COSC 2673 | Machine Learning

Assignment 2: Cell Image Classification

Reuben Abraham s3717497 (contribution 55%)
Orlando Szulc s3897125 (contribution 45%)

Approach

Identify relationships: EDA Visualization

The following EDA visualizations were done to identify strategies for pre-processing and modeling such that we can avoid overfitting, data leakage, enable transfer learning, and identify features for fine tuning our models.
Bar chart identifying cancerous cell types: All cancerous cells are epithelial (main data only). Further research indicates that this is not representative of the overall data, as eg: inflammatory cells can cause DNA damage leading to cancer. We expect that training on main data will develop isCancerous features that can be reused for epithelial cells as part of **transfer learning.** But there may be **class imbalance** issues with the other cell type, as it's not found in every patient, requiring careful stratified sampling (Figure 2).
Patient cell type distribution: In our main data, our 60 patients are missing fibroblast and other cell types. This can cause an uneven data split unless we use stratified sampling, or correct grouping techniques to get as close to a 60-20-20 split (Figure 3).
Image average of cell types: To understand the *morphology* of nuclei type (shape, size, colour, texture), as described in the original paper. Epithelial cells have a clear *colour difference* to other cell types, but a *less defined* nuclei. When modelling, we can account for RGB data by passing in 3 channels, as opposed to 1 channel described in the original paper CNN, to improve the baseline (Figure 4).

Data Pre-processing

Data split: While the original paper [1.1] recommends 2-fold cross validation, we performed a 60-20-20 split into train, test and validation data due to time constraints and to lend more time to training over epochs.

- *data.head()* reveals that patients share a many-to-one relationship with the cell image patches. To avoid **data leakage** per patient, our splitted data must ensure no overlap between the 98 patients. The solution is to use a *GroupShuffleSplit*, based on each patient
- Avoiding **class imbalance** of the 'other' cell type requires correct stratified sampling.
- We also need to ensure the splits are distributed representative of its dataset. Fortunately, as all cancer cells are also epithelial, we only need to worry about distributing the cell type correctly.
- While we could use *StratifiedGroupKFold*, time constraints led us to using *'random_state'* to identify the closest distribution of train, test and validation data that represents the overall dataset, using the split verification graphs.
- The full dataset is acquired by combining each of the main and extra datasets, after grouping.

Split verification: We verified the splits of our main, full and extra data respectively, with the following graphs:
Cell type distribution: these graphs show the best distribution representative to our whole dataset, individually displaying the full data (Figure 5), the main data (Figure 6) and the extra data (Figure 7) respectively. Furthermore these graphs ensure that the train, test and validation sets have similar distributions, by choosing patients with the required data.
Data split per patient histogram: This histogram was used to verify no patient data overlap between sets. The 3 colours represent the training, test and validation data. It was set to 0.5 transparency, so any discolored bins would indicate overlap. We also manually checked the distributions to ensure it followed the 60-20-20 split. These are displayed across three histograms, displaying the main data (Figure 8), the full data (Figure 9) and the extra data (Figure 10).

Other steps: We **normalized** the image pixel values from 255 to the range [0,1] to aid the convergence and generalization of our models. The *Keras* library now takes in string labels directly for determining a model and generator's categorical accuracy and mode, so we can skip **one-hot encoding** for the cell type.

Augmentation: Based on [1.1] (refer to ipynb file), we applied **Image transformations** (90 degrees, vertical and horizontal flip). We used this with all datasets as it helped the accuracy curve. **HSV perturbations** were also attempted, but experiments showed this further affected the class imbalance of other cell types. Since color is crucial to identifying the epithelial cell type, we avoided using these.

Modelling Approach

Evaluation metrics

Weighted average F1: - Assigning weights proportional to their distribution helps class epithelial cells the best, and as these cells are also isCancerous, the best performing model here may be reused for cancer classification too.
multiclass ROC AUC: To address the *class imbalance* above and simply predicting everything as epithelial or isCancerous, we use this with a *One vs Rest* Approach. This defines the model's ability to discriminate between classes, and thus generalize to an external dataset, where most epithelial cells are not just cancerous.
isCancerous recall (classifying isCancerous only): In the medical field, while it is okay to mispredict normal cells as cancerous, correctly identifying cancerous cells is crucial. Recall is best for this as it check *true positives* accounting for *false negatives*.

For model validation: Using a **confusion matrix** helps with identifying constant prediction cases to ensure the model distinguishes features beyond just epithelial cells and cancerous cells, as they take up a disproportionate amount of data. We also use the **loss/validation accuracy curve** over training and validation data over epochs to identify if our model is overfitting/underfitting, and whether it is predicting erratically over a chosen batch size, as other type cells may not be present in each batch. However it is possible to achieve high accuracy by simply predicting everything as cancerous, which is why it is used in conjunction with the confusion matrix.

Model Selection and Baseline

Most research involving supervised classification with cell images rely on Convolutional Neural Networks, which is why both our models of choice are CNNs devised from the following papers:

- **Softmax CNN** from [1.1] **(Baseline)**: This paper recommends a CNN starting with 2 sets of Convolution+Max Pool layers, and 3 fully connected layers at the end (Figure 25). It *only uses 1 channel,* so RGB information is lost. The final classification output layer uses the softmax activation function, although ReLU is used for previous layers.
- **RCCNet** from [1.2]: This builds upon the original paper's model by adding more complexity. The main difference includes adding an extra convolutional layer before each max pooling layer (Figure 26) so that it can incorporate *all 3 RGB channels*. Batch Normalization for each layer, and a higher dropout of 0.5 enhance the *regularization effect* of this model.

Modelling Task Order: Both the above models were originally used to classify cell types with the output layer. So our modeling order will be as follows:

1. Classify cell type with both models above, using the *main data,* and perform hyper-parameter and model tuning to get its improved versions.
2. Based on our evaluation metrics, we will identify the best generalized model for cancer classification by comparing it with the baseline. We make it compatible by swapping the output layer with a 1 unit sigmoid activation layer, using the *full data.*
3. With the same model above, we now use transfer learning to get the learned features via freezing to predict cell type on *main data,* swapping the output layer back to the 4 unit softmax layer.

Training Process (including model layer/hyperparameter tuning)

Classifying Cell type

Softmax CNN: Our baseline was created following the training details based on Section VI C of the paper (see ipynb file for full list). This includes setting dropout to handle regularization, as well as SGD as the optimizer with custom momentum. The initial accuracy curve fits well but plateaus at 69% due to the model capacity only taking 1 channel.

Adapting the model with all 3 channels as the input hyperparameter causes overfitting, as the validation curve drops past 70% (Figure 14). Replacing the SGD optimizer hyperparameter with ADAM works by reducing the learning rate on plateau, achieving 73% accuracy, and better F1-scores. Judging from the final confusion matrix (Figure 15)The main downfall with this algorithm is the **class imbalance** for other type cells with a **max AUC of 85%** (improved softmax in Figure 33), and the **model complexity** of being unable to handle all 3 RGB channels, which is important for the colour of epithelial cells. This likely is the cause for the **lower weighted F1-score of 83%** in the previous figure.
RCCNet: Opposed to the previous, this model with no modifications avoids convergence as the learning rate is too high, plummeting with batches with other cell types (Figure 18). To combat this, I attempted to change the batch size hyperparameter so it would more likely come across these cell types. An issue with ADAM is highly erratic starting learning rates compared to SGD, so these were also annealed on a plateau. Reducing complexity by removing the first set of convolution and max pooling layers, appears to cap the learning capacity similar to the previous model at 70%. This improved RCC showed convergence past 15 epochs to reach the **highest weighted F1-score of 69% and AUC of 89%**, as it is better generalized to all cells, judging by the final confusion matrix (Figure 24).

Classifying Cancerous cells: Our baseline based on Softmax CNN in Figure 39 shows a starting isCancerous recall of 72%. This is eventually improved on by our model of choice (improved RCCNet) (Figure 38). Our data has a strong cancer to epithelial cell correlation, so a model that performs well on epithelial cells (weighted F1 score with highest proportion) would also perform well on cancerous data. This is supported by the **final Recall rising up to 84%**. Oversampling with a rate of 0.6 of the non-cancerous data actually reduced the recall, likely due to overfitting on repeated data. Unlike cell type classification, all cancer models fit correctly with no clear class imbalance, observing the confusion matrix (Figure 28) with similar FN and FP rates. This supports the **85% F1 score and 95% AUC score.** indicates that our model can classify cancerous cells evenly and identify actual cancer cells while avoiding false negatives. A future improvement is to modify the sigmoid prediction to lean towards predicting cells as cancerous.

Transfer learning
Here, we used the cancer classification model trained on full data and improved RCCNet on the cell type. Freezing layers starting from 2 till 10 helped us identify how high-level the features of cancerous cells can adapt to epithelial cells. The learning curves for each of these indicate a short period of plateauing before adapting to the main dataset. The cross matrices (Figures 33, 35, 37) for the lower frozen layers show overfitting to 2 cell types. We identified that freezing up to the *first 10 layers,* with only the fully connected layers trainable in the RCCNet architecture yields the best results over 30 epochs, providing an **F1-score of 65% and an AUC score of 84%** (Figure 37) This tells us that high-level isCancerous features overlap with the cell type. Looking at the cross matrix for this data reveals that it was able to generalize classification to 3 cells, performing best with epithelial and inflammatory cells. While this does not perform as well as the best RCCNet model we found, it may be better suited to other datasets where epithelial cells are not just cancerous.

Final Judgement with Independent Evaluation

Our best models' evaluation metrics are compared with the 2 papers that provided the patch-based CNN architecture (Figure 38). With better weighted **F1 scores of over 90%** , both papers also succeed in classifying epithelial cells better (Figure 40). [1.1] also does better in discriminating between cell types, compared to our models, aided by Neighbouring Ensemble Prediction (NEP) to build resilience to the variability of nuclei, most present in epithelial cells (Figure 4). However we combat this by using the RCCNet architecture in [1.2] to handle 3 channel RGB input.
The main dataset is biased towards epithelial cells having overlapping features with cancer cells. When applying this data to a real world, our best performing model Improved RCCNet (Figure 38) may have this bias overtrained cancerous epithelial cells, while the transfer model may be better suited to other datasets where epithelial cells are not just cancerous, being trained on extra data and overcoming the overclassification from freezing less layers. Despite this, the cross matrix (Figure 40) shows that it suffers from a more extreme class imbalance issue to the softmax CNN, unable to classify other cells, while our improved RCCNet model can discriminate between all 4 cell types with the highest AUC score.

Thus we recommend the **Improved RCCNet model** with an **AUC score of 89% and a weighted F1-score of 69%.**

Independent Evaluation References

(1.1) Sirinukunwattana, K., Raza, S. E. A., Tsang, Y.-W., Snead, D. R. J., Cree, I. A., & Rajpoot, N. M. (2016). Locality Sensitive Deep Learning for Detection and Classification of Nuclei in Routine Colon Cancer Histology Images. IEEE Transactions on Medical Imaging, 35(5), 1196–1206.
https://doi.org/10.1109/tmi.2016.2525803

(1.2) Shabbeer Basha, S. H., Ghosh, S., Kishan Babu, K., Ram Dubey, S., Pulabaigari, V., & Mukherjee, S. (2018). RCCNet: An Efficient Convolutional Neural Network for Histological Routine Colon Cancer Nuclei Classification. 2018 15th International Conference on Control, Automation, Robotics and Vision (ICARCV).
https://doi.org/10.1109/icarcv.2018.8581147

Dataset	Patches	Labels	Patients
main_data	9896	Cancerous	60
extra_data	10384	Cell Type (4 types)	38

Figure 1: Datasets with pixel sized patches

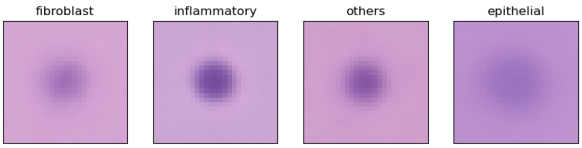


Figure 4: Image average of cell types

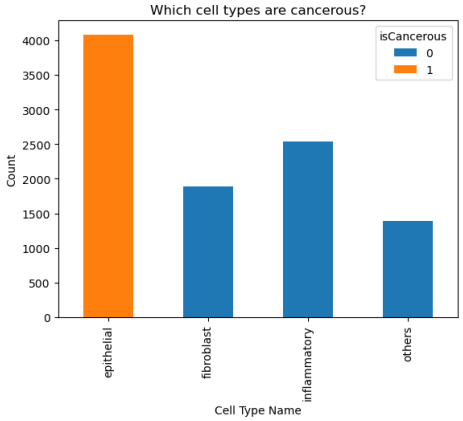


Figure 2: Bar chart identifying cancerous cell types

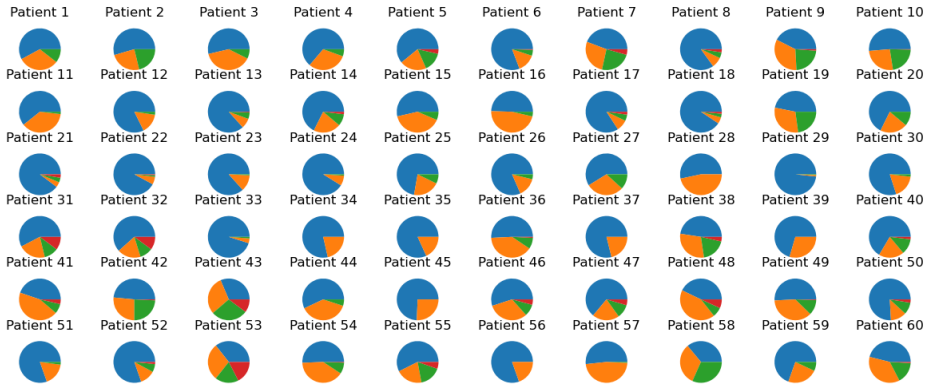


Figure 3: Patient cell type distribution

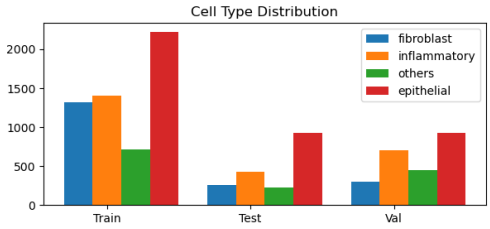


Figure 5: Cell type distribution main data

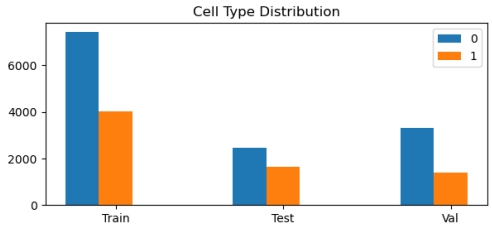


Figure 6: Cell type distribution full data

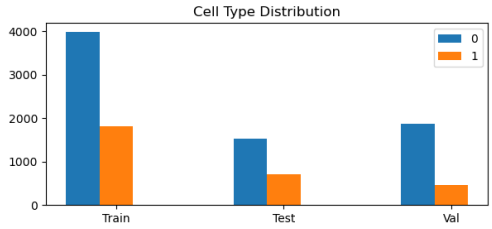


Figure 7: Cell type distribution extra data

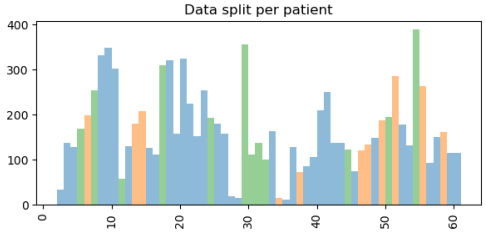


Figure 8: Data split per patient main data

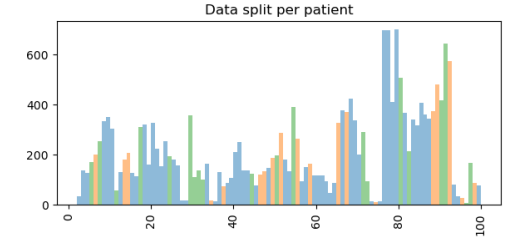


Figure 9: Data split per patient full data

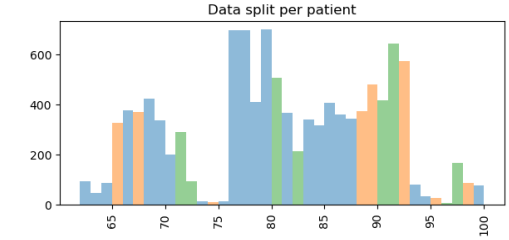


Figure 10: Data split per patient extra data

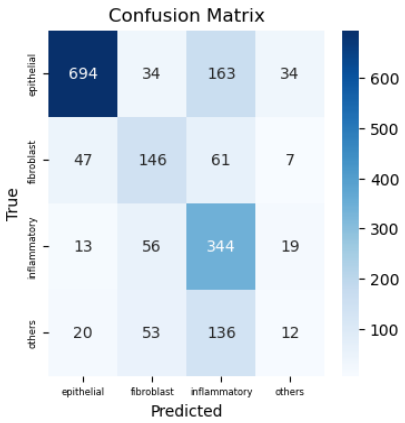


Figure 13: Test cross matrix 1.2

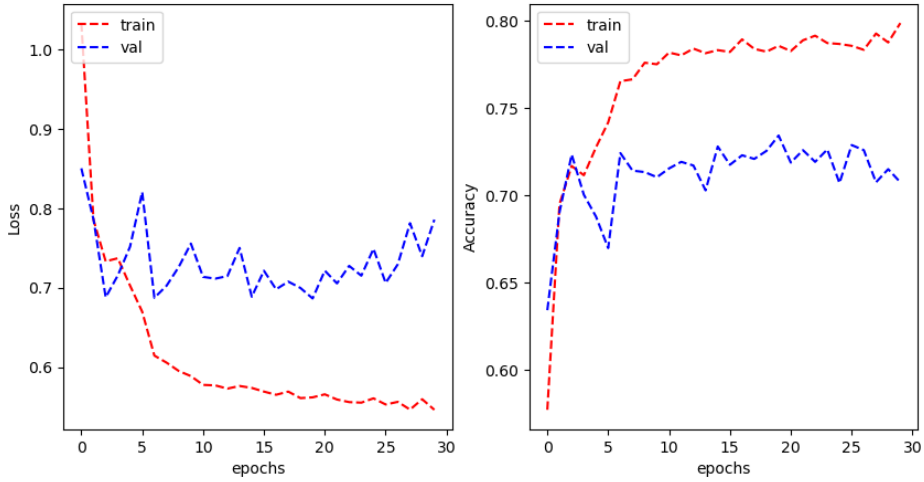


Figure 14: Loss/Accuracy validation curve 1.2

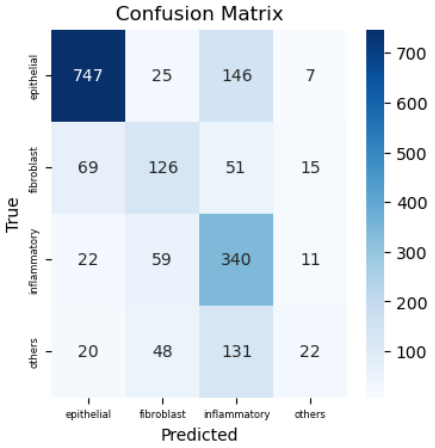


Figure 15: Test cross matrix 1.3

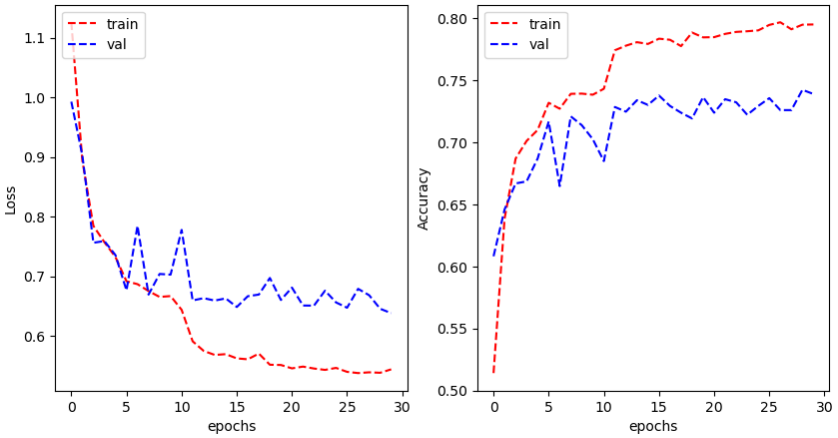


Figure 16: Loss/Accuracy validation curve 1.3

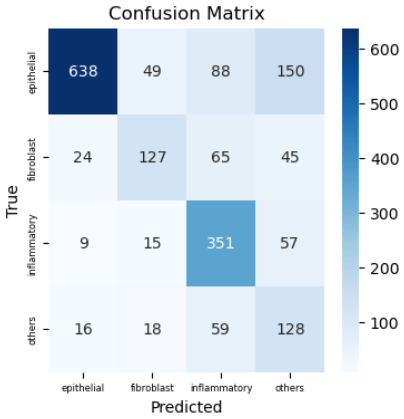


Figure 17: Test cross matrix 2.1

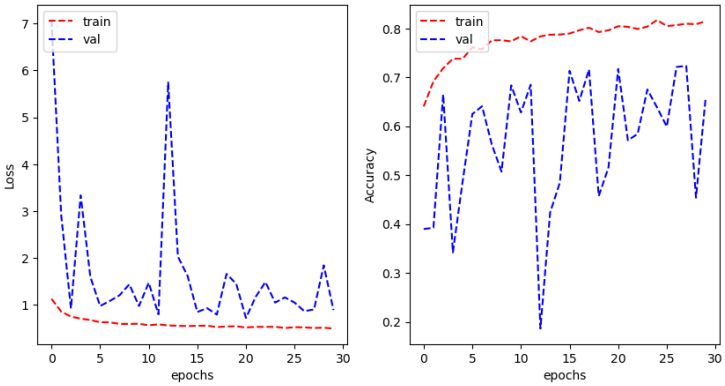


Figure 18: Loss/Accuracy validation curve 2.1

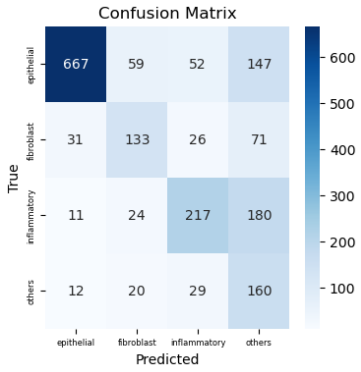


Figure 19: Test cross matrix 2.2

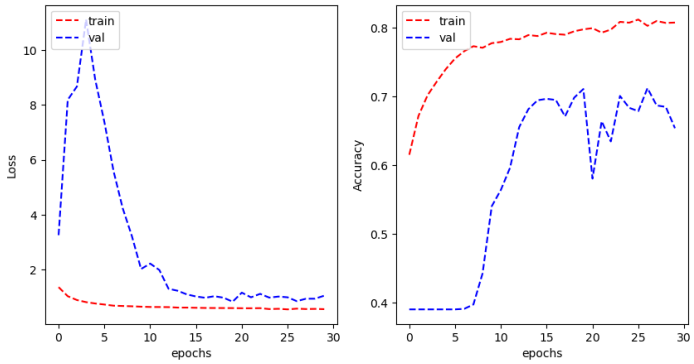


Figure 20: Loss/Accuracy validation curve 2.2

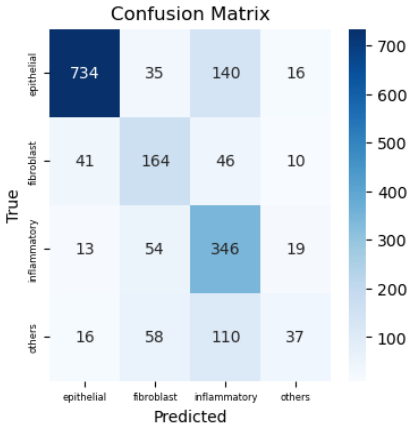


Figure 21: Test cross matrix 2.3

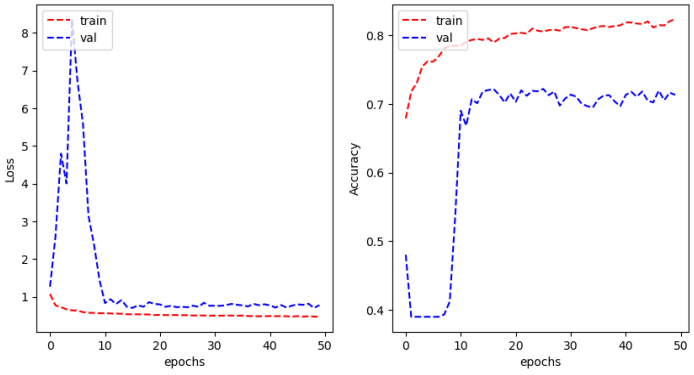


Figure 22: Loss/Accuracy validation curve 2.3

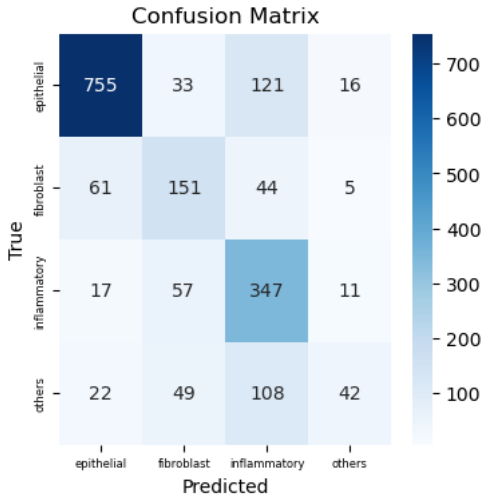


Figure 23: Test cross matrix 2.4

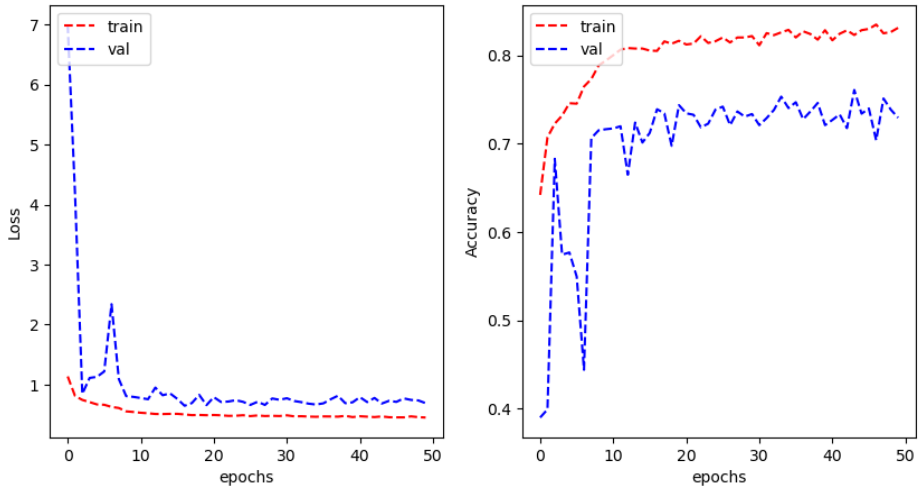


Figure 24: Loss/Accuracy validation curve 2.4

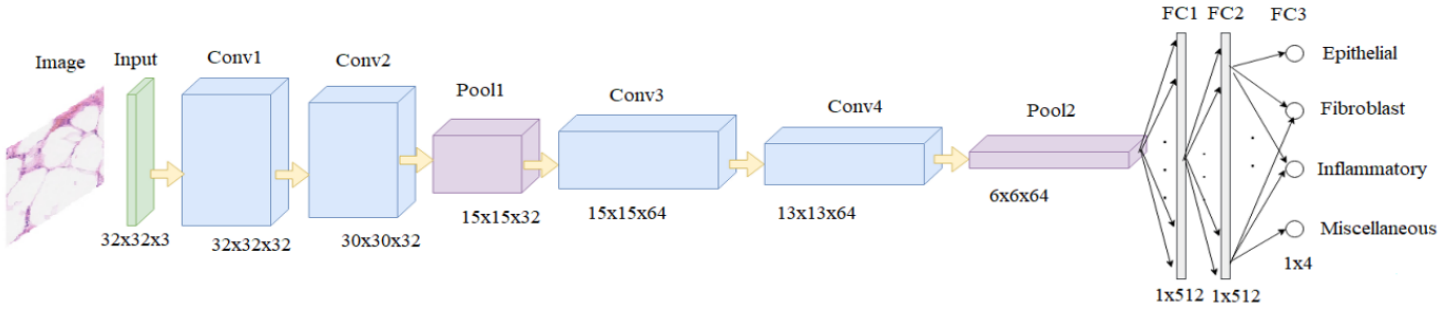


Figure 26: RCCNet architecture

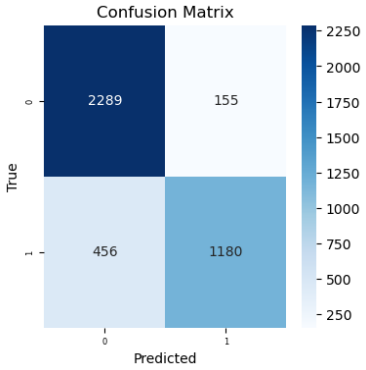


Figure 27: Test cross matrix 3.1

Softmax CNN for classification		
Type	**Filter dimensions**	**I/O dimensions**
Input	4x4x1x36	27x27x1
Convolution	2x2	24x24x36
Max-pooling	3x3x36x48	12x12x36
Convolution	2x2	10x10x48
Max-pooling	5x5x36x48	5x5x48
Fully-connected	2x2	1x512
Fully-connected	5x5x48x512	1x512
Fully-connected	1x1x512x512	1x4

Figure 25: Softmax CNN architecture

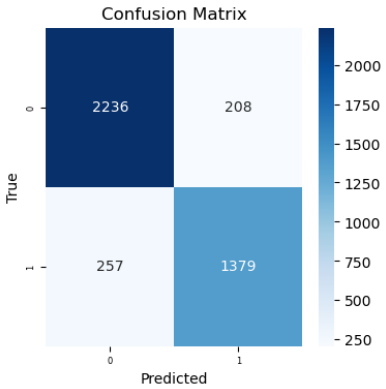


Figure 28: Test cross matrix 3.2

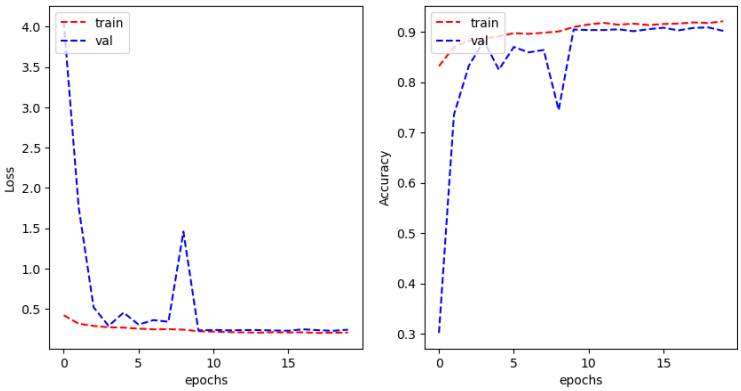
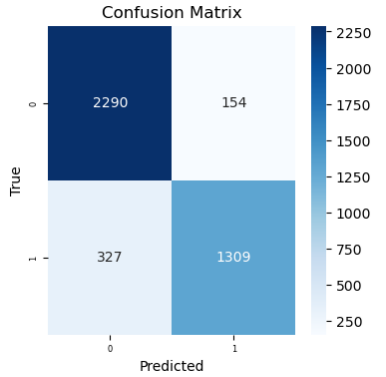
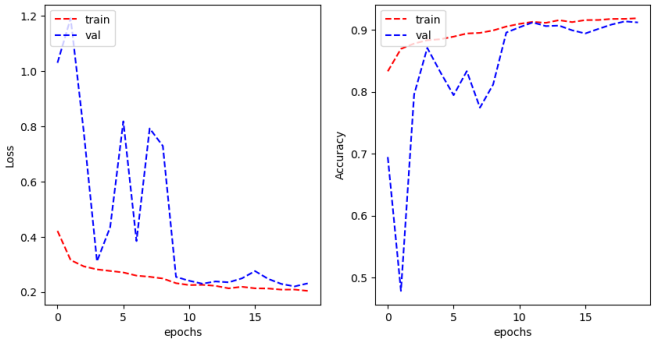


Figure 29: Loss/Accuracy validation curve 3.2





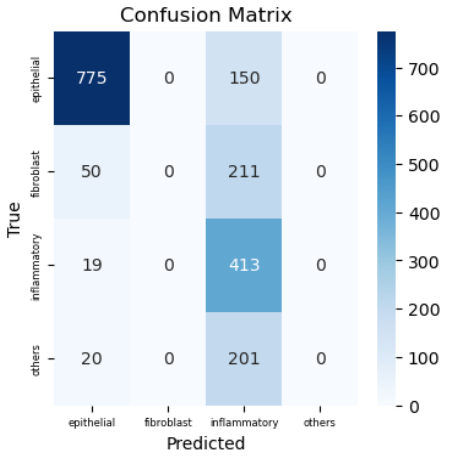


Figure 32: Test cross matrix 4.1

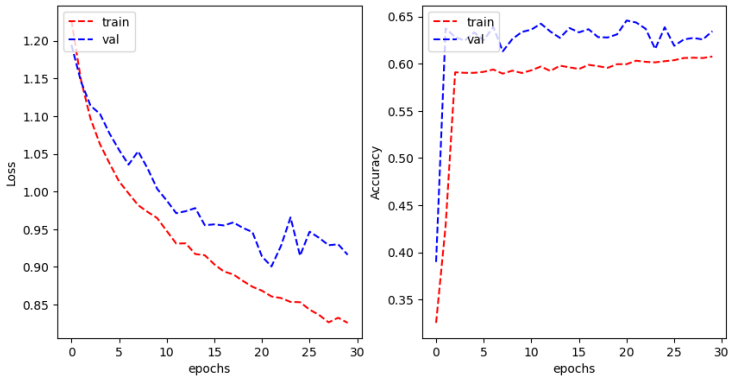


Figure 33: Loss/Accuracy validation curve 4.1

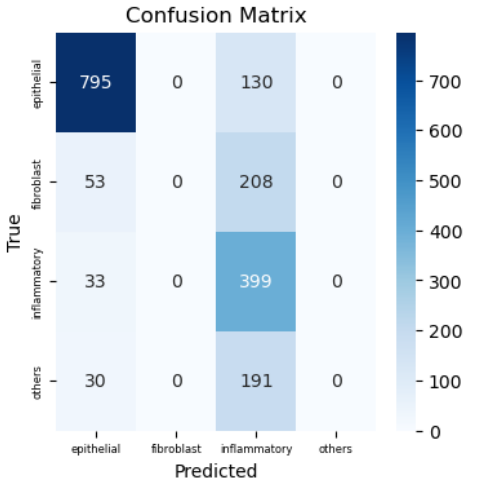


Figure 34: Test cross matrix 4.2

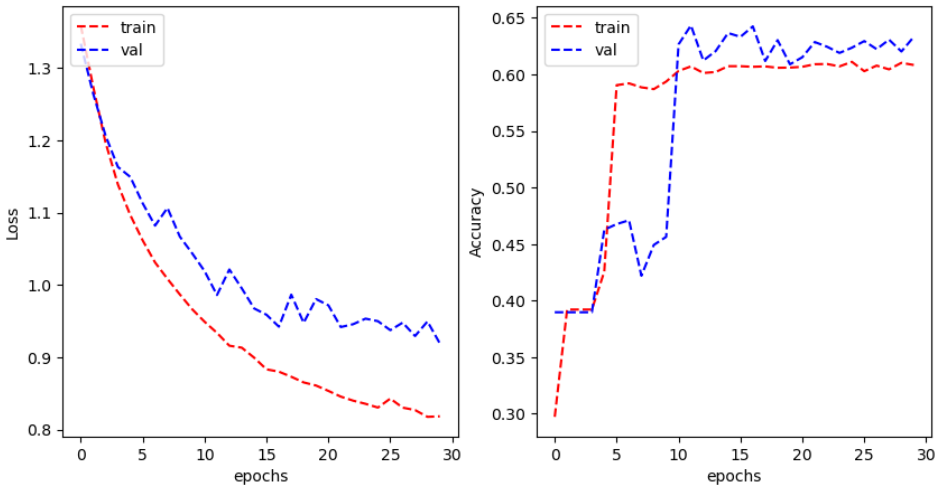


Figure 35: Loss/Accuracy validation curve 4.2

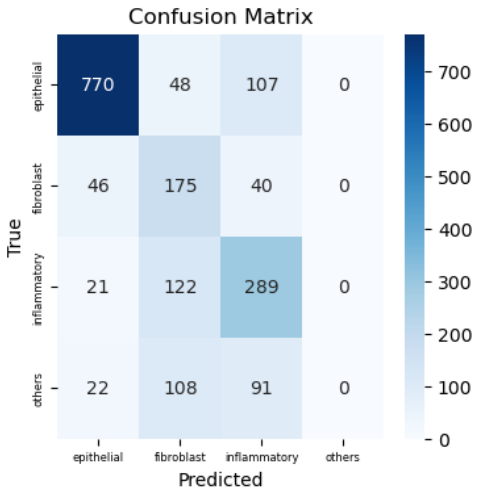


Figure 36: Test cross matrix 4.3

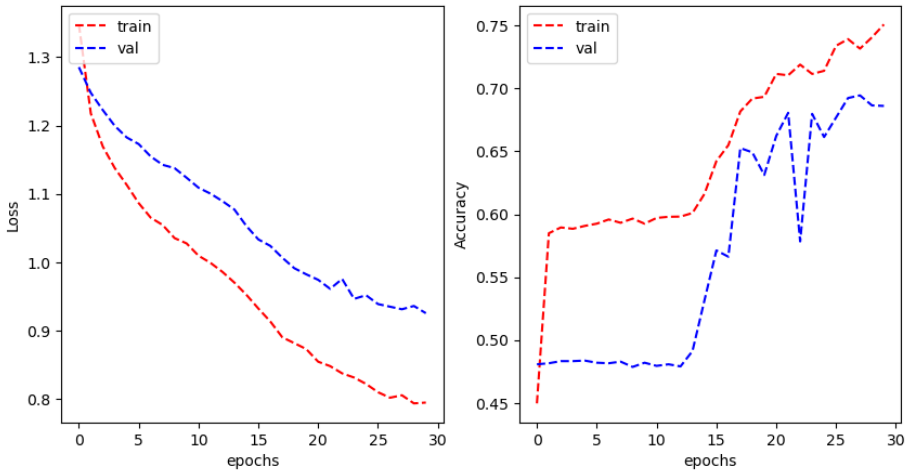


Figure 37: Loss/Accuracy validation curve 4.3

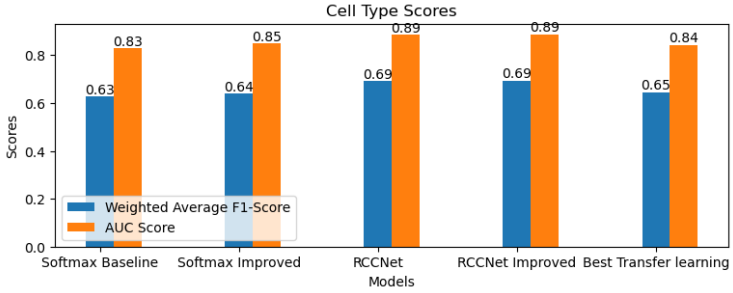


Figure 38: Cell type scores bar chat

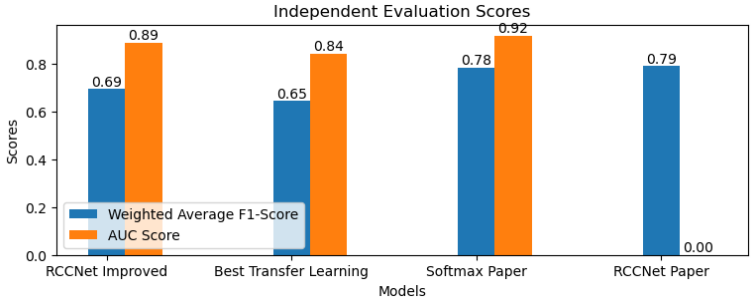


Figure 38: Independent evaluation scores

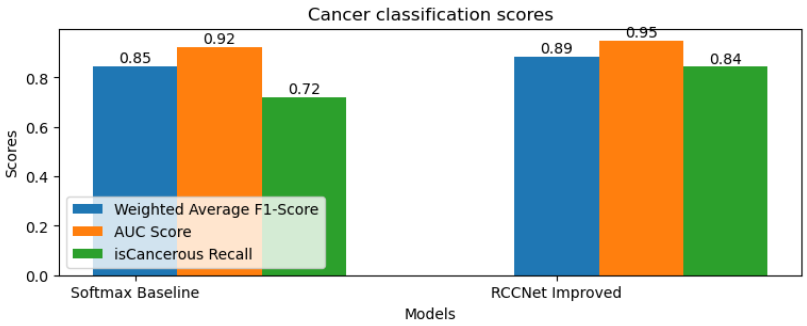


Figure 39: Cancerous classification scores

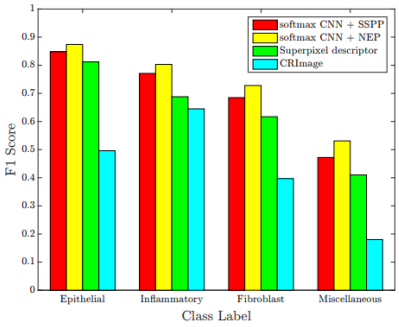


Figure 40: Cell type F1-scores based on the original paper