COSC 2673/2110 Assignment 2

Image Classification Project – Classifying Colon Cancer Cells

Tanvir Azhar (s3797901) Nathan Boc (s3717205)

I. INTRODUCTION

For assignment two, we chose to develop a machine learning system to classify histopathology images of colon cells, determining whether they were cancerous or not and what type of cell they are. We have started with reviewing the state-of-art machine learning techniques for cell classification and in the later stages, use these inspirations to develop our own CNN models followed by professional methodology.

II. Literature Review

Although convolutional neural networks (CNNs) were first introduced in the 1980s, the popularity of deep learning approaches to machine learning began to proliferate throughout the past several years [1]. They have been found to be particularly useful in the assessment and classification of histopathological imaging of colorectal cancer (CRC) tissue samples. CRC, otherwise known as colon cancer, is the fourth most common cause of cancer-related deaths worldwide [2]. However, the early detection of CRC is critical in improving the chances of a successfully treating the disease. Multiple studies explore various approaches towards developing an automated method of diagnosing and detecting CRC by analyzing microscopic colon biopsy images. In [3], Sirinukunwattana, et al. developed a convolutional neural network in 2016 used to classify images of colorectal adenocarcinoma using a spatially constrained CNN with a Neighboring Ensemble Predictor that allowed the model to be trained to predict the probability of a pixel being the center of a nucleus. This design enabled a locality sensitive model that was able to identify the vicinity of the center of nuclei and proved effective with the highest average F1 score being 0.802. Other works improved on CRC detection using multispectral and hyperspectral images, as seen in [4] and [5], respectively. Both their datasets consisted of images which considered a wider wavelength range, being able to capture more detail and characteristics beyond trichromatic (RGB) imagery. This allowed for developed models to have access to an expanded spectral range of optical information to be used for analysis. [4] also implemented the use of a tissue segmentation technique that resulted in separating non-relevant tissues from CRC tissues in the image data. The resulting models with and without tissues segmentation was 0.9917 and 0.7923, respectively. Other techniques explored, which are described in [6], include transfer learning, data augmentation, feature extraction and adaptive CNNs. One noticeable characteristic of adaptive CNN is that the authors are using three layers of convolution, 2 layers of multilayer perceptron along with convolutional filter size of five and subsampling factor of two. Data augmentation is a common practice in the analysis of histopathological images, which allows for the given image dataset to be extended by adding transposed images. This improves the accuracy for the trained model by allowing it to improve its generalization ability on an 'unseen' dataset. The results of explored works indicate that CNN-based machine learning models are able to achieve excellent classification performances when implemented in cohesion with other techniques such as data augmentation and transfer learning. These models prove to be superior when compared to that of base CNN models. This suggests future works to explore the application of such models on various datasets, including those that comprise of multispectral and hyperspectral images, as well as augmented datasets, both of which increase the amount of information to be used in the training of the model.

II. APPROACH

A. Goals, Performance Measure & Target

The given dataset is a modified version of the "CRCHistoPhenotypes" dataset and consists of 27x27 RGB images of colon cells from 99 different patients. The machine learning tasks in our project are the following:

Task 1: Classify images according to whether the given cell image represents a cancerous cell or not ('isCancerous').

Task 2: Classify images into categories of cell type, which include fibroblast, inflammatory, epithelial and others ('cellType').

Our goal is to build supervised machine learning models to accomplish these two tasks successfully.

The given problem is "Solved". In this case, the performance of the model highly depends on the quality of training dataset [7]. The current State-of-art DL models trained on an extensive training set containing more than 15,000 patients of various cancer types, obtained across 45 countries achieved an excellent performance of AUC greater than 0.98 for cell histology task [7]. Besides that, researchers using deep CNNs so far achieved 94.5% accuracy for cell detection and 82%, 58%, 62%, 92% {Classes - normal, HP, TA_LG, CA} for cell type classification [6].

Another research has stated that, they have achieved F1-Score of 0.784 with multiclass AUC score 0.917 for cell type classification [3]. Based on the literature evidence, we have chosen the following performance metrics for the machine learning tasks:

Task 1:

Performance Metrics	F1-Score (macro)
Target Performance	0.91

Task 2:

Performance Metrics	F1-Score (macro)
Target Performance	0.80

B. Data Exploration & Preparation

In our exploratory data analysis, we have first established that there are no null values in our datasets. After that, we performed extensive analysis to gain insight from the dataset. This includes checking data statistics, target classes, and visualization of the data points. During data exploration, we saw an imbalance in the distribution of datapoints by both 'cellType' and 'isCancerous'. Class imbalance was seen in how most cancerous cells in the dataset were of epithelial cell type, which all non-cancerous cells were made up of the other cell types [Fig.2,3]. Due to this, it is expected that any model trained using this dataset will classify all epithelial cells as cancerous and vice versa, while all other cell types will be classified as non-cancerous and vice versa. Another point to take into account is that although it is normally expected that ID values have no influence over the training of a machine learning model, datapoints in this case can originate from the same patient, adding bias to the data [Fig.4]. This would eventuate towards data leaks between the testing and training datasets if patient ID is not taken into account when performing data splitting. In saying this, a possible solution would be to perform splits while ensuring that datapoints from the same patient ID are kept grouped together. For example, all datapoints from patients 1, 2, 3, 4, 5 can be set aside for training and datapoints from patients 6, 7, 8, 9, 10 can be set aside for testing (given that the number of datapoints within the split data sets are proportioned).

Considering training data quality and the size of the dataset, we have split our dataset into 80:10:10 ratio, means that 80% of the data is reserved for training, 10% for validation and 10% for testing the models. After that we have designed data loader function to normalize and create the necessary datasets for model in later stage of the project. We have shown the use of both an image generation function along with manual data manipulation techniques.

C. Evaluation Framework

Custom diagnostic instrumentation such as a learning curve visualization function, a prediction function and a confusion matrix were setup to capture a number of metrics for classification, enabling us to observe, compare and contrast between different training and validation curves. In addition, we have also used sklearn's metrics library to evaluate different models to match our custom generated f1 score so that there are no discrepancies in our results.

D. Baseline Model Architecture

As was demonstrated in our literature review, the use of CNNs are the current state-of-the-art for a number of histopathology image classification tasks, including cell detection and cell identification. We wanted to explore the full gamut of possible model architecture which are inspired form the literatures. We have mainly focused on Bayesian CNN, VGG (inspired form VGG16) and VGG with transfer learning.

Our baseline model is a simple convolution neural network consisting of only two layers. The model is built using tensorflow keras to add sequential layers to the model. It is then compiled with the 'Adam' algorithm optimiser. It returned a result of 0.90 for validation accuracy and 0.99 for training accuracy. It suffers from large overfitting issues. But, the initial purpose of our base line model is to act as a standalone building block for us to understand the general procedure in developing a convolution neural network. Although the result looked promising, the model needed fine tuning to achieve the target performances.

III. MODELS & ISSUES

We encountered a number of issues which required us to perform hyperparameter tuning, data augmentation in order to get these models learn successfully with a good fit.

A. Bayesian CNN

Initially, we opted for probabilistic implementation of CNNs. In this case the difference between a standard CNN and Bayesian CNN is in their first convolutional layer and last dense layer. For Bayesian CNN, we have included a Convolutional2DReparameterization layer. This layer is to tackle the aleatoric uncertainty which can arise from the dataset. As a result, instead of creating an output from the deterministic value like standard CNN, it will create an output which is drawn from a distribution. We also have defined our prior and posterior with kullback-leibler divergence. The output of our model was poor and it was suffering from a huge overfitting for both of the tasks

[Fig.6]. We concluded that our implementation has some lacking for which the model is failing to produce the target performances. At this point we decided to move forward, since most of the literature stating that using CNN architecture like InceptionV3 will result in great accuracy.

B. VGG

Since InceptionV3 is a large neural network, considering the size of our images (27 x 27) we selected VGG. Our VGG architecture is inspired from the actual VGG16. We have constructed a four-layer VGG and trained our model with the given dataset. In our first phase VGG was suffering from large overfitting. We have tried hyper parameter tuning such reducing learning rate, changing batch size, selecting proper optimizer etc. It did not improve our model's performance. We applied data augmentation to the train images and then trained our model once again. For task 1, our VGG model performed really well and exceeded the target performance with a validation F1 score of 0.93 [Fig.9]. When we tested our model with test set, the F1 score was 0.92, with 364 true positives.
For Task 2, the VGG model didn't perform as well as per our expectation. We have tried implementing it with data augmentation, but the results weren't convincing enough, with the validation F1 score being 0.77. We then wanted to push our model further with the transfer learning technique.

C. VGG with Transfer Learning

We have designed two transfer learning techniques to get the best possible result. We have also utilized the extra data set given to us. After an exploratory data analysis, we have observed that there is no cell type information in that data set. Also, there is slight imbalance in target value 'isCancerous'. The first technique is to train a model with the provided extra dataset to create a base model, before freezing some layers in our new model and passing the weights to train and predict with that model. Our validation F1 score for this first approach was 0.78. The second approach was to create a base model using the extra dataset, then grabbing the weights from the global max pooling layer. We passed these weights to our new model to classify the images. In this case the validation F1 score was 0.74.

TABLE I. F1 SCORES FOR FINAL MODELS

Models	***F1 Score***
VGG (Task 1)	Task 1: 0.92
VGG with Transfer Learning (Task 2)	Approach 1: 0.76

Fig. 1. Result Table

It is worth mentioning that we trained approach 1 with augmented data. We have compared both the aproaches and decided to choose the first approach to evaluate the model [Fig: 5]. Our final best F1 score for task 2 was 0.76.

IV. CONCLUSION & ULTIMATE JUDGEMENT

A. Limitations to the real world application

This project is very close to a real-world application, the only limitation being the dataset since histopathological data is often inconsistent and contained with subjectivity. This is quite unavoidable due to the large number of variations in cell morphology. Interestingly, in a real-world setting, most pathologists depend on lower power architecture to get a main impression before using higher power cellular morphological details to confirm the impression. Solely using high power features alone will increase the misclassification. Besides that, the data collection process is costly and difficult to deal with within a lab environment.

B. Possible Improvements

Unsupervised pretraining can be a possible improvement in this project's settings. Since we have a lot of unlabeled data, we can use it to train an unsupervised model such as autoencoder or a generative adversarial network. We can use the lower layers of the autoencoder or the layers of GAN's discriminator, add the output layer for the cell histology task and then fine tune the final network using supervised learning techniques.

C. Ultimate Judgement

At the end we were able to build three models which are close to the reviewed literature. Though we reached our target performance on task one, task two still needs some improvement. Our results indicate that the model we have constructed, has the potential to reach the F1 score of state-of-the art models. Ultimately, we conclude that further work will help us to improve the model and more variety in our dataset would be required to ensure an end-to-end machine learning system.

REFERENCES

[1] R. Draelos. "The History of Convolutional Neural Networks." Glass Box. https://glassboxmedicine.com/2019/04/13/a-short-history-of-convolutional-neural-networks/ (accessed May. 29, 2021).

[2] B. W. Stewart and C. P. Wild, “World cancer report 2014,” Int. Agency Res. Cancer, Lyon, France, Jan. 2014, vol. 3, no. 1, pp. 392–402.

[3] Korsuk Sirinukunwattana, Shan E Ahmed Raza, Yee-Wah Tsang, David R. J. Snead, Ian A. Cree & Nasir M. Rajpoot, 2016, 'Locality Sensitive Deep Learning for Detection and Classification of Nuclei in Routine Colon Cancer Histology Images'

[4] Hawraa Haj-Hassan, Ahmad Chaddad, Youssef Harkouss, Christian Desrosiers, Matthew Toews, Camel Tanougast, 2017, 'Classifications of Multispectral Colorectal Cancer Tissues Using Convolution Neural Network'

[5] Boris Jansen-Winkeln, Manuel Barberio, Claire Chalopin, Katrin Schierle, Michele Diana, Hannes Köhler, Ines Gockel and Marianne Maktabi, 2021, 'Feedforward Artificial Neural Network-Based Colorectal Cancer Detection Using Hyperspectral Imaging: A Step towards Automatic Optical Biopsy'

[6] Junaid Malik, Serkan Kiranyaz, Suchitra Kunhoth, Turker Ince, Somaya Al-Maadeed, Ridha Hamila, & Moncef Gabbouj, 2019, 'Colorectal cancer diagnosis from histology images: A comparative study'

[7] Chetan L. Srinidhi, Ozan Ciga, & Anne L. Martel, 2020, 'Deep neural network models for computational histopathology: A survey'

[8] Sang-Hyun Kim, Hyun Min Koh, Byoung-Dai Lee, 2021, 'Classification of colorectal cancer in histological images using deep neural networks: an investigation'

[9] Muhammad Shaban, Ruqayya Awan, Muhammad Moazam Fraz, Ayesha Azam, Yee-Wah Tsang, David Snead, and Nasir M. Rajpoot, 2020, 'Context-Aware Convolutional Neural Network for Grading of Colorectal Cancer Histology Images'

[10] Mwenge Mulenga, Sameem Abdul Kareem, Aznul Qalid Md Sabri, Manjeevan Seera, Suresh Govind, Chandramathi Samudi, and Saharuddin Bin Mohamad, 2021, 'Feature Extension of Gut Microbiome Data for Deep Neural Network-Based Colorectal Cancer Classification'

APPENDIX

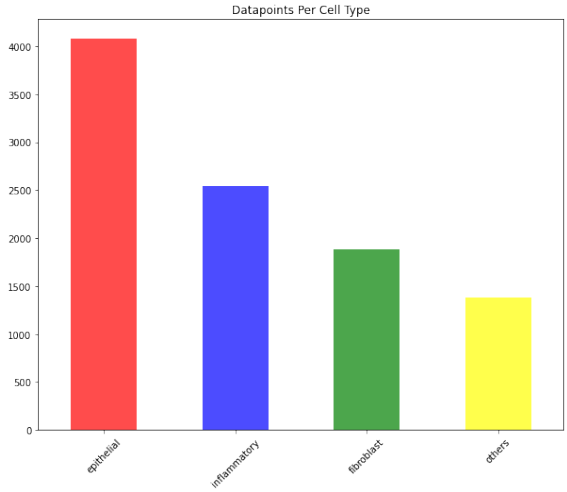

Fig 2: Datapoints per cell type

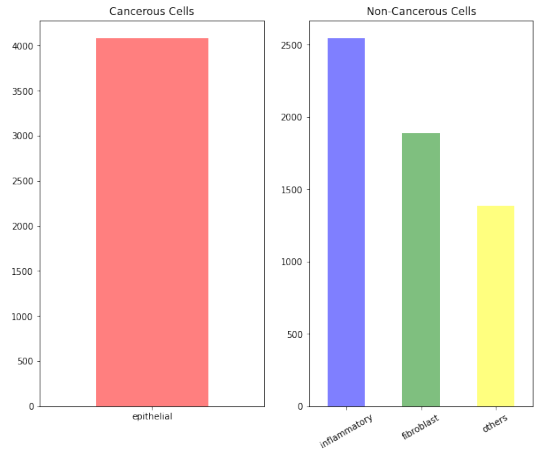

Fig 3: Cell category with cell type

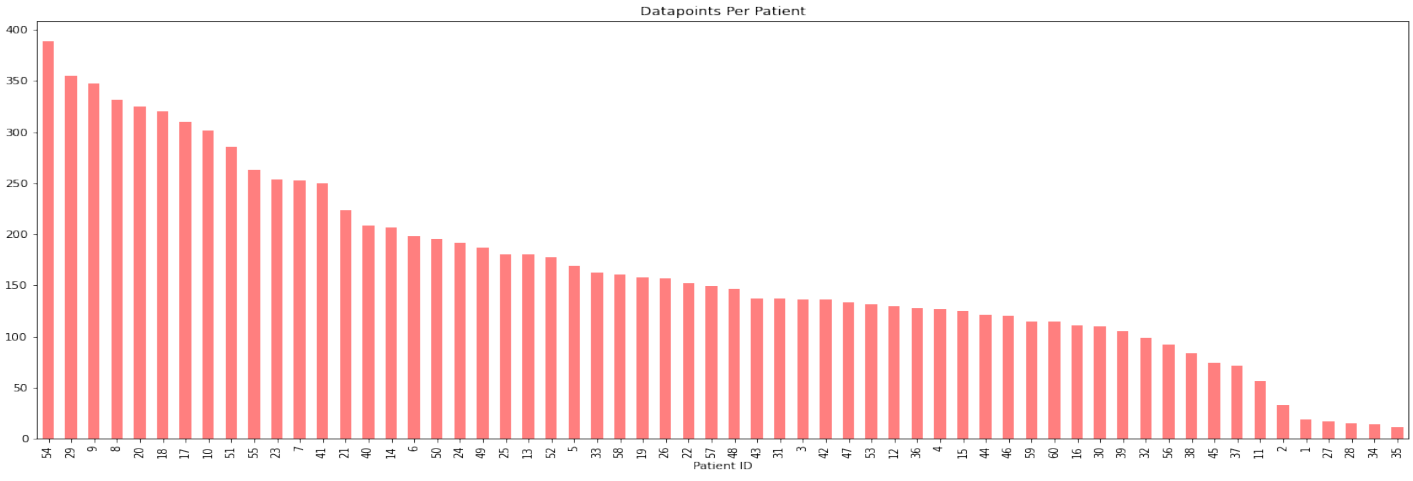

Fig 4: Data Distribution

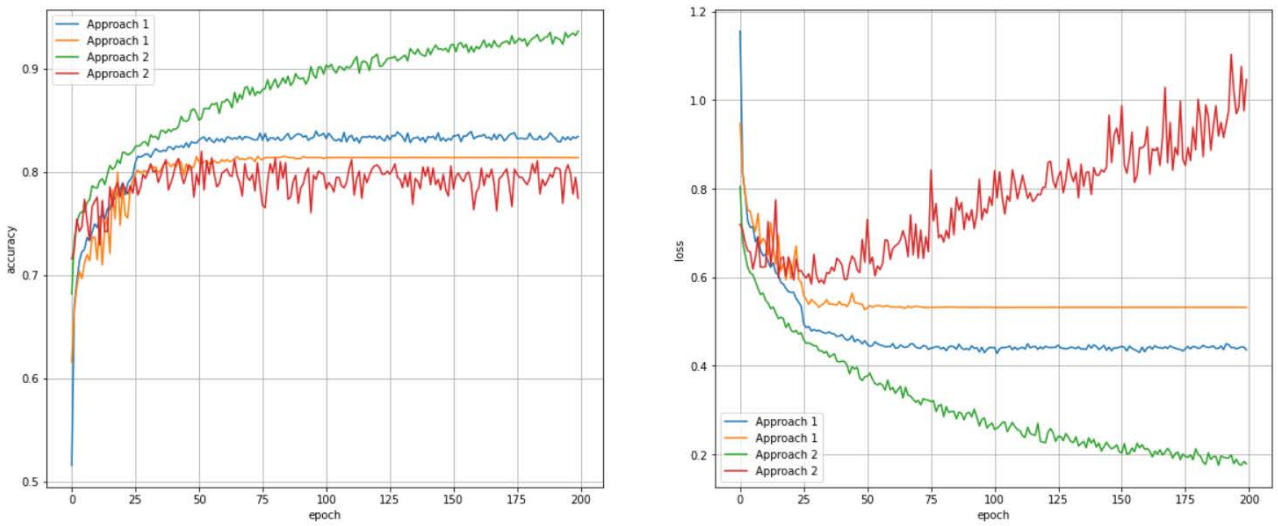

Fig 5: Approach 1 vs Approach 2 learning curve

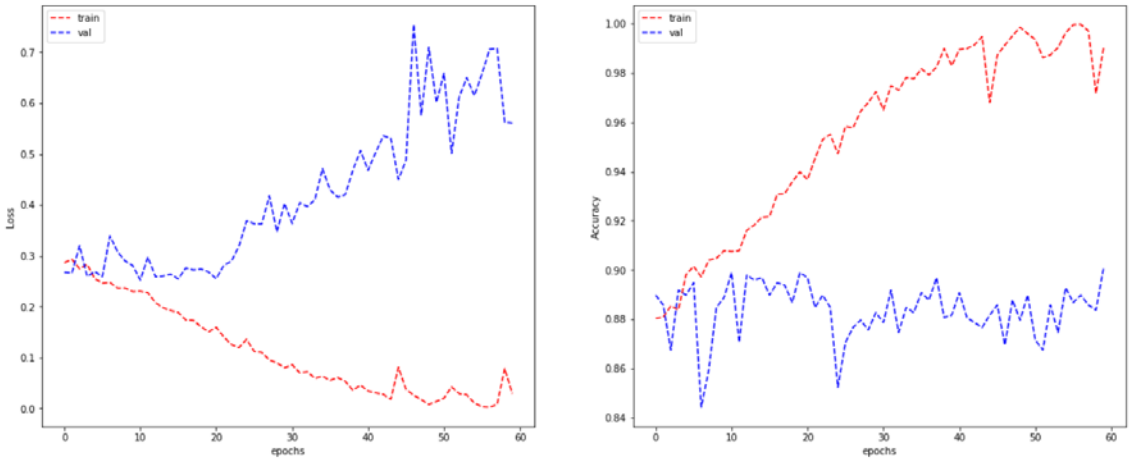

Fig 6: Bayesian CNN learning curve

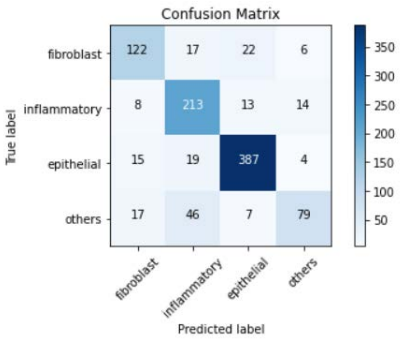

Fig 7: Confusion Matrix for Task 2

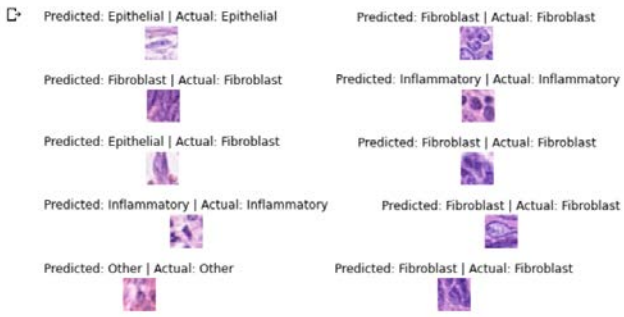

Fig 8: Output of Task 2 Model

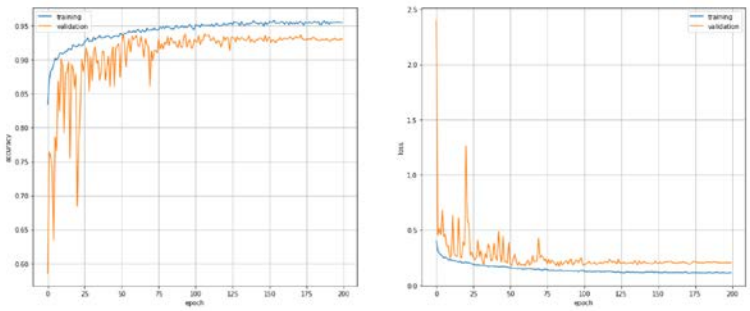

Fig 9: VGG learning curve for task 1