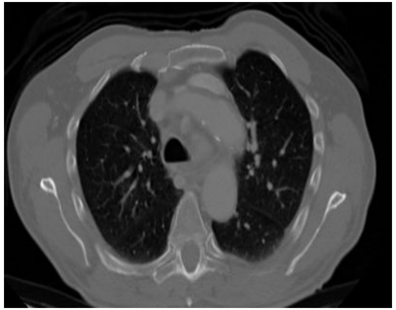
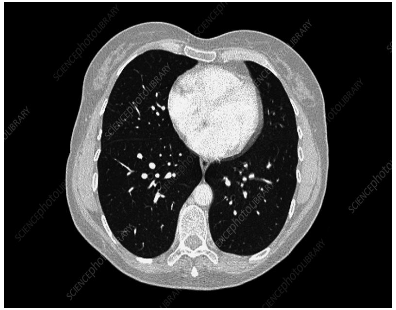
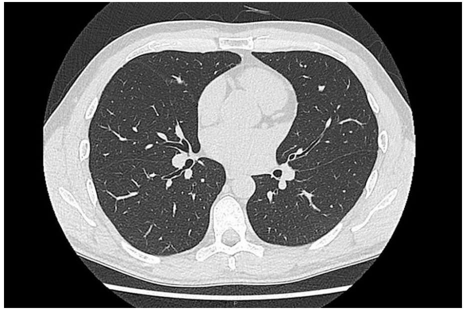
- **Process Flow:**
  - **Input:** Labeled CT scan images
  - **Model:** Feature extraction using a pre-trained VGG16 model
  - **Output:** Predicted class probabilities for each image.

- **Key Steps:**
  1) Data Preparation and Preprocessing.
  2) Model Architecture and Design.
  3) Training and Optimization.
  4) Evaluation and Metrics.

1) Data Preparation and Preprocessing

- **Data Collection:**

  - CT scan images collected and categorized into training, validation, and testing datasets.
  - Images are labeled to enable supervised learning.

1) Data Preparation and Preprocessing

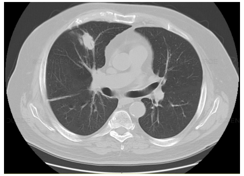
❑ Data Preprocessing

- **Resizing**: All images are resized to 224×224×3 to ensure uniform input dimensions.
- **Normalization**: Pixel values are scaled to a range of [0, 1] or standardized for faster model convergence.
- **Augmentation**: Introduces variations such as rotation, flipping, and zooming to enhance model robustness.

$$z = \frac{x - \mu}{\sigma}$$

Working of CNN layers

Convolutional Layer :



Input Matrix (X)

0	0	0	0	0	0	0
0	0	2	1	2	0	0
0	1	2	2	0	2	0
0	0	2	0	1	1	0
0	0	0	0	1	0	0
0	0	1	2	2	1	0
0	0	0	0	0	0	0

*

Filter (W)

0	1	1
-1	1	-1
0	1	1

f x f

=

Feature Map

1	5	-1	3	0
3	4	4	0	3
1	6	0	3	2
3	5	4	6	1
-1	-1	0	0	-1

5 x 5

$$\text{Output Size} = \left[\frac{n+2p-f}{s}+1\right] \times \left[\frac{n+2p-f}{s}+1\right]$$

n = 5
f = 3
p = 1
s = 1

Working of CNN layers

Convolutional Layer :

$$Z_{i,j} = \sum_{k,l} X_{i+k,j+l} \cdot W_{k,l}$$

(0*0) + (0*1) + (0*1) + (0*-1) + (0*1) + (2*-1) + (0*0) + (1*1) + (2*1) = 1

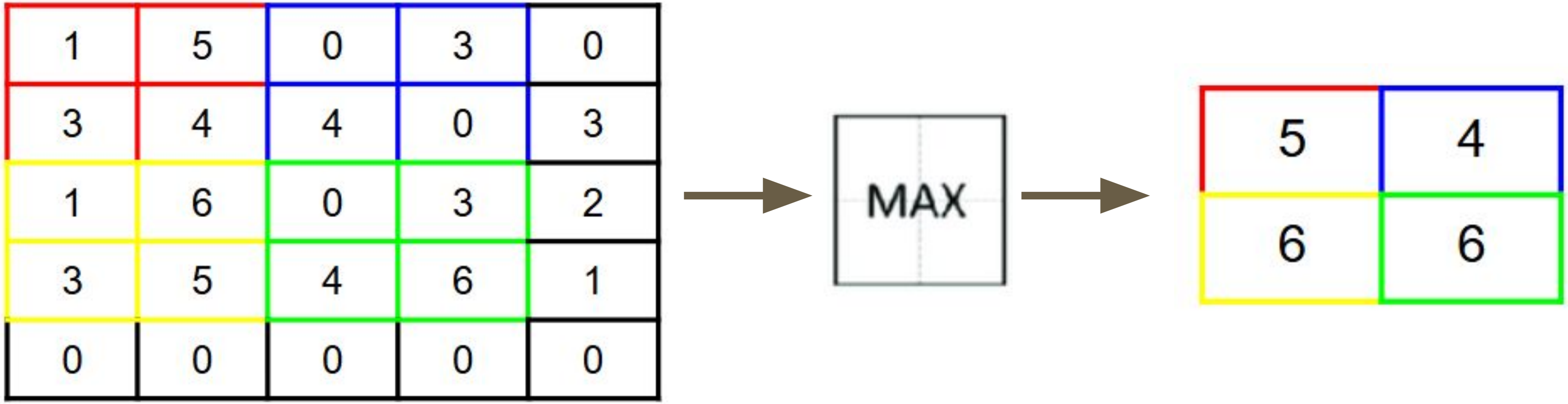
ReLU Activation : The ReLU activation function replaces all negative values in the feature map with 000, while keeping positive values unchanged.

1	5	0	3	0
3	4	4	0	3
1	6	0	3	2
3	5	4	6	1
0	0	0	0	0

Working of CNN layers

Pooling Layer :

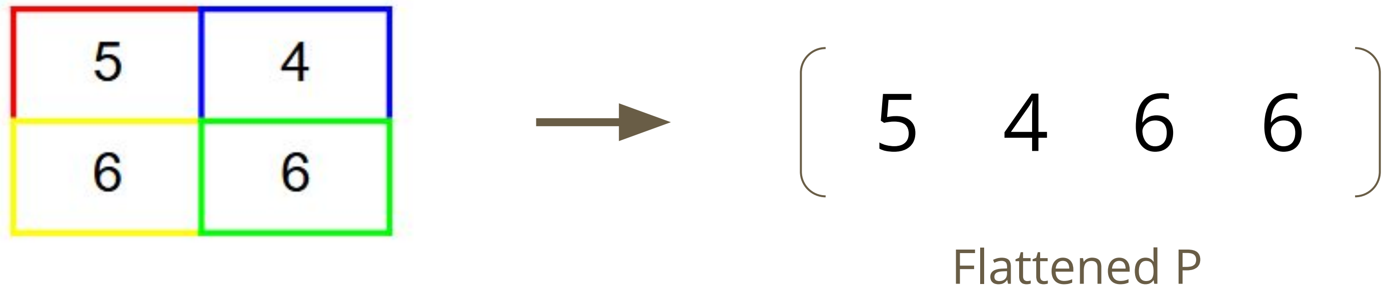
We perform **Max Pooling** with a 2×2 filter and stride 2.



Working of CNN layers

Flattening :

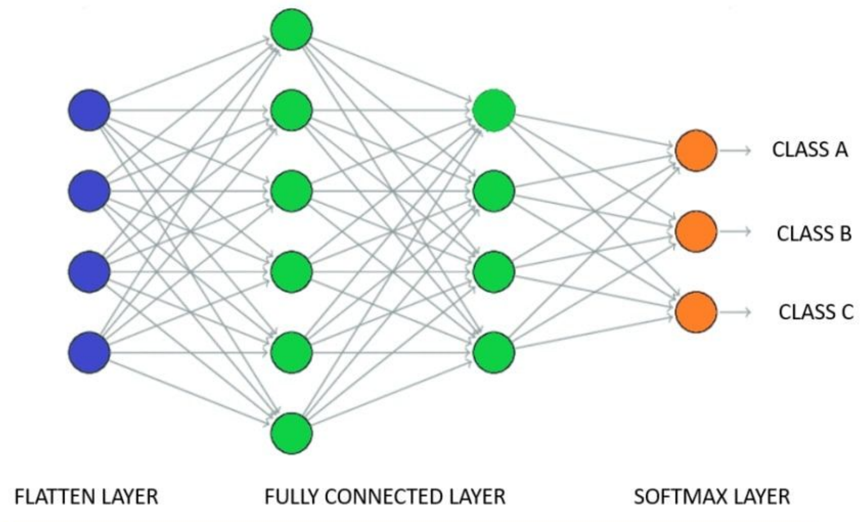
Flatten the pooled feature map into a 1D vector :



Working of CNN layers

Fully Connected Layer :

- Feature Integration
- Dimensional Reduction
- Non-Linear Activation
- Output Probabilities:



https://indiantechwarrior.com/fully-connected-layers-in-convolutional-neural-networks/

Working of CNN layers

Fully Connected Layer :

Suppose:

Edge weights (W) =

0.5	0.2	-0.3	0.8
-0.1	0.4	0.6	-0.2
0.2	-0.5	0.3	0.7
0.4	0.1	-0.2	-0.6

Bias (b) =

0.1	0.2	-0.1	0.05

Working of CNN layers

Fully Connected Layer :

Y = W * Flattened P + b

$$Y = \begin{bmatrix} 0.5 & 0.2 & -0.3 & 0.8 \\ -0.1 & 0.4 & 0.6 & -0.2 \\ 0.2 & -0.5 & 0.3 & 0.7 \\ 0.4 & 0.1 & -0.2 & -0.6 \end{bmatrix} * \begin{bmatrix} 5 & 4 & 6 & 6 \end{bmatrix} + \begin{bmatrix} 0.1 & 0.2 & -0.1 & 0.05 \end{bmatrix}$$

Working of CNN layers

Fully Connected Layer :

Y = W * Flattened P + b

$$y1 = (0.5 \cdot 5) + (0.2 \cdot 4) + (-0.3 \cdot 6) + (0.8 \cdot 6) + 0.1 = 6.4$$

$$y2 = (-0.1 \cdot 5) + (0.4 \cdot 4) + (0.6 \cdot 6) + (-0.2 \cdot 6) + 0.2 = 3.7$$

$$y3 = (0.2 \cdot 5) + (-0.5 \cdot 4) + (0.3 \cdot 6) + (0.7 \cdot 6) - 0.1 = 4.9$$

$$y4 = (0.4 \cdot 5) + (0.1 \cdot 4) + (-0.2 \cdot 6) + (-0.6 \cdot 6) + 0.05 = -2.35$$

Class Scores Y = [6.4, 3.7, 4.9, −2.35]

Working of CNN layers

Softmax Activation :

Convert the scores into probabilities using softmax:

P(y1) = 601.84 / 776.69 = 0.775

P(y2) = 40.45 / 776.69 = 0.052

P(y3) = 134.29 / 776.69 = 0.173

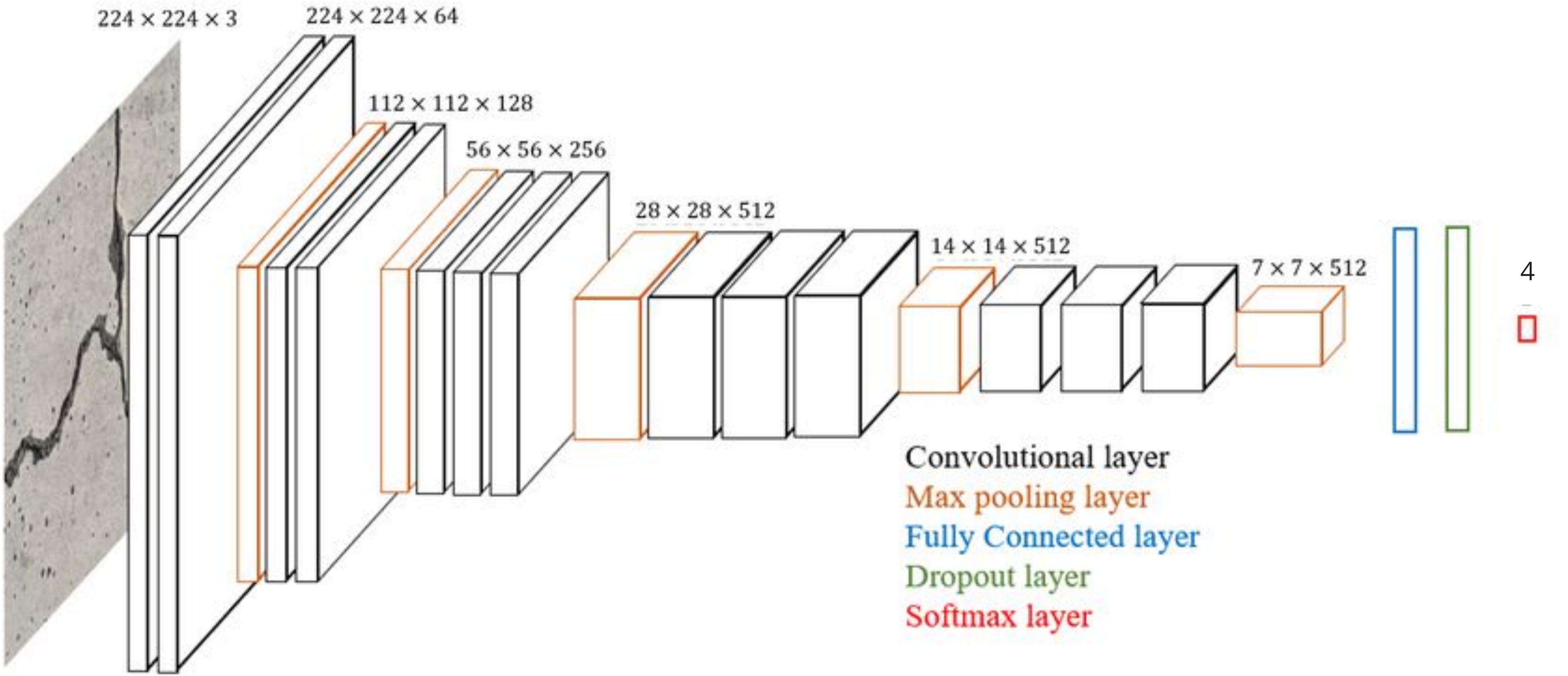
P(y4) = 0.095 / 776.69 = 0.0001

$$P(y_i) = \frac{e^{y_i}}{\sum_{j=1}^{C} e^{y_j}}$$

So, the predicted class is **Class 0** with the highest probability **P(y1) = 77.5%**

P=[0.775, 0.052, 0.173, 0.0001]

2) Model architecture



https://www.researchgate.net/figure/Architecture-of-the-modified-VGG16-model_fig1_350828239

VGG16

Overview

A deep convolutional neural network introduced by the Visual Geometry Group (VGG).

16 weighted layers: 13 convolutional layers and 3 fully connected layers.

Key Features

Small filter sizes - efficient and dense feature extraction.

Enables hierarchical feature learning (from basic edges to complex patterns).

Why VGG16?

- Simplicity
- Transfer Learning
- Feature Extraction
- High Accuracy

3) Training and Optimization

- ❑ The model was trained over **100 epochs** using the training dataset.
  - ➢ **Training Accuracy:** Improved from **38.16% (Epoch 1)** to **93.01% (Epoch 100)**.
  - ➢ **Training Loss:** Reduced from **1.8332 (Epoch 1)** to **0.2139 (Epoch 100)**.

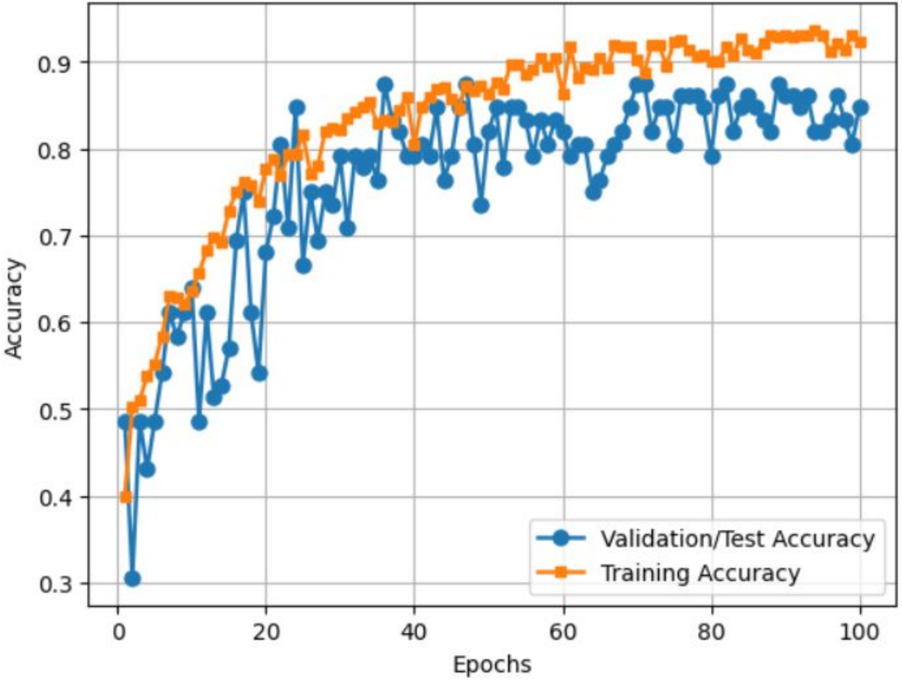
Loss Function: Categorical Cross-Entropy

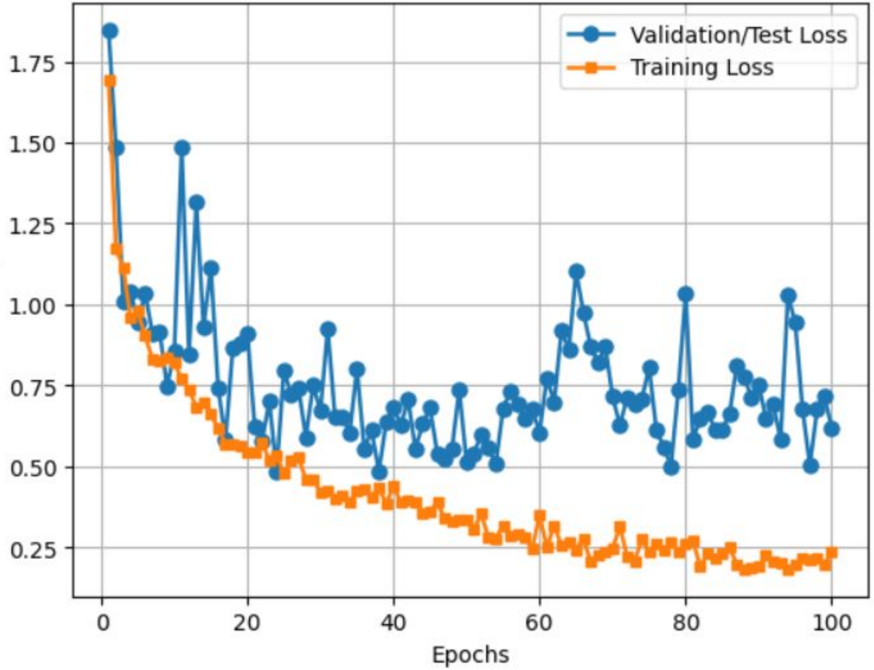
$$L = -\sum_{i=1}^{N} y_i \log(\hat{y}_i)$$

Adam optimizer

$$w_{t+1} = w_t - \eta \nabla L(w_t)$$

3) Training and Optimization





4) Evaluation and metrics

- Final test accuracy achieved: **86.08%**
- Test loss reduced to **0.3796**

$$Accuracy = \frac{TN + TP}{TN + FP + TP + FN}$$

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

$$F1\ Score = 2 * \frac{Precision * Recall}{Precision + Recall}$$

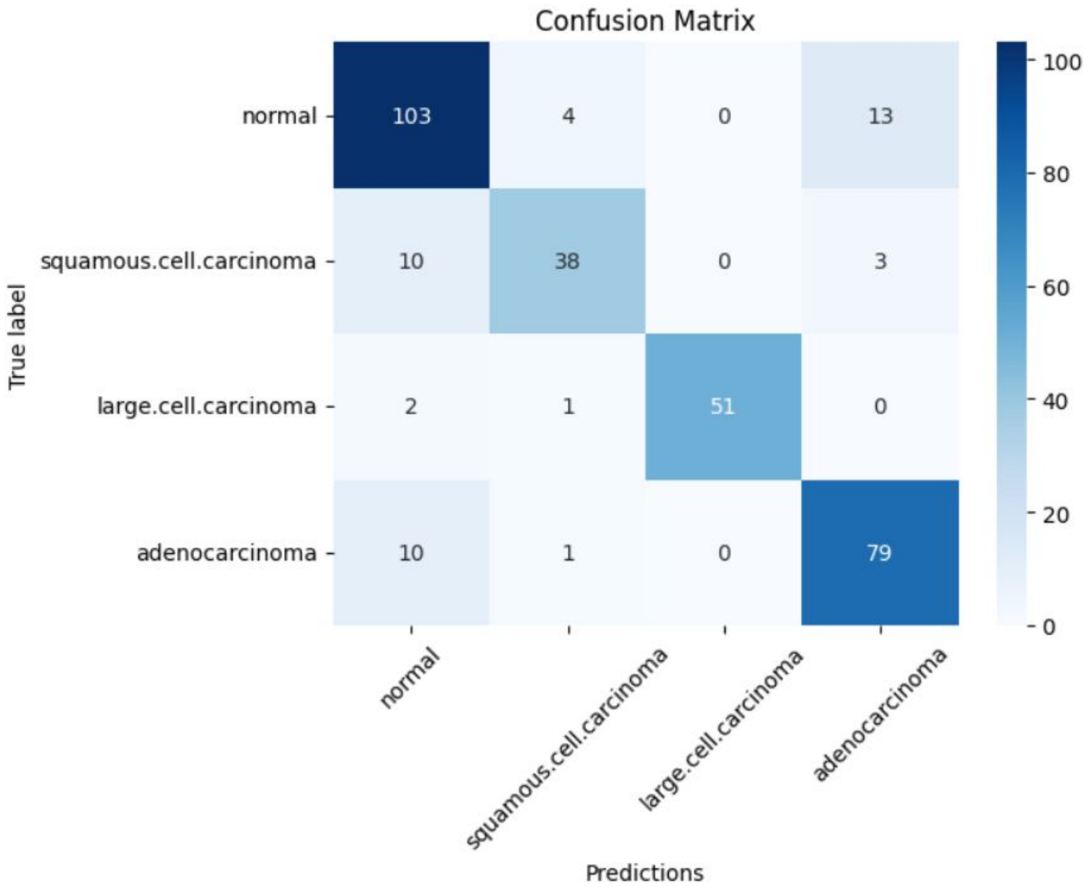
4) Evaluation and metrics

```
20/20 ━━━━━━━━━━━━━━━━━━━━ 5s 187ms/step
              precision    recall  f1-score   support

           0       0.82      0.86      0.84       120
           1       0.86      0.75      0.80        51
           2       1.00      0.94      0.97        54
           3       0.83      0.88      0.85        90

    accuracy                           0.86       315
   macro avg       0.88      0.86      0.87       315
weighted avg       0.86      0.86      0.86       315
```

Classification Report Highlights:

- **Normal Class**: Precision: **82%**, Recall: **86%**, F1-Score: **84%**.
- **Squamous Cell Carcinoma**: Precision: **86%**, Recall: **75%**, F1-Score: **80%**.
- **Large Cell Carcinoma**: Precision: **100%**, Recall: **94%**, F1-Score: **97%**.
- **Adenocarcinoma**: Precision: **83%**, Recall: **88%**, F1-Score: **85%**.

4) Evaluation and metrics



Conclusion

Summary

- Pre-trained CNNs like VGG16 are effective for lung cancer detection.
- Achieved **~86%** accuracy with strong generalization in medical imaging.

Key Outcomes

- Early detection improves treatment outcomes and reduces mortality.
- Automation via CNNs enhances efficiency and scalability in diagnosis

Challenges Addressed

- Robust data preprocessing handled CT scan variations.
- Overfitting reduced [5] with data augmentation and dropout.

Final Thought

- AI-driven diagnostics [6] can transform healthcare with clinical integration.

[5] Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). "ImageNet Classification with Deep Convolutional Neural Networks." In Advances in Neural Information Processing Systems 25 (NeurIPS 2012).

[6] Fintelmann, F., Wohlwend, J., Mikhael, P., Sequist, L. V., & Barzilay, R. (2023). Sybil: A Deep Learning Approach for Lung Cancer Risk Assessment Using CT Scans. Proceedings of the Advances in Neural Information Processing Systems (NeurIPS 2023)

THANK YOU!