

Deep Learning

880663-M-6

Assignment

Using Deep Learning to Perform Multi-Class Classification on the
Lung and Colon Cancer Histopathological
Image Dataset (LC25000)

Report by:

Fabian Waldmann (2106484)

March 2024

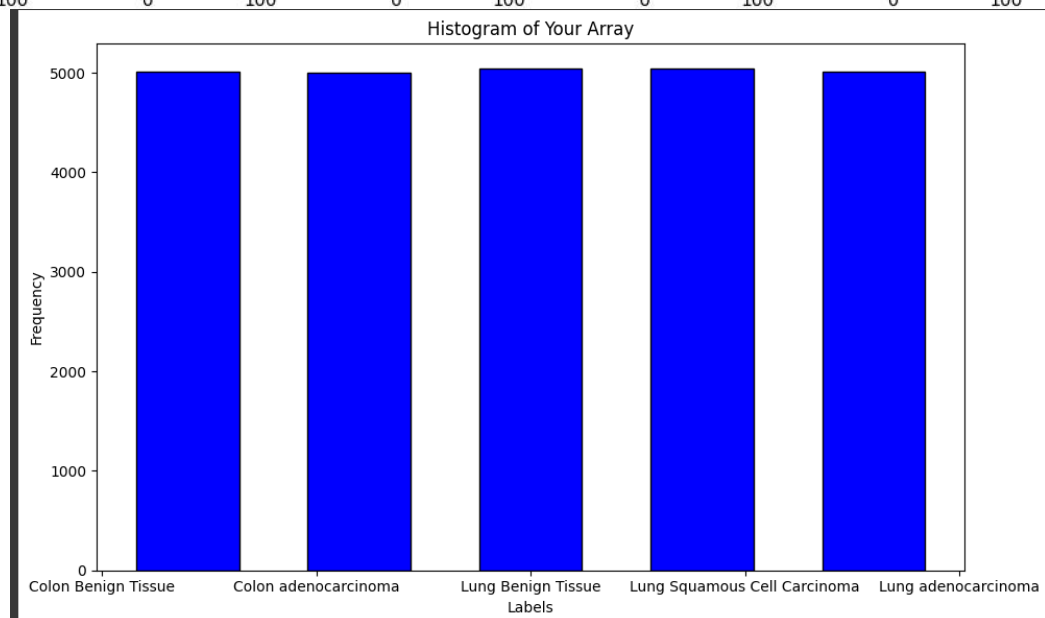
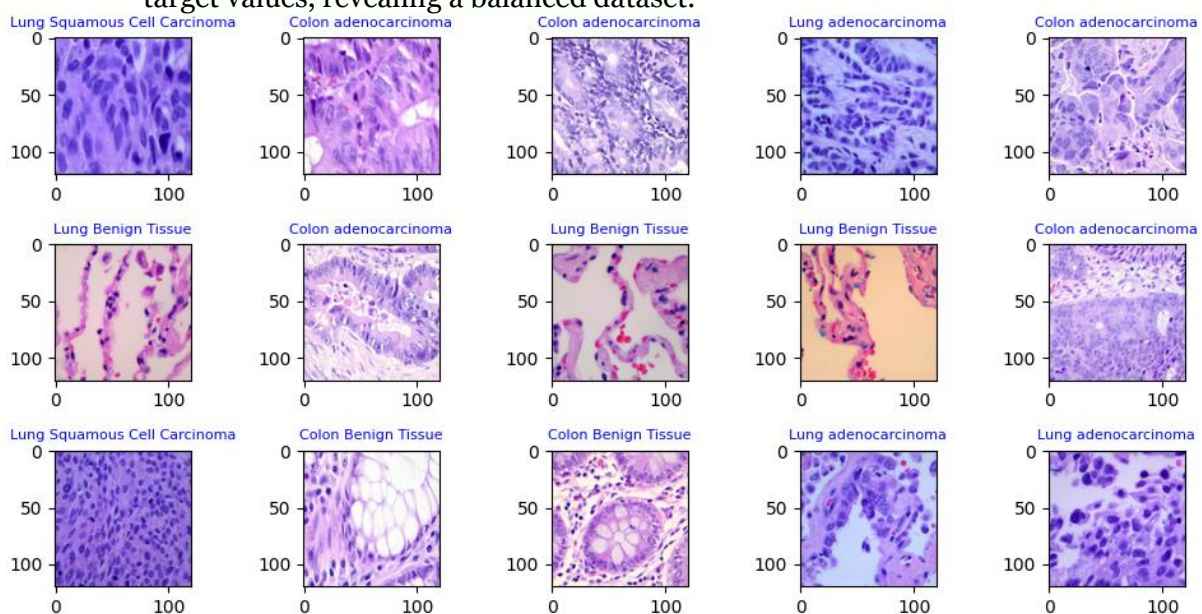
1. Problem Definition

As the field of deep learning has become increasingly important in recent years, it is being used intensively in the domain of computer vision and image analysis. Once trained, the trained network can systematically analyze new medical images and provide diagnostic insights to medical professionals, underscoring its profound societal importance.

The following paper aims to enhance this research by the classification of lung and colon cancer through Deep Learning technique. Therefore, the dataset “Lung and Colon Cancer Histopathological Image Dataset (LC25000)” is utilized. The dataset consists of 25.000 images represented by 5 histologic entities: colon adenocarcinoma, benign colonic tissue, lung adenocarcinoma, lung squamous cell carcinoma, and benign lung tissue. Each class contains 5.000 images, are 768 x 768 pixels in size and is designed for multiclass classification.

2. Exploratory Data Analysis

In the Exploratory Data Analysis (EDA) phase, the initial investigation focused on the shape of the data and labels. To enhance the model's learning efficiency, one-hot encoding was applied to the target values. Further insights into the images and their corresponding labels were gained by visualizing and examining 15 random images. The EDA concluded with a histogram displaying the frequency distribution of the target values, revealing a balanced dataset.

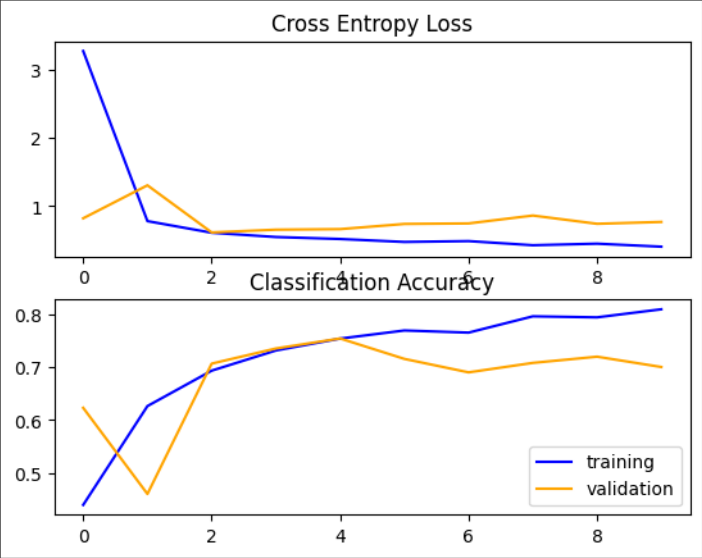


3. Results of the Baseline Model

In the following the results of the baseline model on the validation and test set is presented. In the visualized run it yields an accuracy of 0.7036.

```
Epoch 10/10
471/471 [=====] - 15s 32ms/step - loss: 0.4097 - accuracy: 0.8098 - val_loss: 0.7711 - val_accuracy: 0.7006
157/157 [=====] - 1s 8ms/step - loss: 0.7505 - accuracy: 0.7036
```

```
[20] summarize_diagnostics(history_baseline)
```



Results on Validation set:

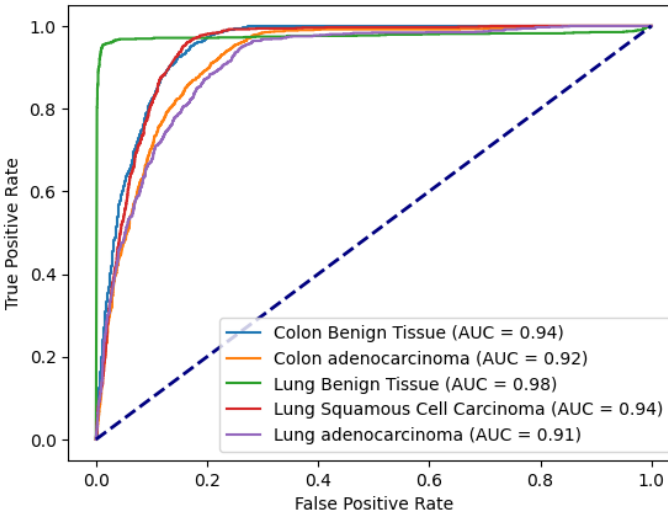
Classification Report:

	precision	recall	f1-score	support
Colon Benign Tissue	0.80	0.33	0.47	1002
Colon adenocarcinoma	0.55	0.88	0.68	1000
Lung Benign Tissue	0.96	0.94	0.95	1008
Lung Squamous Cell Carcinoma	0.71	0.69	0.70	1008
Lung adenocarcinoma	0.64	0.67	0.65	1002
accuracy			0.70	5020
macro avg	0.73	0.70	0.69	5020
weighted avg	0.73	0.70	0.69	5020

Confusion Matrix

True Labels	Colon Benign Tissue	Colon adenocarcinoma	Lung Benign Tissue	Lung Squamous Cell Carcinoma	Lung adenocarcinoma
	329	667	5	0	1
	64	875	5	6	50
	16	23	946	2	21
	0	1	0	691	316
	0	24	25	277	676
Predicted Labels					
Colon Benign Tissue					
Colon adenocarcinoma					
Lung Benign Tissue					
Lung Squamous Cell Carcinoma					
Lung adenocarcinoma					

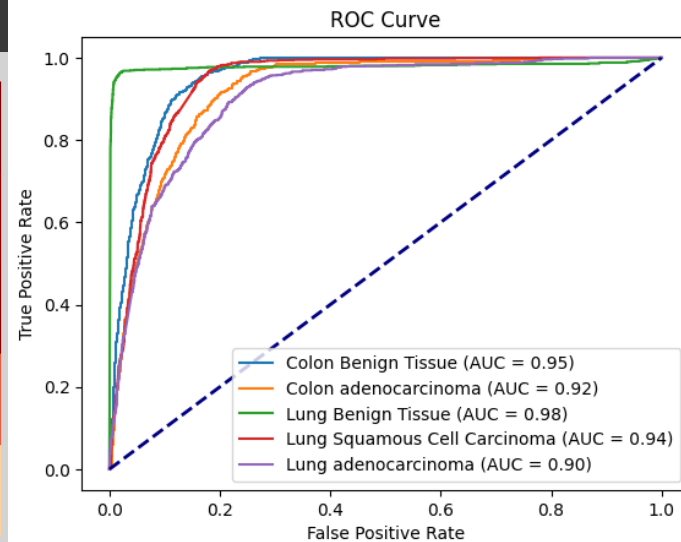
ROC Curve



Results on Test set:

Classification Report:				
	precision	recall	f1-score	support
Colon Benign Tissue	0.84	0.34	0.48	1002
Colon adenocarcinoma	0.56	0.89	0.68	1000
Lung Benign Tissue	0.97	0.93	0.95	1008
Lung Squamous Cell Carcinoma	0.71	0.67	0.69	1008
Lung adenocarcinoma	0.62	0.69	0.65	1002
accuracy			0.70	5020
macro avg	0.74	0.70	0.69	5020
weighted avg	0.74	0.70	0.69	5020

Confusion Matrix					
True Labels	Colon Benign Tissue	Colon adenocarcinoma	Lung Benign Tissue	Lung Squamous Cell Carcinoma	Lung adenocarcinoma
	337	660	5	0	0
	47	890	9	4	50
	14	20	942	1	31
	0	1	0	674	333
	2	29	18	264	689
Predicted Labels					
	Colon Benign Tissue	Colon adenocarcinoma	Lung Benign Tissue	Lung Squamous Cell Carcinoma	Lung adenocarcinoma



4. Improved (Fine-tuned) Model and Its Results

As it can be seen in the results of the baseline the performance of the network improves on the training set continuously. However, the accuracy on the validation set does not improve and even decreases with the amount of epochs. This phenomenon can be seen in many Machine learning models and is caused though the problem of overfitting. Overfitting arises when a model is excessively trained on its training data, leading to a lack of generalization to unseen data. Consequently, the model struggles to make accurate predictions or draw meaningful conclusions beyond the training dataset.

A significant amount of research has been conducted to address overfitting in neural networks. The optimal set of hyperparameters play a crucial role in this context as they directly impact the model's performance. For the utilized convolutional neural networks, the key hyperparameters are briefly explained and implemented in an enhanced model for the present dataset in the following.

One of the most essential hyperparameters is the number of hidden layers between the input and output layers. Consequently, the number of neurons in each layer is also a hyperparameter and determines the structure of the convolutional neural network (CNN). As a general rule, the number of neurons and the number of hidden

layers are increased until the test error no longer improves (Munir, 2023). On the one hand, a small number of layers can lead to underfitting, while on the other hand, too large a number can lead to overfitting, which is why we need to apply appropriate regularization.

A widely employed technique for regularization is dropout. Since each individual neuron has an effect of neurons in the following layers, large networks can become too specialized in a particular type of input data and lead to overfitting. To overcome this issue it is possible to remove a random number of neurons and ignore them in each iteration. That way the neurons do not contribute to the neurons in the next layer and reduces the chance of becoming overspecialized. Literature has shown this in various applications advocating the dropout at 20 - 50% (Moroney, 2020). For the dataset at hand a dropout rate of 20% yielded the highest accuracy.

Another technique to address overfitting is the idea of L2 regularization that penalizes the error function. This leads to reduction in the weight values of the hidden units and thus simplifies the model (Elgendy, 2020). After experimenting with various lambda values, the value “0.001” in the convolutional layers and “0.01” in the Dense layers yielded to the highest accuracy on validation and test set. Furthermore, you can choose different optimizers that change the attributes of your networks such as weights and learning rate. Research has shown that there is no single optimizer that dominates across all domains. However, the optimizer “Adam” is a very popular choice together with “RMSProp” (Schmidt, 2021). After a thorough investigation on the accuracy of the CNN the optimizer “RMSProp” was selected.

As a final hyperparameter in the present network the number of epochs was changed to 15. The number of epochs is the number of times the complete training data is presented during training. Usually, increasing the number of epochs improves the accuracy of a CNN (Munir, 2023). A developer can choose to increase the number of epochs until the validation accuracy starts decreasing.

Below you can see the performance of the trained network on the test set. It yielded an improved accuracy of 0.9422.

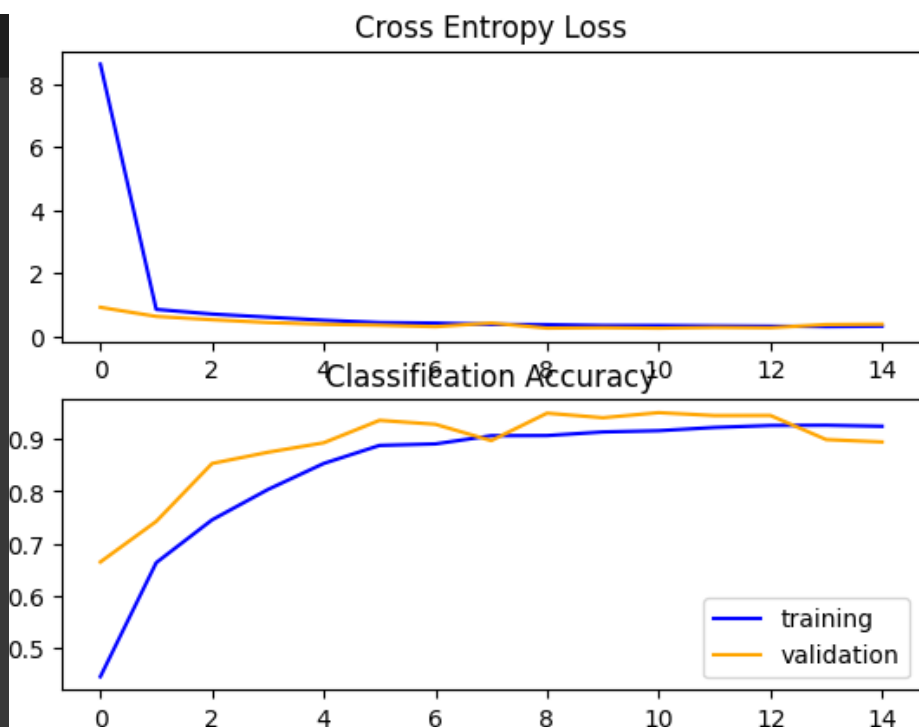
Results on the test set:

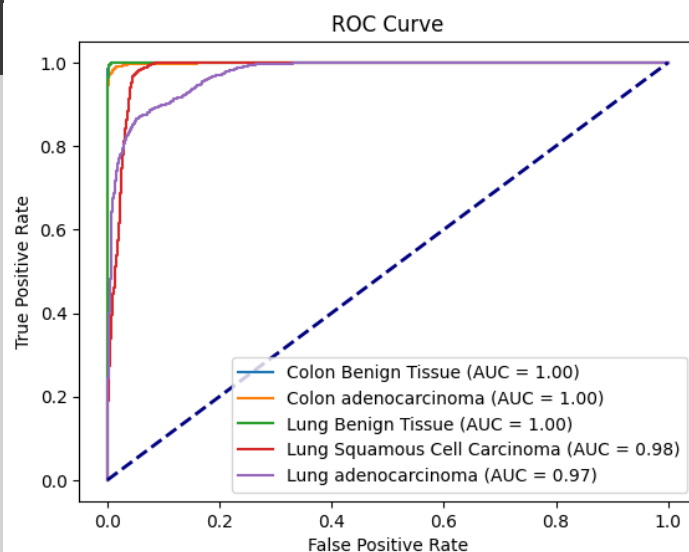
```
model_enhanced = build_enhanced_model()
model_enhanced.summary()
```

Model: "sequential_2"

Layer (type)	Output Shape	Param #
conv2d_6 (Conv2D)	(None, 120, 120, 256)	7168
conv2d_7 (Conv2D)	(None, 120, 120, 128)	295040
max_pooling2d_4 (MaxPooling2D)	(None, 60, 60, 128)	0
dropout_3 (Dropout)	(None, 60, 60, 128)	0
conv2d_8 (Conv2D)	(None, 60, 60, 64)	73792
conv2d_9 (Conv2D)	(None, 60, 60, 64)	36928
max_pooling2d_5 (MaxPooling2D)	(None, 30, 30, 64)	0
dropout_4 (Dropout)	(None, 30, 30, 64)	0
flatten_3 (Flatten)	(None, 57600)	0
dense_9 (Dense)	(None, 128)	7372928
dropout_5 (Dropout)	(None, 128)	0
dense_10 (Dense)	(None, 32)	4128
dense_11 (Dense)	(None, 5)	165

=====
Total params: 7790149 (29.72 MB)
Trainable params: 7790149 (29.72 MB)
Non-trainable params: 0 (0.00 Byte)

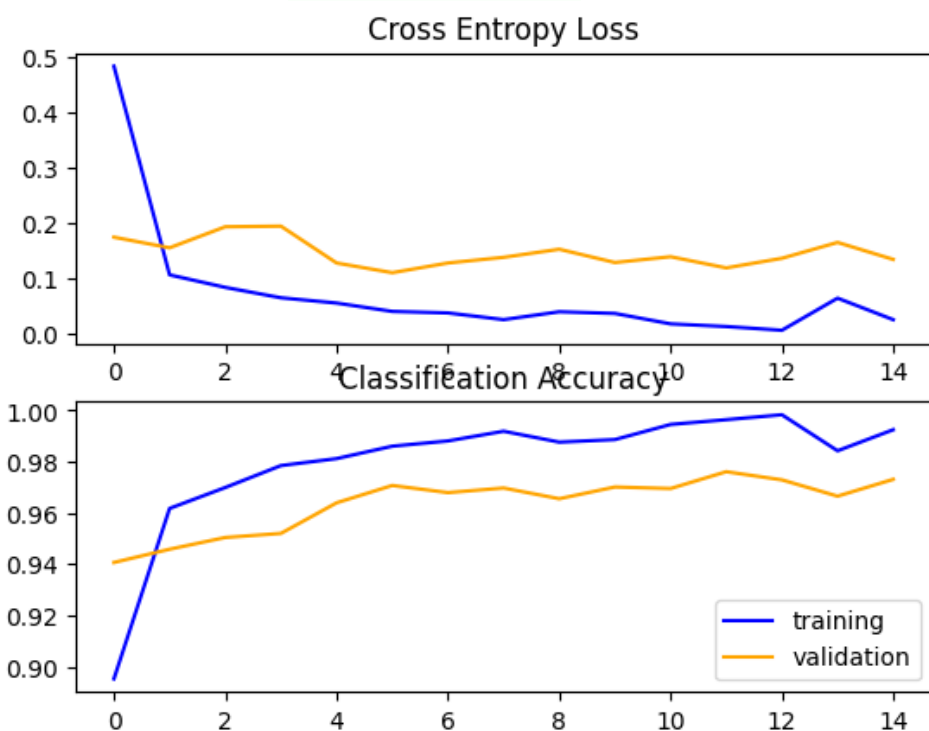




5. Transfer Learning Model and Its Results

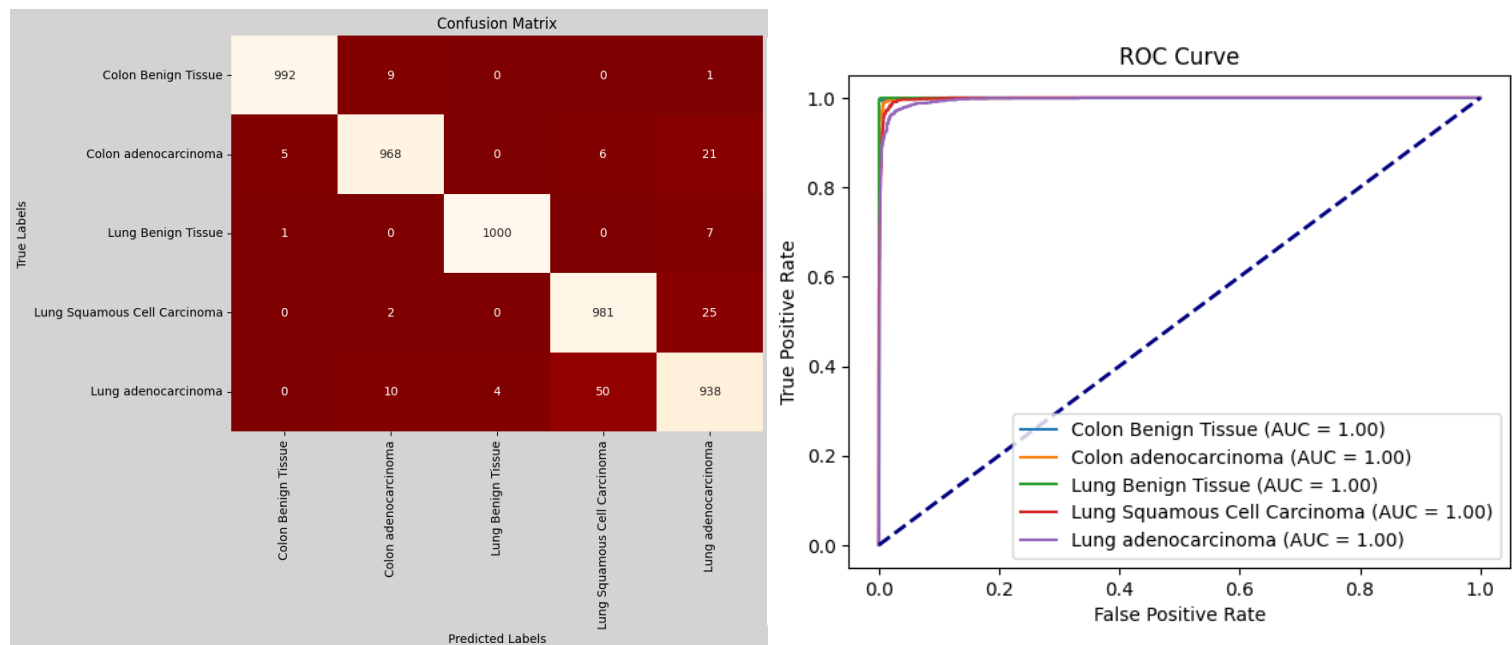
The VGG16 CNN is used in the applied transfer learning model. It is considered as one of the best computer vision models to date. In order to compare it to a similar structure, the same fully connected layers as in the baseline model have been implemented in the transfer learning model, outperforming it substantially. It even yields with an accuracy of 0.9719 a higher accuracy than the enhanced model.

Results on the test set:



Classification Report:

	precision	recall	f1-score	support
Colon Benign Tissue	0.99	0.99	0.99	1002
Colon adenocarcinoma	0.98	0.97	0.97	1000
Lung Benign Tissue	1.00	0.99	0.99	1008
Lung Squamous Cell Carcinoma	0.95	0.97	0.96	1008
Lung adenocarcinoma	0.95	0.94	0.94	1002
accuracy			0.97	5020
macro avg	0.97	0.97	0.97	5020
weighted avg	0.97	0.97	0.97	5020



6. Discussion

In the given visualization it is clearly visible that the transfer learning model has the highest accuracy, followed by the enhanced model and the baseline model with the lowest accuracy. A notable observation in the training and validation accuracy of the base model is the inverse relationship between the two; as training accuracy increases, validation accuracy decreases—an unmistakable symptom of overfitting, a concern not addressed in this particular model. In contrast, the enhanced model exhibits improved generalization, evident in the consistency between training and validation accuracy across epochs. However, a nuanced analysis of the confusion matrix reveals a notable drawback in the enhanced model: it tends to misclassify a significant portion of "lung adenocarcinomas" as "squamous cell carcinomas of the lung." On the other hand, the transfer learning model demonstrates superior performance by avoiding this issue, emerging as the most accurate model among the three for this dataset.

Beyond the discussed enhancements to the network, there exist alternative strategies for refining a CNN. A widely embraced approach involves data augmentation, wherein synthetic data is created and integrated into the training set. This involves operations such as mirroring, rotation, or cropping of image content. Consequently, additional training data is generated, fostering a more resilient model against data noise while enhancing its ability to discern authentic image variations and patterns (Moons, 2018).

In addition, one could also perform a grid search over all possible hyperparameters or introduce even more hidden layers and regularizations, which would make the network even more complex. However, this was not feasible due to the computing resources for this task.

A very promising trend is the implementation of 3D CNNs. Especially in the medical field, 3D CNNs extend the convolution to the temporal dimension and thus enable a more comprehensive understanding of the data. The additional depth information also helps to improve performance, especially for complex structures (Ramya Laxmi, 2020). However, as with any possible structure of the network, which structure is suitable always depends on the characteristics of the problem at hand. Therefore, we need to find a balance between the complexity of the model and the computational resources.

7. References

- Elgendy, M. (2020). *Deep learning for vision systems*. Manning.
- Kumar, S., Rani, S., & Laxmi, K. R. (2020). *Artificial intelligence and machine learning in 2d/3d medical image processing* (R. Raja, Ed.). CRC Press.
doi: 10.1201/9780429354526
- Moons, B., Bankman, D., & Verhelst, M. (2018). *Embedded deep learning: Algorithms, architectures and circuits for always-on neural network processing*. Springer International Publishing.
- Moroney, L. (2020). *Ai and machine learning for coders*. O'Reilly Media.
- Munir, A., Kong, J., & Qureshi, M. A. (2023). *Accelerators for convolutional neural networks*. Wiley. doi: 10.1002/9781394171910
- Schmidt, R. M., Schneider, F., & Hennig, P. (2021, July). *Descending through a crowded valley - benchmarking deep learning optimizers*. In Proceedings of 38th international conference on machine learning (icml) (Vol. 139, pp. 9367–9376). PMLR.