Lung Cancer Detection Using Convolutional Neural Network

S M Firdous Hassan[1] and Dr. Abhishek Das[2]

[1] Aliah University, ll-A/27, Action Area II, Newtown, Kolkata, West Bengal
[2] Dr. Abhishek Das, Associate Prof., Aliah University, Kolkata, West Bengal
lncs@springer.com

Abstract. Lung Cancer is one the leading causes of death in the world. This Lung Cancer Detection project is an approach to transform the lung cancer detection process. This project uses CNN(Convolutional Neural Networks) to obtain the same. CNNs are widely used in the field of image processing and analysis. By using the neural networks, this study aims to automate the hectic process of classifying histopathological images and process the subtle differences between three different types of lung cancer : Adenocarcinoma, Squamous and Normal tissues. Traditional methods of classification require substantial expertise and time to manually process and classify the images and may also lead to diagnostic errors, thus using these deep learning methods helps to automate these tasks and provide us with a quicker and more accurate diagnosis.

Keywords: Lung Cancer, CNN, Histopathological images, Deep Learning.

1 Introduction

Lung cancer, also referred as lung carcinoma, is the second most common type of cancer in the world with over 2 million cases in recent years.(Most common in men and the 3rd most common types in females). Early lung cancer has no symptoms and can only be detected with medical image analysis. Diagnosis of a cancerous tumor cell can be examined by pathology under a microscope and a pathologist can subdivide the cells into three major types namely adenocarcinomas, squamous-cell carcinomas, and large-cell carcinomas.

Deep learning is used for the classification of Histopathological Images into three subtypes – adenocarcinoma, squamous and normal lung tissue. Convolution Neural Networks is used to extract different features from the images through an algorithm. During the training stage of the model, input and output labels are provided and based on that the algorithm analyses the features/patterns for a training data and forms a set of parameters and feature extraction using Convolutional Layers. Pooling layers bring together the computations with similar permutation and reduce the complexity. The convolution filter will form a spatially dense output by assigning a common value to a set of matrix pixels These values decide the output for that image.

2 Literature Survey

Convolutional neural network (ConvNet/CNN), Deep Learning or other machine learning algorithms have been used various researchers to perform experiments on different types of lung cancer detection.[1]

The dataset, titled "Lung and Colon Cancer Histopathological Images.," is obtained from Kaggle and encompasses 15000 different images already augment from 750 images of three classes, adenocarcinoma, squamous and normal.[2]

Öztürk et al., proposed a model where five types of feature extraction techniques were used in individual classification algorithm to predict at which features extraction technique which machine learning algorithm is giving more accuracy.[3]

Sumathipala et al., proposed a model where the image data are taken from LIDC-IDRI, after collecting the image, data image filtration has been implemented, filtration is done based on the patient who went through biopsy.[4]

3 Methodology

3.1 Dataset

The dataset utilized in this project has been curated from Kaggle, a prominent platform for machine learning datasets. The dataset, titled "Lung and Colon Cancer Histopathological Images.," provides us a diverse collection of labelled images, each representing distinct histopathological patterns associated with lung cancer.

This dataset contains 15,000 histopathological images with 3 classes. All images are 768 x 768 pixels in size and are in jpeg file format. The images were generated from an original sample of HIPAA compliant and validated sources, consisting of 750 total images of lung tissue (250 benign lung tissue, 250 lung adenocarcinomas, and 250 lung squamous cell carcinomas).

Original Article : 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

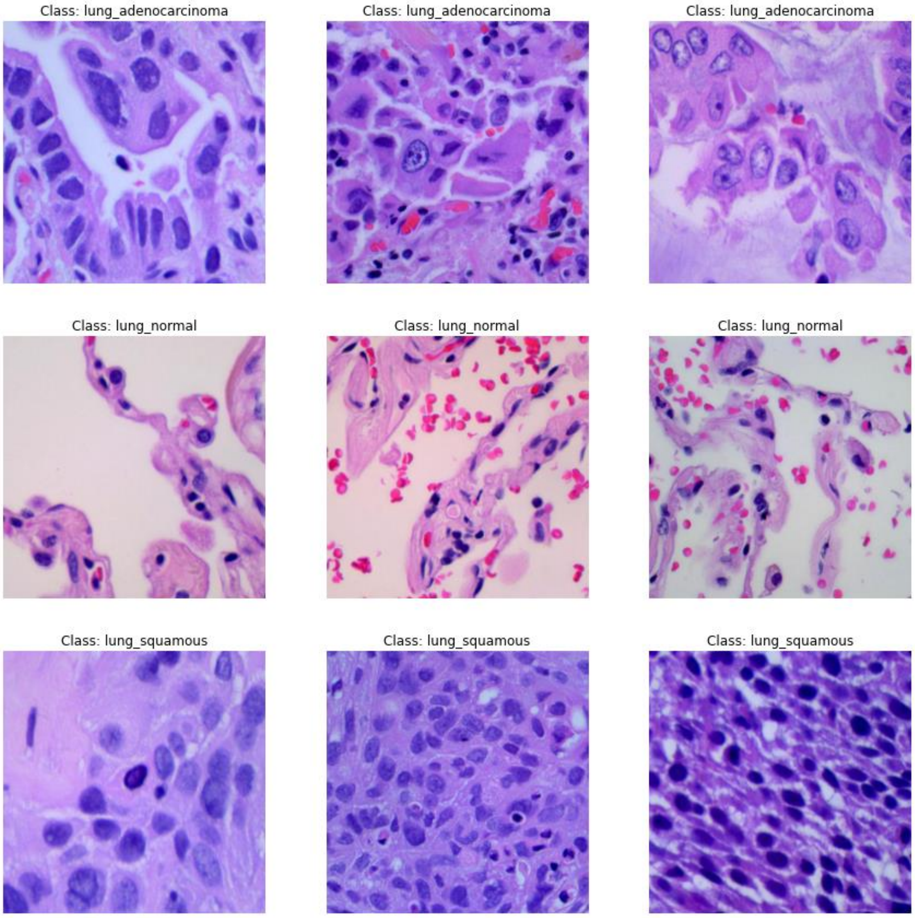
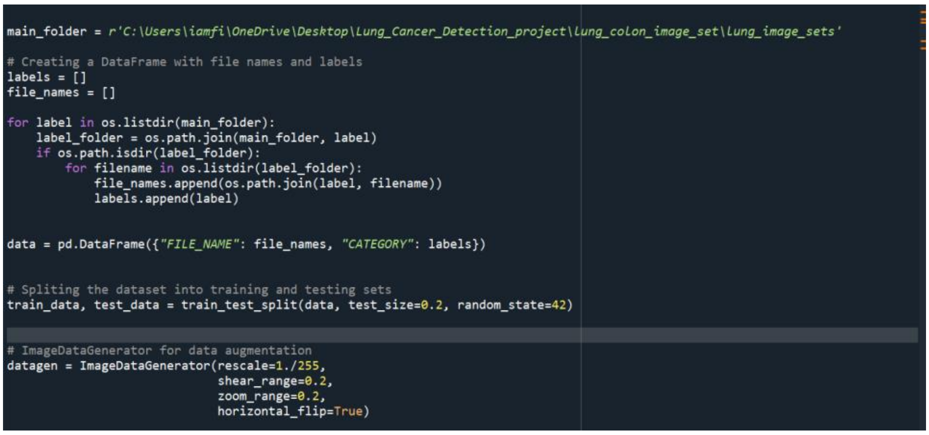


Fig. 1 Images from the lung image dataset

3.2 Image Preprocessing

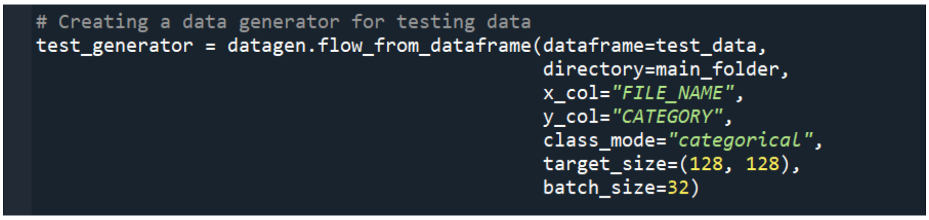
Before feeding the images into the CNN, several preprocessing steps are applied:

- **Resizing**: Standardizing image dimensions to (128, 128) pixels.
- **Normalization**: Scaling pixel values to the range [0, 1] to facilitate model convergence.
- **Data Augmentation**: Employing techniques such as shear, zoom, and horizontal flip to artificially increase the size of the training dataset and improve model generalization.



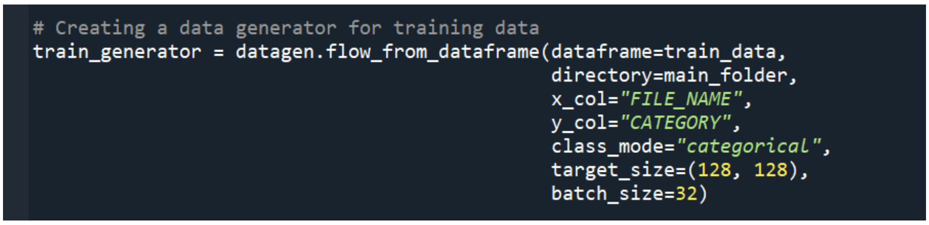
```
main_folder = r'C:\Users\iamfi\OneDrive\Desktop\Lung_Cancer_Detection_project\Lung_colon_image_set\Lung_image_sets'

# Creating a DataFrame with file names and labels
labels = []
file_names = []

for label in os.listdir(main_folder):
    label_folder = os.path.join(main_folder, label)
    if os.path.isdir(label_folder):
        for filename in os.listdir(label_folder):
            file_names.append(os.path.join(label, filename))
            labels.append(label)


data = pd.DataFrame({"FILE_NAME": file_names, "CATEGORY": labels})


# Spliting the dataset into training and testing sets
train_data, test_data = train_test_split(data, test_size=0.2, random_state=42)


# ImageDataGenerator for data augmentation
datagen = ImageDataGenerator(rescale=1./255,
                             shear_range=0.2,
                             zoom_range=0.2,
                             horizontal_flip=True)
```

Fig. 2. Data preparation and Augmentation



```
# Creating a data generator for testing data
test_generator = datagen.flow_from_dataframe(dataframe=test_data,
                                             directory=main_folder,
                                             x_col="FILE_NAME",
                                             y_col="CATEGORY",
                                             class_mode="categorical",
                                             target_size=(128, 128),
                                             batch_size=32)
```

Fig. 3. Test Data Generator



```
# Creating a data generator for training data
train_generator = datagen.flow_from_dataframe(dataframe=train_data,
                                              directory=main_folder,
                                              x_col="FILE_NAME",
                                              y_col="CATEGORY",
                                              class_mode="categorical",
                                              target_size=(128, 128),
                                              batch_size=32)
```

Fig. 4. Train Data Generator

3.3 Convolutional Neural Network (CNN)

A Convolutional Neural Network(CNN) is a class of Deep Learning architecture which specializes in extracting features from complex visual data (here histopathological images) making them suitable for the task of image classification and recognition. CNN is used to identify subtle patterns and spatial hierarchies from the histopathological images provided to classify them into the subtypes. The model does so by applying various convolution filters to the input images and analyze them. The filters helps in recognizing distinct features relevant to adenocarcinoma, squamous and normal lung tissue. The subsequent layers , maxpooling and dense layers with a small dropout feature helps the model to further learn more about the input images thus helping the model to make more accurate predictions based on the learned process

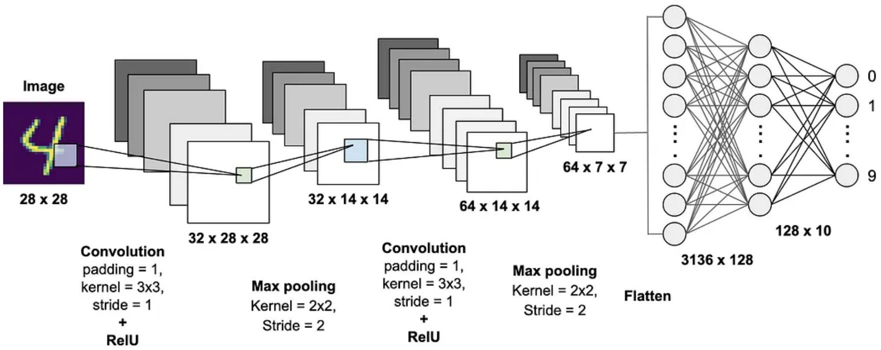


Fig. 5. CNN Model

3.4 Model Architecture

Convolutional Layers:

Convolutional layers play a fundamental role in learning spatial hierarchies of features from the input images. In this project, three convolutional layers are strategically employed. The filter sizes progressively increase, starting from 32 and reaching up to 128. Rectified Linear Unit (ReLU) activation functions are applied to introduce non-linearity, enhancing the model's capacity to capture intricate patterns within the images.

MaxPooling Layers:

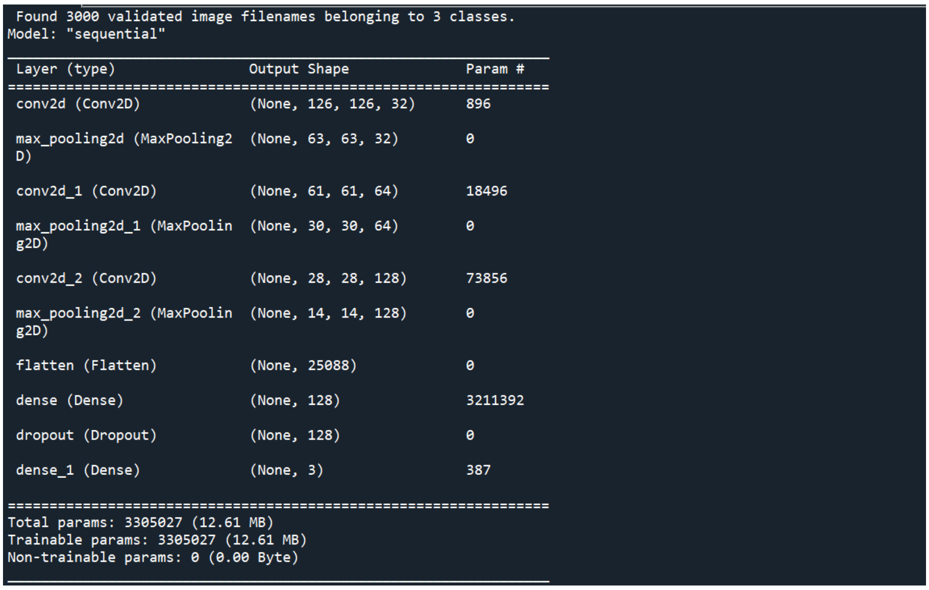
After each convolutional layer, max-pooling layers are employed to downsample the spatial dimensions of the input. This reduction in spatial resolution helps the model to minimize the computational complexity and promoting translation invariance. The application of max-pooling helps the model focus on the most prominent features while discarding less relevant information.

Flatten Layer:

The flatten layer serves as a crucial component in the model architecture by converting the spatial information learned by the convolutional layers into a one-dimensional array. This transformation is essential to transition from the spatial hierarchies captured in the convolutional layers to a format compatible with the subsequent dense layers.

Dense Layer:

Two dense layers are applied after flattening the input images. The first dense layer consists of 128 neurons with Rectifies Linear Unit (ReLU) activation which introduces non-linearity to the dataset. Additionally, a dropout of 0.5 is implemented in this layer to control overfitting. The final dense layer having 3 neurons with softmax activation is applied for multiclass classification. This layer provides the probability and distribute them across the three classes- adenocarcinoma, normal and squamous.



```
 Found 3000 validated image filenames belonging to 3 classes.
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d (Conv2D)             (None, 126, 126, 32)      896

 max_pooling2d (MaxPooling2  (None, 63, 63, 32)        0
 D)

 conv2d_1 (Conv2D)           (None, 61, 61, 64)        18496

 max_pooling2d_1 (MaxPoolin  (None, 30, 30, 64)        0
 g2D)

 conv2d_2 (Conv2D)           (None, 28, 28, 128)       73856

 max_pooling2d_2 (MaxPoolin  (None, 14, 14, 128)       0
 g2D)

 flatten (Flatten)           (None, 25088)             0

 dense (Dense)               (None, 128)               3211392

 dropout (Dropout)           (None, 128)               0

 dense_1 (Dense)             (None, 3)                 387

=================================================================
Total params: 3305027 (12.61 MB)
Trainable params: 3305027 (12.61 MB)
Non-trainable params: 0 (0.00 Byte)
_________________________________________________________________
```

Fig. 6. Model Architecture

3.5 Model Evaluation and Training

Training Process:

The model training process involves the optimization of the CNN's parameters using the Adam optimizer. The dataset is divided into training and validation sets, ensuring the model generalizes well to unseen data. During each epoch, the model iteratively adjusts its internal parameters to minimize the categorical crossentropy loss, which measures the dissimilarity between predicted and actual class distributions. This iterative optimization process aims to enhance the model's ability to accurately classify lung cancer histopathological images.

Image Data Augmentation:

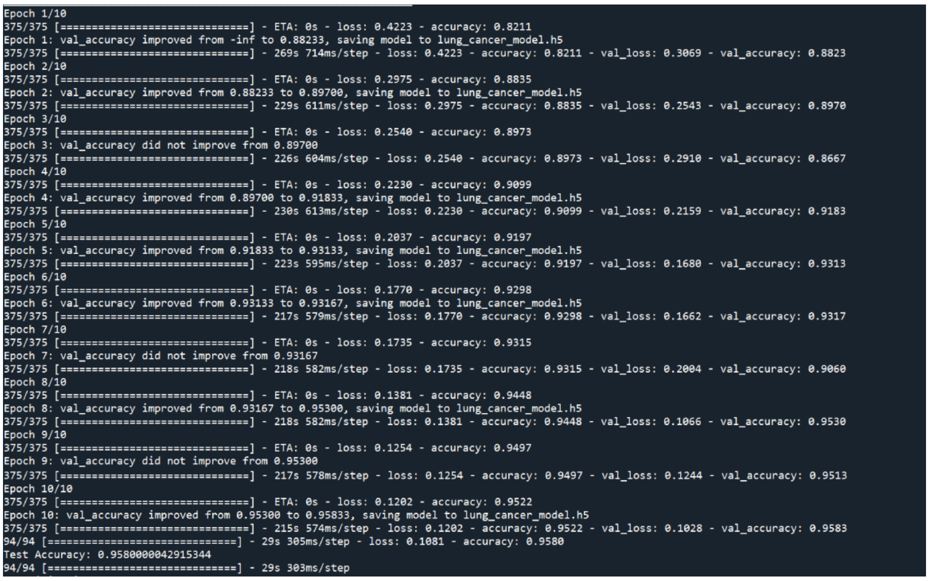
To improve model generalization and control overfitting, image data augmentation techniques are applied during training. These techniques include shear, zoom, and horizontal flip operations, artificially diversifying the training dataset. This augmentation strategy helps the model learn robust features from variations in the input images, enhancing its performance on unseen data. However, the data set used in this research was already augmented so this process was skipped from the training of model.

Validation:

The model's performance is monitored using a separate validation dataset during training. At the end of each epoch, the model is evaluated on the validation set to assess its generalization capabilities. The validation accuracy serves as a crucial metric, indicating how well the model performs on data it has not been explicitly trained on. The training process continues until the model reaches convergence, achieving optimal performance on both the training and validation sets.

Model Evaluation:

Once the training is complete, the model is evaluated on an independent test dataset to assess its real-world performance. Evaluation metrics include accuracy, precision, recall, and F1 score. The confusion matrix provides a detailed breakdown of the model's predictions across the three classes—adenocarcinoma, normal, and squamous. This comprehensive evaluation ensures a thorough understanding of the model's strengths and limitations in lung cancer classification.



```
Epoch 1/10
375/375 [==============================] - ETA: 0s - loss: 0.4223 - accuracy: 0.8211
Epoch 1: val_accuracy improved from -inf to 0.88233, saving model to lung_cancer_model.h5
375/375 [==============================] - 269s 714ms/step - loss: 0.4223 - accuracy: 0.8211 - val_loss: 0.3069 - val_accuracy: 0.8823
Epoch 2/10
375/375 [==============================] - ETA: 0s - loss: 0.2975 - accuracy: 0.8835
Epoch 2: val_accuracy improved from 0.88233 to 0.89700, saving model to lung_cancer_model.h5
375/375 [==============================] - 229s 611ms/step - loss: 0.2975 - accuracy: 0.8835 - val_loss: 0.2543 - val_accuracy: 0.8970
Epoch 3/10
375/375 [==============================] - ETA: 0s - loss: 0.2540 - accuracy: 0.8973
Epoch 3: val_accuracy did not improve from 0.89700
375/375 [==============================] - 226s 604ms/step - loss: 0.2540 - accuracy: 0.8973 - val_loss: 0.2910 - val_accuracy: 0.8667
Epoch 4/10
375/375 [==============================] - ETA: 0s - loss: 0.2230 - accuracy: 0.9099
Epoch 4: val_accuracy improved from 0.89700 to 0.91833, saving model to lung_cancer_model.h5
375/375 [==============================] - 230s 613ms/step - loss: 0.2230 - accuracy: 0.9099 - val_loss: 0.2159 - val_accuracy: 0.9183
Epoch 5/10
375/375 [==============================] - ETA: 0s - loss: 0.2037 - accuracy: 0.9197
Epoch 5: val_accuracy improved from 0.91833 to 0.93133, saving model to lung_cancer_model.h5
375/375 [==============================] - 223s 595ms/step - loss: 0.2037 - accuracy: 0.9197 - val_loss: 0.1680 - val_accuracy: 0.9313
Epoch 6/10
375/375 [==============================] - ETA: 0s - loss: 0.1770 - accuracy: 0.9298
Epoch 6: val_accuracy improved from 0.93133 to 0.93167, saving model to lung_cancer_model.h5
375/375 [==============================] - 217s 579ms/step - loss: 0.1770 - accuracy: 0.9298 - val_loss: 0.1662 - val_accuracy: 0.9317
Epoch 7/10
375/375 [==============================] - ETA: 0s - loss: 0.1735 - accuracy: 0.9315
Epoch 7: val_accuracy did not improve from 0.93167
375/375 [==============================] - 218s 582ms/step - loss: 0.1735 - accuracy: 0.9315 - val_loss: 0.2004 - val_accuracy: 0.9060
Epoch 8/10
375/375 [==============================] - ETA: 0s - loss: 0.1381 - accuracy: 0.9448
Epoch 8: val_accuracy improved from 0.93167 to 0.95300, saving model to lung_cancer_model.h5
375/375 [==============================] - 218s 582ms/step - loss: 0.1381 - accuracy: 0.9448 - val_loss: 0.1066 - val_accuracy: 0.9530
Epoch 9/10
375/375 [==============================] - ETA: 0s - loss: 0.1254 - accuracy: 0.9497
Epoch 9: val_accuracy did not improve from 0.95300
375/375 [==============================] - 217s 578ms/step - loss: 0.1254 - accuracy: 0.9497 - val_loss: 0.1244 - val_accuracy: 0.9513
Epoch 10/10
375/375 [==============================] - ETA: 0s - loss: 0.1202 - accuracy: 0.9522
Epoch 10: val_accuracy improved from 0.95300 to 0.95833, saving model to lung_cancer_model.h5
375/375 [==============================] - 215s 574ms/step - loss: 0.1202 - accuracy: 0.9522 - val_loss: 0.1028 - val_accuracy: 0.9583
94/94 [==============================] - 29s 305ms/step - loss: 0.1081 - accuracy: 0.9580
Test Accuracy: 0.9580000042915344
94/94 [==============================] - 29s 303ms/step
```

Fig. 7. Model Training

4 Results

The trained model demonstrates promising performance on the test dataset, achieving high accuracy and effectively classifying lung cancer histopathological images into adenocarcinoma, normal, and squamous categories.

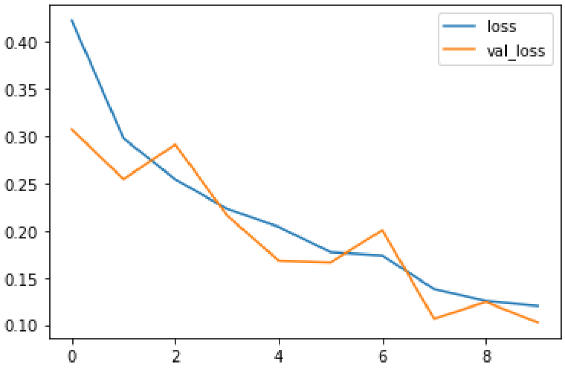


Fig. 8. Loss and Validation Loss

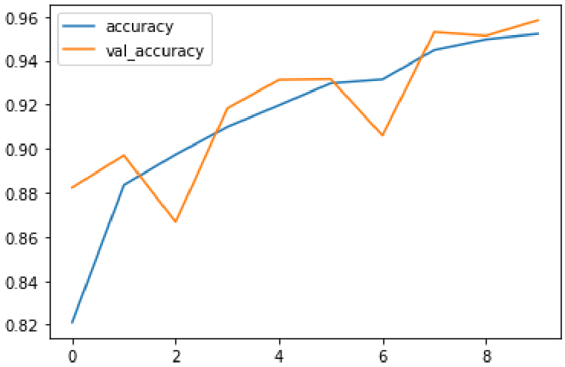
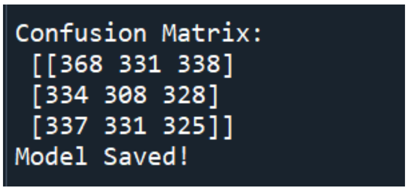
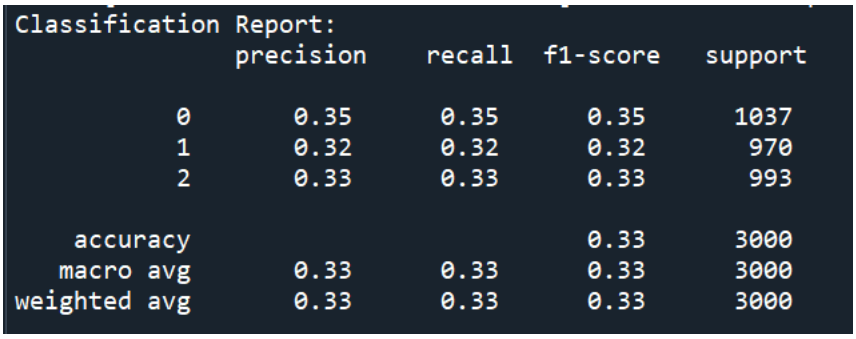


Fig. 9. Accuracy and Validation Accuracy



```
Confusion Matrix:
 [[368 331 338]
 [334 308 328]
 [337 331 325]]
Model Saved!
```

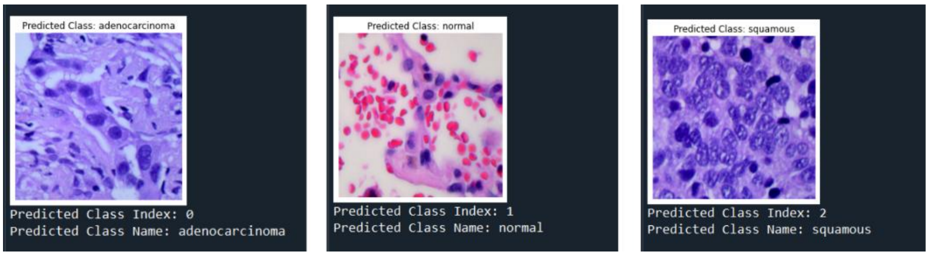
Fig. 10. Confusion Matrix



```
Classification Report:
              precision    recall  f1-score   support

           0       0.35      0.35      0.35      1037
           1       0.32      0.32      0.32       970
           2       0.33      0.33      0.33       993

    accuracy                           0.33      3000
   macro avg       0.33      0.33      0.33      3000
weighted avg       0.33      0.33      0.33      3000
```

Fig.11. Classification Report

4.1 Sample Output :



```
Predicted Class Index: 0
Predicted Class Name: adenocarcinoma

Predicted Class Index: 1
Predicted Class Name: normal

Predicted Class Index: 2
Predicted Class Name: squamous
```

Fig.12. Outputs of three classes – adenocarcinoma, squamous and normal lung tissue.

5 Future Work

The model is working well but it can be further fine-tuned through experimenting with more hyperparameter tuning and some architecture adjustments. Additionally, combination of different predictions from multiple models can be a good approach also considering the use of pre trained models like VGG16 or ResNet for transfer learning to leverage the knowledge gained from other image classification tasks.

6 Conclusion

This project represents a pivotal step towards the capabilities of deep learning for the automated classification of lung cancer histopathological images. The CNN architecture in addition to the diverse dataset has provided a detection model with a commendable accuracy rate in distinguishing the subtypes – adenocarcinoma, normal and squamous. However, many more fine tuning and other architecture methods should be implemented in the future to increase the efficiency of the model further.

7 References

1. 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

2. Krishnaiah, V., G. Narsimha, and Dr N. Subhash Chandra. "Diagnosis of lung cancer prediction system using data mining classification techniques." International Journal of Computer Science and Information Technologies 4.1 (2013): 39-45

3. Daoud, Maisa, and Michael Mayo. "A survey of neural network-based cancer prediction models from microarray data." Artificial intelligence in medicine (2019).

4. Masud, Mehedi, Niloy Sikder, Abdullah-Al Nahid, Anupam Kumar Bairagi, and Mohammed A. AlZain. "A machine learning approach to diagnosing lung and colon cancer using a deep learning-based classification framework." Sensors 21, no. 3 (2021): 748.

5. Hage Chehade, A., Abdallah, N., Marion, J.M., Oueidat, M. and Chauvet, P., 2022. Lung and colon cancer classification using medical imaging: A feature engineering approach. Physical and Engineering Sciences in Medicine, 45(3), pp.729-746.

6. Goodfellow, I., Bengio, Y., Courville, A. (2016). Deep Learning. MIT Press.

7. Pedregosa, F. et al. (2011). Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research, 12, 2825–2830.