Multiclass prediction of tumors with Convolutional Neural Networks (CNN)

[redacted]

Dominik Duc Duy Nguyen - Transfer Learning, Hyperparameter Optimization, Baseline Section

Baseline results

The baseline model shows potential signs of overfitting, as the training loss decreases while the validation loss remains constant. Additionally, the training accuracy increases with each epoch, while the validation accuracy stays unchanged. This indicates the model memorizing training instances, leading to poor generalization of the validation set. The baseline model had an *F1-score of 0.818* on the validation set**,** whereas on the test set an *F1-score of 0.757*. Notably, the meningioma class followed by the glioma class was more difficult to distinguish for the model than the no-tumor and pituitary classes (Table 2: F1-score: 0.75, 0.55, 0.85, 0.85 respectively). The model accuracy was *0.82* for the validation set, and *0.77*.for the test set.

Figure 1 | *ROC curve validation set*

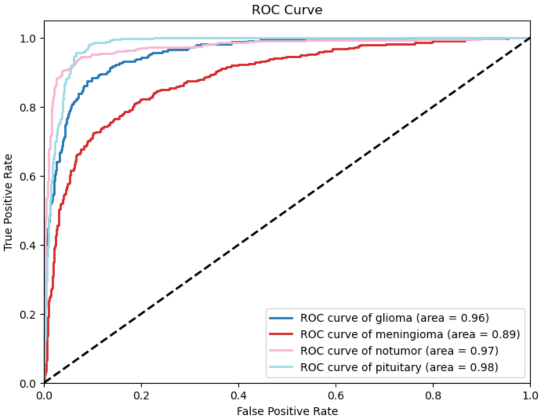


Figure 2 | *ROC curve test set*

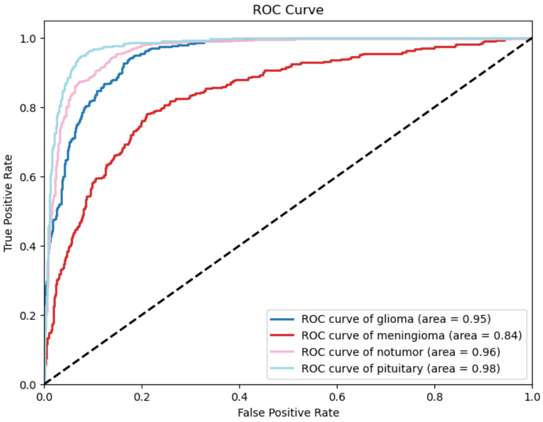


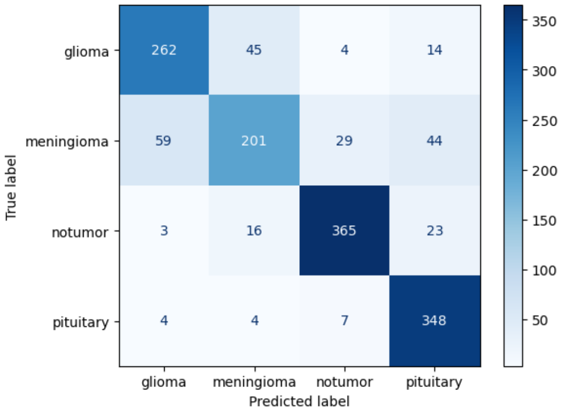
Table 1 | *Validation set metrics*

	Precision	**Recall**	**F1-score**
Glioma	0.80	0.81	0.80
Meningioma	0.76	0.60	0.67
No-Tumor	0.90	0.90	0.90
Pituitary	0.81	0.96	0.88

Table 2 | *Test set metrics*

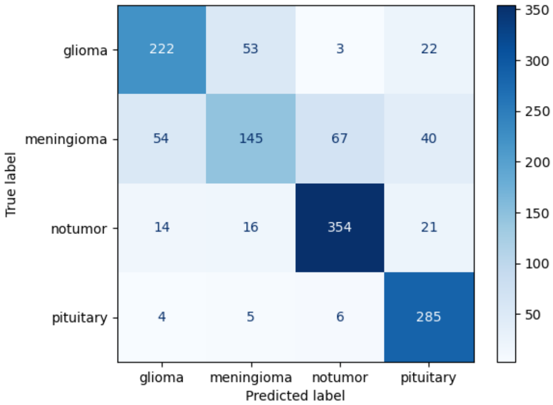
	Precision	**Recall**	**F1-score**
Glioma	0.80	0.81	0.80
Meningioma	0.76	0.60	0.67
No-Tumor	0.90	0.90	0.90
Pituitary	0.81	0.96	0.88

Figure 3 | *Confusion matrix validation set*



	glioma	meningioma	notumor	pituitary
glioma	262	45	4	14
meningioma	59	201	29	44
notumor	3	16	365	23
pituitary	4	4	7	348

Figure 4 | *Confusion matrix test set*



	glioma	meningioma	notumor	pituitary
glioma	222	53	3	22
meningioma	54	145	67	40
notumor	14	16	354	21
pituitary	4	5	6	285

Figure 5 | *Training and validation Loss & Accuracy*

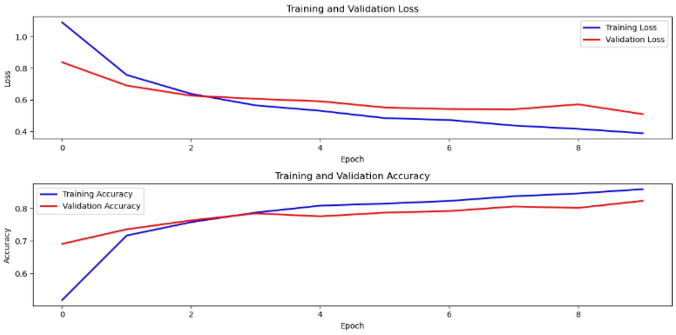
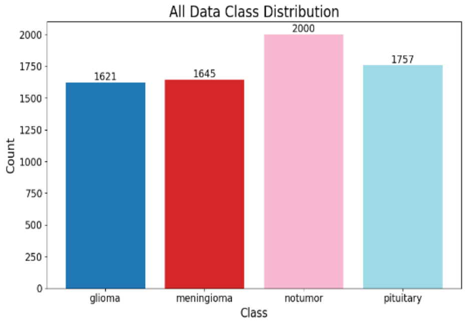


Figure 6 | *Class Distribution*



Enhanced model summary

The enhanced model is a CNN consisting of three convolutional layers, each using ReLU activation along with batch normalization and max-pooling layers to improve feature extraction and stabilize the model. Additionally, the input to each convolutional layer is zero-padded to minimize information loss. The number of filters doubles with each layer (32, 64, and 128, respectively) to progressively capture more complex patterns. After flattening the output from the convolutional layers, the model includes two dense layers with ReLU activation to learn non-linear relationships. The final layer has four nodes with softmax activation for multi-class classification. The model's best performance was achieved using the Adam optimizer with categorical cross-entropy as a loss function, which is suitable for multi-class classification. Increasing the number of epochs to 30, including the use of early stopping, allowed the model to go through more learning steps, in turn, enhancing its performance. The batch size was adjusted to balance training speed and learning stability, with the original batch size of 32 providing the best result.

Justification of implementation choices

The baseline model was improved by tuning its hyperparameters, such as the number of convolutional layers, dense layers, the number of filters, and the implementation of batch normalization. These hyperparameters were selected based on their performance on the validation set and by monitoring for minimal signs of overfitting in the training curves. Evaluating the baseline model, we observed that the training and validation accuracy curves had not yet reached their peak. We increased the number of *training epochs* to improve model performance, which caused the training curves to reach their maximum but also resulted in overfitting. To prevent this, we implemented.*early stopping* based on validation loss, restoring the best model weights if the validation loss increased.

Additionally, the baseline confusion matrix showed poor classification of glioma and meningioma compared to no tumor and pituitary classes.To improve feature learning and class distinction, we *added multiple convolutional and dense layers* and increased the *number of filters*. Unlike the baseline model with 32 filters per layer, we doubled the filters in each layer to extract more abstract features. We used fewer filters in the first layer due to the high signal-to-noise ratio. The number of layers and filters significantly impacted the validation F1-score. We attribute the improved F1-scores to the model's ability to learn more complex features with added layers and filters.

However, this increased complexity often led to overfitting, as seen in the large differences between training and validation accuracy. While the F1-score of the meningioma' class improved, other class scores sometimes decreased. By adjusting the number of layers and filters, we balanced feature learning and overfitting, resulting in the enhanced model's architecture. To minimize information loss due to the small image resolution, we applied *zero-padding* in each convolutional layer, which improved classification performance for all classes and increased the overall validation F1 score by *0.08*. This increase is likely due to the richer feature extraction.

We also added batch normalization layers after each convolutional layer to facilitate training and potentially increase generalizability. While *batch normalization* slightly improved the validation F1-score, the train and validation F1-curves did not converge more. Other methods of increasing the generalizability were explored by adding a *dropout layer*, for which the *dropout chance* was tuned as well as increasing the *batch size*, and adding *L2 regularization*. While these implementations did lead to a smaller difference between the train and validation accuracy training curves, they generally also led to a lower validation F1-score. These changes in F1-score were not class-specific and were observed for all four classes. In the end, we deemed the sole validation F1-score more important than the small difference in the train and validation F1-scores. Therefore, we chose not to implement the mentioned regularization methods. In turn, it is important to note that our current enhanced model could be at risk for overfitting.

Lastly, to further fine-tune the enhanced model, *leaky ReLU* and *Nadam optimizer* were investigated. Leaky ReLU was considered for further enhancement, as Mastromichalakis (2020) mentions that this activation function can lead to small improvements in predicting the right class in comparison to the classic ReLU function. In addition, the Nadam optimizer was considered to help speed up convergence (Saqib et al., 2020). However, both of these hyperparameters did not yield better model performance.

The results achieved a test F1-score of *0.803*, and a validation F1-score of *0.804***,** which is a *0.046* increase in test F1-score compared to the baseline model. Moreover, the enhanced model was able to more accurately classify the meningioma cases compared to the baseline model (test F1-score of 0.55 vs. test F1-score of 0.71, respectively), while also achieving small increases in test F1-score for the no-tumor and pituitary classes (Table 4). Interestingly, the classification of the Glioma cases was not improved by the enhanced model compared to the baseline model, suggesting model performance could be further improved with more hyperparameter tuning. From the training curves of the enhanced model, we can see that the training and validation loss & accuracy reached a maximum during training (Figure 12). Moreover, the difference in the validation and training accuracy curves, suggests that the model is slightly overfitting on the training data. The model showed an overall accuracy of *0.8* on both the validation and test set.

Figure 7 | *ROC curve validation set*

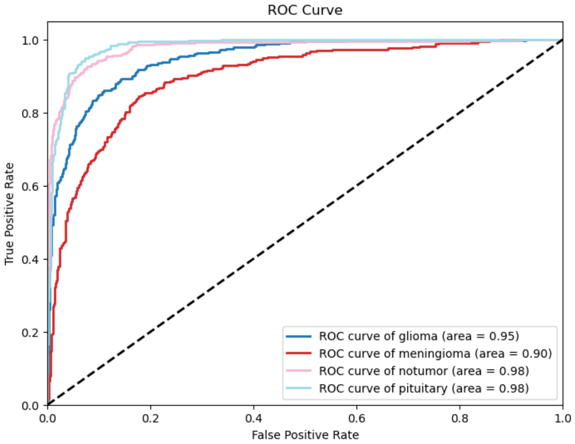


Figure 8 | ROC curve test set

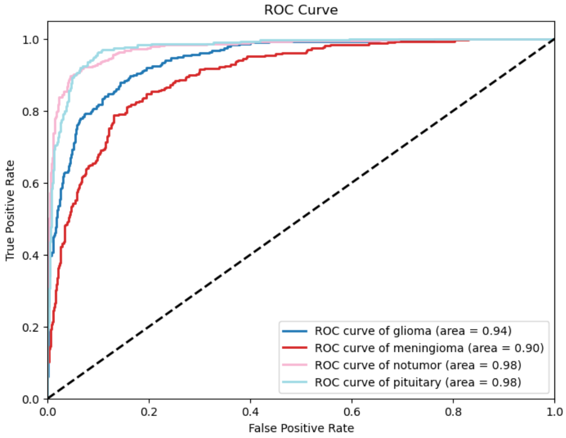


Table 3 | Enhanced validation set metrics

	Precision	**Recall**	**F1-score**
Glioma	0.84	0.65	0.73
Meningioma	0.64	0.74	0.69
No-Tumor	0.87	0.88	0.88
Pituitary	0.87	0.91	0.89

Table 4 | Enhanced test set metrics

	Precision	**Recall**	**F1-score**
Glioma	0.86	0.61	0.72
Meningioma	0.64	0.78	0.71
No-Tumor	0.89	0.89	0.89
Pituitary	0.84	0.90	0.87

Figure 10 | *Confusion matrix validation set*

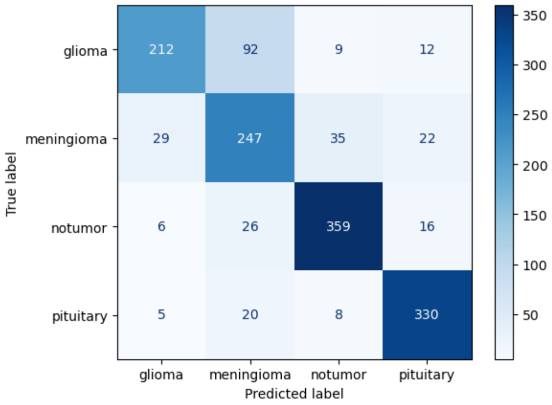


Figure 11 | *Confusion matrix test set*

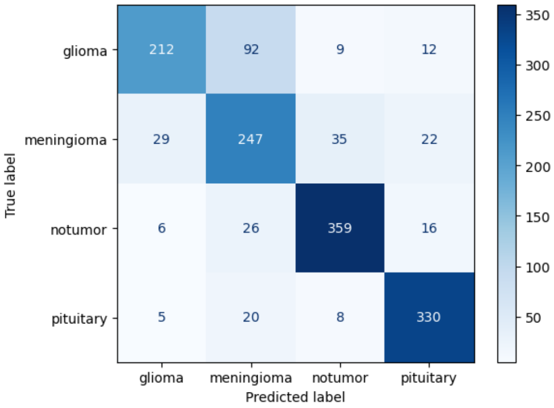


Figure 12 | *Training and validation Loss & Accuracy*

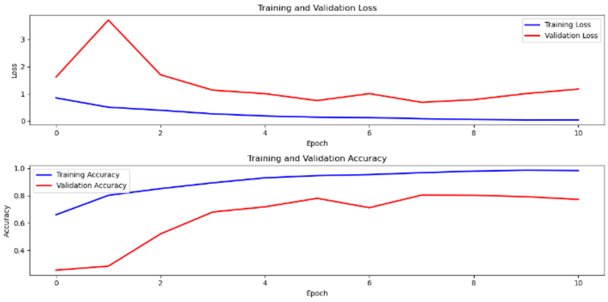
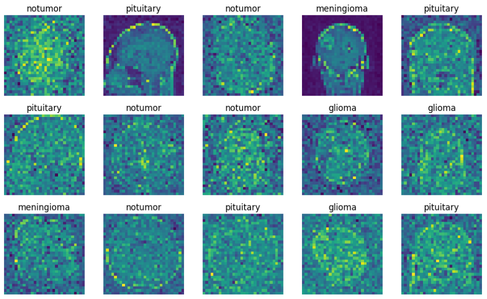


Figure 13 | *15 Random training samples*



Discussion of possible further improvements

This paper aimed to find optimal configurations given the specified tensor size for the input of the CNN. Better learning trajectories are probably possible with a different input size. This paper used downscaled images with an image size of 30x30 as shown in Figure 10. Downscaling, however, can lead to a decrease in predictive performance by a loss of information in the process. In addition, Richter et al. (2021) point out that image size is indicative of the object of interest, in turn, this can depreciate how well an algorithm can differentiate the object of interest. While image size might help the model in learning, there is a tradeoff between speed and performance. Next to different input sizes, data augmentation methods, i.e. creating alternate images by e.g. flipping, rotations, and noise addition, might assist the CNN in learning better representations of the object of interest (Taylor & Nitschke, 2018). Especially as Taylor and Nitschke (2018) state that such algorithms require larger sets of data to be able to effectively predict. Specifically to the dataset used in this paper, data augmentation might help in reducing the slight imbalance as shown in Figure 6.

The learning process, i.e. the hyperparameter tuning, was mainly done manually. A possible improvement might entail using search methods such as Grid Search (GS) or Random Search (RS). Among these methods, RS has gained popularity due to its strategy of sampling from the parameter space based on specified distributions over each parameter. This approach generally outperforms GS and even more complex methods at a lower computational cost, especially in high-dimensional settings (Andonie et al., 2020). This project could be optimized further and gain a better overall model performance by implementing even larger hyperparameter ranges with methods such as RS and GS, as these methods enhance the likelihood of discovering configurations that lead to superior generalization on unseen data.

Next to pre-processing methods and search methods, future work could investigate the use of more advanced (hybrid) models to further enhance performance. Liang et al. (2018) proposed a model that combined CNN and RNN in the task of blood cell image classification. The combination of CNN and RNN makes it capable of extracting both spatial and temporal features, which significantly raises the classification results of medical image data. Their approach, however, suffers from low speed, where the classification time is 3.8 seconds per image, which may impede practical usage. Other types of hybrid models use for instance a support vector machine as the final layer instead of the normally used softmax function for multi-class classification, which led to a small increase in performance (Kibriya et al., 2021).

While already mentioned that image size is important for predictive performance, it is especially important when making use of transfer learning with models that require larger image resolutions and RGB channels. Apart from that, upscaling the images to higher resolution and changes to RGB format could enhance this further, thereby improving models that tend to capture the underlying features and patterns in the data (Richter et al., 2021). While this line of thought holds up, Richter et al. (2021) state that it is better to accommodate the architecture of the algorithm to the input size of the original image rather than down- and upscaling to fit it to a transfer learning model's preferred input size.

Transfer learning

In this section of the report, we investigate the use of transfer learning by employing three pre-trained architectures VGG16, ResNet50, and DenseNet121 for feature extraction. Since our dataset consists of grayscale images (30x30x1), we addressed the channel discrepancy by converting the single-channel grayscale images to three-channel RGB using TensorFlow's grayscale_to_rgb function. Additionally, we re-sized the images to 90x90 to satisfy the input requirements of these models. After freezing the layers of each pre-trained model up to the fully connected layer, we experimented with various configurations of custom dense layers, employing a manual search for hyperparameter optimization. The metrics we obtained from this approach revealed suboptimal performance, particularly in the meningioma class, which consistently proved to be the most challenging to classify accurately. Notably, precision, recall, and F1-scores were the lowest for this class, both in training and test sets. For instance, the F1-score for meningioma was *0.39* on the test set (Table 4), compared to higher scores for the other classes.

Our analysis suggests that the performance gap between the transfer-learning, baseline, as well as enhanced models, is the significant difference in the resolution and characteristics of the images used for pre-training these models on ImageNet, compared to the images in our dataset. These pre-trained models may be extracting features from our images that are not relevant or informative, possibly identifying irrelevant patterns, leading to incorrect classifications, particularly in the meningioma class. To test this hypothesis, we conducted additional experiments where we allowed certain layers of the pre-trained models to be trainable. This adjustment yielded significantly better results across all evaluation metrics, confirming our assumption. By allowing some of the layers to adapt to our dataset, the models could learn more relevant feature representations, leading to improved performance. This provides strong evidence that freezing all layers might prevent the model from adjusting to the domain-specific features in our data, whereas unfreezing some layers allows the model to learn more useful representations tailored to the dataset.

Figure 14 | *ROC curve validation set*

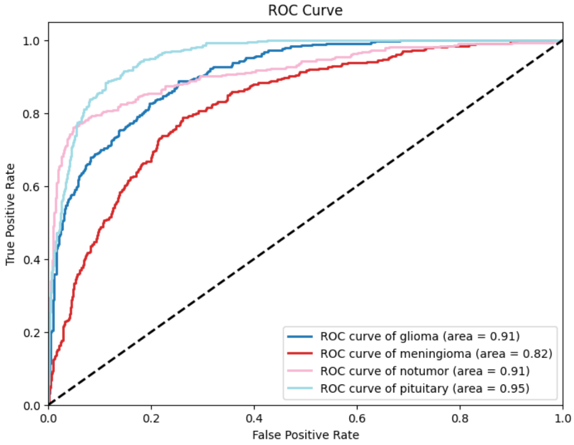


Figure 15 | ROC curve test set

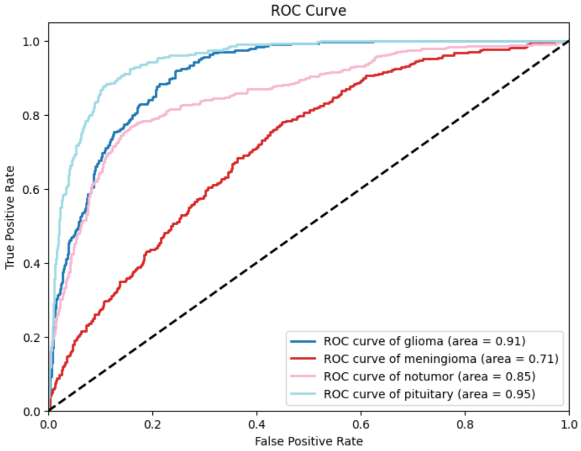


Table 5 | Transfer Learning validation set metrics

	Precision	**Recall**	**F1-score**
Glioma	0.76	0.63	0.69
Meningioma	0.55	0.58	0.57
No-Tumor	0.85	0.75	0.80
Pituitary	0.72	0.89	0.80

Table 6 | Transfer Learning test set metrics

	Precision	**Recall**	**F1-score**
Glioma	0.71	0.55	0.62
Meningioma	0.39	0.39	0.39
No-Tumor	0.72	0.72	0.72
Pituitary	0.71	0.87	0.78

Figure 16 | *Confusion matrix validation set*

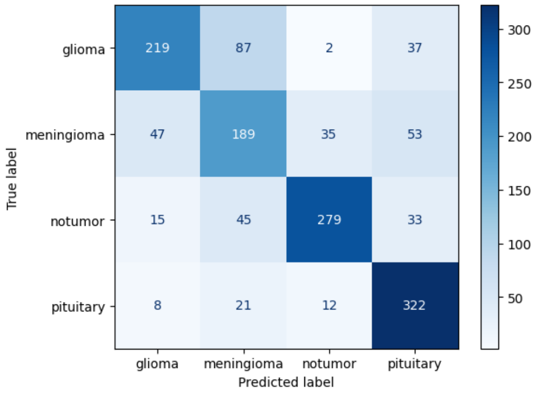


Figure 17 | *Confusion matrix test set*

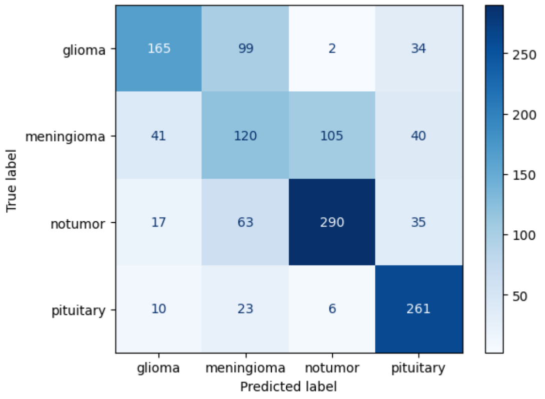
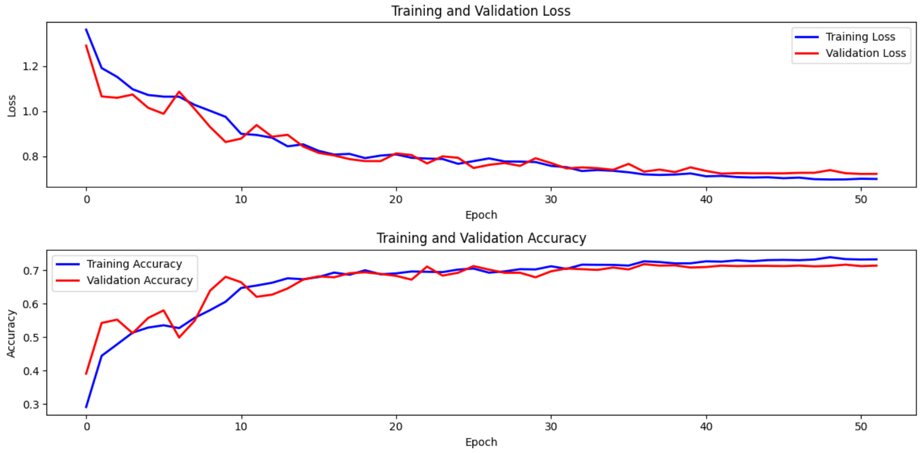


Figure 18 | *Training and validation Loss & Accuracy*



References

Andonie, R., & Florea, A.C. (2020). Weighted Random Search for CNN Hyperparameter Optimization. International Journal of Computers Communications & Control, 15(2), 3868. https://doi.org/10.15837/ijccc.2020.2.3868

Gómez-Guzmán, M. A., Jiménez-Beristaín, L., García-Guerrero, E. E., López-Bonilla, O. R., Tamayo-Perez, U. J., Esqueda-Elizondo, J. J., Palomino-Vizcaino, K., & Inzunza-González, E. (2023). Classifying Brain Tumors on Magnetic Resonance Imaging by Using Convolutional Neural Networks. *Electronics*, *12*(4), 955. https://doi.org/10.3390/electronics12040955

Kibriya, H., Masood, M., Nawaz, M., Rafique, R., & Rehman, S. (2021). Multiclass Brain Tumor Classification Using Convolutional Neural Network and Support Vector Machine. *Mohammad Ali Jinnah University International Conference On Computing (MAJICC)*. https://doi.org/10.1109/majicc53071.2021.9526262

Liang, G., Hong, H., Xie, W., & Zheng, L. (2018). Combining Convolutional Neural Network With Recursive Neural Network for Blood Cell Image Classification. IEEE Access, 6, 36188-36197. https://doi.org/10.1109/ACCESS.2018.2846685

Mastromichalakis, S. (2020). ALReLU: A different approach on Leaky ReLU activation function to improve Neural Networks Performance. *arXiv (Cornell University)*. https://doi.org/10.48550/arxiv.2012.07564

Richter, M. L., Byttner, W., Krumnack, U., Wiedenroth, A., Schallner, L., & Shenk, J. (2021). (Input) Size Matters for CNN Classifiers. In *Lecture notes in computer science* (pp. 133–144). https://doi.org/10.1007/978-3-030-86340-1_11

Saqib, N., Saqib, N., Rafiquzzaman, G. M., & Rafiquzzaman, G. M. (2020). Image Classification using DNN with an Improved Optimizer. *2020 IEEE Region 10 Symposium (TENSYMP)*. https://doi.org/10.1109/tensymp50017.2020.9230585

Taylor, L., & Nitschke, G. (2018). Improving Deep Learning with Generic Data Augmentation. *IEEE Symposium Series On Computational Intelligence (SSCI)*. https://doi.org/10.1109/ssci.2018.8628742