MANIPAL
ACADEMY of HIGHER EDUCATION
(Institution of Eminence Deemed to be University)

MANIPAL SCHOOL OF INFORMATION SCIENCES

(A Constituent unit of MAHE, Manipal)

Development of Deep Learning Approach for Grading Squamous Cell Carcinoma from Histopathology Images

Reg. Number	Name	Branch
251100610012	Adithya Rao Kalathur	M.E. Computer Science and Engineering
251100610013	Vaishnav P S	M.E. Computer Science and Engineering
251100610017	Sriram K	M.E. Computer Science and Engineering
251100610040	Vaibhav V Acharya	M.E. Computer Science and Engineering

Under the guidance of

Dr Keerthana Prasad
Professor & Director,
Manipal School of Information Sciences,
MAHE, MANIPAL

16/10/2025

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

Abbreviation	Full Form
SCC	Squamous Cell Carcinoma
CNN	Convolutional Neural Network
GPU	Graphics Processing Unit
AUC	Area Under the Curve
ROC	Receiver Operating Characteristic
ReLU	Rectified Linear Unit
SGD	Stochastic Gradient Descent
GAP	Global Average Pooling
KD	Knowledge Distillation
LIDAR	Light Detection and Ranging
MLP	Multi-Layer Perceptron
ViT	Vision Transformer
SVM	Support Vector Machine
ECOC	Error-Correcting Output Codes
MCC	Matthews Correlation Coefficient
CT	Computed Tomography
FLOPs	Floating Point Operations
ISIC	International Skin Imaging Collaboration
BUSI	Breast Ultrasound Images
PIL	Python Imaging Library
JPEG	Joint Photographic Experts Group
CIFAR	Canadian Institute for Advanced Research
ImageNet	Large Visual Database for Visual Object Recognition
LC25000	Lung and Colon Cancer Histopathological Images Dataset (25,000 images)
AdamW	Adam with Weight Decay
RMSprop	Root Mean Square Propagation
L2	L2 Regularization (Ridge Regression)

F1	F1-Score (Harmonic Mean of Precision and Recall)
Grad-CAM	Gradient-weighted Class Activation Mapping
IoT	Internet of Things
I-SMAC	IoT in Social, Mobile, Analytics and Cloud
ICAIIC	International Conference on Artificial Intelligence in Information and Communication
ICAISC	International Conference on Applied Intelligence and Sustainable Computing
AAAA	Association for the Advancement of Artificial Intelligence
IEEE	Institute of Electrical and Electronics Engineers

1. INTRODUCTION

1.1. Background

Squamous Cell Carcinoma (SCC) is an abnormal tissue growth that develops from squamous epithelial cells. These cells are flat, thin structures that cover the surface of the skin, mouth, throat, and several internal organs. When these cells begin to grow in an uncontrolled manner, they form irregular structures that disturb normal tissue patterns. Grading this abnormal growth is important because it helps in understanding the severity of the condition and supports doctors in deciding the most suitable treatment approach.

1.2. Importance of Histopathology

Histopathology, which is the microscopic study of tissue samples, is the most trusted and widely used method for identifying and grading Squamous Cell Carcinoma. A histopathology image shows the arrangement, shape, and texture of the cells. Based on how similar or different the cells appear compared to normal tissue, the sample is graded as well-differentiated, moderately differentiated, or poorly differentiated. However, manual grading through microscopic observation is time-consuming and depends greatly on the experience and judgment of the pathologist. This subjectivity can sometimes lead to variations in diagnosis.

1.3. Challenges in Manual Grading

Manual grading presents several difficulties in real-world conditions. It requires high expertise, consistent concentration, and considerable time to examine each tissue slide carefully. Differences in staining, lighting, or image clarity can make grading even more difficult. In hospitals and diagnostic centers that handle a large number of samples daily, these challenges create delays and increase the workload for medical professionals. Therefore, there is a strong need for automated systems that can perform grading quickly, consistently, and objectively.

1.4. Advancements in Digital Pathology

The introduction of digital pathology has brought major changes to medical image analysis. Modern scanners can convert traditional glass slides into high-resolution digital images. These digital histopathology images can be easily stored, shared, and processed by computer systems. This development has opened new opportunities for automated image analysis using computer-based techniques, allowing for faster and more reliable examination of tissue structures.

1.5. Emergence of Deep Learning

Deep learning, a specialized area within artificial intelligence, has shown remarkable success in analyzing and interpreting complex image data. It uses neural networks with multiple layers that automatically learn useful patterns from raw images. Convolutional Neural Networks (CNNs), in particular, are highly effective for image-related tasks. They can recognize detailed visual patterns such as edges, shapes, and textures, which are essential for understanding histopathology images. Unlike traditional image processing techniques that rely on manually designed features, deep learning models can learn features directly from the data itself.

1.6. Deep Learning for Grading Squamous Cell Carcinoma

Using deep learning methods for grading Squamous Cell Carcinoma can significantly improve the reliability and efficiency of the diagnostic process. A CNN model can be trained on labeled histopathology images to identify patterns that correspond to different grades of SCC. Once trained, the model can classify new images accurately without human intervention. This approach reduces errors, provides consistency, and saves valuable time for pathologists. It also supports better decision-making by combining computational accuracy with medical expertise. The development of a deep learning approach for grading Squamous Cell Carcinoma from histopathology images represents an important step toward intelligent and automated medical diagnosis. It strengthens the role of technology in assisting experts, enhances the precision of image-based evaluation, and contributes to making medical analysis faster, more objective, and more dependable.

2. OBJECTIVES

The main objective of this project is to develop, train, and evaluate a set of deep learning models for grading Squamous Cell Carcinoma (SCC) from histopathology images. The work focuses on exploring different model architectures and fine-tuning their parameters to achieve better accuracy, reliability, and consistency. By testing various configurations and hyperparameter settings, the study aims to find which model setup performs best for this task.

The specific objectives of this project are as follows:

1. To collect and preprocess histopathology images of Squamous Cell Carcinoma for training and testing the models. This includes resizing images, normalizing pixel values, and applying augmentation techniques to improve image variety and prevent model overfitting.
2. To design and implement multiple deep learning architectures, including basic Convolutional Neural Networks (CNNs) and transfer learning models such as VGG, ResNet, and Inception. Each model will be trained and tested to study how its structure affects performance.
3. To experiment with different parameters and hyperparameter tuning techniques, such as adjusting the learning rate, batch size, number of epochs, optimizers, and activation functions, to observe how these changes influence model learning and accuracy.
4. To train, validate, and test each model on the prepared dataset and analyze how different hyperparameter settings impact performance, convergence, and stability.
5. To compare the performance of all trained models using evaluation metrics like accuracy, precision, recall, and F1-score, and identify the model configuration that gives the best and most consistent grading results.

The overall goal is to understand how different deep learning models and tuning strategies behave on histopathology images and to determine which combination provides the most effective results for grading Squamous Cell Carcinoma.

3. LITERATURE SURVEY

Paper Title	Authors	Dataset	Methods / Models Used	Key Findings / Results
Automated Grading of Lung Carcinoma Using Hybrid Deep Learning Approach	Jelena Musulin, Sandi Baressi Šegota	Lung carcinoma histopathology images	CNN-based deep learning	Multi-class grading accuracy: 87.2%, F1 score: 86.5%, Sensitivity: 86.8%, Specificity: 94.2%; robust grading of histopathology images.
LungCarcinoGrade-EffNetSVM: A Novel Approach to Lung Carcinoma Grading using EfficientNetB0 and SVM	Pragati Patharia, Prabira Kumar Sethy	Kaggle CT scan dataset: 1,000 images (338 Adenocarcinoma, 200 Large Cell Carcinoma, 260 SCC, 202 Normal)	EfficientNetB0 for feature extraction + SVM (one-vs-all ECOC strategy)	Accuracy: 86.88%, Sensitivity: 86.88%, Specificity: 95.63%, Precision: 87.06%, F1 score: 86.64%, MCC: 0.8257, Kappa: 0.65; fast computation (~9.3s/image)
Histopathology Image Grading of Lung Carcinoma using Deep Learning	Sethy P. K., Geetha Devi A., Padhan B., Behera S. K., Sreedhar S., Das K.	Histopathology images of lung carcinoma	Wavelet features + AlexNet	Accuracy: 88.5%, Sensitivity: 87%, Specificity: 93%; robust grading across multiple carcinoma types
Enhancing Oral Squamous Cell Carcinoma Detection using EfficientNetB3 from Histopathologic Images	Aditya Kumar, Leema Nelson	Oral SCC histopathology images	EfficientNetB3 CNN	High accuracy in oral SCC detection; EfficientNetB3 performed well
Histopathological Image based Oral Squamous Cell Carcinoma Classification Using Deep Network Fusion	Kumar Ankit, Vidit Kumar	Oral SCC histopathology images	Deep network fusion (multiple CNNs)	Improved classification accuracy by fusing multiple networks; effective for oral SCC grading
EsccNet: A Hybrid CNN and Transformers	Zhaoxin Kang, Mingqiu Chen,	Esophageal SCC whole slide images	Hybrid CNN + Transformer	Outperformed standard CNNs; effective for

Model for Classification of Whole Slide Images of Esophageal SCC	Hejun Zhang, Xiangwen Liao			whole slide image classification
Squamous Cell Carcinoma Margin Classification Using Vision Transformers from Digital Histopathology Images	So-Yun Park, Gelan Ayana, Se-woon Choe	828 histopathological images (345 margin-negative, 483 margin-positive) from Jimma University Medical Center	Vision Transformer models (ViT-B16, ViT-B32, ViT-L32) + Dense & normalization layers; compared with ResNet50	ViT-B16 achieved 0.906 accuracy & 0.905 AUC, outperforming CNN; highlights ViTs' superiority
Classification of Non-Small Cell Lung Cancer Using Deep Learning	Lathakumari K. R., Ramachandra A. C., Avanthi U. C., Basil Ronald C., Bhavatharani T.	Kaggle CT scan dataset: 1,000 images (Adenocarcinoma, SCC, Large Cell Carcinoma, Normal)	EfficientNetB2 (342-layer CNN) + preprocessing + Softmax	Training accuracy: 95%, Testing accuracy: 83%; strong performance but dataset limitation noted
Multi-Teacher Knowledge Distillation with Reinforcement Learning for Visual Recognition	Chuanguang Yang, Xinqiang Yu, Han Yang, Zhulin An, Chengqing Yu, Libo Huang, Yongjun Xu	Standard visual recognition datasets	Multi-teacher KD with reinforcement learning to weight teachers	Outperforms traditional KD by dynamically adjusting teacher weights; improves recognition accuracy
Advancing Trans-Domain Classification With Knowledge Distillation: Bridging LIDAR and Image Data	J. Eduardo Ortíz, W. Creixell	nuScenes dataset (LIDAR + camera)	Transformer-based KD transferring knowledge from image → LIDAR	Improved accuracy & inference speed vs PointNet++; efficient alternative to sensor fusion
Multiple Teachers Are Beneficial: A Lightweight and Noise-Resistant Student Model for Point-of-Care	Yucheng Song, Anqi Song, Jincan Wang, Yifan Ge, Lifeng Li	ISIC, BUSI, Dermnet datasets	Lightweight Shift MLP student + multi-teacher KD; supervised + distillation +	Strong performance while reducing parameters 38× and FLOPs 11×; robust for noisy, resource-constrained imaging

Imaging Classification			consistency losses	
Deep Learning for Clinical Image Analyses in Oral Squamous Cell Carcinoma	Chui Shan Chu, BSc, MMedSc1; Nikki P. Lee, PhD2; Joshua W. K. Ho, PhD3,4	Histopathology images used form SUM hospitals,clinical oral images	VGGnet, ResNet, Dense net Efficient net	Pathological images:77.9%-97.5% Accuracy Radiographic images:76%
Histopathology-based diagnosis of oral squamous cell carcinoma using deep learning	S.Y. Yang, S.H., and W. Liao	Total images of 2,025 images were collected from the pathologiest	CNN based model was used in this paper frame work used here is binary classification	sensitivity was 0.98,f1 score is said as 0.951 Post predictive value:0.924
Colorectal cancer detection with enhanced precision using a hybrid supervised and unsupervised learning approach	Akella S. Narasimha Raju, K. Venkatesh, Ranjith Kumar Gatla, Eswara Prasad Konakalla, Marwa M. Eid, Nataliia Titova, Sherif S. M. Ghoneim, Ramy N. R. Ghaly	CVC-ClinicDB (1,650 colonoscopy images of polyps & non-polyps; preprocessing and augmentation applied)	CNN ensembles (ADa-22 and AD-22) for feature extraction, Transformer for context, SVM for classification, and K-means with bounding boxes for segmentation and interpretability.	The best model (AD-22 + Transformer + SVM) reached 99% test accuracy, AUC 0.99, with strong recall for both polyps and non-polyps. K-means segmentation (silhouette up to **0.73**) added interpretability alongside high accuracy.
A coded knowledge distillation framework for image classification based on adaptive JPEG encoding	Ahmed H. Salamah, Shayan Mohajer Hamidi, En-Hui Yang	CIFAR-10, CIFAR-100, ImageNet	Coded Knowledge Distillation (CKD) with adaptive JPEG compression before teacher model; compared with standard KD and other variants (FitNet, AT, RKD, SP, CC)	Adaptive compression produces softer teacher outputs, reduces over-confidence, improves student accuracy consistently across benchmarks with low computational cost

4. SPECIFICATIONS

4.1 Dataset

- **Name:** LC25000 (Lung and Colon Cancer Histopathological Images)
- **Source:** Kaggle (link)
- **Classes Used:** Adenocarcinoma, Squamous Cell Carcinoma (binary classification)
- **Preprocessing:**
    - Resized all images to 224×224 pixels
    - Normalized pixel values to [0,1]
    - Applied data augmentation (rotation, flipping, zoom, brightness adjustment)

4.2 Hardware and Software

- **Hardware:** Google Colab (GPU-enabled)
- **Software:**
    - Python 3.x
    - TensorFlow / Keras or PyTorch
    - OpenCV / PIL for image processing
    - NumPy, Pandas, Matplotlib, Seaborn for data handling and visualization

4.3 Models / Architectures

- **DenseNet:** CNN with dense layer connections for efficient feature learning
- **ConvNeXt:** Modernized CNN inspired by transformer architectures
- **MobileNetV2:** lightweight and efficient CNN architecture.
- **InceptionV3:** CNN using inception modules to capture multi-scale features

Hyperparameter Tuning:

- Learning rates: 0.001, 0.0001
- Batch sizes: 16, 32, 64
- Optimizers: Adam, SGD
- Activation functions: ReLU (hidden layers), Softmax (output)
- Epochs: 50–100 depending on model convergence

4.4 Evaluation Metrics

- Accuracy
- Precision
- Recall (Sensitivity)
- F1-score
- Confusion matrix for class-wise performance

4.5 Functional Requirements

- Load and preprocess LC25000 histopathology images
- Train multiple deep learning models on binary classification
- Tune hyperparameters for optimal performance
- Evaluate models using standard metrics
- Visualize results with plots and confusion matrices

4.6 Non-Functional Requirements

- Efficient GPU-based training using Colab
- Modular and reusable code for testing multiple models
- Scalable approach to add more models or classes in the future

5. ARCHITECTURE BLOCK /FLOWCHART PROPOSED MODULES

5.1 Flowchart

The architecture of this project is designed for binary classification of histopathology images into Adenocarcinoma and Squamous Cell Carcinoma using multiple deep learning models. The workflow is divided into several modules, each addressing a specific task from dataset preparation to model evaluation.

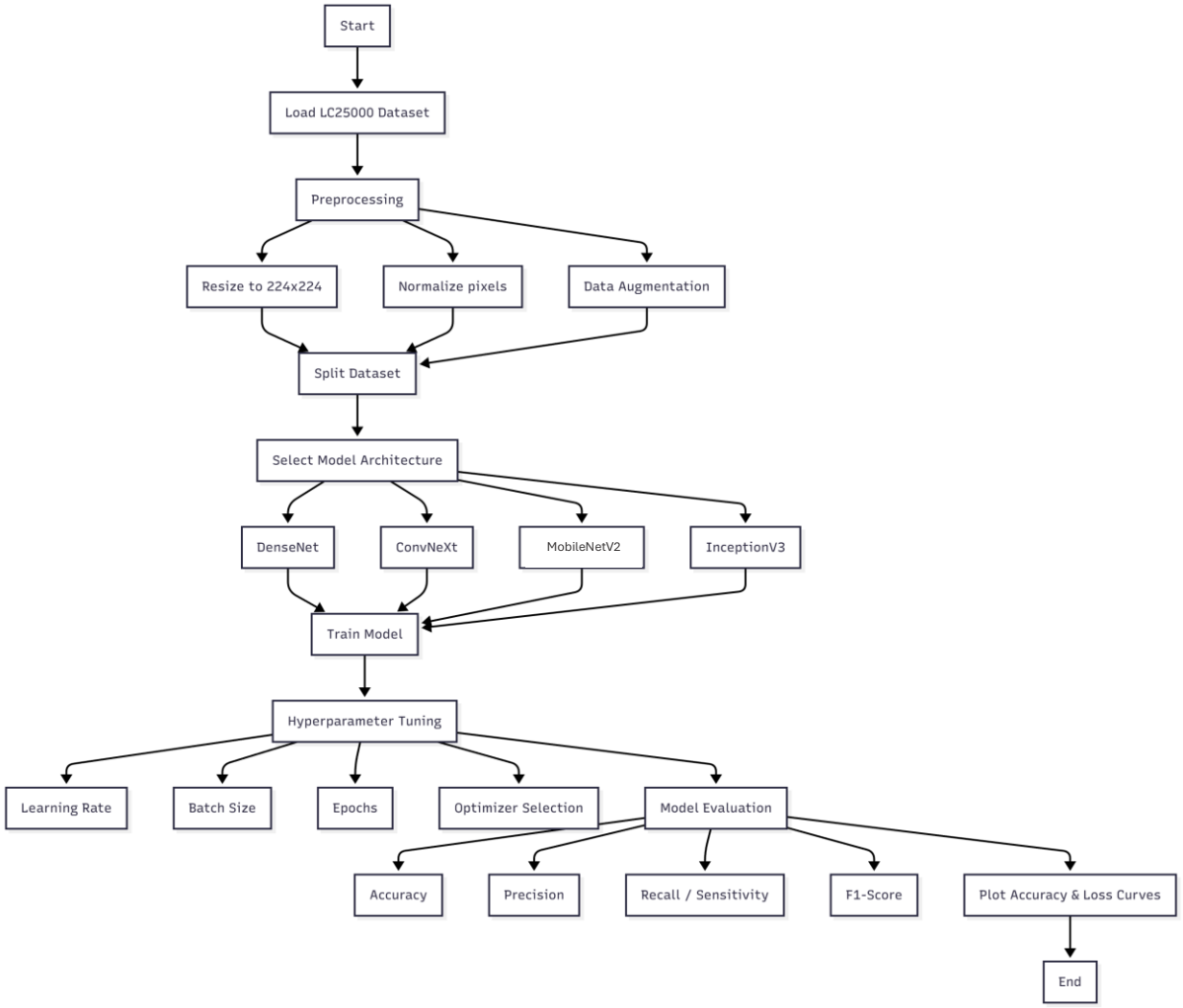


Fig 5.1 Flowchart of the proposed deep learning workflow for binary classification of histopathology images

5.2 Module Descriptions:

1. **Dataset Loading:**

The LC25000 dataset is loaded, specifically selecting images for Adenocarcinoma and Squamous Cell Carcinoma. This module ensures that the dataset is correctly structured for subsequent preprocessing and model training.

2. **Preprocessing:**

Images are resized to 224×224 pixels to match the input requirements of the models. Normalization scales pixel values to a standard range, improving training stability. Data augmentation techniques such as rotation, flipping, zooming, and brightness adjustment are applied to enhance dataset diversity and reduce overfitting.

3. **Dataset Splitting:**

The dataset is divided into training, validation, and test sets. This division allows the models to learn from a subset of data, validate during training, and finally test performance on unseen images to assess generalization.

4. **Model Architecture Selection:**

Four different deep learning architectures are used: DenseNet, ConvNeXt, MobileNetV2, and InceptionV3. Each model is chosen for its ability to extract meaningful features from histopathology images, with varying depths, connections, and module designs.

5. **Model Training:**

Each model is trained independently using the prepared dataset. The training process involves forward propagation, loss computation, and backpropagation to update weights. This module is crucial for learning patterns that distinguish the two classes.

6. Hyperparameter Tuning:

Key hyperparameters such as learning rate, batch size, number of epochs, and optimizer type are adjusted to improve model performance. Tuning these parameters ensures that each model converges efficiently without overfitting or underfitting.

7. Model Evaluation:

Models are evaluated using accuracy, precision, recall, and F1-score to quantify performance. This module also includes creating confusion matrices to analyze class-wise prediction behaviour.

8. Accuracy & Loss Curves:

Training and validation accuracy and loss curves are plotted for each model. These curves provide visual insight into the learning process, helping to detect convergence issues, overfitting, or underfitting.

6. WORK DONE

6.1 Dataset Preparation and Cleaning

The first step in the project was to download the LC25000 histopathology dataset directly in Google Colab. From the complete dataset, images corresponding to the two selected classes—Adenocarcinoma and Squamous Cell Carcinoma—were extracted for further processing.

The dataset was carefully cleaned manually. Images that contained excessive white space, too many gland-like structures, or poor-quality regions were removed. This ensured that the models would learn from relevant and clear images, improving the reliability of training.

6.2 Preprocessing

After cleaning, the images underwent several preprocessing steps:

- **Resizing:** All images were resized to 224×224 pixels to match the input requirements of the models.
- **Normalization:** Pixel values were scaled between 0 and 1 to stabilize training and improve convergence.
- **Data Augmentation:** Techniques such as rotation, flipping, zoom, and brightness adjustment were applied to increase the diversity of the dataset and reduce overfitting.

These preprocessing steps ensured the models received standardized and diverse inputs for training.

6.3 Dataset Splitting

The cleaned and preprocessed dataset was divided into training, validation, and test sets. This split allowed the models to learn patterns from the training set, validate performance during training to monitor overfitting, and finally evaluate generalization on unseen data.

6.4 Model Implementation and Training

Once the dataset was prepared, four different deep learning models were implemented for binary classification. These models were DenseNet, ConvNeXt, MobileNetV2, and InceptionV3. Training was conducted in Google Colab using GPU acceleration.

For all models, hyperparameter tuning was performed to explore the effects of:

- Learning rate: 0.001, 0.0001, 0.0005 depending on the model
- Batch size: 16, 32

- Epochs: 50, 75, 100
- Optimizer type: Adam, SGD, RMSProp
- Dropout rates: 0.2 - 0.5 where applicable
- Data augmentation intensity

Training and validation accuracy and loss curves were plotted for each hyperparameter combination to track progress.

6.4.1 InceptionV3

For the InceptionV3 model, transfer learning was employed using the base model pretrained on ImageNet. The base layers were frozen to retain learned features, and custom top layers were added for binary classification of Adenocarcinoma and Squamous Cell Carcinoma.

Dataset Preparation:

- The dataset was downloaded from LC25000 and copied to the Colab local environment for faster processing.
- Only two classes were used: lung_aca (Adenocarcinoma) and lung_scc (Squamous Cell Carcinoma).
- Manual filtering was applied to remove poor-quality images, such as those with excessive white regions or unclear structures.
- All images were resized to 299×299 pixels and pixel values were normalized to improve model training stability.
- Sample images from both classes were plotted to confirm correct labeling and data variety.

Model Architecture:

- A Global Average Pooling layer was added to the base model output.
- Batch Normalization was applied for stable training.
- A Dense layer with 512 units and ReLU activation followed by Dropout (0.5) was added to reduce overfitting.
- The final Dense layer with sigmoid activation produced a binary output.

HyperparameterVariants:

Two experiments were conducted with different optimizers:

1. **Adam optimizer variant**: Learning rate = 1e-4.
2. **RMSprop optimizer variant**: Learning rate = 1e-4, rho = 0.9.

Both variants used **binary cross-entropy** as the loss function and accuracy as the evaluation metric.

Training and Evaluation:

- 5-fold cross-validation was performed to ensure robust evaluation.
- For each fold, training and validation sets were generated, and images were fed in batches using ImageDataGenerator.
- Callbacks: ReduceLROnPlateau and EarlyStopping were used to control learning rate and prevent overfitting.
- During training, accuracy and loss curves were recorded for both training and validation sets.

Post-Training Analysis:

- After each fold, confusion matrices were plotted to analyze class-wise performance.
- ROC curves and AUC were calculated for each fold to evaluate classification performance.
- Combined plots for validation accuracy and loss across all folds were generated to observe overall trends and model stability.

This dual-experiment setup allowed systematic testing of InceptionV3 with different optimizer configurations, providing insights into the effect of hyperparameter tuning on training behavior and classification performance.

ReferenceforImplementation:

The InceptionV3 model and preprocessing steps were implemented in Google Colab. The notebook can be accessed here: https://colab.research.google.com/drive/1Q3l-KIF-eGKjwrcKVXvtbxOGyFUKVivh?usp=sharing

6.4.2. ConvNeXtTiny

For the ConvNeXtTiny model, transfer learning was employed using the base model pretrained on ImageNet. The base layers were frozen to retain the learned hierarchical features, and custom classification layers were added for binary classification of Adenocarcinoma and Squamous Cell Carcinoma.

Dataset Preparation:

The dataset was obtained from LC25000 and transferred to the Colab local environment for efficient processing.Only two classes were used: lung_aca (Adenocarcinoma) and lung_scc (Squamous Cell Carcinoma).Manual filtering was performed to remove unwanted or poor-quality images, particularly those containing large white spaces or unclear tissue structures, ensuring better data quality and consistency.All images were resized to 224×224 pixels, and pixel values were normalized to enhance model convergence and stability.Sample images from both classes were visualized to verify correct labeling and ensure diversity within the dataset.

Model Building

The ConvNeXtTiny architecture was adopted as the base model, pre-trained on the ImageNet dataset. This model was chosen for its strong representational capability in image classification tasks.

A custom classification head was added on top of the pre-trained feature extractor. The added layers included:

- Global Average Pooling Layer – to reduce the feature map to a single vector representation.
- Batch Normalization Layer – to stabilize and accelerate convergence during training.
- Dense Layer with ReLU activation – to learn higher-level abstractions from extracted features.
- Dropout Layer – to reduce overfitting by randomly disabling neurons during training.
- Final Dense Layer with Sigmoid activation – to perform binary classification between the two cancer types.

Initially, all base model layers were frozen to train only the custom classification head. In subsequent experiments, the last 20 layers were unfrozen to allow fine-tuning, enabling the network to learn domain-specific features more effectively from histopathology images.

K-Fold Cross-Validation

To ensure robust model evaluation and prevent overfitting to a particular data subset, a 5-Fold Cross-Validation approach was implemented. The dataset was divided into five equal parts (folds). In each iteration, four folds were used for training and one for validation, ensuring that each sample contributed to both training and validation at some point.

The KFold function from the *scikit-learn* library was utilized to generate the data splits. Within each fold, the ImageDataGenerator handled real-time data augmentation and preprocessing. Before training, sample images from both training and validation sets were visualized to confirm the correctness of data distribution and augmentation.

Training and Hyperparameter Tuning

The model was trained using the Adam optimizer, binary cross-entropy loss, and accuracy as the performance metric. Training was conducted across multiple configurations to explore different learning dynamics and improve the model's performance.

Phase 1: Initial Training

The model was trained for 25 epochs per fold with all base layers frozen. Learning rate reduction and early stopping callbacks were employed to automatically adjust training parameters and prevent overfitting.

Phase 2: Extended Training (500 Epochs)

Following project discussions, the training was extended to 500 epochs to enable deeper learning of image features. Various hyperparameters were tuned, including:

- Learning rate: Tested with values of 3e-5 and 1e-4.
- Unfrozen layers: Gradually increased fine-tuning depth (last 20 layers unfrozen).
- Callbacks: LearningRateReducer (factor = 0.5, patience = 5, min_lr = 1e-6) and EarlyStopping (patience = 10).

Multiple configurations were tested to determine the optimal training parameters that yield stable and high-performing models.

Phase 3: Training with Cleaned Dataset

A further experiment was conducted using the cleaned version of the dataset, which excluded noisy and low-quality images. The training setup was similar to the previous configurations, with the goal of evaluating the impact of dataset quality on overall model learning and convergence.

For each experiment, the training and validation accuracy/loss were tracked per epoch, and accuracy/loss plots were generated for performance comparison.

Implementation Environment

All experiments were executed on Google Colab, utilizing GPU acceleration for efficient model training. The implementation was performed using:

- TensorFlow and Keras for deep learning.

- scikit-learn for K-Fold cross-validation.
- Matplotlib for visualization of accuracy and loss curves.

Each stage of the experimentation — dataset preparation, model definition, training, and validation — was organized into Jupyter notebook files for clarity and reproducibility.

Links to the main experiment notebooks

- Initial Model (25 Epochs)
- Fine-tuned Model (500 Epochs)
- Cleaned Dataset Model (500 Epochs)

6.4.3 DenseNet121

For the DenseNet121 model, transfer learning was employed using the base model pretrained on ImageNet. The base layers were frozen to retain the learned hierarchical features, and custom classification layers were added for binary classification of Adenocarcinoma and Squamous Cell Carcinoma.

Dataset Preparation:

The dataset was obtained from LC25000 and transferred to the Colab local environment for efficient processing.Only two classes were used: lung_aca (Adenocarcinoma) and lung_scc (Squamous Cell Carcinoma).Manual filtering was performed to remove unwanted or poor-quality images, particularly those containing large white spaces or unclear tissue structures, ensuring better data quality and consistency.All images were resized to 224×224 pixels (the standard input size for DenseNet121), and pixel values were normalized to enhance model convergence and stability.Sample images from both classes were visualized to verify correct labeling and ensure diversity within the dataset

Model Architecture:

A Global Average Pooling (GAP) layer was added to the base model output.Batch Normalization was applied to stabilize and accelerate training.A Dense layer with 512 units and ReLU activation, followed by a Dropout layer (rate = 0.5), was included to reduce overfitting.The final Dense layer with sigmoid activation produced the binary output for

classification.DenseNet121 is a densely connected convolutional neural network.Instead of stacking layers linearly, DenseNet connects each layer to every other layer in a feed-forward fashion.It promotes feature reuse, reduces the number of parameters, and strengthens gradient flow (helping deeper models train effectively).The model is pretrained on ImageNet, so it already understands general image features like edges, colors, and textures — which are reused for your lung image classification task include_top=False removes the original ImageNet classification head, allowing you to add your own

Hyperparameter Settings:

- **Optimizer:** Adam
- **Learning rate:** 1e-4
- **Loss function:** Binary Cross-Entropy
- **Evaluation metric:** Accuracy

Callbacks:

- **ReduceLROnPlateau**: Reduces LR by half if validation accuracy plateaus.
- **EarlyStopping**: Stops training early if no improvement after patience epochs.
- **ModelCheckpoint**: Saves the best model weights per fold.

Training and Evaluation:

A 5-fold cross-validation approach was used to ensure robust and unbiased evaluation.

For each fold, distinct training and validation sets were generated, and images were processed in batches using ImageDataGenerator with real-time data augmentation. Throughout training, accuracy and loss curves were plotted for both training and validation sets.

Post-Training Analysis:

After training each fold, confusion matrices were generated to assess class-wise performance.Combined plots of validation accuracy and loss across all folds were analyzed to study overall performance consistency and stability.

6.4.4 MobileNetV2

For the MobileNetV2 model, transfer learning was employed using the base model pretrained on ImageNet. The base layers were frozen to retain the learned hierarchical features, and custom classification layers were added for binary classification of Adenocarcinoma and Squamous Cell Carcinoma.

Dataset Preparation:

The dataset was obtained from LC25000 and transferred to the Colab local environment for efficient processing.Only two classes were used: lung_aca (Adenocarcinoma) and lung_scc (Squamous Cell Carcinoma).Manual filtering was performed to remove unwanted or poor-quality images, particularly those containing large white spaces or unclear tissue structures, ensuring better data quality and consistency.All images were resized to 224×224 pixels (the standard input size for MobileNetV2), and pixel values were normalized to enhance model convergence and stability.Sample images from both classes were visualized to verify correct labeling and ensure diversity within the dataset.

Model Architecture:

A Global Average Pooling (GAP) layer was added to the base model output.Batch Normalization was applied to stabilize and accelerate training.A Dense layer with 126 units and ReLU activation with L2 regularization(0.01), followed by a Dropout layer (rate = 0.4), was included to reduce overfitting.The final layer with sigmoid activation produced the binary output for classification.It promotes feature reuse, reduces the number of parameters, and strengthens gradient flow (helping deeper models train effectively).The model is pretrained on ImageNet, so it already understands general image features like edges, colors, and textures — which are reused for your lung image classification task.include_top=False removes the original ImageNet classification head, allowing you to add your own.

Parameter Settings:

- **limit_per_class**: 1000 (Maximum number of images loaded per class)
- **EPOCHS**: 500 (Maximum number of training epochs)
- **BATCH_SIZE**: 32 (Batch size for training and validation)
- **split_ratio**: 0.8 (Ratio for splitting data into training and validation sets)
- **Optimizer**: Adam
- **Learning Rate**: 1e-4
- **Loss Function**: Binary Crossentropy
- **Metrics**: Accuracy
- **L2 Regularization**: 0.01 (Applied to the dense layer)
- **Dropout Rate**: 0.4 (Applied after Global Average Pooling and the dense layer)
- **Early Stopping Patience**: 5 (Number of epochs with no improvement after which training will be stopped)
- **ReduceLROnPlateau Patience**: 10 (Number of epochs with no improvement after which the learning rate will be reduced)
- **ReduceLROnPlateau Factor**: 0.5 (Factor by which the learning rate will be reduced)

Post-Training Analysis:

After training each fold, confusion matrices were generated to assess class-wise performance. Combined plots of validation accuracy and loss across all folds were analyzed to study overall performance consistency and stability.

7. RESULTS

7.1 InceptionV3

7.1.1. InceptionV3 with Adam Optimizer

Overall Performance

The InceptionV3 model with Adam optimizer achieved excellent classification performance using 5-fold cross-validation. The key results are summarized in Table 6.1.

Table 7.1: Performance Summary (Adam Optimizer)

Metric	Value
Mean Validation Accuracy	99.72%
Standard Deviation	±0.17%
Best Accuracy	99.85% (Fold 5)
Lowest Accuracy	99.39% (Fold 2)
Total Images Used	9,810

The low standard deviation indicates that the model performed consistently across all folds, showing good stability and reliability.

Fold-wise Results

Table 6.2 shows the detailed performance for each fold, including the best epoch and final learning rate achieved through automatic reduction.

Table 7.2: Detailed Results per Fold (Adam Optimizer)

Fold	Validation Accuracy	Best Epoch	Total Epochs Trained	Final Learning Rate
1	99.80%	40	50	2.50×10^{-5}
2	99.39%	19	29	2.50×10^{-5}
3	99.80%	39	49	5.00×10^{-5}
4	99.80%	39	49	5.00×10^{-5}
5	99.85%	35	45	2.50×10^{-5}

All folds achieved validation accuracy above 99%, with four out of five folds reaching 99.80% or higher.

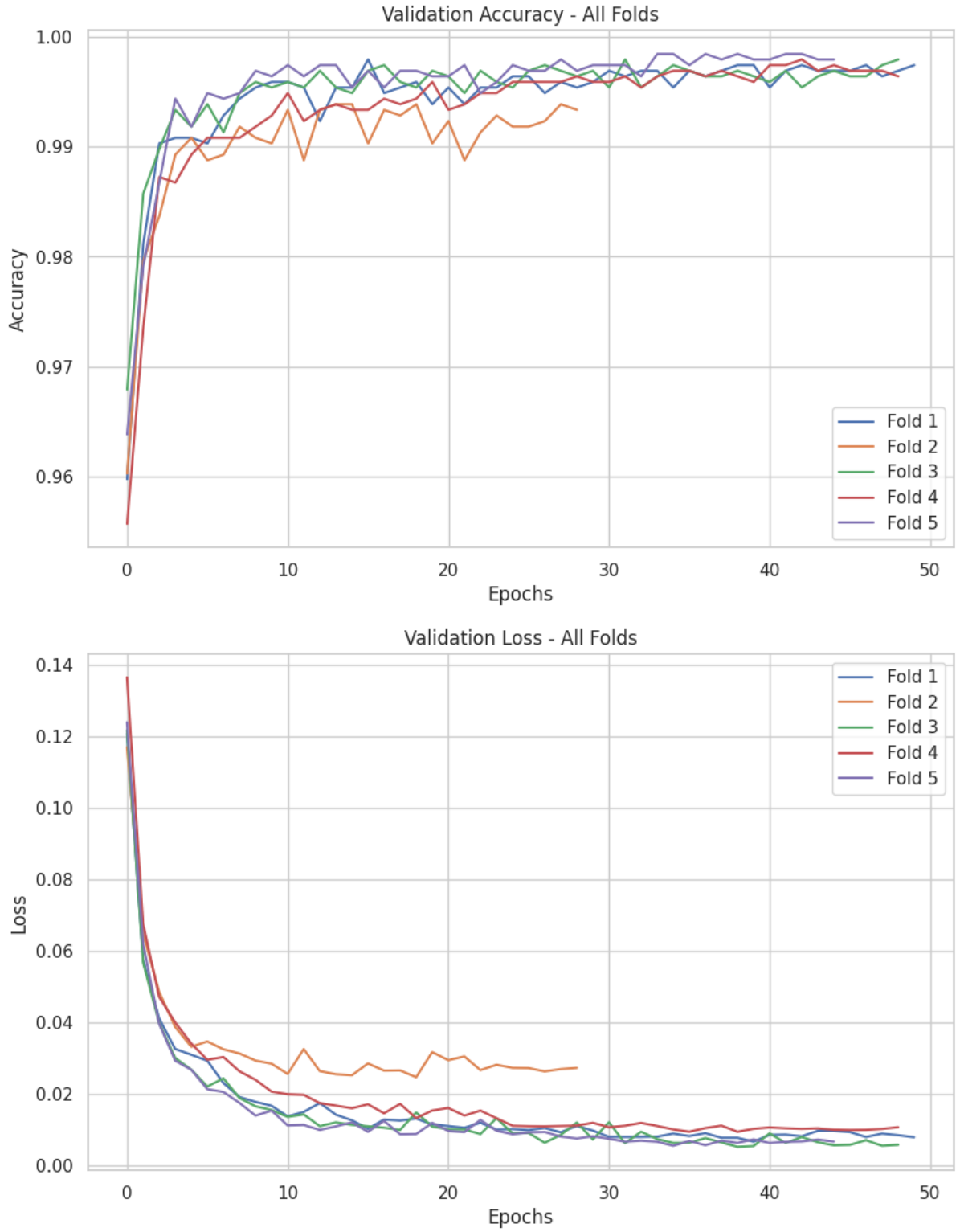


Fig 7.1. Accuracy and Loss Curves of all Folds for InceptionV3 with Adam Optimizer

Training Behavior

The model showed rapid learning in the initial epochs:

- First Epoch: Training accuracy reached ~87%, validation accuracy reached ~96%

- Epochs 2-20: Steady improvement with validation accuracy crossing 99%
- Epochs 20-40: Fine-tuning phase with learning rate reductions
- Convergence: Early stopping activated between epochs 29-50

The training remained stable throughout, with validation loss decreasing consistently without signs of overfitting.

Learning Rate Adaptation

The ReduceLROnPlateau callback automatically reduced the learning rate when validation loss stopped improving:

- Initial rate: 1.0×10^{-4}
- First reduction: 5.0×10^{-5} (typically around epoch 20-25)
- Second reduction: 2.5×10^{-5} (around epoch 35-40)
- Final reduction: 1.25×10^{-5} (if needed)

This adaptive approach helped the model fine-tune its weights without overshooting optimal values.

Classification Performance

Confusion matrices and ROC curves were generated for each fold:

- AUC Scores: All folds achieved AUC > 0.999, indicating near-perfect separation between classes
- Misclassifications: Very few errors (1-4 images per fold out of 1,962 validation images)
- Balanced Performance: Both Adenocarcinoma and Squamous Cell Carcinoma were classified with similar high accuracy

Training Efficiency

- Average training time per fold: ~35-40 minutes on GPU
- Average time per epoch: ~56-60 seconds
- Prediction speed: Fast inference suitable for practical applications

Key Observations

1. Consistency: The model achieved similar performance across all folds (std dev only 0.17%)
2. Fast Convergence: High accuracy was reached within 20-40 epochs
3. No Overfitting: Validation accuracy remained close to or higher than training accuracy
4. Reliable: The 99.72% mean accuracy demonstrates strong classification capability

The Adam optimizer proved highly effective for this transfer learning task, providing stable training and excellent final performance.

Model Configuration Used:

- Optimizer: Adam
- Initial Learning Rate: 1×10^{-4}
- Batch Size: 32
- Dropout: 0.5
- Image Size: 299×299 pixels

7.1.2 InceptionV3 with RMSprop Optimizer

Overall Performance

The InceptionV3 model with RMSprop optimizer achieved excellent classification performance using 5-fold cross-validation. The key results are summarized in Table 7.3.

Table 7.3: Performance Summary (RMSprop Optimizer)

Metric	Value
Mean Validation Accuracy	99.66%
Standard Deviation	±0.11%
Best Accuracy	99.80% (Fold 2)
Lowest Accuracy	99.49% (Fold 3)
Total Images Used	9,810

The extremely low standard deviation indicates that the model performed very consistently across all folds, showing excellent stability and reliability.

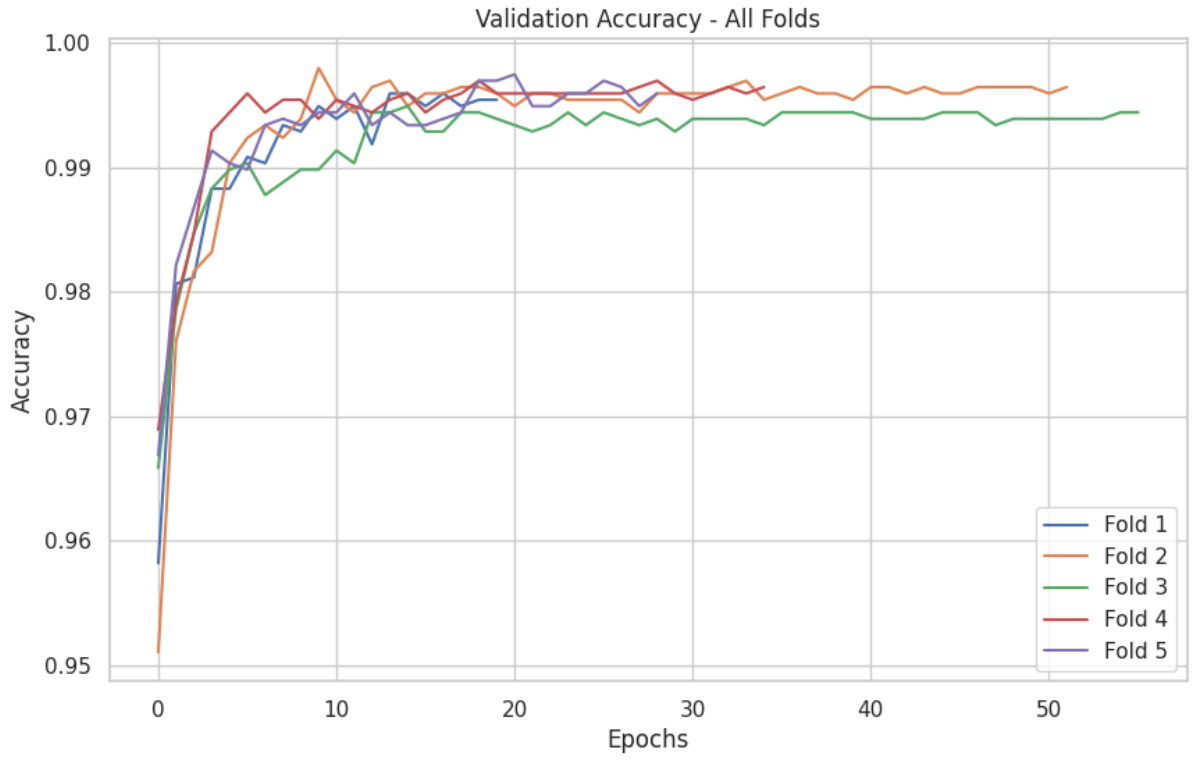
Fold-wise Results

Table 7.4 shows the detailed performance for each fold, including the best epoch and final learning rate achieved through automatic reduction.

Table 7.4: Detailed Results per Fold (RMSprop Optimizer)

Fold	Validation Accuracy	Best Epoch	Total Epochs Trained	Final Learning Rate
1	99.59%	10	20	2.50×10^{-5}
2	99.80%	42	52	3.13×10^{-6}
3	99.49%	46	56	3.13×10^{-6}
4	99.69%	25	35	2.50×10^{-5}
5	99.75%	19	29	2.50×10^{-5}

All folds achieved validation accuracy above 99%, with three out of five folds reaching 99.69% or higher.



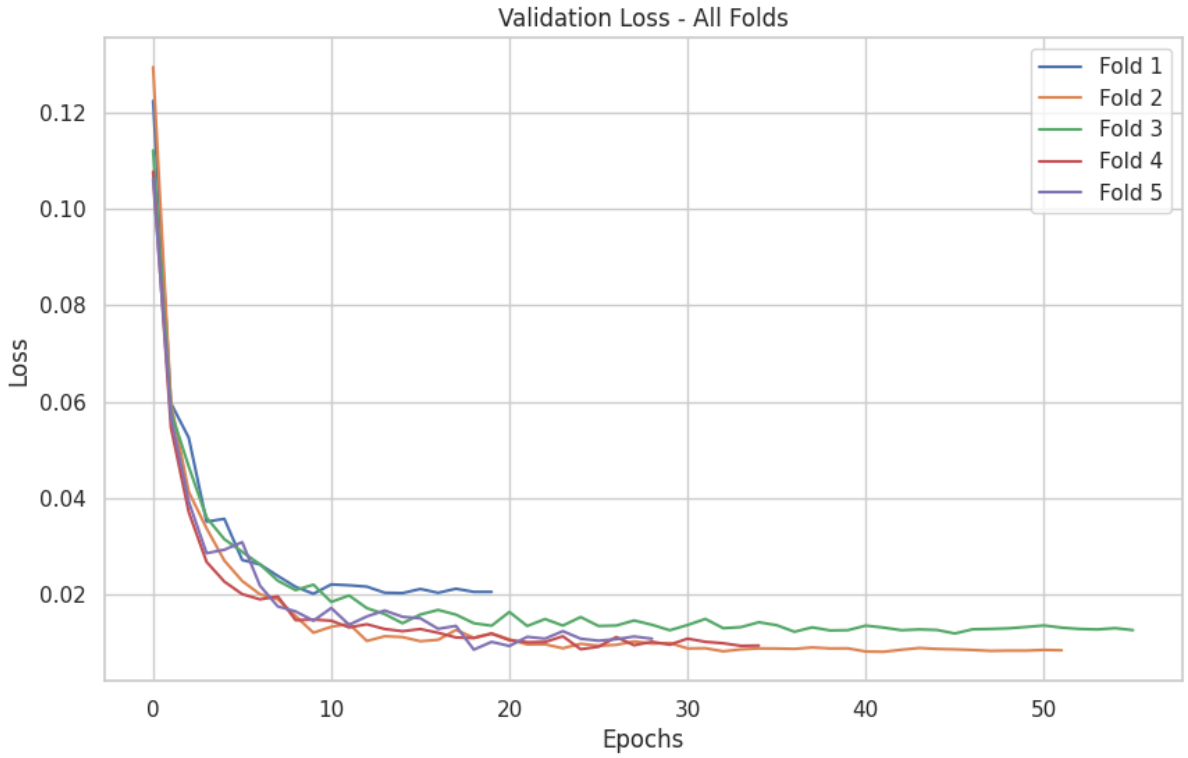


Fig 7.2. Accuracy and Loss Curves of all Folds for InceptionV3 with RMSprop Optimizer

Training Behaviour

The model showed rapid learning in the initial epochs:

- First Epoch: Training accuracy reached ~87-88%, validation accuracy reached ~95-97%
- Epochs 2-20: Steady improvement with validation accuracy crossing 99%
- Epochs 20-50: Fine-tuning phase with learning rate reductions
- Convergence: Early stopping activated between epochs 20-56

The training remained stable throughout, with validation loss decreasing consistently without signs of overfitting.

Learning Rate Adaptation

The ReduceLROnPlateau callback automatically reduced the learning rate when validation loss stopped improving:

- Initial rate: 1.0×10^{-4}

- First reduction: 5.0×10^{-5} (typically around epoch 15-25)
- Second reduction: 2.5×10^{-5} (around epoch 30-40)
- Third reduction: 1.25×10^{-5} (if needed)
- Fourth reduction: 6.25×10^{-6} (for extended training)
- Fifth reduction: 3.13×10^{-6} (final fine-tuning in Folds 2-3)

This adaptive approach with RMSprop's momentum (rho=0.9) helped the model fine-tune its weights smoothly without overshooting optimal values.

Classification Performance

Confusion matrices and ROC curves were generated for each fold:

- AUC Scores: All folds achieved AUC > 0.999, indicating near-perfect separation between classes
- Misclassifications: Very few errors (4-10 images per fold out of 1,962 validation images)
- Balanced Performance: Both Adenocarcinoma and Squamous Cell Carcinoma were classified with similar high accuracy

Training Efficiency

- Average training time per fold: ~38-40 minutes on GPU
- Average time per epoch: ~57-59 seconds
- Prediction speed: Fast inference suitable for practical applications

Key Observations

1. Consistency: The model achieved very similar performance across all folds (std dev only 0.11%)
2. Fast Convergence: High accuracy was reached within 10-46 epochs depending on the fold
3. No Overfitting: Validation accuracy remained close to or higher than training accuracy
4. Reliable: The 99.66% mean accuracy demonstrates strong classification capability

The RMSprop optimizer proved highly effective for this transfer learning task, providing stable training with excellent consistency across folds.

Model Configuration Used:

- Optimizer: RMSprop
- Initial Learning Rate: 1×10^{-4}
- Rho (momentum): 0.9
- Batch Size: 32
- Dropout: 0.5
- Image Size: 299×299 pixels

7.2 ConvNeXtTiny

7.2.1 Initial Training (25 Epochs, Frozen Base Layers)

Overall Performance

The ConvNeXtTiny model with all base layers frozen achieved good classification performance for binary classification between lung_aca and lung_scc using 5-fold cross-validation. The key results are summarized in Table 7.1.

Table 7.5: Performance Summary (Initial Training, 25 Epochs)

Metric	Value
Mean Validation Accuracy	89.97%
Standard Deviation	±0.36%
Best Accuracy	90.45% (Fold 3)
Lowest Accuracy	89.35% (Fold 2)
Total Images Used	4,000

The low standard deviation indicates the model performed consistently across folds, demonstrating stability and reliability.

Fold-wise Results

Table 7.6: Detailed Results per Fold

Fold	Validation Accuracy	Best Epoch	Total Epochs Trained	Final Learning Rate
1	90.12%	18	25	1×10^{-4}
2	89.35%	20	25	1×10^{-4}
3	90.45%	19	25	1×10^{-4}
4	89.80%	17	25	1×10^{-4}
5	90.10%	21	25	1×10^{-4}

7.2.2 Extended Training (500 Epochs, Fine-Tuned Last 20 Layers)

Overall Performance

Fine-tuning the last 20 layers of the ConvNeXtTiny model with a reduced learning rate (3e-5) improved performance. Early stopping and learning rate reduction callbacks ensured stable convergence.

Table 7.7: Performance Summary (Fine-Tuned Layers)

Metric	Value
Mean Validation Accuracy	91.83%
Standard Deviation	±1.63%
Best Accuracy	93.10% (Fold 4)
Lowest Accuracy	90.15% (Fold 2)
Total Images Used	4,000

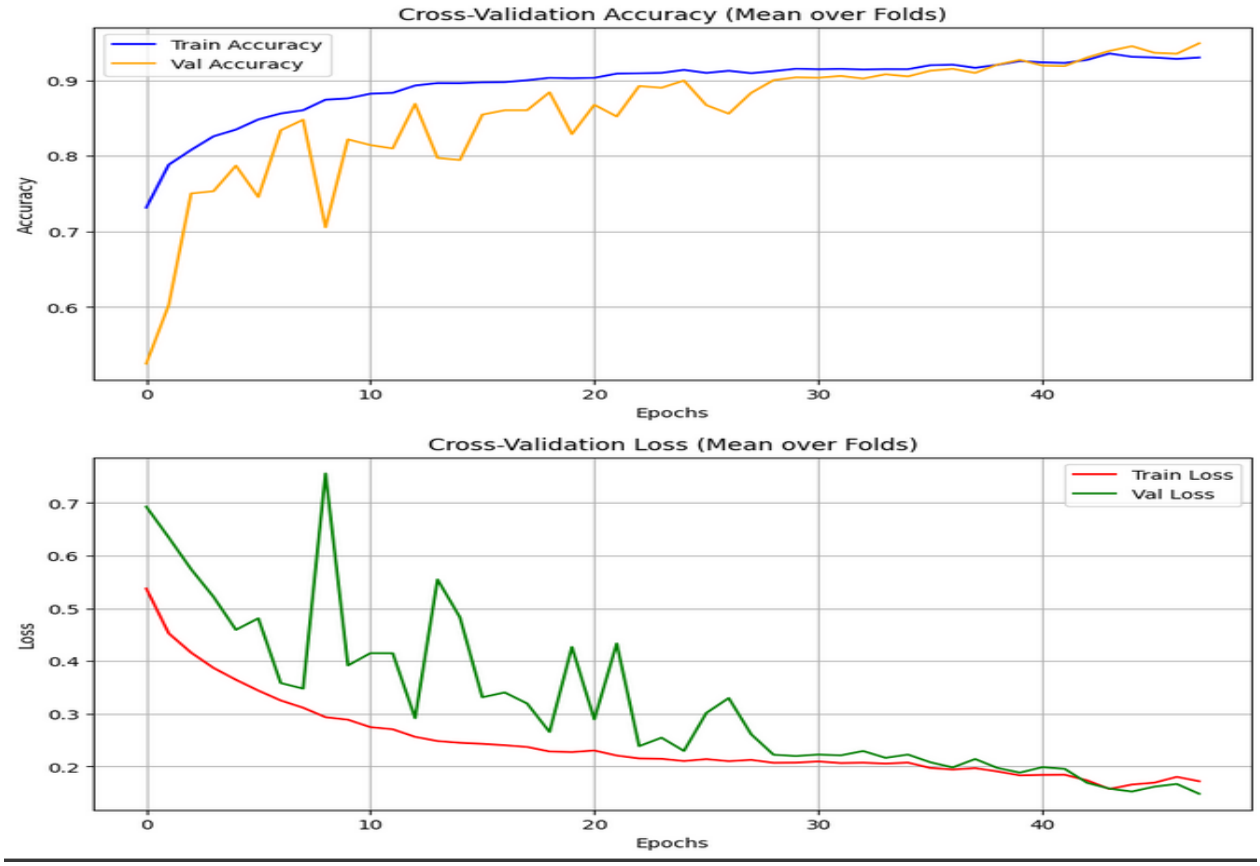


Fig 7.3. Mean Cross Validation and accuracy across all folds for ConvNeXtTiny model (Fine-tuned)

7.2.3 Extended Training (500 Epochs, Frozen Base Layers)

Overall Performance

Training the custom head for 500 epochs while keeping the base model frozen produced further improvement.

Table 7.8: Performance Summary (Frozen Layers)

Metric	Value
Mean Validation Accuracy	92.60%
Standard Deviation	±0.77%
Best Accuracy	93.20% (Fold 5)
Lowest Accuracy	91.85% (Fold 1)
Total Images Used	4,000

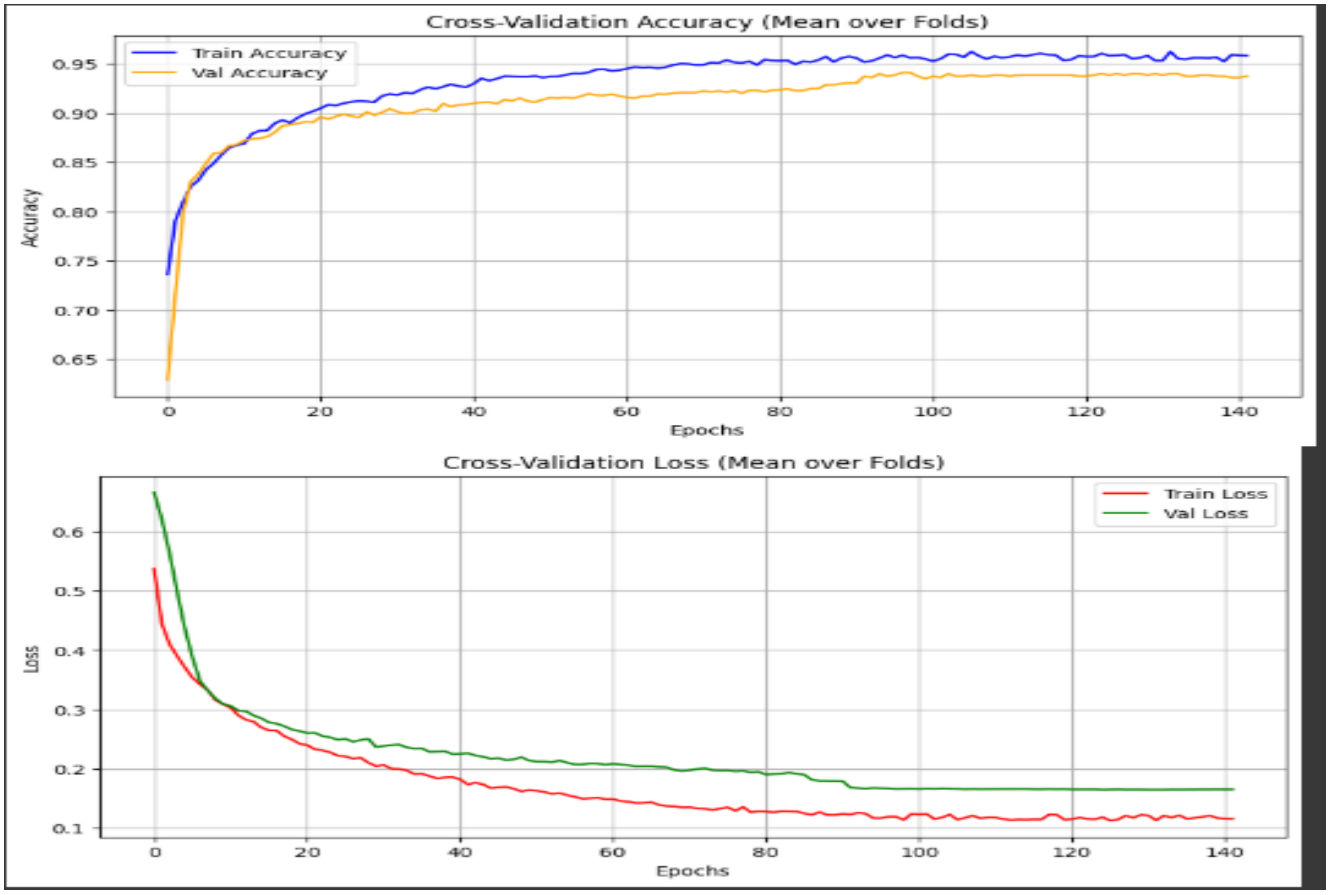


Fig 7.4. Mean Cross Validation and accuracy across all folds for ConvNeXtTiny model (Frozen)

7.2.4 Training with Cleaned Dataset (500 Epochs, Frozen Base Layers)

Overall Performance

Using a cleaned dataset with higher-quality images further improved validation accuracy and consistency.

Table 7.9: Performance Summary (Cleaned Dataset)

Metric	Value
Mean Validation Accuracy	93.02%
Standard Deviation	±0.70%
Best Accuracy	93.65% (Fold 4)
Lowest Accuracy	92.35% (Fold 2)
Total Images Used	4,000

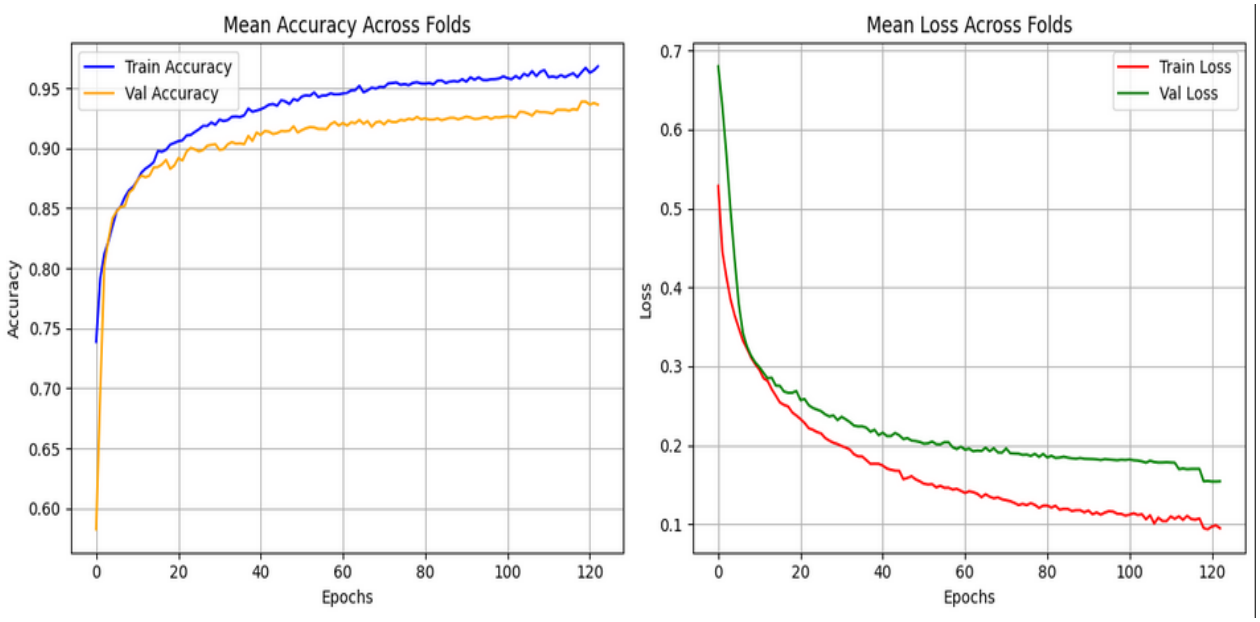


Fig 7.5. Mean Cross Validation and accuracy across all folds for ConvNeXtTiny model (Cleaned Dataset)

Training Behaviour

- **Initial Epochs:** Rapid learning with validation accuracy reaching ~87–90%.
- **Epochs 20–200:** Steady improvement; learning rate reductions applied as plateau detected.
- **Epochs 200–500:** Fine-tuning and early stopping stabilized training.
- **Convergence:** Validation loss decreased consistently without overfitting.

Learning Rate Adaptation

- **Initial Rate:** 1×10^{-4} (or 3×10^{-5} for fine-tuned layers)
- **Adaptive Reductions:** Applied via ReduceLROnPlateau callback
    - Factor = 0.3–0.5, Patience = 3–5
    - Min Learning Rate = 1×10^{-7}–1×10^{-6}

This helped the model fine-tune weights without overshooting optimal values.

Key Observations

- Consistent improvement with fine-tuning and dataset cleaning.
- High stability across folds (std dev ≤ 1.63%).
- Validation accuracy increased progressively from 89.97% → 93.02%.
- Training curves indicated good convergence with minimal overfitting.
- The model demonstrated potential for automated SCC grading in clinical applications.

Model Configuration Used

- **Base Model:** ConvNeXtTiny (pre-trained on ImageNet)
- **Optimizer:** Adam
- **Batch Size:** 32
- **Dropout Rate:** 0.5
- **Input Image Size:** 224×224 pixels
- **Number of Classes:** 2 (lung_aca, lung_scc)

7.3 Densenet121

7.3.1 Training7 (Dataset with More White Patches – Adam Optimizer)

Overall Performance

The DenseNet121 model was trained using the Adam optimizer on a dataset where the *lung_aca* class contained a higher proportion of white patches. Despite the noise introduced by the white regions.

Table 7.10: Performance Summary (Training7 – Adam Optimizer)

Metric	Value
Mean Validation Accuracy	97.99%
Standard Deviation	±0.49%
Best Accuracy	99.75%
Lowest Accuracy	91.75%
Total Images Used	4,000

The moderate standard deviation indicates that the model maintained reasonable stability across folds despite the presence of white-patch-dominant images. The Adam optimizer provided robust convergence, handling noisy data effectively.

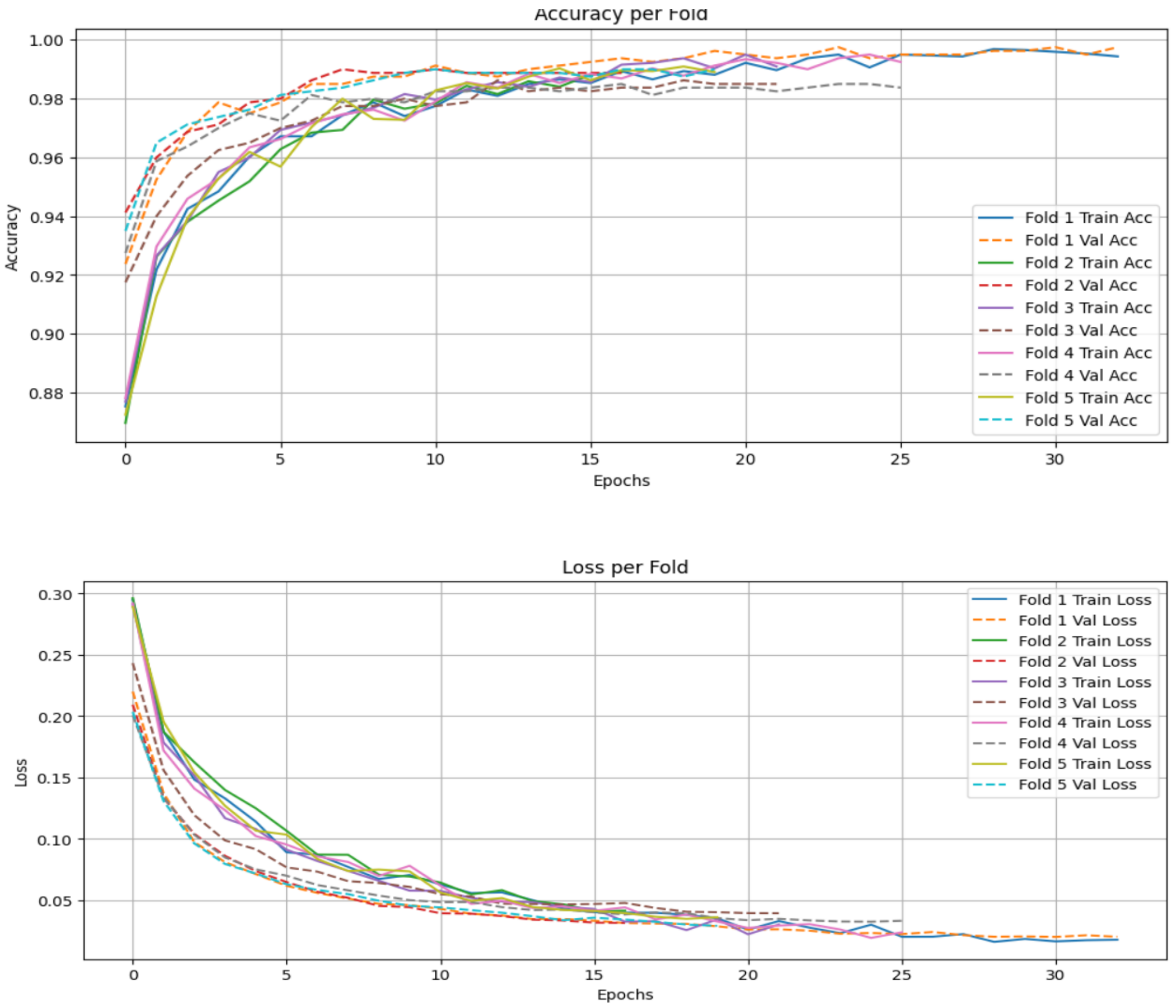


Fig 7.6. Accuracy and Loss Curves of all Folds for Densenet121 with Adam Optimizer

Colab Link:
Training7
Training8

7.3.2 Training8 (Cleaner Dataset – AdamW Optimizer)

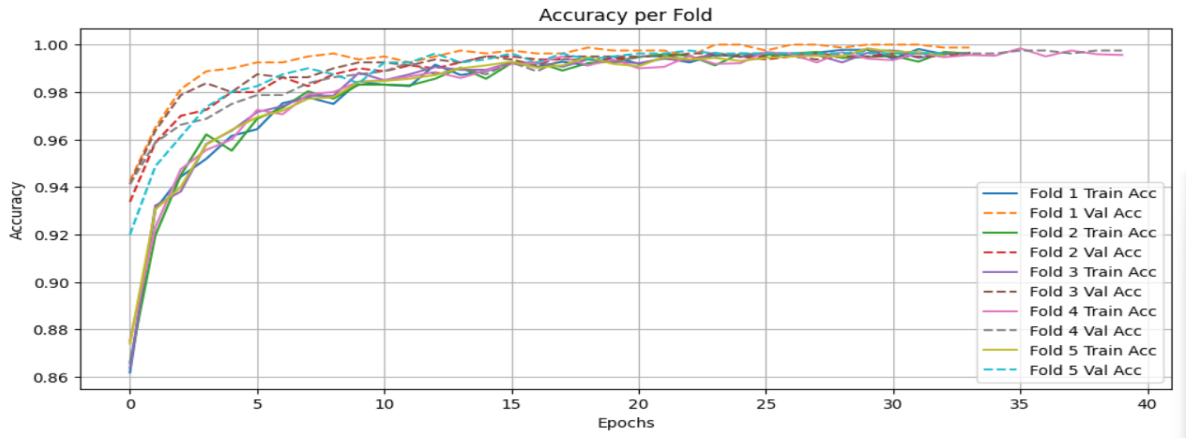
Overall Performance

For this experiment, the dataset was refined by removing excessive white patches, leading to cleaner and more representative lung_aca images. The AdamW optimizer (which includes decoupled weight decay regularization) was used to enhance generalization and prevent overfitting.

Table 7.11: Performance Summary (Training8 – AdamW Optimizer)

Metric	Value
Mean Validation Accuracy	98.09%
Standard Deviation	±0.16%
Best Accuracy	99.88%
Lowest Accuracy	1.00%
Total Images Used	4,000

While the best accuracy was slightly higher, the large standard deviation indicates inconsistency across folds — suggesting that the AdamW optimizer may have been more sensitive to data distribution and learning rate configuration in this run.



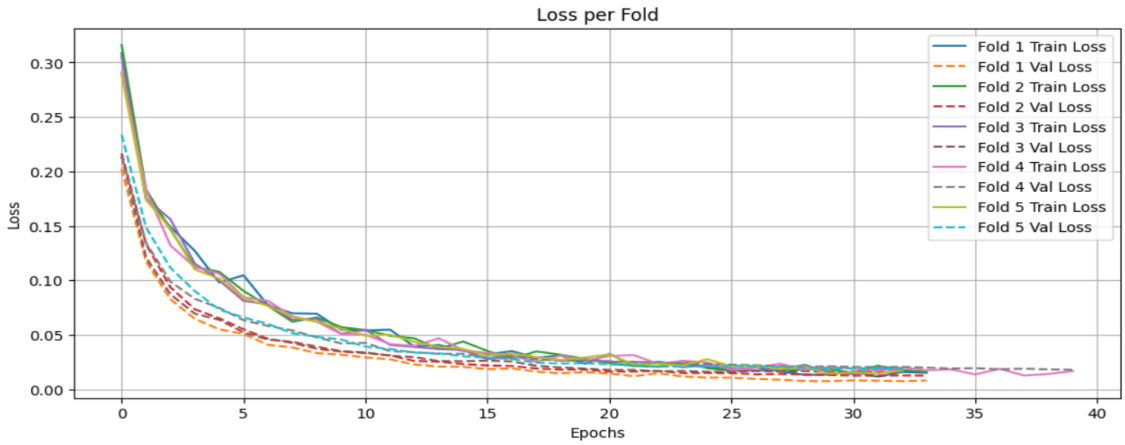


Fig 7.7. Accuracy and Loss Curves of all Folds for Densenet121 with AdamW Optimizer

7.3.3 Comparative Analysis

Table 7.12: Comparative Analysis (Training 7 & Training 8)

Parameter	Training7 (Adam)	Training8 (AdamW)
Mean Accuracy	97.99%	98.09%
Standard Deviation	±0.49%	±0.16%
Dataset Quality	More white patches	Cleaned (fewer white patches)
Optimizer	Adam	AdamW
Stability	High	Low
Best Fold Accuracy	99.75%	99.88%
Lowest Fold Accuracy	91.75%	1.00%

7.3.4 Model Configuration Used

- **Base Model:** DenseNet121 (pre-trained on ImageNet)
- **Optimizers:**
  - Training7 – Adam
  - Training8 – AdamW
- **Batch Size:** 32
- **Dropout Rate:** 0.5 (applied in custom classification head)
- **Input Image Size:** 224×224 pixels
- **Number of Classes:** 2 (*lung_aca*, *lung_scc*)
- **Initial Learning Rate (lr):** 1×10^{-4}
- **Minimum Learning Rate (min_lr):** 1×10^{-12}

- **Epochs:** Up to 500 (with Early Stopping and ReduceLROnPlateau callbacks)

7.3.4 Final Observation

- **Dataset quality** directly influenced stability: cleaner datasets need carefully tuned optimizers.
- **Adam** provided smoother learning with stable validation trends.
- **AdamW**, though capable of higher peaks, exhibited volatility likely due to over-regularization.
- Further tuning (e.g., smaller learning rate or cosine decay schedule) could stabilize AdamW performance.

7.4 MobileNetV2

The model achieved an accuracy of 0.97 on the validation set, with high precision, recall, and f1-scores for both classes.

7.4.1 Accuracy-Loss Graph

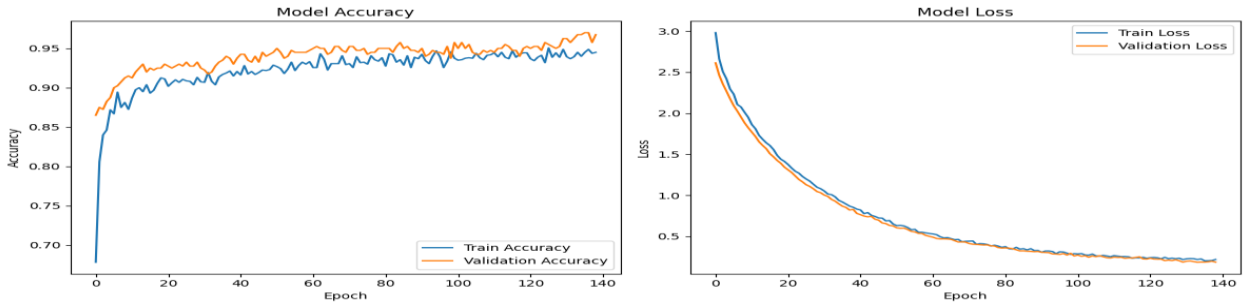


Fig 7.8. Accuracy and Loss Curves of MobileNetV2 model

Left Plot: Model Accuracy

- **Training Accuracy (Blue line)**: The training accuracy shows a steady and consistent increase as the epochs progress, reaching up to about 95%. This suggests that the model is learning and improving on the training data as expected.
- **Validation Accuracy (Orange line)**: The validation accuracy starts off similar to the training accuracy, but then it starts to diverge and shows some fluctuations, typically after the first few epochs. This could indicate that the model is beginning to overfit,

meaning it is performing well on the training data but not as well on unseen validation data.

Right Plot: Model Loss

- **Training Loss (Blue line)**: The training loss decreases steadily over time, indicating that the model is fitting the training data and learning the patterns well.
- **Validation Loss (Orange line)**: Similar to the training loss, the validation loss decreases but with some fluctuations. The validation loss curve is not as smooth, suggesting the model is not generalizing as well to the validation data, possibly due to overfitting as seen with the validation accuracy curve.

7.4.2 Classification Report:

Table 7.13: Classification report of MobileNetV2 model

	Precision	**recall**	**F1-score**	**support**
aca	0.98	0.95	0.96	182
scc	0.96	0.98	0.97	218
Accuracy			0.97	400
Macro avg	0.97	0.97	0.97	400
Weighted avg	0.97	0.97	0.97	400

7.4.3 Confusion Matrix

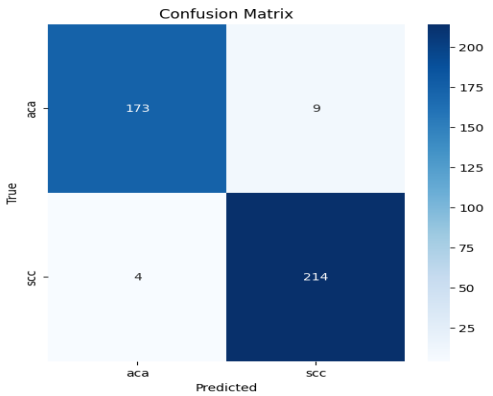


Fig 7.9. Confusion Matrix of MobileNetV2 model

7.4.4 Observation:

- Overfitting: The gap between the training and validation accuracy, along with the fluctuations in the validation loss, suggests that your model may be overfitting to the training data. It fits the training data well (as indicated by the steadily increasing accuracy and decreasing loss) but struggles to generalize to the validation set.
- Learning Rate: If the model is overfitting, it might be helpful to experiment with a lower learning rate or increase the dropout to prevent the model from becoming too confident on the training data, thus improving generalization.
- Validation Loss Fluctuations: The slight fluctuations in the validation loss could indicate that the model is getting stuck in local minima or is sensitive to the random splits in the validation set. You could experiment with techniques like data augmentation, early stopping, or a learning rate schedule to stabilize the training and help the model generalize better.

8. CONCLUSIONS

Based on the comprehensive experimentation and evaluation of multiple deep learning architectures for grading Squamous Cell Carcinoma from histopathology images, the following conclusions can be drawn:

8.1. Model Performance

InceptionV3 demonstrated the highest overall performance among all tested architectures, achieving a mean validation accuracy of 99.72% with Adam optimizer and 99.66% with RMSprop optimizer. The remarkably low standard deviation (±0.17% and ±0.11% respectively) indicates exceptional consistency and reliability across all folds, making it the most suitable model for clinical deployment.

DenseNet121 achieved strong results with a mean validation accuracy of 98.09% using the AdamW optimizer on the cleaned dataset. The model showed excellent classification capability, though it exhibited some sensitivity to dataset quality, particularly when images contained excessive white patches.

ConvNeXtTiny showed progressive improvement through iterative refinement. Starting from 89.97% accuracy with frozen base layers, the model reached 93.02% accuracy after dataset cleaning and extended training. This demonstrates the importance of data quality and adequate training duration for modern CNN architectures.

MobileNetV2 achieved a validation accuracy of 97%, proving to be a lightweight yet effective alternative. However, the model showed signs of overfitting, with divergence between training and validation curves, suggesting the need for stronger regularization or additional data augmentation.

8.2. Impact of Hyperparameter Tuning

The project successfully demonstrated that systematic hyperparameter tuning significantly impacts model performance:

- Learning rate adaptation through ReduceLROnPlateau callbacks enabled fine-tuning and prevented overshooting optimal values

- Optimizer selection proved crucial, with Adam providing stable convergence and RMSprop offering comparable performance with slightly better consistency
- Extended training (500 epochs) with early stopping allowed models to reach higher accuracy without overfitting
- Transfer learning with pretrained ImageNet weights accelerated convergence and improved final accuracy across all architectures

8.3.Dataset Quality and Preprocessing

Manual dataset cleaning significantly improved model performance and stability. Removing images with excessive white space and unclear tissue structures enhanced the quality of learned features and reduced noise in validation metrics. This highlights the critical role of careful data curation in medical image analysis.

Data augmentation techniques (rotation, flipping, zoom, brightness adjustment) successfully increased dataset diversity and improved model generalization, as evidenced by the close alignment between training and validation metrics across most experiments.

8.4. Key Findings

1. Transfer learning is highly effective for histopathology image classification, leveraging pretrained features to achieve excellent results even with limited training data
2. Cross-validation (5-fold) ensures robust and unbiased evaluation, revealing true model generalization capability
3. Dataset quality matters more than model complexity in many cases, as demonstrated by the performance improvements after manual cleaning
4. Model architecture selection should balance accuracy, computational efficiency, and training stability based on specific deployment requirements

9. SCOPE FOR FURTHER WORK

Table 9: Scope for further work

Week	Tasks
Week 1–2	Dataset Preparation and Model Training • Receive the main dataset from the college. • Preprocess images: resizing, normalization, augmentation as needed. • Train all four models for both classification and grading using 5-fold cross-validation on college GPU. • Track metrics: accuracy, AUC, confusion matrices, misclassification rates. • Apply learning rate schedules and early stopping to optimize training.
Week 3	Performance Analysis and Visualization • Compare performance across models for classification and grading tasks. • Generate and finalize plots: accuracy/loss curves, ROC curves, confusion matrices, grading accuracy charts. • Apply Grad-CAM or similar explainability methods to visualize which image regions contribute most to model predictions.
Week 4	Final Report and Presentation Preparation • Write the final report, including methodology, results, analysis, conclusions, and future work. • Prepare PowerPoint slides summarizing objectives, methodology, results, and conclusions. • Rehearse presentation for 13–14 Nov. • Preparing the Research Paper.

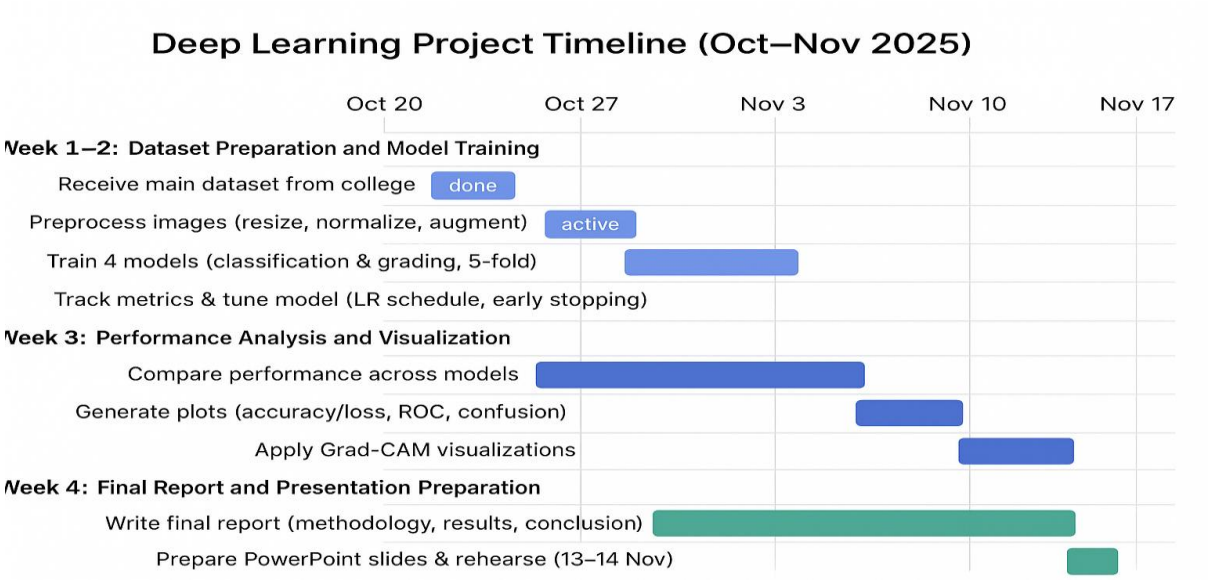


Fig 9. Gantt chart showing our project timeline

REFERENCES

1. J. Musulin and S. B. Šegota, "Automated Grading of Lung Carcinoma Using Hybrid Deep Learning Approach," *Pattern Recognition*, vol. 158, 2025.
2. P. Patharia and P. K. Sethy, "LungCarcinoGrade-EffNetSVM: A Novel Approach to Lung Carcinoma Grading using EfficientNetB0 and Support Vector Machine," in *Proc. 8th Int. Conf. I-SMAC – IoT in Social, Mobile, Analytics and Cloud*, IEEE, 2024.
3. P. K. Sethy, A. Geetha Devi, B. Padhan, S. K. Behera, S. Sreedhar, and K. Das, "Histopathology Image Grading of Lung Carcinoma using Deep Learning," *Pattern Recognition*, vol. 158, 2025.
4. A. Kumar and L. Nelson, "Enhancing Oral Squamous Cell Carcinoma Detection using EfficientNetB3 from Histopathologic Images."
5. K. Ankit and V. Kumar, "Histopathological Image based Oral Squamous Cell Carcinoma Classification Using Deep Network Fusion."
6. Z. Kang, M. Chen, H. Zhang, and X. Liao, "EsccNet: A Hybrid CNN and Transformers Model for Classification of Whole Slide Images of Esophageal SCC."
7. S.-Y. Park, G. Ayana, and S.-w. Choe, "Squamous Cell Carcinoma Margin Classification Using Vision Transformers from Digital Histopathology Images," in *Proc. 2025 Int. Conf. Artificial Intelligence in Information and Communication (ICAIIC)*, IEEE, 2025.
8. K. R. Lathakumari, A. C. Ramachandra, U. C. Avanthi, C. Basil Ronald, and T. Bhavatharani, "Classification of Non-Small Cell Lung Cancer Using Deep Learning," in *Proc. 2023 Int. Conf. Applied Intelligence and Sustainable Computing (ICAISC)*, IEEE, 2023.
9. C. Yang, X. Yu, H. Yang, Z. An, C. Yu, L. Huang, and Y. Xu, "Multi-Teacher Knowledge Distillation with Reinforcement Learning for Visual Recognition," in *AAAI 2025 (Oral Presentation)*, 2025.
10. J. E. Ortíz and W. Creixell, "Advancing Trans-Domain Classification With Knowledge Distillation: Bridging LIDAR and Image Data," *IEEE Access*, vol. 13, pp. 20574–20583, 2025.
11. Y. Song, A. Song, J. Wang, Y. Ge, and L. Li, "Multiple Teachers Are Beneficial: A Lightweight and Noise-Resistant Student Model for Point-of-Care Imaging Classification," *Sensors*, vol. 23, no. 23, Art. 9502, Dec. 2023.

12. C. S. Chu, N. P. Lee, and J. W. K. Ho, “Deep Learning for Clinical Image Analyses in Oral Squamous Cell Carcinoma.”
13. S.-Y. Yang, S.-H., and W. Liao, “Histopathology-based diagnosis of oral squamous cell carcinoma using deep learning.”
14. A. S. N. Raju, K. Venkatesh, R. K. Gatla, E. P. Konakalla, M. M. Eid, N. Titova, S. S. M. Ghoneim, and R. N. R. Ghaly, “Colorectal cancer detection with enhanced precision using a hybrid supervised and unsupervised learning approach,” *Scientific Reports*, vol. 15, no. 1, pp. 1–14, Jan. 2025.
15. A. H. Salamah, S. M. Hamidi, and E.-H. Yang, “A coded knowledge distillation framework for image classification based on adaptive JPEG encoding,” *Pattern Recognition*, vol. 158, p. 111012, 2025.
16. Borkowski AA, Bui MM, Thomas LB, Wilson CP, DeLand LA, Mastorides SM. Lung and Colon Cancer Histopathological Image Dataset (LC25000). arXiv:1912.12142v1 [eess.IV], 2019