CS904 Assignment 3

Callum Marvell
u1606521

March 18, 2020

1 Introduction & Overview

This brief report outlines the results obtained by the code written for the third CS904 assignment, focused on image segmentation in pathology, using classical features in a SVM, as well as CNN-based classification.

2 Task 1 - Classical Image Features

The objective of this first task was to build a SVM capable of classifying small 150x150 patches/images into one of the 8 types of cells provided.

The features selected and used were based on those used in the paper referenced in the coursework specification (1). These were:

- Histogram measures; mean, variance, minimum and maximum values, kurtosis, skewness and fifth central moment
- The same exact measures on the LBP histogram produced with radius 1, 8 points
- GLCM measures (through scikit-image greycoprops)
- The mean and variance of the average Gabor filter response of the image (for a number of wavelengths and angles)

All of these measures are concatenated into a single large feature list and used for training. Results are standardised prior to training. Over 10-Fold cross validation, accuracy scores average to 0.8 (or 80%) - a fairly respectable score.

```
Average classification score is: 0.8071999999999999
[0.848 0.782 0.796 0.806 0.81  0.832 0.8   0.796 0.794 0.834]
```

Figure 1: Accuracy scores of SVM on 10-fold cross validation

SelectKBest feature analysis was performed on the resulting SVM to gauge performance of specific features.This revealed that by far the most relevant/important was the mean of the average filtered gabor image, followed by the variance, kurtosis, skew and maximum of the grayscale histogram and then the same measures on the LBP histogram. Least useful of all were the values from the GLCM. Some features were entirely useless (or at least effectively useless) - those being the mean and minimum values for both the histogram and LBP histograms.

3 Task 2 - Transfer Learning

For this task, fastai was used with a vgg_19 core in order to facilitate quick and easy transfer learning. All layers except the last were frozen and the 5000 training images were used to adjust the weights of the final layer.

The optimal learning rate was found using built-in fastai methods, and the best number of epochs was found empirically to be 3. This resulted in a CNN that generally exhibits no signs of overfitting (training error exceeds validation error) and has a fairly high degree of accuracy of about 90%.

epoch	train_loss	valid_loss	accuracy	time
0	0.735039	0.344467	0.884000	00:57
1	0.411801	0.275996	0.899000	00:53
2	0.279223	0.246492	0.909000	00:53

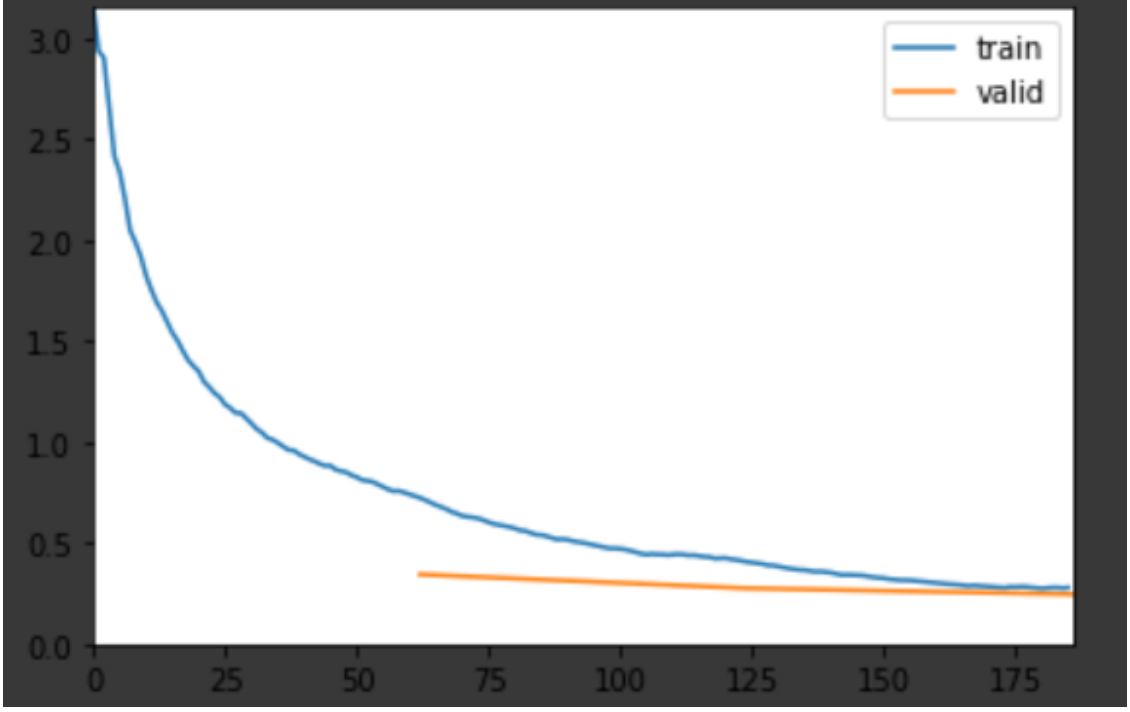

Figure 2: Training/Validation Loss during training of CNN

4 Task 3 - WSI Segmentation

The aim of this task was to use the SVM and the CNN trained/developed in the previous tasks to classify segments of the WSIs provided into one of the 8 classes. To keep the colormaps consistent, Set1 from matplotlib was used throughout - meaning that the classes are coloured as follows:

1. TUMOR - Red

2. STROMA - Blue
3. COMPLEX - Green
4. LYMPHO - Purple
5. DEBRIS - Yellow
6. MUCOSA - Brown
7. ADIPOSE - Pink
8. EMPTY - Grey

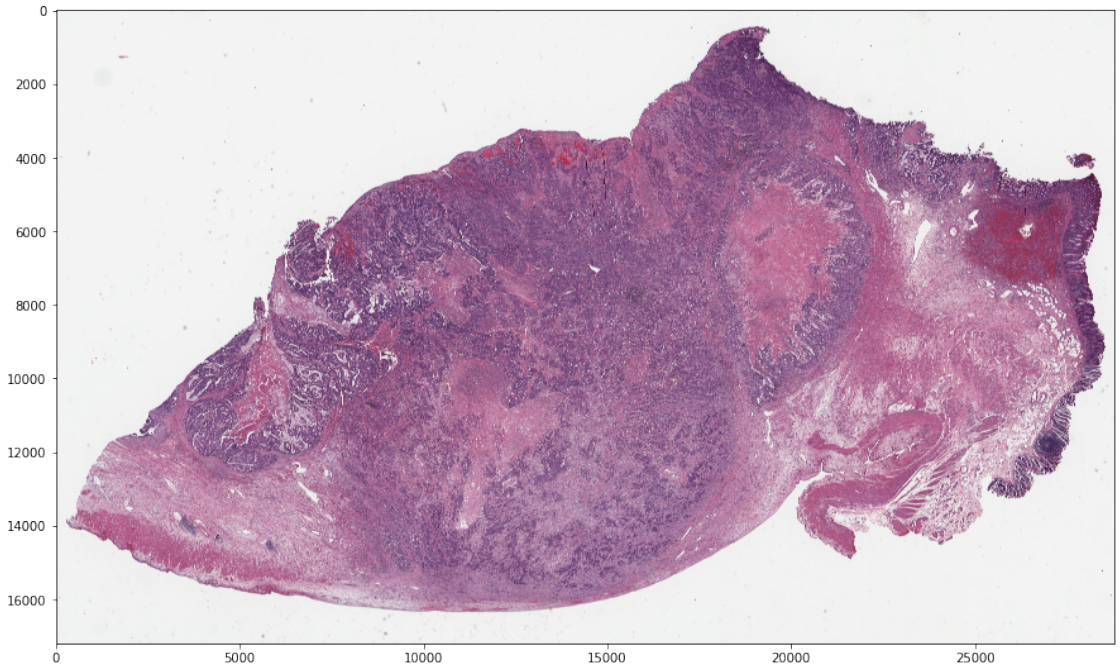

Figure 3: WSI 1

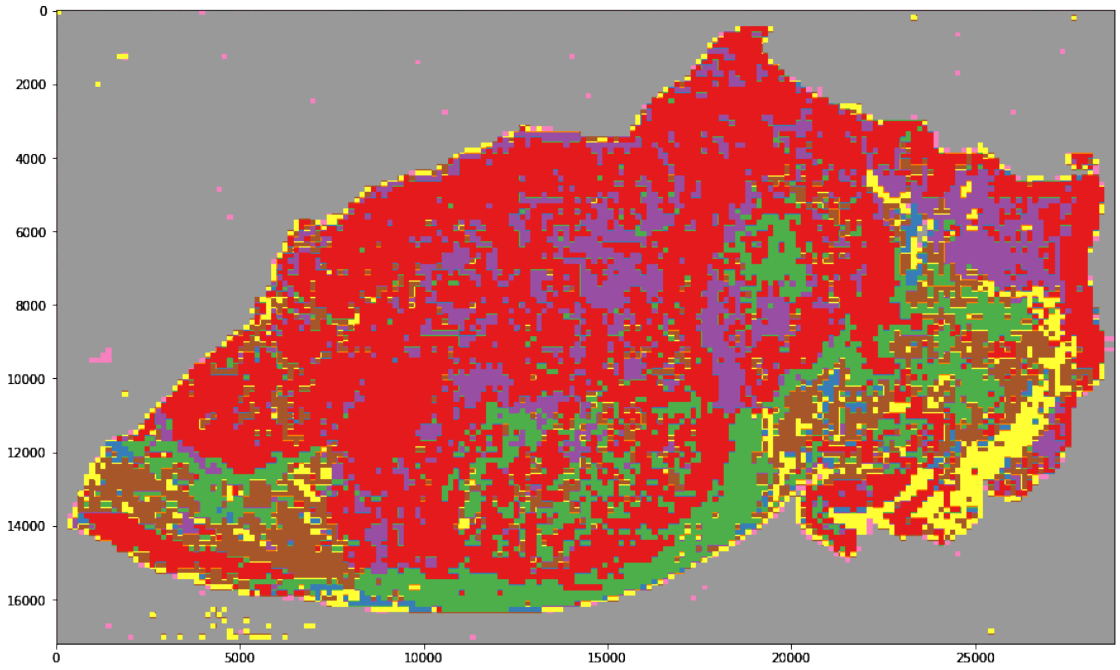

Figure 4: WSI 1 (Segmented by SVM)

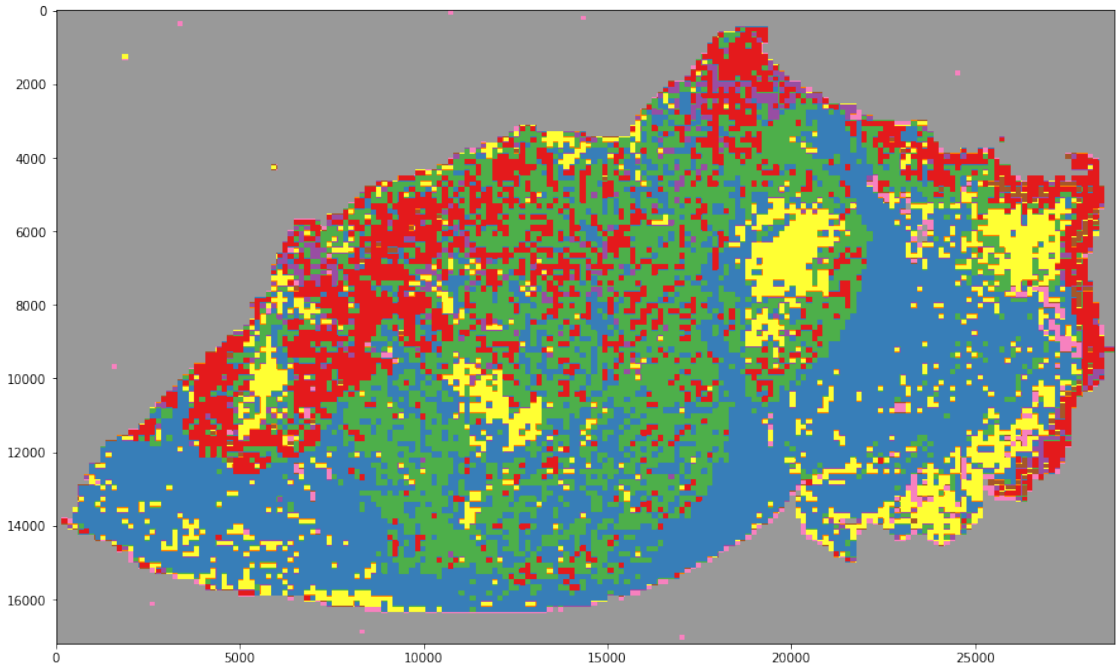

Figure 5: WSI 1 (Segmented by CNN)

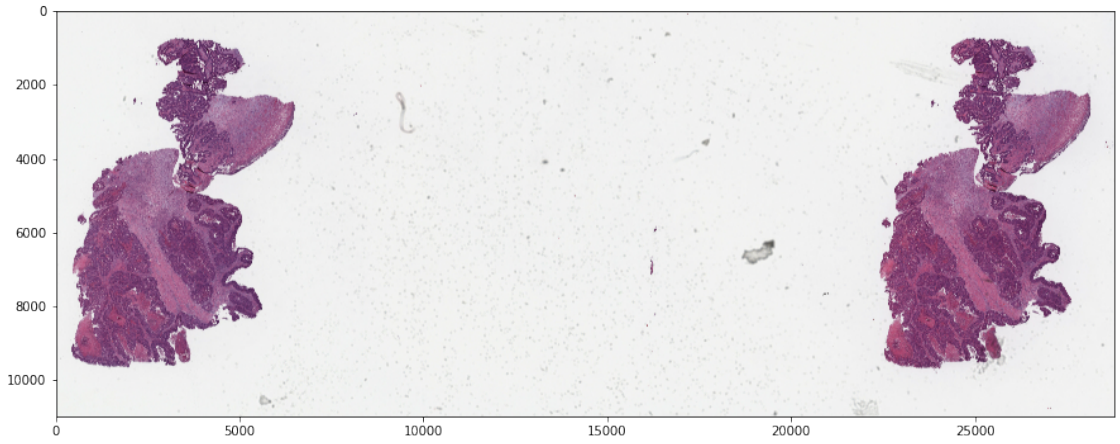

Figure 6: WSI 2

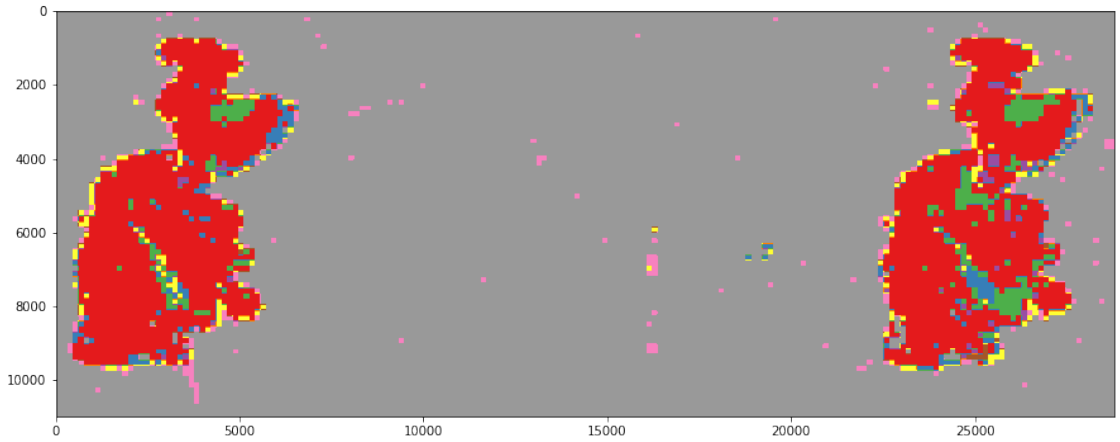

Figure 7: WSI 2 (Segmented by SVM)

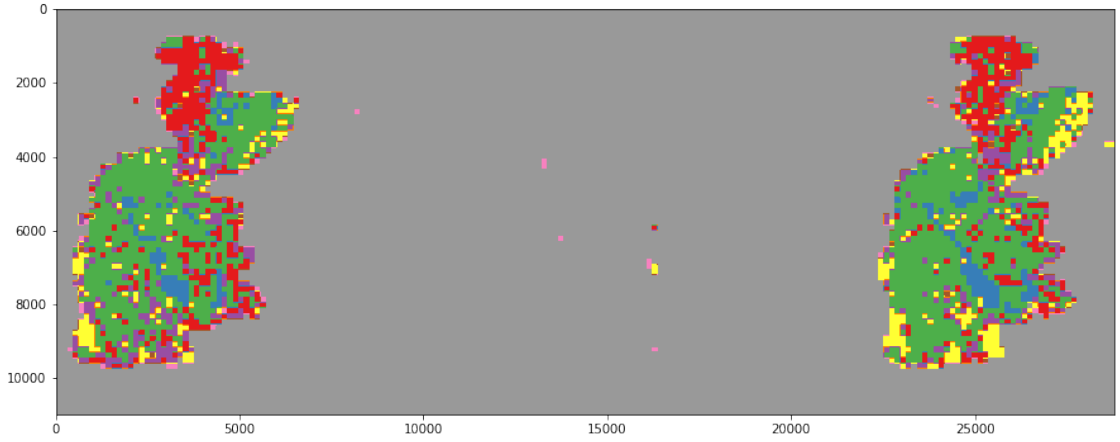

Figure 8: WSI 2 (Segmented by CNN)

Several conclusions can be quickly drawn here:

- Both approaches were very good at recognizing and classifying empty space, so both could likely be used to reduce image file storage sizes easily.
- The SVM was more "trigger happy" to classify cells as tumors than the CNN.
- The SVM tended to "clump"/"group" patches together more often, resulting in larger contiguous areas of a specific classification than on the CNN (despite the identical resolution).

The first image shows some struggle between the two in terms of differentiating STROMA (blue), COMPLEX (green) and to some extent LYMPHO (purple), and the SVM classifies much more as TUMOR (red). Over multiple runs, SVM also seems to often misclassify the EMPTY (grey) space as ADIPOSE (pink) in image 1. This may be due to the specific staining of the image breaking it's interaction with the data standardisation process, and points to the SVM needing a larger source of data for training to perform well on all whole slide images. Otherwise they broadly agree. A similar pattern emerges in image 2, with the SVM classifying much more as TUMOR (red) than the CNN does.

Given the higher accuracy exhibited by the CNN during training, the lack of overfitting and the seemingly more "fine-grained" analysis, it seems to be giving broadly better results. To top that off, it also ran more quickly (at least on the colabs environment used throughout this project). That being said, if more data were to be given as input for training both approaches, the SVM may pull ahead in terms of runtime, and accuracy at such low sample sizes may not be the best assessor. of overall performance. In reality, an approach using multiple variants of the SVM and CNN and taking a "majority vote" from them could be optimal, leading to fewer misclassifications and emulating the real world situation of asking for a "second opinion" from another pathologist.

5 Conclusion

In conclusion, given the small sample set provided and the specific approaches used within these coursework tasks, the CNN-based classifier seems to perform notably better than the SVM-based approach. However, without a significantly larger test set to verify this on, or the opinion of a professional expert, this is hard to guarantee. Either way, both methods perform very well in cross-validation with other parts of the data set, so both seem to be useful automated predictors for the classification of tissue. By combining the two (and perhaps some additional variants) it is very likely that an optimal classifier could be produced that handily outperforms a single "real-person" professional expert, in a shorter amount of time.

All of the code for this coursework is included in the single python notebook submitted. All test runs were performed on google colabs in a GPU hardware accelerated session.

References

[1] J. N. Kather, C.-A. Weis, F. Bianconi, S. M. Melchers, L. R. Schad, T. Gaiser, A. Marx, and F. G. Zöllner, "Multi-class texture analysis in colorectal cancer histology," *Scientific reports*, vol. 6, p. 27988, 2016.