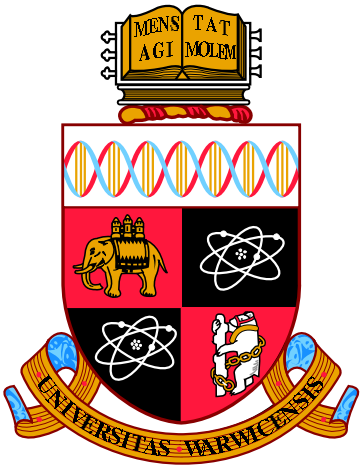

Preprocessing Algorithms for the Digital Histology of Colorectal Cancer

by

Caroline Mary Shapcott

Thesis

Submitted to the University of Warwick
for the degree of

Doctor of Philosophy

Department of Computer Science

March 2020

Contents

List of Tables

List of Figures

Acknowledgments

Especial thanks to my supervisor, Professor Nasir Rajpoot, for his excellent support and guidance in this part-time project. He set the path and has guided the project with accurate judgment and comprehensive knowledge of digital pathology.

I have enjoyed the many wide ranging and stimulating mathematical conversations with Professor David Epstein who has been a wonderful source of inspiration. Thanks too to Rona, David's wife, for giving insights into all sorts of issues and being amazingly hospitable (along with David, of course).

Thanks to all past and present members of the TiaLab, especially Korsuk Sirinukunwattana, Nicholas Trahearn, Adnan Khan and Violeta Kovacheva, Najah Alsubaie and Talha Qaiser.

Thank you too to more recent members, particularly Ruqayya, Navid, Jev, Simon and Javid.

All the staff in Computer Science have been most supportive. Thanks to Roger Packwood who has been a solid technical rock and to Sharon Howard who has been both kind and efficient!

Thanks to Sarab Anand who started me on this path.

I would like to thank the NHS and its patients.

Finally, thanks PG2 - to Conall, Elinor, Art and Mr L. for patience, commas and Latex wrangling.

Declarations

This thesis is submitted to the University of Warwick in support of my application for the degree of Doctor of Philosophy. It has been composed by myself, except where acknowledged, and has not been submitted in any previous application for any degree.

Mary Shapcott

March 8th 2020

Abstract

Pre-processing techniques were developed for cell identification algorithms. These algorithms which locate and classify cells in digital microscopy images are important in digital pathology. The pre-processing methods included image sampling and colour normalisation for standard Haemotoxilyn and Eosin (H&E) images and co-localisation algorithms for multiplexed images. Data studied in the thesis came from patients with colorectal cancer. Patient histology images came from 'The Cancer Genome Atlas' (TCGA), a repository with contributions from many different institutional sites. The multiplexed images were created by TIS, the Toponome Imaging System.

Experiments with image sampling were applied to TCGA diagnostic images. The effect of sample size and sampling policy were evaluated. TCGA images were also used in experiments with colour normalisation algorithms. For TIS multiplexed images, probabilistic graphical models were developed as well as clustering applications. NW-BHC, an extension to Bayesian Hierarchical Clustering, was developed and, for TIS antibodies, applied to TCGA expression data.

Using image sampling with a sample size of 100 tiles gave accurate prediction results while being seven to nine times faster than processing the entire image. The two most accurate colour normalisation methods were that of Macenko and a 'Naïve' algorithm. Accuracy varied by TCGA site, indicating that researchers should use several independent data sets when evaluating colour normalisation algorithms. Probabilistic graphical models, applied to multiplexed images, calculated links between pairs of antibodies. The application of clustering to cell nuclei resulted in two main groups, one associated with epithelial cells and the second associated with the stromal environment. For TCGA expression data and for several clustering metrics, NW-BHC improved on the standard EM algorithm.

Sponsorships and Grants

No sponsorships or grants were associated with this programme.

Acronyms

AJCC American Joint Committee on Cancer.

BHC Bayesian Hierarchical Clustering.

BHC-NW Bayesian Hierarchical Clustering with Normal-Wishart Distribution.

CAF Cancer Associated Fibroblast.

CCD Charge Coupled Device.

CDSA Cancer Digital Slide Archive.

CIMP CpG island Methylator Phenotype.

CNN Convolutional Neural Network.

COAD Colon Cancer Data (TCGA).

CPG Cytosine followed by Guanine in DNA.

CRC Colorectal Cancer.

CSC Cancer Stem Cell.

E Eosin.

ECM Extracellular Matrix.

EM Expectation Maximisation.

FN False Negative.

FP False Positive.

GAN Generative Adversarial Network.

GDC Genomic Data Commons.

GPU Graphics Processing Unit.

H Haemotoxilyn.

IBD Inflammatory Bowel Disease.

IHC Immunohistochemistry.

KNN k Nearest Neighbours.

MLH1 Mismatch repair protein.

MSI Microsatellite Instability.

NHS National Health Service.

OD Optical Density.

PGM Probabilistic Graphical Model.

PM Pathologist assignment of M in TNM staging.

PN Pathologist assignment of N in TNM staging.

PT Pathologist assignment of T in TNM staging.

READ Rectal Data (TCGA).

ReLU Rectified Linear Unit.

RS Random Sampling.

SRS Systematic Random Sampling.

TA Transit Amplifying Cell.

TCGA The Cancer Genome Atlas.

TIL Tumour Infiltrating Lymphocyte.

TIS Toponome Imaging System.

TME Tumour Microenvironment.

TN True Negative.

TNM Staging System - T, tumour N, node M, metastasis.

TP True Positive.

WHO World Health Organisation.

WSI Whole-Slide Image.

Chapter 1

Introduction

Radical improvements in camera technology, huge expansions in storage capacity and the development of super-fast graphics processing units have turned *digital pathology* into a rapidly maturing discipline (Kayser [93]). Digital pathology enables researchers to extract knowledge from images stored in the resulting repositories (which may be distributed or centralised).

This thesis analyses image data originating from patients with cancer of the lower intestine, *bowel cancer*. Bowel cancer is the third most common cancer in the UK [28] and occurrences of this cancer in the US [29] follow a similar pattern. In addition, advanced colon cancer, characterised by *metastasis*, its spread to distant sites in the body, has a high mortality rate ([160]).

Data produced by the *The Cancer Genome Atlas* project (*TCGA*), are used extensively in this thesis [89]. TCGA was a project sponsored by the US government in which the aim was "to profile genomic changes in 20 different cancer types" [167]. Data collection commenced in 2005. In the case of colon cancer twenty-four sites from across the USA contributed data which included clinical and demographic data as well as results of *sequence analysis*, i.e. the determination of DNA coding and *protein expression* data which measured the presence of proteins in tissue. Although the TCGA project has been wound up the TCGA data repository remains accessible via the *Genomic Data Commons* portal [131] which as of October 30th 2020 contained data from "67 projects covering 68 primary sites, 23,399 genes and 3,376,130 mutations from over 84,000 cases" [64].

Bowel cancers are also referred to as *colorectal cancers* or *CRCs*. Data related to colorectal cancers that have been extracted in surgery are stored in the TCGA

COAD (colon cancer) and READ (rectal cancer) data collections.

Much of this thesis is devoted to the analysis of diagnostic images from TCGA, images of cancer tissue that have been captured using high-definition digital microscopes. The aim has been to identify the individual cells in an image using digital techniques and to uncover interesting relationships between cell features and other variables of interest such as clinical variables (gender, age, pathology scores, etc.) and molecular data.

This introductory chapter is structured as follows. Section 1.1 contains a brief overview of the role of pathology in the diagnosis and treatment of colorectal cancer, focussing on *histology*, the examination of diseased tissue under the microscope. The section describes various standard techniques for processing histology images. Section 1.2 introduces digital histology: the use of advanced devices, software and computational techniques in histology. The remaining sections address topics covered in the three substantive chapters of this thesis. The first topic, described in Section 1.3 is the application of sampling to cell identification algorithms. Section 1.4 introduces the second topic, the use of *colour normalisation* in processing histology images: the adjustment of colour intensities so that they match specified colour distributions. Section 1.5 describes the third topic, the analysis of multiplexed images output by a robot, the *Toponome Imaging System (TIS)*, which applies multiple reagents to tissue samples. In addition this chapter describes a Bayesian clustering algorithm and its extension to data with more complex structures. Finally, Section 1.6 summarises the contributions to knowledge made in this thesis.

1.1 Histology - Tissue under the Microscope

This section describes the contents of the first part of Chapter 2, the background chapter. One of the primary responsibilities of the pathology laboratory is to analyse tissue and to report on any findings. The tissue may come from a *biopsy*, whereby a small sample is extracted from a suspicious location in the body, or from tissue extracted during surgical excision of the cancer. After surgical excision the tumour mass and surrounding tissue are sent to the pathology laboratory which then reports on gross characteristics, such as tumour size and appearance and on results of examination under the microscope. Both the COAD and READ data sets contain the results of such analysis.

After laboratory analysis the resulting pathology report is sent to other mem-

bers of the medical team. The report is used by physicians for diagnosis and to decide on treatment options [31]. These include surgery (particularly in the case of biopsies), chemotherapy, radiotherapy, or even no active treatment - 'watchful waiting' [130].

As well as results generated in the pathology laboratory a typical pathology report (Ayesha Azam [11]) contains information supplied by the patient's medical team: demographic information such as gender and age, plus observations made in surgery such as the location of the extracted tumour and radiology results such as the degree of metastasis. Individual contents of a typical pathology report for colon cancer are discussed in Section 2.3

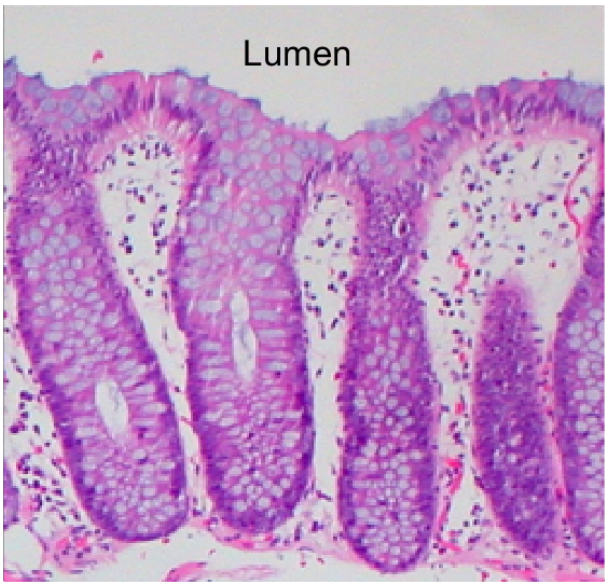

Figure 1.1: View of normal colon tissue under the microscope. (Ed Uthman [50])

In general, for normal healthy, tissue taken from the gut wall the biological structures appear well organised: cells are positioned with a high degree of regularity, and their sizes and shapes are well defined. Figure 1.1 is an example of normal colon tissue. The empty region at the top of Figure 1.1 is part of the gut *lumen* where the waste products are carried. The surface of the gut wall is marked by a single layer of connected dark cells, a layer which is folded like a part of a glove that has fingers. In Figure 1.1 there are three fingers that start at the lumen and have their tips at the bottom of the image. Each of these fingers has darkish purple cells along its edge. In the three-dimensional volume from which the tissue section was taken these cells are in a sheet that is one cell thick. The finger-like regions are called *crypts*

and they are the place where liquid is removed from the matter in the gut. The cells forming the surfaces of crypts are mostly *epithelial* cells.

An important result of microscopic examination which is included in the pathology report is the cancer *grade*, an indicator of the abnormality of the tumour's microscopic appearance. In normal tissue cells develop from precursor cells called *stem* cells: cell *differentiation* occurs when cells take on their function in an organ, such as forming part of a layer of skin or fighting infection: the cancer grade is an assessment of the degree of differentiation seen in the tissue under the microscope. *Low grade* cancers are *well differentiated* and resemble normal healthy tissue while *high grade* cancers are *poorly differentiated*, looking disordered and unlike normal tissue.

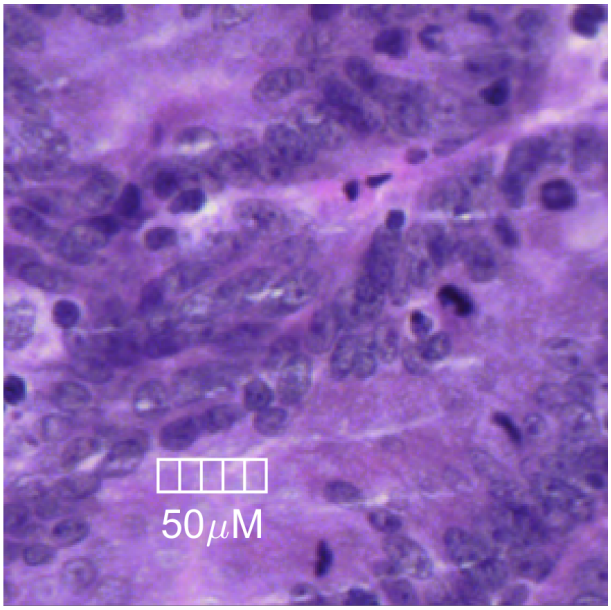

Figure 1.2: Tile from a moderately differentiated tumour (TCGA COAD:Patient AA-3543, Tile 1490). (Tile numbering is explained in detail in Subsection 3.5.2.)

Cancer grade is described in more detail in Section 2.6, in the background chapter of this thesis. Here it is discussed in the context of colorectal cancer.

In contrast, to Figure 1.1 the colon cancer tissue shown in Figure 1.2 appears disorganised and cells tend to be large and irregular in shape. Here the epithelial cells are only roughly lined up in the shape of crypts. They are clumped together in some areas and scattered about in others. The tissue displayed in Figure 1.2 is from the diagnostic image for patient AA-3543 in the TCGA repository. The cancer grade

assigned to the tissue was “moderately differentiated”. Further note that the term *tile* in the caption of Figure 1.2 refers to a region in the diagnostic image which is defined in detail in Subsection 3.5.2.

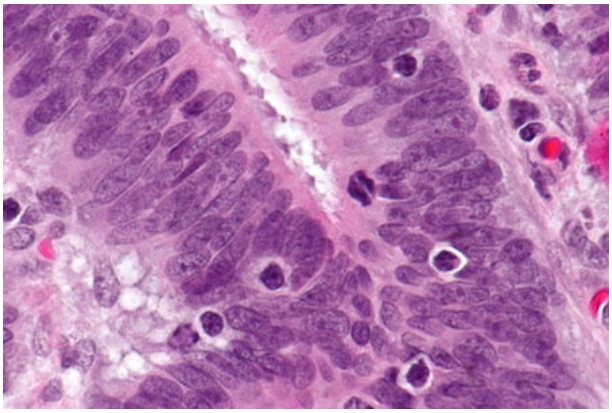

Figure 1.3: Tumour Infiltrating Lymphocytes. (Libre Pathology [116])

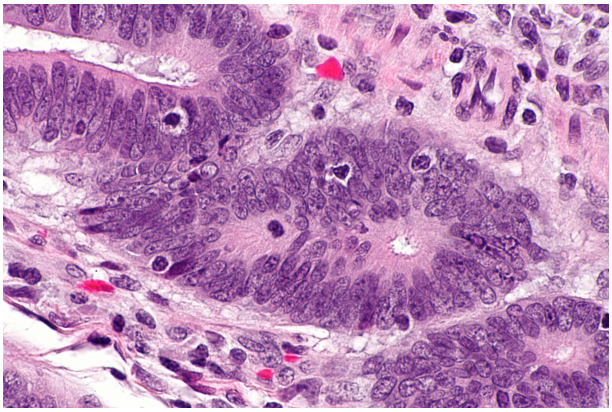

Figure 1.4: Tumour-Infiltrating Lymphocytes (From TCGA COAD)

Inflammatory cells are cells produced by the body in response to infection or other threats such as cancer. Figures 1.3 and 1.4 are images of colon cancer tissue in which *tumour infiltrating lymphocytes (TILs)* are visible as well as epithelial cells. A *lymphocyte* is a type of inflammatory cell that is made in the bone marrow and is found in the blood and in lymph tissue (National Cancer Institute [128]). Some lymphocytes make antibodies while others kill tumour cells and help control immune responses. TILs are lymphocytes that are interspersed among epithelial cells in the tumour. In Figures 1.3 and 1.4 TILs may be identified by their round shape, dark with a light-coloured halo.

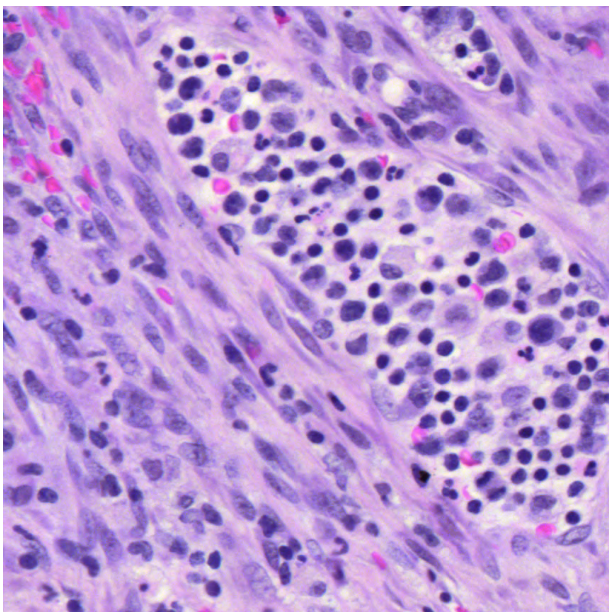

Figure 1.5: Tissue Section Showing Fibroblasts - long spindle-shaped cells surrounded by pink staining (Patient:AA3543, Tile 1067)

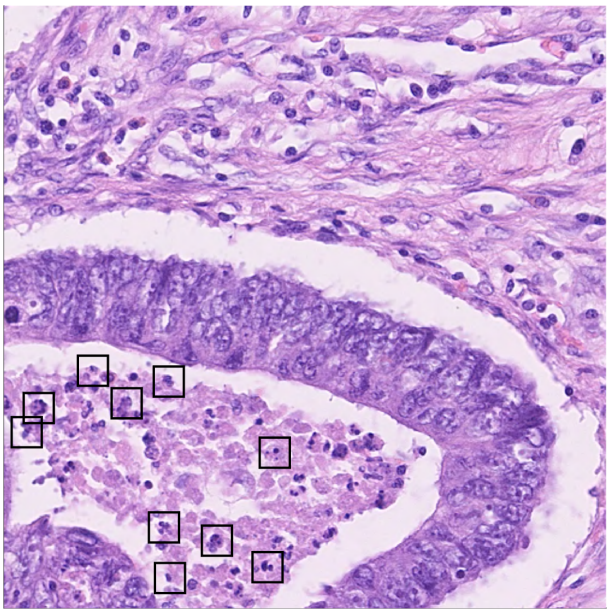

Figure 1.6: Tissue Section Showing 'other' cells ('Cell' Training set: Image 86) (3.3)

So far two categories of cells, *epithelial* cells and *inflammatory* cells have been introduced. Two other categories analysed in the thesis are *fibroblasts* and a miscellaneous category of *other* cells. Figure 1.5 displays fibroblasts from the TCGA repository while 'other' cells are displayed in Figure 1.6. The 'other' cells are framed by black boxes in the figure. Cell types are described in more detail in Section 2.4.

When pathologists individually assign values to cancer grade the level of agreement is relatively low. There are usually only three or four different classes in the grading scheme (see the related discussion in Chapter 2), which also lowers the accuracy of predictions, but cancer grade does summarise what is seen under the microscope and is a measure of the severity of the underlying disease.

1.2 Digital Modelling of Histology Images

Since the mid-sixties when Prewitt and Mendelsson [139] applied computerised image analysis to histology slides much effort has been devoted to the extraction of meaningful information from such slides. The aim has always been to formulate a *model* of the image data, a model which has predictive power.

The term *model* refers to a mathematical formula which uses input variables to output *predictions*. For example, in digital histology the input to the model is generally an image while the predicted output is some useful quantity, often related to diagnosis or prognosis. Observe that in the literature the term 'model' may also refer to an *algorithm* which does the actual calculations.

Early work in digital pathology used *hand-crafted* models whereby in the modelling process features of the image are selected for use by experts in the field and input to a mathematical formula that predicts quantities of interest. The aim may be to predict quantities for low level tasks such as counting cells or distinguishing regions of tumour from regions of normal tissue or the objective may be to predict high level indicators for diagnosis or prognosis. For example Yuan [187] modelled the spatial distribution of lymphocytes in triple negative breast cancers, finding a formula that was a predictor of patient survival. This example, where domain specific knowledge was used to select the variables of interest, in order to identify cell locations is typical of handcrafted approaches.

Frequently an image processing model breaks the prediction task into *feature identification*, followed by the prediction of a quantity of interest using those features. The features may be low-level quantities such as texture features, edges of objects, or even colours. High level features may represent complex biological entities such as cell nuclei. In colon histology not only cells but also other objects are of interest. These include crypts and regions of tumour and in these cases the output of feature

identification is a spatial map of these objects.

The map is then summarised in a *profile*, a set of *summary features* such as counts or correlations. In the case of Yuan [187] two summary features were extracted from each histology image. The first feature was was the number of intratumoral lymphocytes and was the second feature was the number of cancer cells. These were combined in a single value: the ratio of feature 1 to feature 2 which could be regarded as single-feature *profile* of the patient. In the work just described the patient profile output by the model was a good predictor of both survival time and the level of cytotoxic T lymphocyte protein.

More recent work in digital pathology has focussed on *Deep Learning*, the application of *Convolutional Neural Networks (CNNs)* to image processing. A convolutional neural network is an assembly of processing components which is designed to take advantage of local spatial correlations in the input data. A CNN has a form which allows it to use general features, not just pre-selected ones. When presented with an input object, a CNN passes data through various processing *layers*, outputting predictions in the last layer. For example, in the case of cell location algorithms, the input may be an image of a cell, and the output may be the cell type. Figure 1.7 is a diagrammatic representation of such a network, used in Chapter 3. Note that the network is based on the well-known CIFAR-10 image classifier, available from the Tensorflow web site [10]. In the example shown in Figure 1.7 input to the model is an image containing an inflammatory cell. The model processes the input, and a prediction of the cell type is output (e.g. one of 'Epithelial', 'Inflammatory', 'Fibroblast' or 'Other'.

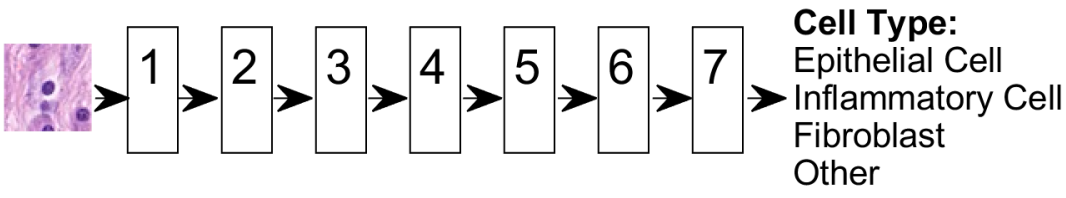

Figure 1.7: Schematic of CNN Model. The model classifies an image according to one of the four categories shown on the right. The predicted cell type is 'Inflammatory'. Seven processing layers are shown.

To predict the cell type on the right of Figure 1.7 the image on the left is input to the layer $L1$ which extracts many small patches from the image, transforms them, and passes them to the next layer $L2$. Data passes through the layers of the

network, and the final layer $L7$ outputs the predictions made by the network. The operation of each layer is specified in detail using a collection of *parameters*, numeric values which define the transformations applied to the data input to the layer. In the case of CNNs these parameters are usually referred to as *weights*. The sheer number of weights make CNNs very flexible [51] compared with hand crafted models but determining the best weights to use presents challenges.

1.3 Sampling

Chapter 3 explores the effects of sampling in the CNN prediction process. Most applications of CNNs use the entire histology image to make predictions. To do this the whole-slide image is subdivided into *tiles* using a *grid* of constant size, the CNN is applied to each tile individually, and the per-tile results are stitched together in order to compute the whole-image features. Tiles are typically large enough to contain hundreds of cells, but small enough so that computations are not too demanding of memory or processing power.

Deep learning models can be trained and deployed in reasonable time-scales using the power increasingly offered by inexpensive graphics processing units. However, using the entire image in prediction remains expensive and any performance improvements are useful. By using only a small proportion of the image in the prediction phase of modelling, sampling can improve performance without significant loss of accuracy. In addition, sampling gives insights into how features of interest are distributed in the image.

In Chapter 3 two sampling policies, *Random Sampling*, and *Systematic Random Sampling* were examined. These policies were applied to both the *'Cell'* identification algorithm of Sirinukunwattana et al. [157] and the *'HoverNet'* algorithm of Graham et al. [71] which segmented an image into regions corresponding to single cells.

After training using a locally available data set the trained 'Cell' model was applied to tiles sampled from diagnostic images that belonged to the TCGA data repository. The output, a list where each entry was a coordinate pair accompanied by a classification tag, was not useful without further processing. Instead, it was necessary to extract *profiles* from the list, summary features which captured the distribution of cells within the diagnostic image. In this study straightforward statistics were used: for each whole slide image (WSI) the average numbers of cell types per unit area was calculated, a profile descriptive of the WSI.

Like the 'Cell' model, the 'Hovernet' model accepts tiles as input. In contrast to the 'Cell' model which outputs cell locations tagged by cell type, the 'Hovernet' model outputs a segmentation of the tile into disjoint regions labelled by cell classification. The network is trained by optimising the detection weights and the classification weights simultaneously.

Experiments were carried out using with different sample sizes. It was found for the calculation of profiles, that sampling led to a nine-fold improvement in speed, with little degradation in performance. This applied to the use of Sysematic Random Sampling with a nominal sample size of 100 tiles. Systematic Random Sampling was more accurate than Random Sampling.

As an application, associations between profile values and various clinical variables were calculated and several of these were found to be significant.

1.4 Colour Normalisation

Chapter 4 describes exeriments with *colour normalisation* using TCGA data.

Differences in laboratory preparation techniques, including variations in the stain manufacturing process, and differences microscope software lead to colour variations in histology images. Although these variations are handled quite easily by the human eye they challenge digital image processing quite seriously. For example, in Chapter 4 an experiment with TCGA diagnostic images, the use of raw unnormalised images for cell classification with the 'Cell' algorithm described in Chapter 3, resulted in an average classification accuracy of less than 40%.

In the TCGA COAD repository, patient data have been uploaded from twenty-four different sites and the diagnostic images vary markedly with respect to average colour intensity. It is necessary to cater for such batch effects. *Colour normalisation* pushes the input image into the training colour space, where it can be processed more accurately. Colour Normalisation was used in preprocessing, transforming the image before it was presented to the 'Cell' algorithm.

Two experiments with colour normalisation were conducted, one for cell detection and the other regarding cell classification.

Tiles containing cells were selected from ten different TCGA contributing sites, hand-marked and normalised using five different colour normalisation techniques.

Cell detection, that is the prediction of the locations of nuclei was applied to the normalised images. The locations of cells were hand marked and compared with predictions. No improvement was observed, possibly because the cell detection algorithm carried out stain normalisation internally before applying the detection CNN.

In contrast it was found that colour normalisation markedly improved classification accuracy. However, the improvement was quite site-dependent, indicating that studies of colour normalisation should include several sites, rather than only one or two.

'Naive' colour normalisation and Macenko stain normalisation were the winning techniques: for each site, the best performing algorithm was one of these two methods. The computational cost of 'Naive' Standardisation is much lower than the Macenko technique, indicating that this straightforward technique should be seriously considered for colour normalisation.

1.5 Molecular Analysis

Chapter 5 deals with multiplexed images created by an imaging robot [105]. Input to the robot, the *Toponome Imaging System (TIS)* is a tissue section (extracted from a tumour block in the case of colorectal cancer). The robot contains a caddy (library) of *antigens* chemicals that indicate the presence of different types of molecules, particularly proteins. In successive processing rounds each antigen is applied to the tissue section and the resulting patterns of adhesion are captured under fluorescent light using a charge coupled device. Each complete execution of a robot program results in a *stack* of images that record spatial maps of the molecules associated with the antigens.

In the thesis analysis of stack data included the following. Pearson *colocalisation* analysis was extended to the multivariate case with *Probabilistic Graphical Models*, extracting undirected graphs from the set of image stacks. *Partial correlations* are extracted from such graphs: providing better explanatory power than marginal correlations extracted directly from the covariance matrix. In one application of graphical modelling the data points were pixels and the features were the per-pixel intensities of the tags (antigens). In the second application, the data points were cell

nuclei and the features were average intensities across the cell nuclei.

The second method used to analyse stack data was clustering, applied to intensity vectors, calculated on a per cell basis. In the case of images classed as 'normal' tissue the EM clustering algorithm identified regions which mostly corresponded to crypts and stroma.

The second part of Chapter 5 considered the use of clustering in the analysis of gene expression data, for proteins used in the TIS stacks. *Bayesian Hierarchical Clustering, BHC* was extended from uncorrelated data to modelling correlated data. The extended algorithm *BHC-NW* was applied to protein expression data from the TCGA data repository. Using various clustering metrics *BHC-NW* performed well, beating the EM algorithm for Gaussian mixture models on many criteria.

BHC-NW found two significant clusters in the gene expression data; the smaller cluster being associate with MLH1 underexpression. Comparison of clinical variables according to cluster found associations with tumour location. For example, the smaller cluster was associated with right-sided tumours. These results are in accord with the smaller cluster containing CIMP-high, MSI-high tumours - that molecular grouping being regarded as significant for outcome and treatment.

1.6 Contributions

In cell identification the role of sampling was explored. For two well known algorithms, 'Cell' and 'Hovernet' when sampling was used the computational effort was reduced by a factor of about 9 or 10 without significant loss of accuracy. An application of sampling to the TCGA COAD dataset found significant associations between the densities of cell types and various clinical variables.

Colour normalisation was found to be advantageous when determining the type of a cell, though it did not prove to be useful for cell detection. In addition to the well-known Macenko algorithm a straightforward implementation without stain decomposition performed well experimentally. Different sites varied greatly: researchers should beware of results that apply to only one of two sites.

Molecular data obtained from multiplexed fluoerescence images was analysed using both probabilistic graphical models and clustering. The BHC algorithm was extended to data with off-diagonal elements in the cluster covariance matrices and

to hyperparameter optimisation. The extension BHC-NW was used to cluster gene expression data from the COAD and READ data sets, outperforming other clustering algorithms. In addition various clinical variables were found to be associated with cluster assignments.

Chapter 2 covers the background to the three research chapters.

Chapter 2

Background

Research made possible by developments in digital camera technology, networking and GPU technology has led to considerable progress in histology. This chapter introduces conventional histology and describes recent work in digital histology, providing the background for Chapters 3, 4 and 5.

The development of digital camera technology has enabled histology slides to be recorded digitally and the resulting image files can be saved in digital repositories. Thanks to networking technology, users can view remote images and the information associated with them, such as markings made in the laboratory to draw attention to specific regions in images, as well as technical details of the procedures used, such as microscope manufacturer and pixel resolution. The Cancer Digital Slide Archive (CDSA) (Gutman et al. [74]) is such a repository allowing TCGA images to be viewed using a web browser (See the CDSA website [2]).

Digital pathology replaces shelves of glass slides - and the postal service - with digital storage and networking technology, but its advantages go beyond routine patient care. With digital pathology it becomes easier to search for interesting associations between histology images and biological and molecular features. Already, research using manual pathology has found many such associations, some of which are described in this chapter.

If it is assumed that the view under the microscope (either direct or digitised) contains useful information about underlying biological processes then we may explore different ways of accessing that information. Digital image processing can be used for many different purposes: it has the potential to identify biological objects such as cells and crypts, to distinguish abnormal from normal tissue and to identify the molecu-

lar subtype of a cancer, for example to determine what sorts of mutations are present.

This background chapter is organised as follows:

Section 2.1 is a brief introduction to the biology of colorectal cancer. Standard histology procedures in the pathology laboratory are discussed in Section 2.2 while Section 2.3 describes the structure and contents of a standard pathology report. Section 2.4 discusses the appearance of tissue under the microscope.

Section 2.5 describes various measures that have been found to be useful in diagnosis, prognosis and in making treatment decisions. Some of these measures, such as cancer 'grade' are already employed in routine pathology (Section 2.6). Others have been discovered through research but are not in general use.

Increasingly, information about molecular processes is being used in medical decision making. Section 2.7 describes relationships between mutational data, protein expression data and colorectal cancer.

Section 2.8 discusses developments in digital pathology: the use of image processing models which make useful predictions for patient care. Models based on convolutional neural networks (CNNs) are discussed in Section 2.9. We describe the structure of CNNs, their use in image processing applications and issues concerning the training of CNNs. In particular we discuss various optimisation strategies used in CNN training. CNN models used in cell identification are discussed.

2.1 Colon Cancer Biology

The colon is the part of the gastrointestinal tract which is responsible for absorbing water from human waste products. Figure 2.1 shows how the gastrointestinal tract is organised.

The innermost layer of the gut wall is called the *mucosa*. The mucosa is surrounded by a layer of connective tissue (the *lamina propria*). Epithelium can be seen in the image of the mucosa in Figure 2.2. The epithelial layer is a single sheet of columnar *epithelial cells* that are clearly visible in Figure 2.2. Folds in the sheet take the form of finger-like invaginations called the *crypts of Lieberkuhn* - about 14,000 per cm^2, also visible in Figure 2.2. Figure 2.3 is a diagram of a crypt, dating from 1892. Note that crypts are also found in the small intestine. The mucosa is

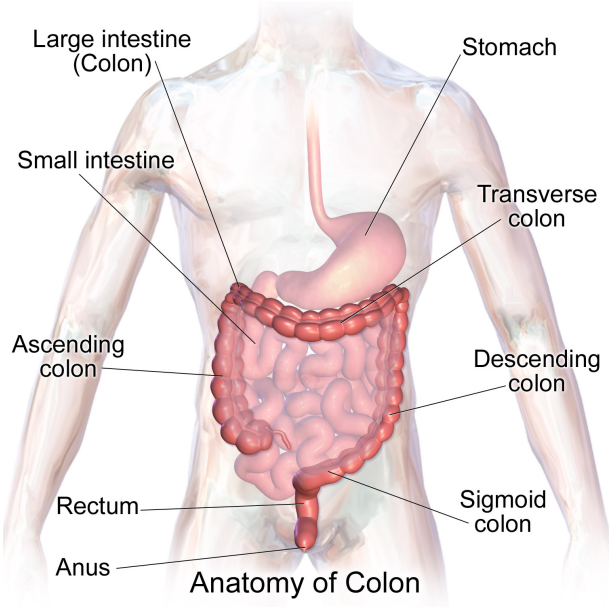

Figure 2.1: The Colon [162]

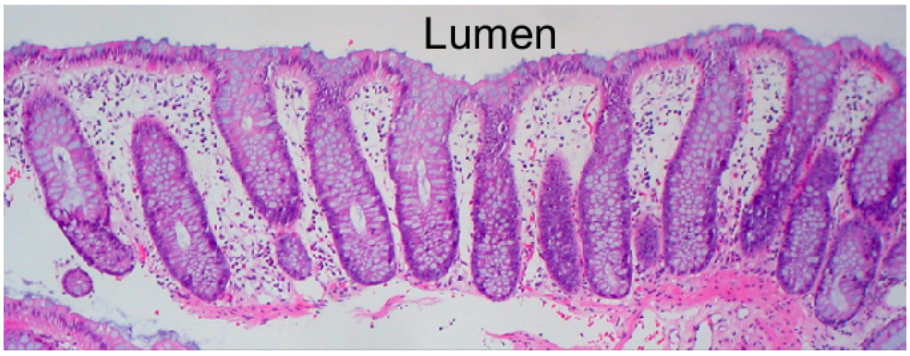

Figure 2.2: Mucosa of the colon. [50]

surrounded by the *submucosa* and the *muscularis externa* (the external muscle layer) while the outermost layer is called the *adventitia* or *serosa*. Note that the empty region in the centre of the tract is called the *lumen* and is where the waste is carried.

Figure 2.4 shows a section of normal tissue containing crypts. The section is a transverse cut through the crypts which therefore appear circular in the figure. The boundary of each crypt is formed by a single layer of epithelial cells. The lumina at the centres of the crypts are also visible as white coloured regions. Some of the epithelial cells are *goblet* cells which are responsible for secreting *mucus* into the lumen. The mucus is a jelly-like substance which aids the progress of waste products through the bowel. The mucus builds up in the goblet cells, in the region between

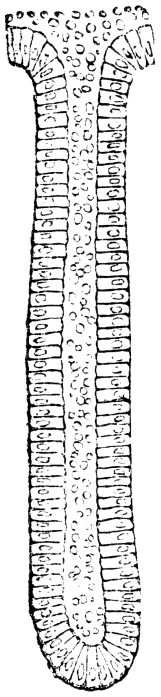

Figure 2.3: A Crypt of Lieberkuhn

epithelium and lumen: the region that resembles a cut through an orange.

Most cancerous tumours develop from polyps that form on the inside of the gut. The majority of polyps are benign and do not form cancers ([80], [1]), but those polyps that do develop into tumours are characterised by uncontrolled cell division and proliferation. Less commonly, non-polypoid areas of *neoplasia* (often associated with inflammatory bowel disease (IBS)) develop into tumours (Zisman and Rubin [190]).

If a patient's symptoms or screening results suggest the presence of colon cancer, a *biopsy* may be taken: a section of tissue from a suspicious polyp or from an area of *dysplasia*, the abnormal growth of cells. The sample is sent to the pathology lab for screening. If the pathology report suggests that surgery is needed then the tumour is removed and the surgical section, (i.e. the tumour and any surrounding tissue) is preserved. In the pathology laboratory, the tumour is examined to check how far it has penetrated the bowel: a measurement recorded by the pathologist - the *T stage*. See Section 2.3 for more details.

Figure 2.5 is a schematic diagram showing two crypts. The top third of a crypt contains terminally differentiated epithelial cells which are continually extruded

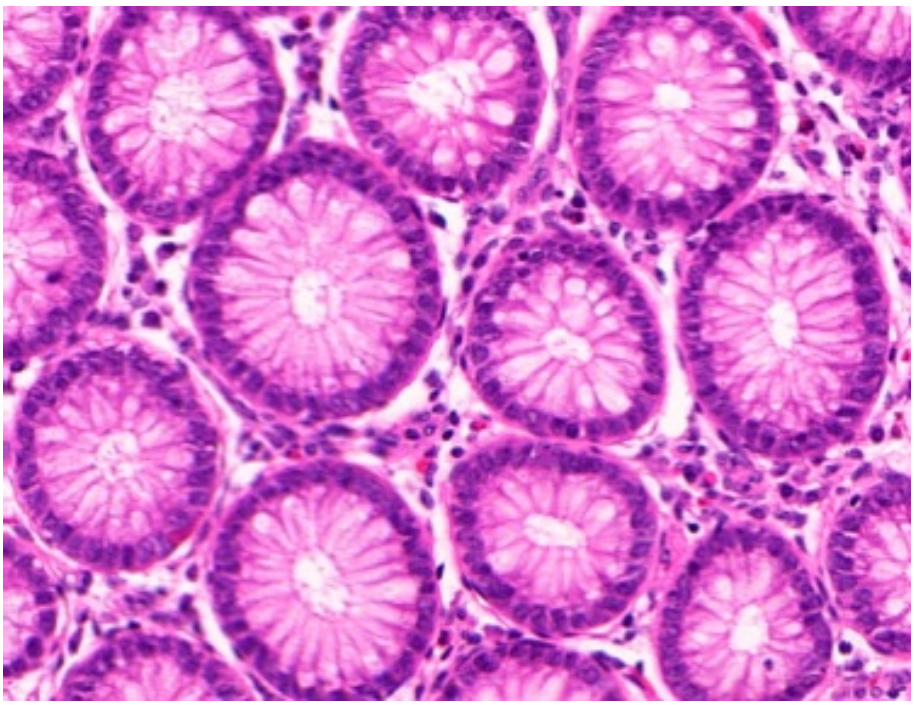

Figure 2.4: Normal Tissue Containing Crypts. Original image obtained from Wikipedia contributors [185]

into the lumen [144]. There are three main types of such cell: *coloncytes* (columnar absorptive cells also called absorptive enterocytes), mucus-secreting *goblet* cells, and *enteroendrocine* cells.

Stem cells at the bottom of the crypt are responsible for producing these differentiated cells [96]. They have two main properties: they are able to perpetuate themselves through extended time periods and they are *pluripotent*: they can generate differentiated cells of the tissue of origin. The stem cells generate transit-amplifying (TA) cells that proliferate and differentiate into one of the epithelial cell types. The exact location of the stem cells is undetermined, but Ricci-Vitiani et al. [144] state that it is believed that stem cells are: "interspersed among more differentiated daughter cells" making it difficult to identify them.

Others identify the +4 position in the crypt (i.e cell number four in the crypt wall, counting away from the bottommost cell in the crypt) as the location of stem cells [148]. In a normal human crypt there are about 2,000 cells and, it is believed, about 19 stem cells [96].

Stem cells are responsible for the self-renewal of the crypt and many cell

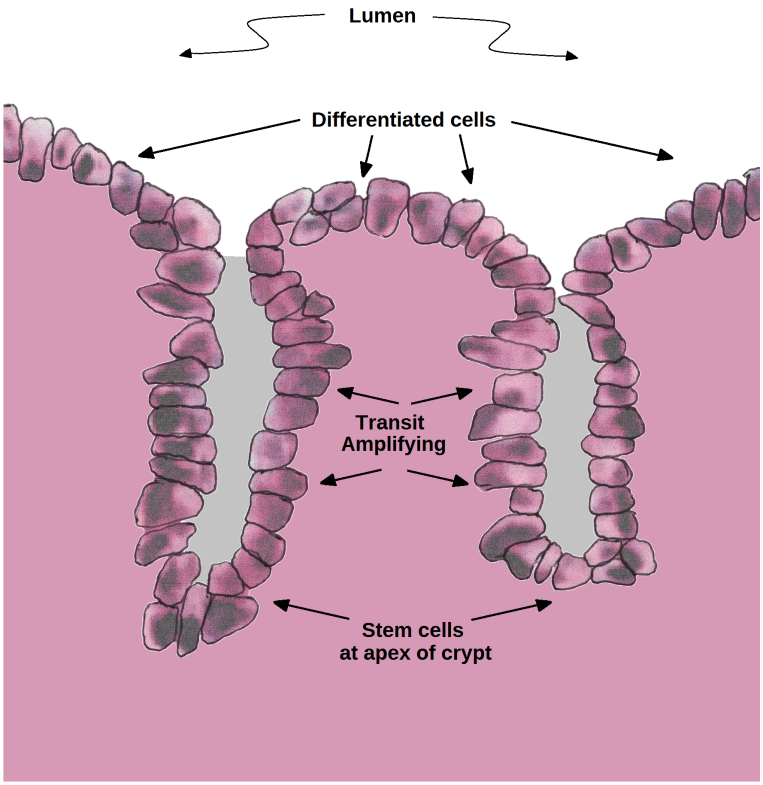

Figure 2.5: Colonic crypt organisation. Stem cells are located at the bottom of the crypts. Upon asymmetrical division, the daughter cells undergoing differentiation migrate upward to give rise in turns to transit-amplifying (TA) precursors and terminally differentiated cells. Ricci-Vitiani et al. [144]

processes are involved ([164]) , in particular in the *Wnt signalling pathway*. Schatoff et al. [149] state:

"The WNT signaling pathway is a critical mediator of tissue homeostasis and repair, and frequently co-opted during tumor development. Almost all colorectal cancers (CRC) demonstrate hyperactivation of the WNT pathway, which in many cases is believed to be the initiating and driving event."

Colon cancer is one of the best-studied cancers because of studies of hereditary cases. In the USA such cases account for between 10% and 30% of colorectal cancers (National Cancer Institute [129]).

Cancer has its origins in the cell nucleus, in DNA. When dangerous mutations in DNA accumulate, cancer results from unregulated cell proliferation (Munro et al. [122]).

There are two theories concerning the genesis of colon cancer: the *stochastic* theory and the *cancer stem cell (CSC)* theory [144]. The stochastic theory postulates that all cells in the tumour have the potential to proliferate and that tumour growth results from the accumulation of cell division in randomly selected cells. In contrast, the CSC theory holds that the colon cancer tumour is hierarchically organised and only a proportion of cells are capable of supporting tumour growth. In their survey paper on CSCs [96] state:

"According to the CSC hypothesis, it can be assumed that the first mutational hit occurs in a colonic stem cell located at the crypt bottom that, being long lived, can accumulate oncogenic mutations over years or decades. Once transformed, mutated stem cells can divide symmetrically or asymmetrically giving rise to other CSCs and progenitors, which in turn generate other cancer cells devoid of self-renewal ability. Eventually, the entire niche will be colonised by mutant stem cells, and the crypt will be filled with their progeny. The proliferating cancer cells will be subjected to further changes that may result in the progression of cancer."

The CSC theory postulates that when the entire crypt is full of cancer cells or their progeny - an event called *monoclonal conversion* has occurred. In recent times evidence has been accumulating for the CSC theory [7]. The importance of CSCs was highlighted in the study by [122]. Five stem cell markers were all detected in CRC tumours and found to be associated with tumour grade. It was possible to discriminate between normal tissue, low grade adenocarcinomas and high grade carcinomas using the markers.

2.2 Colorectal Cancer and Pathology - Biopsies and Tumour Specimens

A systematic discussion of the pathologic procedures connected with colorectal cancer can be found in [55]. When a patient presents with indicative symptoms or as a result of screening and cancer is suspected they may be subjected to a colonoscopy in which a surgical instrument is used to examine the interior of the bowel. The examining physician snips any suspicious polyps off and sends the tissue biopsy off to the pathology laboratory for examination. If the decision to operate is made, and the tumour is removed from the patient it is sent to the pathology laboratory for processing and examination.

The glass slides used in histopathology are produced by a sequence of standard operations. The first step is to place the tissue in a chemical fixative such as formalin in order to prevent decay. The sample is then progressively dehydrated with the use of alcohol, before being infiltrated with molten wax, then cooled. Finally, the wax block is sliced with a piece of equipment called a microtome (Figure 2.6) and selected tissue sections are placed on glass slides. Manual microtomes require skilled operators, although more recent models have automated settings.

Figure 2.6: Microtome (Veterinary Pathology [178])

The tissue sections are nearly transparent in appearance, and for purposes of examining cellular structures they are stained with reagents that react differently with different molecules, hence cellular components. In the most common type of staining, H&E staining, a combination of *haemotoxilyn* (H) and *eosin* (E) is used. Haemotoxilyn stains nuclear material dark blue, and eosin stains cytoplasm and extracellular connective tissue pink. A cover slip is placed over the stained tissue slice and the glass slide is stored for later analysis. Subsequently the slides are referred to as H&E slides.

Pathologists examine the slide under the microscope, and prepare reports for the rest of the medical team. These findings feed into decisions concerning treatment options: decisions to be made by the clinical team responsible for the patient. Experts extract useful information rapidly in these routine examinations.

Procedures for the production of digital images from tissue sections are well-defined, but differences in materials and manual operations can result in substantial variations in the appearance of the images, posing significant challenges for automated methods: hence the discussion here.

The pathologist examines the tissue sections under the microscope. Where the microscope includes a camera, the slide may be photographed and the resulting image stored, alongside other information concerning the patient.

Figure 2.7: Microscope (Veterinary Pathology [178])

2.3 The Pathology Report

The pathology laboratory's post-operative report is used for diagnosis and prognosis, and to inform decisions regarding future treatment. Table 2.1 outlines the structure of such a pathology report - information kindly provided by Ayesha Azam [11]. The report outlined here is applicable to colon cancer, issued after laboratory analysis of a surgically extracted tumour. Note that besides items obtained by microscopic examination, other information is also present, some of which is provided by sources outside the laboratory: for example the radiology department usually contributes information on metastasis.

Note that the items in part A (clinical information) include demographic variables and the patient's clinical history:

"The clinical history includes the location of the tumour in the body, because it is known that left-sided and right-sided cancers have different behaviours." [11]

Part B records details of the surgical procedure.

In Part C, the results of macroscopic examination (measurements taken using the naked eye) are recorded.

Part D includes the results of microscopic examination. According to [11]:

"Part D includes our findings based on *histology*, the examination of slides under the microscope (or digital images). Mostly in the form of a synoptic report which includes items (a) to (j)."

If the growth is confirmed to be a colorectal cancer it is categorised: there are two main categories *adenocarcinomas* and *mucinous adenocarcinomas* as well as rarer types of cancer such as *signet ring* cancers. This standard WHO classification is recorded as item (a).

Part E is the final diagnosis, including the TNM components, which are described in Subsection 2.6.1.

Most of the fields in TCGA clinical data are standard and map directly onto the structure in Table 2.1.

Various components of the pathology report are discussed in more detail in later sections of this chapter.

Table 2.1: Structure of Pathology Report - After Ayesha Azam [11]

Part	Description	Comments
A	**Clinical information**	
a	Patient Details	e.g. patient name, age, gender, hospital ID, lab ID, NHS number, date specimen sent.
b	Clinical History	Can include information about previous biopsy diagnosis, site of tumour, preoperative stage of the tumour, any pre-operative chemotherapy or any other co-morbidities
B	**Specimen Details**	
a	Type of procedure	e.g. right hemicolectomy or other types
b	Nature of specimen	e.g. small bowel, large bowel, caecum
C	**Macroscopic description**	
a	Site of tumour in the specimen	
b	Maximum tumour diameter	
c	Distance to the nearest resection margin	Longitudinal and circumferential
d	Any tumour perforation	As it will upstage the tumour
D	**Microscopic description**	
a	Tumour type	WHO classification. Such as: adenocarcinoma, mucinous carcinoma, signet ring carcinoma
b	Tumour grade	Higher grade related to poor prognosis
c	Extent of local invasion	Helps decide pT
d	Distance of the tumour from resection margins	To assess whether the tumour has been completely taken out or not
e	Any vascular/peri-neural/ lymphatic invasion	Adverse prognosis indicators
f	Tumour deposits	Status predicts prognosis
g	Tumour budding	
h	Lymph nodes examined	
i	Metastasis	Any histological evidence of distant metastasis.
j	TNM stage	Recorded as pT, pN, pM
E	**Diagnosis**	pT, pN, pM

2.4 The Slide under the Microscope

In histology images there are many objects that pathologists have identified as useful biological entities [62]. Under the microscope these are recognisable at varying resolutions. For example at coarse resolutions regions of tumour, *stroma*, and fat are visible.

Regions of tumour are usually purple in colour with lighter-coloured tree-like structures traversing them. *Stroma* is tissue which supports the functioning cells in an organ and is light pink in colour with cells contributing to its grainy texture. Fat is usually whitish in colour. Note that the term *Tumour Microenvironment (TME)* is often used interchangably with the term 'stroma' to emphasise the role of non-cancer cells in the vicinity of tumour cells (Chen and Song [34]).

At higher magnifications, crypts or distorted variants can be distinguished, as illustrated previously in Figure 1.1. The appearance of crypts is important in cancer grading, discussed below.

As the microscope's magnification is increased individual cells come into view [62]. For example, in digitised slides from TCGA, at maximum magnification each pixel is nominally 0.25 micrometres in width. This magnification is termed *40X* . The magnification of *20X* has pixels which are 0.5 micrometres in width. (Note that these numbers are approximate, and in the TCGA data, the actual pixel widths vary by a few percent from the nominal value)

In this thesis we concentrate on *cells*, the loci of most biological activity. Under the microscope many different types of cell are visible, but in practice the four categories of cells mentioned in the introductory chapter are adequate for many purposes. The following subsections discuss these in more detail.

2.4.1 Epithelial Cells

Epithelium is one of the four basic tissue types in the human body [62]. (The others are connective tissue, muscle and nervous tissue.) Epithelium is made up of epithelial cells and is present as sheets of contiguous cells such as skin or as glands. Crypt boundaries are composed of epithelium.

There are three principal shapes of epithelial cell: *squamous*, *columnar*, and *cuboidal*. These can be arranged in a single layer of cells as simple epithelium or in layers which are two or more cells deep. Functions of epithelial cells include secretion,

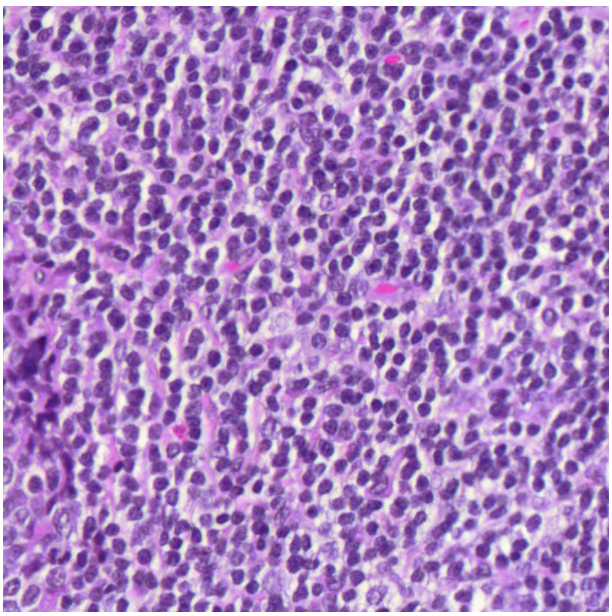

Figure 2.8: Tissue Section Showing Lymphocytes - Dark circular nuclei are surrounded by bright rings (AA-3864, tile 1504)

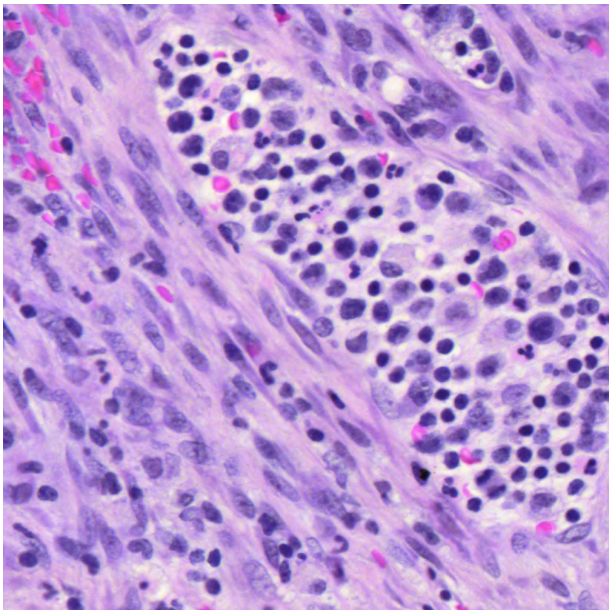

Figure 2.9: Tissue Section Showing Fibroblasts - Long spindle-shaped cells surrounded by pink staining (AA3543, tile 1067)

selective absorption, protection, transcellular transport and sensing.

As already discussed in the introduction, in normal tissue, the crypts are regular and well-defined. Figure 2.2 is of normal tissue. It is a longitudinal view of

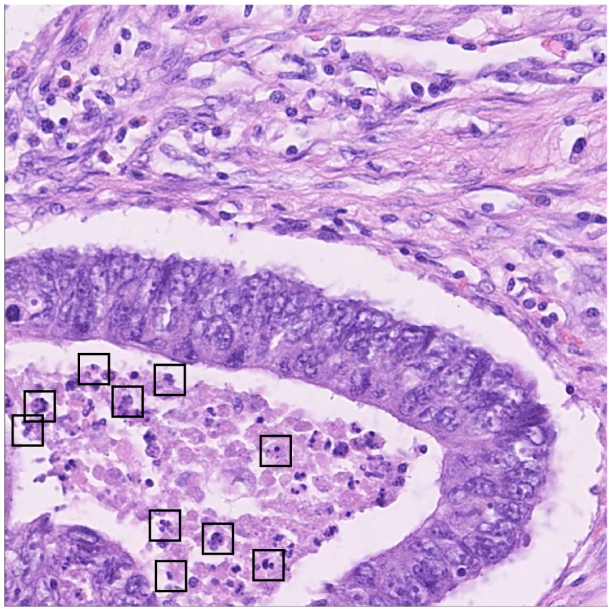

Figure 2.10: Tissue Section Showing 'Other' cells ('Cell' Training set: Image 86)

crypts while Figure 2.4 is a transverse view of normal tissue containing crypts. In a colon tumour, it is usually, though not always, possible to make out structures that resemble crypts (Figures 2.13 and 2.14).

2.4.2 Inflammatory Cells

White blood cells are associated with inflammation which is the body's natural response to cancer. *Lymphocytes* are inflammatory cells that are produced in lymph nodes. Lymphocytes are darker than other cells, with dense round nuclei. According to [62]:

"The peripherally situated cytoplasm stains a light blue and contains azurophilic granules."

Inflammatory cells are mostly less than about $10\mu m$ in diameter. A region occupied nearly entirely by lymphocytes is shown in Figure 2.8.

2.4.3 Fibroblasts

Fibroblasts are spindle-shaped cells found in the stroma. Because the image is a two-dimensional section of the tissue, a fibroblast may appear elliptically shaped in the image. Figure 2.9 displays fibroblasts. They appear as long purplish-grey cells against pink-coloured stroma. Note that Figure 2.9 also contains a region occupied by inflammatory cells similar to those in Figure 2.8. Fibroblasts are discussed in

more detail in 2.6.6.

2.4.4 'Other' Cells

In the analysis described in later chapters, the category 'Other' is used to assign a category to cells not already classified as one of the three main types. In this analysis, these cells comprise a small percentage of all cells: about 2%. Figure 2.10 contains 'other' cells. Ten 'other' cells are highlighted, surrounded by black squares. They are mostly dark coloured like inflammatory cells, but in Figure 2.10 they generally are smaller, and the surrounding 'halo' is not present.

2.5 Grading

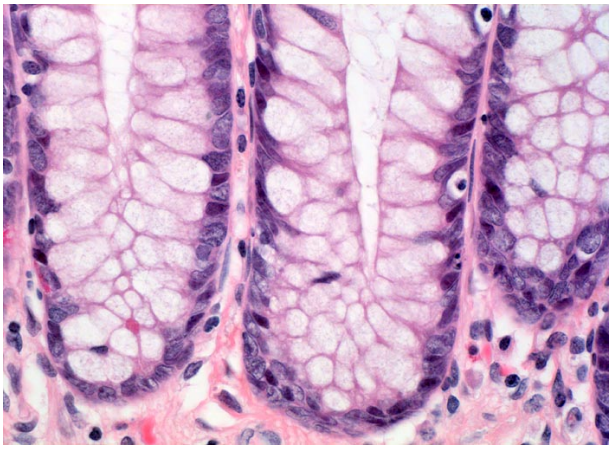

Figure 2.11: Normal colon tissue

It is generally accepted that the appearance of tissue under the microscope is a significant predictor of outcome, and the examining pathologist routinely records the appearance of the biopsy material or surgical material as the tumour *grade*. In normal tissue the crypts are easily visible, and their structure is orderly. It is easy to assign cells to their type. In contrast, in late-stage cancer, the crypt structures are not discernible, or, if they can be made out, they are very distorted.

Grade is assigned a value on a discrete scale by the pathologist. Cancers that resemble normal tissue are graded as *well differentiated* or equivalently as *low grade*. The image in Figure 2.12 is an example of a low grade cancer. The crypts are clearly visible and well formed. Low grades are associated with less aggressive cancers and good outcomes for patients.

At the other end of the scale are cancers that look highly abnormal, *high grade* cancers. Figure 2.14 is a region sampled from a high grade whole-slide image. Crypts are visible, but deformed in appearance. The WHO grading scheme of 2010 defines three categories of differentiation: *well differentiated*, *moderately differentiated* and *poorly differentiated* [23]. An example of moderately differentiated tissue can be seen in Figure 2.13. The labels are defined in terms of the percentage of the image with gland formation. Swapping between categories when referring to them can be confusing, because the *maximum* value of one labelling set, say 'high grade' corresponds to the *minimum* value of the other: 'poorly differentiated'.

The latest WHO guidelines [126] use the following grades:

Table 2.2: WHO Grading Guidelines

Grade 1	Well Differentiated
Grade 2	Moderately Differentiated
Grade 3	Poorly Differentiated
Grade 4	Undifferentiated

In the UK the WHO grades are routinely combined, with Grades 1 and 2 together and Grades 3 and 4 together [24].

Manual procedures are slow and the time of a fully-trained pathologist is expensive. In contrast, digital image processing is fast and cheap to operate. If digital pathology can capture useful relationships between computable features of the histology image and clinical variables of interest then these relationships can be used predictively. Thus if a computer can successfully extract information from a tissue sample then automation has clear benefits beyond mere data capture and storage.

Unfortunately the assignment of tumour grade is a subjective process, resulting in low levels of both intra-observer agreement and inter-observer agreement. Chandler and Houlston [30] report on a study in which 104 consultant pathologists (sampled from the register of the UK College of Pathologists) examined twenty images of CRC. Intra-observer agreement was measured by asking respondents to grade the same image on two separate occasions. Both three-grade (as described in an earlier WHO classification scheme) and two-grade classification schemes were used. There was substantial intra-observer agreement for both systems (two-grade system $\kappa = 0.809$, three-grade system $\kappa = 0.704$). However, the level of agreement between the pathologists was low : (two-grade system $\kappa = 0.358$, three-grade system

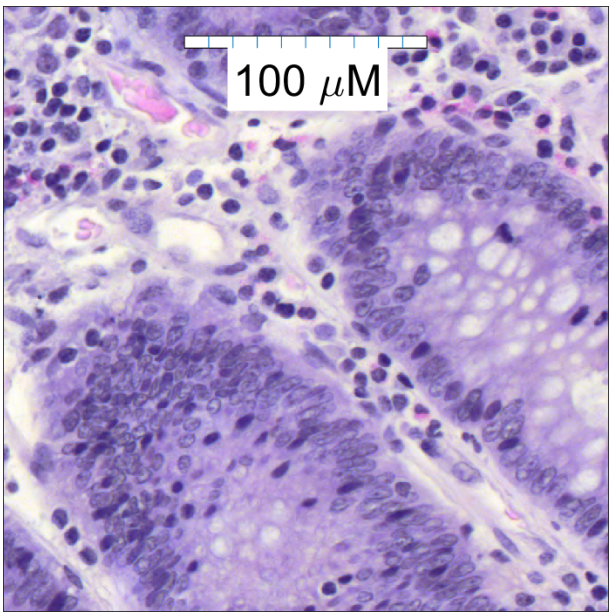

Figure 2.12: Example of a low-grade (well differentiated) colon cancer (TCGA COAD: Patient AA-3845, tile 1412)

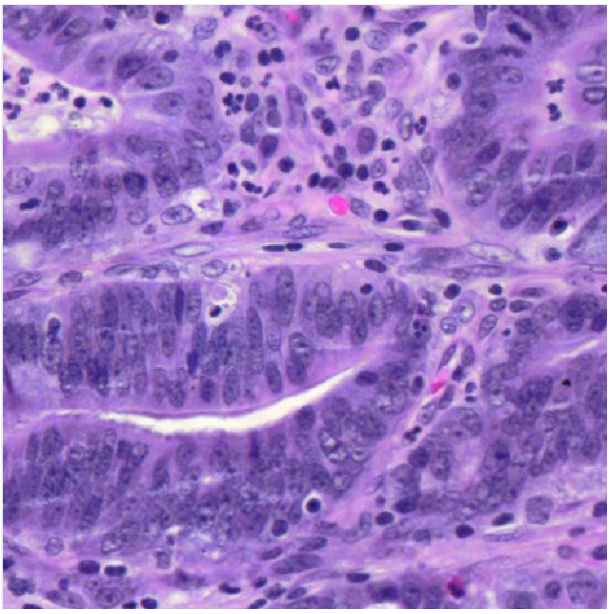

Figure 2.13: Patch from an intermediate grade (moderately differentiated) colon cancer (TCGA COAD: Patient AA-3543, tile 854)

$\kappa = 0.351$). The most disagreement occurred for moderate differentiation. The individual κ scores were: 0.467, 0.255, 0.358 for the well-, moderately- and poorly-differentiated categories.

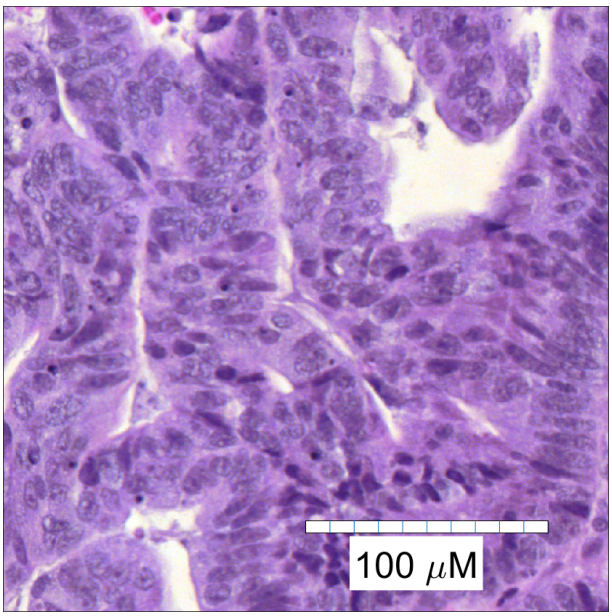

Figure 2.14: Example of a high grade (poorly differentiated) colon cancer (TCGA COAD: Patient AA-A02J, tile 3169)

Hassan et al. [78] report a significant relationship between grade and outcome ($P < 0.05$) in a meta-analysis that pooled thirty-one studies with 1,900 patients in all.

2.6 Pathological Indicators in Colorectal Cancer

In addition to cancer grade many other pathological indicators have been studied. Some of these are recorded routinely, while others are still the subject of research.

The American Joint Committee on Cancer (AJCC) recording system defines standards for the fields in pathology reports. It is updated every few years as new research findings become available [31], and thus can be regarded as embodying those indicators which are generally accepted as significant.

For example, Version 8 of the AJCC recording system for CRC included lymphovascular invasion, both small and large, as well as genomic markers relevant to precision medicine. Microsatellite Instablity (MSI) status, a molecular measure, was added to the prognostic factors for use in clinical care. Note that the TCGA study found that MSI status was a significant genomic marker [167].

In this section we describe various histopathological characteristics which have

been studied. These features have been identified as being significant for diagnosis, prognosis and treatment decisions. In most cases the features were initially identified using manual pathology - for example cells may be counted under the microscope. In other cases digital pathology has also been used, for example, in the study of tumour-infiltrating lymphocytes.

2.6.1 TNM Staging

One of the aims of medical reporting is to produce reports which can make useful predictions. The patient's report should be *reproducible*: ideally the same report should result, no matter which laboratory or which personnel contribute.

Defined by the American Joint Committee on Cancer (AJCC), the TNM staging system is the standard system for describing cancer. The TNM version for colorectal cancer has three stages (T:tumour, N:node and M:metastasis). The *T stage* records the extent of the tumour. It indicates how far the cancer has grown into the wall of the colon or rectum. The degree to which the cancer has spread to nearby lymph nodes is recorded in the *N stage*. The *M stage* relates to metastasis: whether or not the cancer has spread to distant organs or glands.

In practice, in cases of colorectal cancer, as well as the cancer grade, *TNM stages* are also reported as noted in Table 2.1. In Table 2.1 the T, N, M stages are output as part of the final diagnosis, underlining their importance. A single TNM stage may be reported by the pathologist, recorded as a combination of the T, N, and M stages: a summary of the pathological indicators. Corresponding to increasing severity, the stages used in the TCGA clinical data are recorded as 0, I, IIA, IIB, IIC, IIIA, IIIB, IIIC, IVA, IVB and IVC.

In the pathology report structure shown in Table 2.1 the T, N and M stages are recorded separately. The T stage is obtained through the assessment of any tumour invasion of the bowel wall. Ayesha Azam [11] remarks:

"The degree of tumour invasion into or through the bowel wall will help us decide the pT stage."

As for the N Stage, [11] remarks that regarding lymph nodes:

"the number examined and how many involved helps us to define the N stage of the tumour, higher number of involved LN nodes has been found to be associated

with higher risk of tumour recurrence." (ibid.)

Concerning the M stage:

"Pathologists can only base assessment of distant metastatic disease on submitted specimens. Sometimes the included specimen does include certain aspects that would qualify for the criteria of distant metastasis. However, in most cases we are unable to record the M stage in the histology report and this information comes from the radiology data (e.g. distant metastasis in liver)."

2.6.2 Mucin

In the case of colorectal cancers, the presence of *mucin* is an important factor. Mucin is produced as part of normal bowel functions and carcinomas with an excess of mucin are recorded as *mucinous carcinomas* in the TCGA COAD data set. Mucinous carcinomas are associated with varying outcomes and with other symptoms. Symonds and Vickery Jr [163] found that in 132 of 893 colorectal cancer cases the tumours were mucinous carcinomas and were associated with poorer outcomes. In a study of 6,475 patients Park et al. [137] found various differences between mucinous and non-mucinous carcinomas. Patients with mucinous carcinomas were younger, had larger tumour size and later T stage. Five-year disease-free survival was lower: 76.5% versus 83.2% (p=0.008) as was five-year overall survival: 81.4% versus 87.4% (p=0.005). In the analysis of colon cancer mucinous histology was an independent factor (p=0.026).

2.6.3 Venous, Lymphovascular, and Perineural Invasion

Venous, lymphovascular and perineural invasion are all associated with poor outcomes and their status is recorded in the TCGA clinical data used in this thesis.

Dawson et al. [41] report:
"Venous invasion is a surrogate marker of the risk of metastatic disease."

In addition the detection of venous invasion in Stage II CRC:
"may prompt oncologists to consider adjuvant chemotherapy."

2.6.4 Presence of Epithelial Cells

The number of (epithelial) *tumour* cells was found to be associated with improved survival by West et al. [182].

2.6.5 Tumour Infiltrating Lymphocytes

Regarding the different types of cells, inflammatory cells have long been recognised as important cancer battlers, and their presence is indicative of the on-going fight within the body.

Tumour-infiltrating lymphocytes (TILs) are inflammatory cells implicated in killing tumour cells. Their presence is often associated with better clinical outcomes (after surgery or immunotherapy). Galon et al. [59] found that TIL concentrations predicted clinical outcomes in colon cancer. Denkert et al. [47] found that the increased concentration of TILs was associated with improved survival in HER2+ and TNBC breast cancer, but in luminal HER2- breast cancer TILs were negative for survival. In Figure 2.15 round lymphocytes are clearly visible, both inside and outside the epithelium.

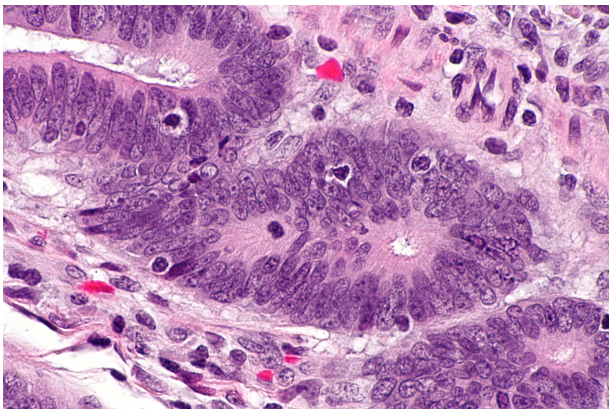

Figure 2.15: Tumour-Infiltrating Lymphocytes (From TCGA COAD)

2.6.6 Fibroblasts and Stroma

Fibroblasts are associated with the body's response to injury: the process of wound healing. Fibroblasts are the most common cells in stroma (Section 2.4), the connective tissue found in organs as opposed to the functional tissue *parenchyma*. The role of fibroblasts in cancer progression and metastasis is complex: "with both cancer-promoting and cancer-restraining actions" according to Kalluri [92] who refers

to fibroblasts as the "cockroaches of the human body".

Normally injury to tissue results in a wound healing response: the formation of scar tissue. The growth of stroma in scar tissue formation is referred to *desmoplasia* [82]. The response to cancer is similar and is called cancer fibrosis or cancer stroma. The immune cells, capillaries, basement membrane, activated fibroblasts and *extracellular matrix (ECM)* surrounding the cancer cells constitute the tumour stroma [92]. Tumour cells interact with the stroma which is referred to as the *tumour microenvironment (TME)*. The interaction is two-way: tumour cells can influence the chemistry of fibroblasts which in turn may change the behaviour of the tumour cells.

Cancer Associated Fibroblasts (CAFs) form a dominant component of the tumour stroma and are believed to play an important role in tumour progression. Qian et al. [140] analysed data from patients undergoing screening colonoscopy and concluded that higher levels of circulating fibroblast growth factor 21 (FGF21) were associated with increased risks of disease. In an earlier study Tommelein et al. [170] found links between CAFs and metastasis. In particular they concluded that positive expression of CAF-related genes was "significantly correlated with distant recurrence and poor probability of recurrence-free and overall survival".

A recent review by Chen and Song [34] described the role of CAFs in cancer pathology. According to the authors current cancer treatments often fail because the TME surrounding tumour cells may prompt relapse and therapeutic resistance. The mechanisms favouring tumour progression involve cell to cell contact and CAFs are believed to be responsible. According to [34]:

"Mechanistically CAFs build up and remodel the *extracellular matrix (ECM)* which enables the tumour cells to invade through the TME ... CAFs are larger, harbour multiple branches of cytoplasm and have indented nuclei under light microscopy."

There are numerous potential sources of CAFs as well as normal fibroblasts, including mesenchymal stem cells. At least six are mentioned by [34]. Correspondingly, although there are many biological markers for CAFs none of them uniquely identifies CAF subpopulations [34]. Tumour cells reprogram normal fibroblasts into CAFs using molecular mechanisms such as hypoxia, activation of active growth factors and epigenetic. In turn, CAFs can turn normal epithelial cells into tumour cells causing tumorigenesis, promote the formation of blood vessels in the tumour,

and cause metastasis.

There are many molecular markers for CAFs. According to Han et al. [76]: "Traditional CAF biomarkers such as α-smooth muscle actin (αSMA), g activation protein (FAP), S100A4, platelet-derived growth factor receptors (PDGFRα/β) or vimentin have been well-studied despite none of them (being) specific to CAFs."

Fibroblasts have been proposed as biomarkers for diagnosis and prognosis in CRC. Tsujino et al. [172] reported that the abundance of myofibroblasts in the tumour stroma was an indicator of disease recurrence. Vitamin D expression from CAFs was also positive for survival in [34].

2.6.7 Tumour Budding

A tumour bud is a "single cancer cell or cell cluster of up to four tumour cells". Tumour budding has recently been included as an additional prognostic factor in the Union for International Cancer Control's TNM classification and in guidelines issued by the College of American Pathologists (Zlobec et al. [191]). Tumour budding was not in UK guidelines as of May 2020 [121].

According to Ayesha Azam [11]:
"There is considerable interest in the phenomenon of tumour budding at the advancing edge of colorectal cancers. There is some published evidence that presence of tumour budding can help predict the risk of metastatic spread in early stage cancers). "

Tumour buds are associated with poor patient outcomes. Lang-Schwarz et al. [114] analysed 576 low-grade CRCs and included tumour buds in a metric that they found to be good for prognosis. Unfortunately they do not quote the regression coefficients of their statistical model and the contribution of tumour budding to the metric is not quoted.

2.6.8 Poorly Differentiated Clusters

Poorly differentiated clusters (PDCs) are "malignant clusters with five or more cells lacking glandular differentiation" [175]. Studies indicate that the presence of PDCs strongly predicts lymph node metastasis and therefore can be useful in decision-making about treatments in early stage colorectal cancer.

2.6.9 Region of Submucosal Invasion

Toh et al. [169] studied 207 pT1 colorectal cancers, looking for high-risk features associated with lymph node metastasis. Lymph node metastasis was noted in 19 patients (9.2%). These cancers had a significantly wider area of invasion ($p = 0.001$) and greater area of submucosal invasion ($p < 0.001$) compared with pT1 stage cancers without lymph node metastasis. Differentiation and vascular and lymphatic invasion were also significant predictors of lymph node metastasis ($p < 0.0001$, $p = 0.039$, and $p = 0.018$). Submucosal measures were good predictors, but patient numbers were small in this study.

2.6.10 Serrated Carcinomas

Figure 2.16: Serrated Adenoma (Wikimedia user Nephron [184]).

Another useful morphological feature is the presence of epithelial material whose section has a *serrated* appearance. It has been demonstrated by [91] that "sessile serrated lesions" found in the colon are associated with a distinct molecular

pathway to colorectal cancer. These lesions develop into serrated adenomas (polyps) and eventually into serrated carcinomas. Of various molecular groupings associated with mutational status, Group I, defined by [91] is CIMP-High, MSI-High and associated with silencing of the mismatch repair gene, MLH1. The group examined approximately two hundred CRC carcinomas and found that approximately 20% of them were hyper-mutated. They were strongly associated with CIMP-High status, MSI-High status and MLH1 silencing. This implies that these carcinomas are associated with the serrated pathway [167].

Serrated carcinomas are believed to develop from serrated adenomas (Figure 2.16). In an analysis of colorectal cancer data Felipe De Sousa et al. [54] found three distinct subtypes of cancer distinguished by their molecular profiles. The group that was associated with poor prognosis was characterised by histological images that were poorly differentiated. The term *serrated pathway* refers to the steps by which serrated adenomas change into carcinomas.

2.7 Molecular Aspects of Colon Cancec

For cancer to develop from polyps and adenomas it is necessary to disrupt the molecular pathways that prevent its development. By studying familial cases of CRC, various oncogenes and tumour suppressor genes associated with mutations were discovered, in particular APC, KRAS and P53 [53].

Molecular findings for the TCGA colon and rectal data sets were published in [167]. It was found that 16% of colorectal cancers were hypermutated. Hypermutation was strongly associated with MSI-high status (*microsatellite instability*) and *CIMP*-high status (CIMP:CpG island methylator phenotype). Additionally, it was associated with suppression of the mismatch repair protein MLH1. These are the characteristics of consensus molecular group 1 described by authors such as Jass [91] and Guinney et al. [73].

Board [19] contains a discussion of the genetics of colorectal cancer. Much effort has been devoted to the understanding of the molecular pathways which underlie carcinogenesis. Aims here are the discovery of both all-purpose therapies and also targeted therapies. For example in the treatment of colorectal cancer the use of anti-EFGR drugs is common, but these drugs are not effective for all cancers.

Guinney et al. [73] describe a large study which aimed to establish consensus molecular subtypes among different research groups. They remark:

"All groups identified one tumour sub-type enriched for microsatellite instability (MSI) and one subtype characterised by high expression of mesenchymal genes, but failed to achieve full consistency among the other subtypes."

Their analysis (six subtyping algorithms and 4,151 patients) found four molecular subtypes, listed as consensus groups CMS1, CMS2, CMS3 and CMS4.

Regarding CMS1:

"CMS1 encompassed the majority of MSI tumours and had over-expression of proteins involved in DNA damage repair in reverse phase protein array (RPPA) analysis, consistent with defective DNA mismatch repair."

"CMS1 is characterised by increased expression of genes associated with a diffuse immune infiltrate, mainly composed of TH1 and cytotoxic T cells, along with strong activation of immune evasion pathways, an emerging feature of MSI CRC1."

This consensus group had a population frequency of 14%. Tumour characteristics included a high infiltration of immune cells and an average level of stromal infiltration.

The CMS2 group was the most common consensus type: 37%. There were low levels of stromal infiltration:

"We detected more frequent copy number gains in oncogenes and losses in tumour suppressor genes in CMS2 than in the other subtypes."

"CMS2 tumours displayed epithelial differentiation and strong up-regulation of WNT and MYC downstream targets, classically implicated in CRC carcinogenesis."

The consensus subtype CMS3, was 13% of the population. There was a low/average infiltration of immune cells accompanied by a low level of stromal infiltration:

"Notably, CMS3 samples had a distinctive global genomic and epigenomic pro-

file as compared with other CIN tumors: (i) consistently fewer SCNAs an association not explained by differences in tumor purity; (ii) nearly 30% were hypermutated, which overlapped with MSI status; and (iii) higher prevalence of CpG Island Methylator Phenotype (CIMP) low cluster in TCGA samples, with intermediate levels of gene hypermethylation."

"Enrichment for multiple metabolism signatures was pronounced in CMS3 epithelial CRCs, in line with the occurrence of KRAS activating mutations described as inducing prominent metabolic adaptation. CMS3 tumors displayed similarities with a 'metabolic', genomically stable subtype recently described in gastric cancer".

Consensus type CMS4, with a frequency of 23% had an average/high infiltration of immune cells and a high level of stromal infiltration.

"CMS4 samples exhibited a gene expression profile compatible with stromal infiltration, ... and higher admixture with non–cancer cells."

2.7.1 Immunohistochemistry

Immunohistochemistry is the experimental discipline of of localising proteins and other molecules in tissue sections using labelled antibodies and markers. These are applied as dyes. Some dyes are designed for the visible sprectrum while others fluoresce under ultra-violet light.

In visible light microscopy the resulting coloured precipitate colours the tissue where the molecules of interest are located. The reaction pattern is viewed under the microscope. According to [141]:

"interpretation of IHC results requires familiarity with the expected pattern of immunoreactivity based on location of the antigen in the cell of interest."

Galon and Lanzi [58] use immunohistochemistry in work which is based on the observation that presence or absence of T Cells (immune cells) is associated with patient survival. A biomarker is created by measuring the quantities of two types of T cells in the tumour. The antibodies used are CD3 and CD8. Concentrations of these cells are measured in two regions: the tumour core and the tumour margin. The biomarker is obtained as follows. Adjacent tissue slices from the microtome are stained, one slice with CD3, the other with CD8. Segmentation of the into

core and margin is done automatically, and stain intensity measurements result in four quantities which comprise the biomarker: $< CD3, margin > < CD3, core >$, $< CD8, margin >, < CD8, core >$. For TNM stages I, II and III the authors report that the biomarker predicts both disease-free survival time and overall survival time better than TNM staging.

In *fluorescence* microscopy fluorophores are added to antigens (the molecules that lock to entities of interest in the tissue).

Analysis has concentrated on localising proteins to particular cell components (usually in defined cell phenotypes), starting with isolated single cells and more recently progressing to images containing many cells. For an excellent overview of *locational proteomics*, the study of proteins in their locations, the reader is referred to [132]. Much work in this field has been carried out by Robert Murphy and colleagues at Murphy Labs where results from subcellular image analysis have been used to create models of cellular structure [25].

A good example of the application of machine learning to protein localisation is described in the report by Boland and Murphy [21]. Fluorescent images of HeLa cells were used to learn a neural network classifier. Given an image associated with a new protein, the classifier could extract a set of features from the image and carry out a calculation which assigned a specific organelle to the image (such as the nucleus, the nucleoli, the cytoskeleton, etc). The classifier is obtained from a training data set containing images and their class. Each member of the data set is an image consisting of exactly one cell and the cell has been treated with a protein which selectively associates with a particular organelle. The first step is to extract morphological features from the images. There are eight types of these *subcellular location features (SLFs)*, including geometrical features such as roundness, eccentricity, edge features such as brightness, homogeneity, texture features and wavelet features and more. If a parallel image of the cell DNA is available then extra features may be used such as the amount of overlap with the DNA region, and the average distance from the nucleus. 22 features were input to a neural network classifier. The authors found that the neural network performed well with test data and was superior to linear discriminant analysis, decision trees and kNN classifiers.

Huang and Murphy [85] describe how subcellular patterns may be recognised within an image containing many cells without using segmentation into single cells. Each image in the training set was generated synthetically by merging single Hela

cell images that belonged to one of ten major subcellular location patterns. A DAG Gaussian kernel was trained to recognise the location pattern of an image. The most discriminative features were selected by step-wise discriminant analysis and of the top fifteen discriminative features, ten are Haralick texture features which can be calculated without requiring cell segmentation.

Chen and Murphy [33] applied graphical models to subcellular location patterns in multi-cell images. It was assumed that the cells being sampled were individuals which resulted from the growth of an original population of ancestor cells. The ancestor cells were a mixture of the cell classes of interest and the descendant cells were clustered in regions. Segmentation was used to divide the image into individual cells. Two sorts of graphical model were used: the first type was the feature space model whereby cells in an image were from mixed populations which had not had time to proliferate. In second type of model the ancestor cells were mixed and daughter cells had had time to form and were clustered near each other. In each case a cell was assumed to belong to a class with a prior probability but the class was adjusted by the likely class probabilities of its neighbours. Priors were obtained using a support vector machine. Test and training data were created by aggregating single-cell data from the well-known HeLa data set. Good improvements were obtained: the classification error of the base classifier was decreased by about a third.

Feature vectors have been used to predict subcellular location in fluorescence images from the Human Protein Atlas [133]. Each data point consisted of four images - from protein, nucleus, micro-tubules and endoplasmic reticulum (ER). The protein image was obtained by immunofluorescence whereas conventional staining was used to obtain the other channels. Three different cell lines were used and 1,902 proteins imaged. As well as the morphological features mentioned above, the authors used features that linked images together - for example the correlation coefficient between the protein and nuclear channels was employed as a feature. The authors also used a modified watershed algorithm to segment the images into single cell regions: 29,099 of these. Two types of classifier were used: a support vector machine and a random forest model. Both classifiers gave similar results. In both cases the best features for classification included measures of interaction between channels, such as the Pearson correlation between DAPI and protein.

2.8 Digital Analysis of Histopathology Images

Pathologists routinely and rapidly extract useful information from histology slides. They summarise this information in their reports to the physicians responsible for treatment decision. Snead et al. [159] have examined how well these standard procedures are carried out using digital images. They conclude that digital pathology is as good as manual methods.

The next step is to use digital image processing to extract information that has clinical value. Broadly, there are two general approaches to this problem. The first approach is to use *hand crafted* algorithms, and the second approach is to use *deep learning algorithms* which use convolutional neural networks such as the one outlined in the introductory chapter of this thesis. Various hand crafted algorithms were discussed in Subsection 2.7.1, in the description of immunohistochemistry. Here we briefly discuss how hand crafted algorithms operate before proceeding to describe modelling with CNNs.

Hand crafted algorithms were used in early experiments with digital image processing in pathology. Researchers concentrated on choosing features that were likely to be useful. Feature selection could be guided by manual pathology research or from research in general image processing.

For many hand crafted algorithms prediction is carried out in two stages. The first stage extracts a set of selected feature values from an image while the second stage is a regression step which combines the features to produce the final prediction. This process was described in the introductory chapter of this thesis using Yuan [187] as an example.

In the first stage consider the set S containing J functions g_j:

$$S = \{1 \leq j \leq n_J : g_j\} \tag{2.1}$$

Each function g_j operates on the image I to compute the value $g_j(I)$ of feature j.

In the second stage of the algorithm a regression function h operates on the feature values and predicts the value of output variable y:

$$y = h(g_1(I), g_2(I), ...g_J(I)) \tag{2.2}$$

The overall model f is the *composition* of h and g:

$$y = f(I) = h(g(I)) \tag{2.3}$$

In hand crafting the broad aim is to find good features: that are reproducible, that have good predictive power and that can be explained in terms of known biology. For example, it is known from manual pathology that the presence of TILs is indicative of good survival, so it might be expected that a measure of TIL concentration, computed from the H&E image would be a predictor of survival.

The $g_j()$ are functions of the image intensity, *handcrafted* features which capture visual aspects of the image. Examples include textural features such as those defined by Haralick [77], statistics of 'colour intensity' and morphological features such as regions found by watershed algorithms (Preim and Botha [138]), often used to segment nuclei. The resulting statistical models often achieve a good fit between prediction and observation but they can be unstable to minor changes in input and fail to generalise to new data sets (Janowczyk and Madabhushi [90]). An advantage of hand crafting is that the parameters of the model can be interpreted as weightings that reflect the importance of the different features, but this can be offset by the need for specialists to make the selection of features which can be costly [90].

Examples of hand crafted models have already been discussed in this thesis. These include the fully automated model of TIL densities formulated by Yuan [187] and the partially automated model of West et al. [182] that calculates the abundance of epithelial cells and uses this feature to predict survival.

Deep learning models that do not require hand crafting are discussed in the next section.

2.9 Convolutional Neural Networks

As discussed in Section 1.2 deep learning models that rely on convolutional neural networks (CNNs) have become increasingly popular (Bera et al. [16]). In recent years there has been a strong trend towards using convolutional neural networks in digital pathology rather than hand crafted models.

In this section CNN models are introduced as mathematical formulae; a CNN model is the composition of a set of processes, or layers. Layers take different forms

and the main types of layer that comprise a CNN are introduced. It is not enough to specify the structure of a CNN, it is also necessary to use numeric constants in the formulae. The numeric constants, also known as *parameters* or *weights* are not specified in advance but instead are obtained by *training* the CNN. Training is usually done by labelling a set of images using manual marking then finding the values of weights that give a good fit between labels and the values that the CNN predicts. A discussion of techniques used in training is included in this section.

The term *model* was introduced in Section 1.2. Formally, a CNN may be regarded as a statistical *model*, a function f of observed data x and a set of parameters θ (equivalently, weights W). Calculation of f results in a set of predictions y (Efron and Hastie [51]). The notation used is:

$$y = f(x; \theta) \tag{2.4}$$

In a few cases the values of θ are available from physical or chemical knowledge, but in most situations they are obtained by *training* the model using data which contains cases where both x and y are known.

Convolutional neural networks are models which grew out of experimental knowledge of the operation of the visual cortex. LeCun et al. [115] designed an image processing model based on neural networks which successfully recognised digits from handwritten US Zip codes. The successful classification of images by Krizhevsky et al. [111] led to CNNs becoming hugely popular for image recognition. Note that CNNs are also used with other types of data such as temporal data, but the discussion here focuses on image data.

CNNs have proved very successful in extracting a wide variety of biological objects from pathology images. For example, [90] applied deep learning to seven different use cases including segmentation (into nuclear regions, epithelium, tubules and lymphocytes), detection (of mitosis events, invasive ductal carcinoma) and classification (lymphoma type). For an application of CNNs in histopathology see Xu et al. [186].

The widespread adoption of CNNs in digital pathology has been made possible by the use of *Graphics Processing Units (GPUs)*, processors originally designed for handling scenes in 3D games. GPUs specialise in array operations and are very suitable for image processing [3].

CNNs utilise spatial properties when processing an image. For example we expect that pixels that are close together should be treated together. The first stage of a CNN is a bank of *filters*: the image is broken up into small patches which are input to the filters, and each filter applies a set of weights to an incoming patch. In later layers the filter banks may assemble patches together, with the last bank outputting a set of features that contains enough information to predict the final output. This output may be a set of activation maps, maps of probabilities of objects of interest, a tag assigning a class to the object, or indeed one of many possible output types, such as survival predictions. CNNs are very flexible: they allow for complex non-linear transformations, and make no assumptions about which features of an image are significant: this means that modellers who are not domain experts can implement CNNs successfully.

The introductory chapter included a schematic of the CIFAR-10 network. Another influential network is 'Alexnet' which was introduced by Krizhevsky et al. [111]. Figure 2.17 shows a version of 'Alexnet' which was used by Janowczyk and Madabhushi [90] in an introduction to deep learning in pathology. The original 'Alexnet' network contains eleven computational *layers*, connected in series. There are five convolutional layers, labelled C1 to C5, three pooling layers labelled M1, M2 and M3, and three fully connected layers, labelled F1 to F3. The version used in [90] and shown in Figure 2.17 uses fewer layers - mainly because the images are much smaller and fewer parameters are needed in the model.

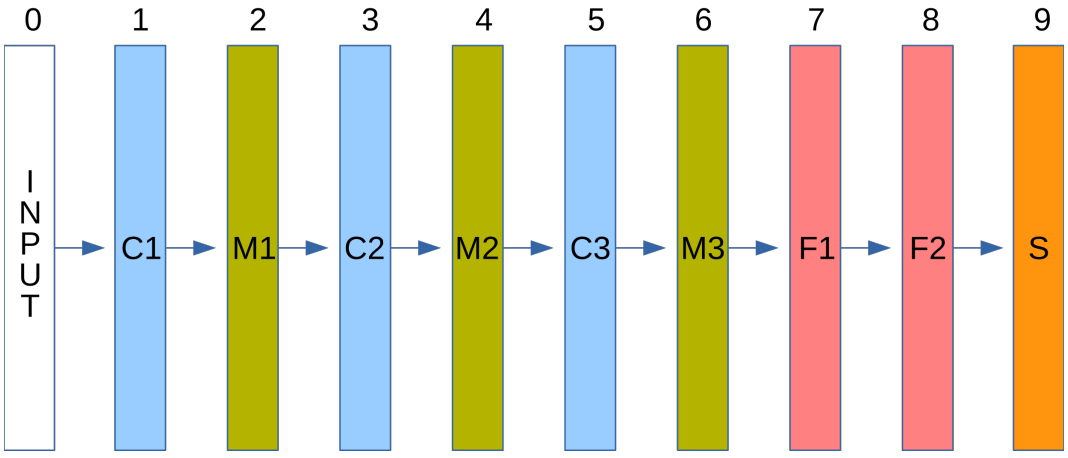

Figure 2.17: Layers used by Janowczyk and Madabhushi [90] adapted from 'Alexnet'

Now let us consider the flow of data through the network.

Assume that each layer l is associated with a function f_l. If there are L

layers connected in series then the CNN predicts output $y = f(x)$ where f is the composition of the f_l:

$$f = f_L(f_{L-1}(..., f_1(I; W_1)...; W_{L-1}); W_L) \tag{2.5}$$

Each f_l operates on the output of the previous transformation f_{l-1} and its output is fed into the next transformation f_{l+1}.

Each transformation is a mathematical formula containing a set of input values x and a set of constant values, the set of weights W_l applicable to layer l. For example, consider the first layer with transformation f_1, then x is the input image I. If I is an RGB image of height h and width w, then input x is instantiated as three arrays of size $h \times w$, I_{red}, I_{green} and I_{blue}. Considering colour $c \in \{red, green, blue\}$, then for a pixel $< i, j >$ in the image, at row i and column j, the intensity of the pixel is the array value $I_c(i, j)$.

$$I_c(i, j) = intensity\ of\ pixel < i, j > \in\ I_c \tag{2.6}$$

For the first layer, the expression is:

$$f_1 = f_1(I; W_1) \tag{2.7}$$

Note that layers do not have to be connected in series as in this example; they can be connected using more complex structures such as trees and recursive structures containing loops.

There are several types of layers in CNNs: convolution layers, pooling layers, rectified linear layers and fully connected layers.

2.9.1 Convolution Layers

In a convolution layer the input to the layer is an image (or a set of image patches) and the output is a collection of *activation maps*. Figure 2.18 illustrates the flow of data in the first convolution layer of 'Alexnet' in the original formulation by Krizhevsky et al. [111]. In this example the input is an RGB image, stored as three (227×227) intensity maps. The first step in constructing the convolution layer consists of defining a grid of points: the distance between points on the grid is known as the *stride*. The patch at each grid point is input to a bank of 96 *filters* (in the example the size of the patch is (11×11)). Therefore each filter is defined by a $(11 \times 11 \times 3)$ array of weights. At each pixel in the patch the product of the pixel

intensity [r, g, b] and the weight array $[w_g, w_g, w_b]$ defines an output pixel in the output patch. In the diagram, the size of the grid is (55×55), meaning that 3025 (55×55) output patches are created for each filter. The patches are stitched together, forming *activation maps* as output.

2.9.2 Pooling Layers

Pooling layers reduce the size of the activation maps. Contiguous patches in an activation map are grouped together and used to create new, smaller patches. Figure 2.19 shows *max pooling* (Nagi et al. [125]) being applied to four 2×2 patches. The maximum value in each patch is computed, and the four max values are used to form a new patch. In this example, the height and width of the activation map have been reduced by two. There are two important advantages of pooling. In the first place pooled patches contain information from larger regions in the original image than do the individual input patches so that the network gets this information at an earlier stage. Secondly, fewer weights need to be used in the next convolutional layer: the number of weights in successive stages is reduced. Both these properties of pooling reduce the training burden.

As well as the *max()* function described here, other aggregation functions may be used. These include *average* pooling where the average patch value is used, and *global* pooling. However Scherer et al. [150] conclude that, in general, max pooling is superior to other forms of pooling.

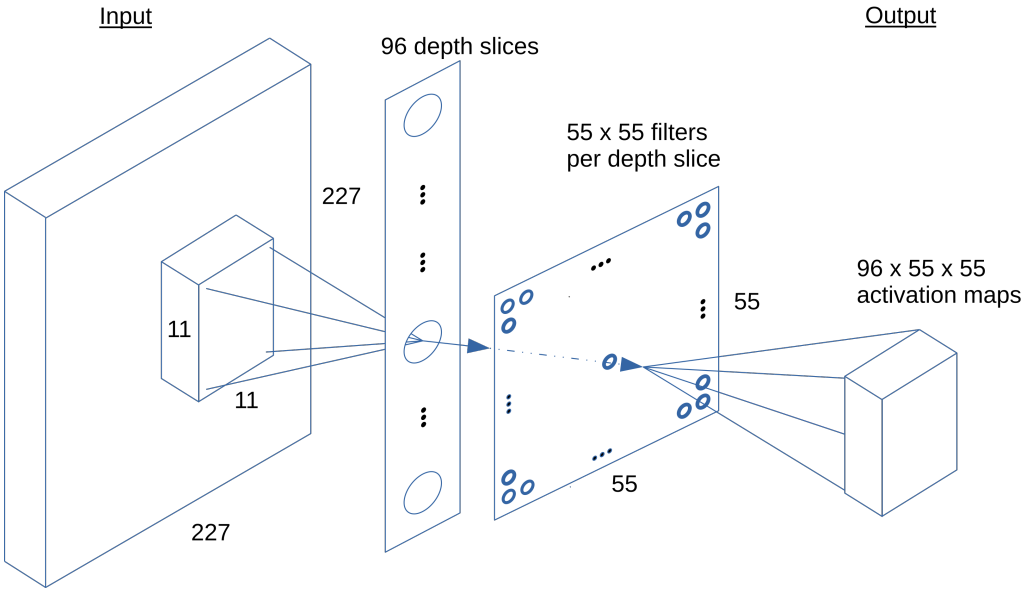

Figure 2.18: Convolutional Layer - First Layer with Image as Input

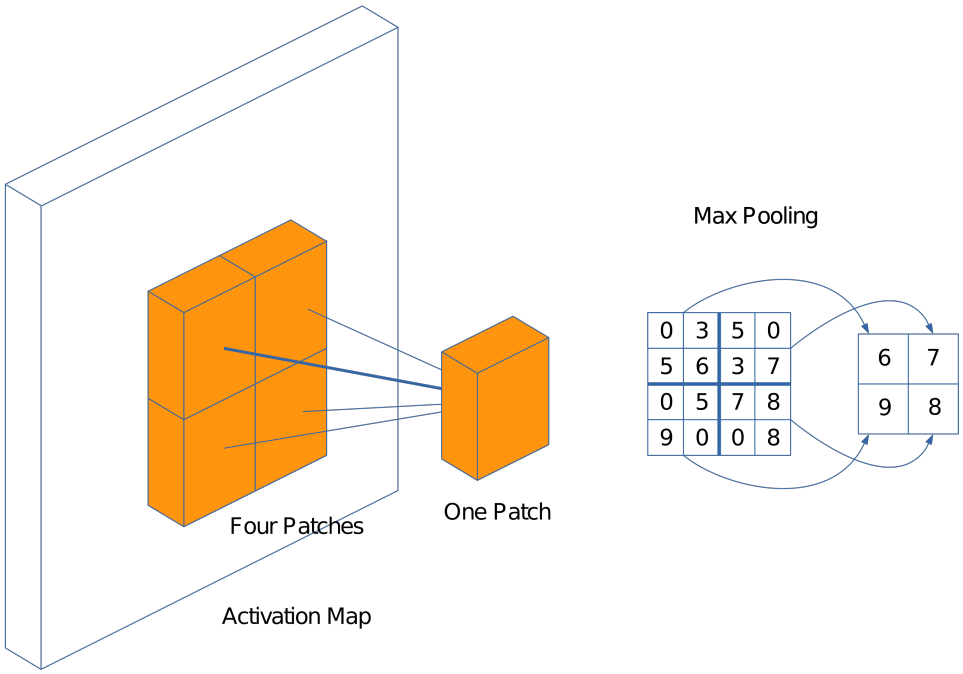

Figure 2.19: Max Pooling. Here each new patch is created using the maximum value in four patches. The righthand side of the figure shows pooling operating on sixteen patches to produce four new patches.

2.9.3 Rectified Linear Unit (ReLU) Layer

The usual approach is to employ layers in series: a mix of convolution, pooling and *Rectified Linear Units (ReLU)s*. ReLU units put non-linear transformations in the path followed by data as it travels through the network. The ReLU transformation is a threshold operation whereby every element less than zero is set to zero and elements greater than or equal to zero are left unchanged:

$$f(x) = max(0, x) \tag{2.8}$$

One advantage of ReLUs is that they improve network training [69].

2.9.4 Fully Connected Layers

The final stages of a CNN contain *fully connected layers* which predict the final result(s). It is assumed that the CNN, pooling and ReLU layers build a set of features which capture the implicit information in the image. These features are input to the final fully connected layers which may be instantiated by conventional neural networks. Note that other computational models may be hooked in here, such as

decision trees, or even logistic regression.

2.9.5 The CNN as a Composition of Layers

It is usual to group the convolutional, pooling and regularisation layers together during network construction, building the network in series. For example the network of interest may be represented as shown in Figure 2.17, the network used by Janowczyk and Madabhushi [90].

2.10 Deep Learning in Digital Pathology

Deep learning has been extensively used in digital pathology.

Janowczyk and Madabhushi [90] have written a tutorial introduction to deep learning in digital pathology with many examples. A recent review by Bera et al. [16] cites many applications of deep learning using histology images, in particular standard H&E images.

Deep learning in pathology usually includes the identification of biological entities or their surrogates. Biological entities that have been used include cells, TILs, cellular regions, glands and other regions of interest, such as stroma and tumour. For example [32] used CNNs for automatic lymphocyte detection. Bychkov et al. [26] describe the use of CNNs to classify regions in the image as either tumour or cancer and found that this classification could be used to predict outcomes.

In this thesis, in Chapters 3 and 4 deep learning is applied to the task of *cell identification*. In a given deep learning algorithm the job of cell identification may include several tasks. The *detection* task is that of finding cells - finding the locations of cells in an image. The *classification* task is to assign a cell type to an cell. Another approach to cell identification is to *segment* the input image: to find a set of non-overlapping regions such that each region is associated with a cell, and each cell is labelled with a region number.

In the thesis the focus is on two cell identification models. The first model is termed 'Cell' in this discussion. In 'Cell' the detection task is based on [158] while the classification task is based on CIFAR-10 [10] The second cell identification model is the 'Hovernet' model introduced by Graham et al. [71]. 'Hovernet' employs a CNN to segment the input image and outputs regions which are labelled with the cell type.

The 'Hovernet' training data included diagnostic images of colorectal cancer from TCGA and may be suitable for identifying colorectal cancer cells.

The next section of this background chapter briefly introduces the two models. The models are used as exemplars in describing the creation of data sets to be used in training which is discussed in Section 2.12 of this chapter. The models are also discussed in Chapter 3, which deals with sampling in whole slide images.

2.11 Deep Learning Models: 'Cell' and 'Hovernet'

The 'Cell' procedure is carried out using two convolution neural networks, applied in series, linked by an intermediate step. The first CNN detects nuclei while the second CNN, based on the CIFAR-10 algorithm, classifies a nucleus into a category defined in training. The intermediate step links detection and classification.

The second algorithm 'Hovernet' also has three components, but in contrast to 'Cell', a single cost function is used in training to estimate goodness of fit. This unified form of the cost function enables the model weights of the three components to be optimised simultaneously. The 'Hovernet' algorithm is a segmentation algorithm, taking a tile as input, and outputting a set of non-overlapping regions, each of which is predicted to contain a single cell of a certain type. Rather than work on small patches within a tile, the 'Hovernet' CNN integrates three branches, 'distance' , 'detection', and 'cell type' into a single predictive model. Some post-processing, namely boundary computation, is required in order to separate clumps of cells: the watershed algorithm is used here. The gradients of the distance functions show sharp changes at boundaries and this property of the distance map helps in calculating boundaries.

2.12 Training CNNs

Training is the process of choosing the best parameters (weights) for a prediction model. Training has been referred to in broad terms in Section 1.2; here it is explained in more detail.

The first step in the training process is to create a *training data set* D_{train}, a set of data for which the values of the dependent variables are known. D_{train} is of

the form:

$$D_{train} = \{1 \leq i \leq n_{obs} :< x_i, y_i^{obs} >\} \tag{2.9}$$

In Equation 2.9, n_{obs} refers to the total number of items in the training set, i refers to a particular item in the training set, x_i refers to the values of the input (predictive) variables and y_i refers to the value(s) observed for item i.

In many cases, human users create the training data sets needed for cell prediction. If the model is a *detection* algorithm, that is, if it predicts the locations of cells, then the user can view a training image on a computer screen and mark cells with the mouse. The exact method used depends on the type of model: for example the centres of cells may be dotted with the mouse as in 'Cell' or segment boundaries may be traced as in 'Hovernet'. Images with their sets of observed detection data are stored as training data.

In the case of a *classification* model which accepts an image of a cell as input and predicts the cell type, the human user can label such images with their cell type. Usually some expertise is required: ideally a trained pathologist should do this. The training data is the set of images tagged with their cell type assignments.

Considering cell classification the training data set is of the form:

$$D_{train} = \{1 \leq i \leq n_{obs} :< I_i, c_i^{obs} >\} \tag{2.10}$$

Here the model $f(I_i; W)$ is applied to each image I_i using the specified weights W. The model outputs a predicted value of the cell type c_i.

$$c_i^{pred} = f(I_i; W) \tag{2.11}$$

The list of predictions S^{pred} is denoted by:

$$S^{pred} = \{1 \leq i \leq n_{obs} : c_i^{pred}\} \tag{2.12}$$

The network is trained by finding the weights W for which the prediction list best matches the observation list.

To quantify the match between predictions and observations a *loss* function L is defined. The loss function compares predictions with observations and the aim of training is to minimise the loss function. In the case of *cell classification* a suitable loss function is the *cross entropy* loss function. Briefly, the cross entropy loss

function is the difference between the information needed to know the observations (the *ground truth*) and the information obtained by applying the model f to the input data. If the value of the cross entropy is low the difference is low and the loss function is low, which is the correct behaviour. Conversely, if the difference is high the loss function is also high.

To compute the classification loss function for an image the output of the last layer of the CNN is used, a vector z of logistic regression values. For example, if there are four classes, z is of the form:

$$z = [z_1, z_2, z_3, z_4] \tag{2.13}$$

The z values are converted to probabilities using:

$$q(z_j) = \frac{e^{z_i}}{\sum_j' e^{z_{j'}}} \tag{2.14}$$

Assume that the observed value for data item i is $j = c_i^{obs}$. This is converted to a binary vector v which is 1 in position j and 0 in the remaining positions. For example, if the cell type 2 ('inflammatory') is observed then the vector v is [0100]. The cross entropy loss function L_i for observation i is the scalar product of v and q:

$$L_i = -\sum_j v_j log(q_j) \tag{2.15}$$

And summing over all data points in the training set we obtain the training loss function:

$$L = \sum_i^{n_{train}} L_i \tag{2.16}$$

The network is trained by finding the minimum value of the loss function L with respect to W. This is done by *optimisation*, by calculating those weights which minimise L.

Note that it is assumed implicitly that images in D_{train} are a representative sample of the images for which predictions will be made. This is rarely spelled out in the literature and if the training images are selected from a small number of WSIs important cases may be missed. The 'Cell' algorithm was trained on ten WSIs and 16 WSIs were used to train the 'Hovernet' model. These are quite small numbers and do not guarantee that interesting cases such as those with MLH1 suppression

are included. In addition, it may be important to identify rare types of cell for classification. Here stratified sampling may be appropriate.

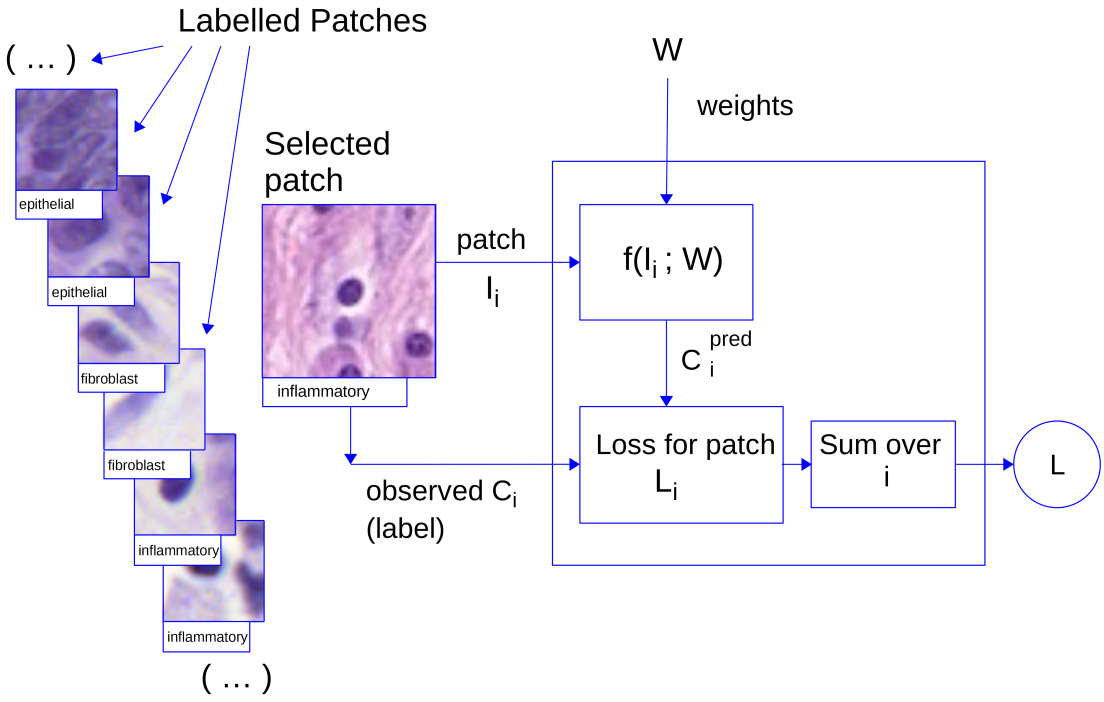

Figure 2.20: Loss Function Calculation. $f(I_i; W)$ predicts c_i^{pred} for labelled image I_i, outputs loss L_i. Sum losses to get total loss L.

2.12.1 Optimisation

Optimisation of the loss function is an iterative process in which the loss function is repeatedly calculated until it reaches a minimum.

Figure 2.20 is a schematic of the computation of the loss function. The training data set is represented by the set of labelled patches on the left: epithelial cells, inflammatory cells and fibroblasts are illustrated. The current patch I_i is input to the model $f(I_i; W)$ which outputs the prediction c_i^{pred}. The loss function L compares c_i^{pred} with the observed value C_i, outputting L_i (the cross-entropy loss function described above is often used). The L_i values are aggregated to form the overall loss function L for the training set and weights. Figure 2.20 is specialised to

cell classification but the overall structure still applies to detection and segmentation.

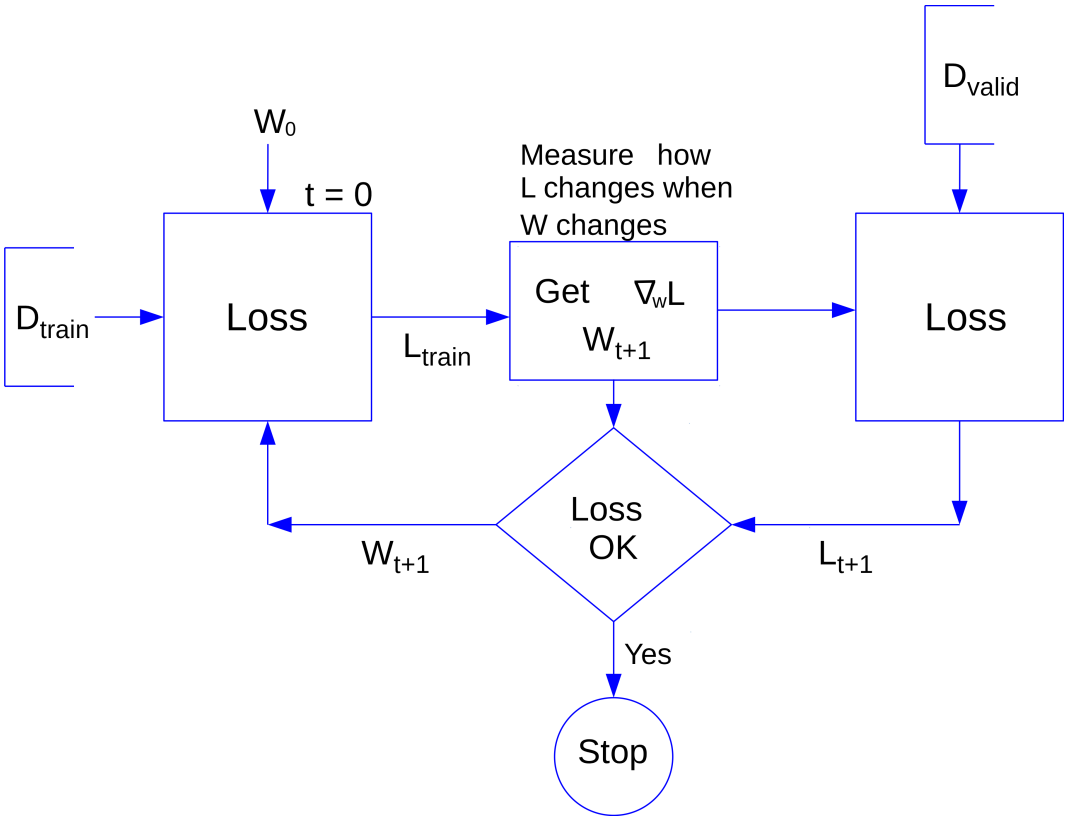

Figure 2.21: Loss Optimisation.

Figure 2.21 is a diagram of the optimisation process. Optimisation is *iterative*: the same steps are repeated until a stopping criterion is satisfied. In the diagram variable t denotes the current step number. The weights W_0 are initialised randomly, images I_i are selected from the training data D_{train} and the aggregate loss function L for the training data is calculated as illustrated in Figure 2.20.

In each iteration t upgraded values of the weights W_{t+1} are obtained by *stochastic gradient descent*. The partial derivatives of the loss function with respect to the weights are calculated and the new weights are obtained by travelling along the surface of L in weight by L space a fixed distance η in the downward direction defined by the partial derivatives. (η is called the *learning rate* and is a hyperparameter chosen by the user.)

Algorithms 1 and 2 contain outline code for the optimisation process illustrated in Figure 2.21. The training data set is denoted by D_{obs} and an independently tagged *validation* data set is denoted by D_{val}. The network structure is denoted by f_{net}. The aim is to return a good set of weights W. In the pseudocode the terms D^x_{train} and D^x_{valid} refer to the sets of predictor variables x in Equation 2.9. Similarly, the term D^{obs}_{valid} refers to y the predicted values in that equation.

Before the main loop the weights are initialised to random values and the epoch t is set to zero.

Calculation of the partial derivatives of the weights is carried out using the chain rule (Algorithm 2). In a forward pass through the layers of the network the intermediate per layer activation maps are stored, plus the derivatives of the loss function with respect to the weights in each layer. The forward pass is followed by a backward pass in which the chain rule is used to compute the partial derivatives from the stored activation maps and derivatives.

A second, independently obtained set of labelled patches, the *validation* set is used to obtain the validation loss function L_{valid}. When L_{valid} converges then iterations stop and the weights W_t are returned by the optimisation function.

Algorithm 1 Train Network

procedure TRAINNETWORK(D_{train}, D_{valid}, f_{net}, ϵ, η)
  W_0 = random() ▷ Initialise weights randomly
  $t = 0$ ▷ Epoch zero
  $L^0_{valid} = 0$
  while $True$ **do**
    t = t + 1 ▷ Next epoch
    $\nabla_W L = GETDERIVS(D^x_{train}, f_{net}, W_{t-1})$ ▷ Derivs. loss w.r.t. weights
    $W_t = W_{t-1} + \eta \nabla_W L$ ▷ Update weights
    $f^{pred}_{valid} = f_{net}(D^x_{valid}; W_t)$ ▷ Predictions for validation set
    $L_{vt} = L(D^{obs}_{valid}, f^{pred}_{valid})$
    if $|L_{vt} - L_{v(t-1)}| \leq \epsilon$ **then return** W_t
    end if ▷ Return if convergence
  end while
end procedure

Algorithm 2 Compute Partial Derivatives

procedure GETDERIVS(D, f_{net}, W)
  for all layers l ascending **do** ▷ Forward pass to compute predictions
    z_l = weighted average of inputs to f_l
    $a_l = f_l(z_l)$ ▷ Predict activation map
    $(f_l)'$ = derivative of f_l w.r.t. z_l ▷ Save derivs,
  end for
  for all layers l descending **do** ▷ Backward pass. Use chain rule.
    $\delta_l = (f_l)' W_{l+1}^T \delta_{l+1}$
    $\nabla_{W_l} L = \delta_l a_{l-1}^T$ ▷ Compute gradients. (T denotes transpose.)
  end for
end procedure

Typically when L is considered as a function over weight space there are many local minima and heuristics need to be used in order to ensure a thorough search. If too coarse a grid is used then minima will be missed; conversely, if the grid defined by the value of η is too fine the search will take too long. In practice the research community tends to use a standard value for the learning rate, based on experience and experimentation. For example, in the 'Cell' detection algorithm, for the size of the step in $W - space$, the learning rate η, the value 0.001 was used.

2.12.2 Batching

The training data set may contain a very large number of points and rather than process the entirety of D_{train} it is usual to divide it into a set of *batches*. Batches are processed in series: the output from batch b is the input to batch $b + 1$. The weights are updated during each batch: convergence to the minimum is faster.

Algorithm 3 ProcessBatches

procedure PROCESSBATCHES(D, f_{net}, W_0, $n_{batches}$, η)
  $S_B = \{1 \leq b \leq n_{batches} : D_b\}$ ▷ Partition data into batches
  $b = 1$
  while $b \leq n_{batches}$ **do**
    $\nabla_W L = GETDERIVS(D_b^x, f_{net}, W_{b-1})$ ▷ Partial derivs.:L w.r.t. W
    $W_t = W_{t-1} + \eta \nabla_W L$ ▷ Follow the slope of W for distance η
    $b = b + 1$
    if $|L_{bt} - L_{b(t-1)}| \leq \epsilon$ **then return** W_t
    end if
  end while
end procedure

The pseudo-code in 3 should replace the contents of the main processing loop in Algorithm 1. In the algorithms used in this thesis a batch size of 256 was typical.

2.12.3 Momentum

To use an analogy with walking in a landscape of peaks and valleys: we compute the maximum slope of the ground where we are, and walk down that slope, predicting the height of the ground at a displacement which is a distance η from the current coordinates. This calculation does not take the trajectory followed by W into account. To do this we may add the term $\alpha \Delta W_{t-1}$:

$$W_t = W_{t-1} - \eta \nabla_W L_t + \alpha \Delta W_{t-1} \tag{2.17}$$

In Equation 2.17 the weights are updated, by a linear combination of the standard update term and the previous weight values . Note that α is a hyperparameter called the *momentum*

2.12.4 Varying the learning rate

The Adagrad algorithm (Duchi et al. [48]) varies the learning rate η: increasing η for sparse parameters and decreasing it for denser parameters.

We denote the partial derivative of the loss function L w.r.t the jth weight W_j, computed at epoch t as g_{tj} where:

$$g_{tj} = (\nabla_W L)_{tj} \tag{2.18}$$

Then we may define the quantity G_{tjk} as follows:

$$G_{tjk} = \sum_{f_t} g_{jt} g_{kt} \tag{2.19}$$

At epoch $t+1$ we consider the diagonal elements of G and use them to modify the calculation of W_{t+1}:

$$W_{(t+1)j} = W_{tj} - \frac{\eta}{\sqrt{G_{tjj}}} g_{tj} \tag{2.20}$$

It may be observed that $\sqrt{G_{jj}}$ is the $\ell 2$ norm of previous derivatives, so weights with a smaller value of this quantity change more.

2.12.5 The Adam Optimiser

The Adam optimiser (Kingma and Ba [102]) keeps running averages of the gradient $\nabla_W L$ at each iteration t and uses them to compute the changes to the weights. The algorithm:

"updates exponential moving averages of the gradient and the squared gradient where the hyperparameters $\beta_1, \beta_2 \in [0, 1)$ control the exponential decay rates of these moving averages."

Using moving averages ensures that the gradient change at each iterative step is calculated more accurately than in Algorithm 1.

Algorithm 4 Adam Optimiser

procedure TRAINADAM(D_{train}, D_{valid}, f_{net}, ϵ, ϵ_2, η, β_1, β_2)
  $W_0 = \text{random}()$
  $t = 0$
  $L^0_{valid} = 0$
  $stopping = false$
  while True **do**
    $t = t + 1$
    $\nabla_W L = GETDERIVS(D^x_{train}, f_{net}, W_t)$
    $g_t = (\nabla_W L)_t$
    $m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t$ ▷ Update biassed moving average of moment 1
    $v_t = \beta_2 v_{t-1} + (1 - \beta_2)(g_t)^2$ ▷ Update biassed moving average of moment 2
    $\hat{m}_t = m_t / (1 - \beta_1^t)$ ▷ Correct first moment for bias
    $\hat{v}_t = v_t / (1 - \beta_2^t)$ ▷ Correct second moment for bias
    $W_t = W_{t-1} - \eta \hat{m}_t / (\sqrt{\hat{v}_t} + \epsilon_2)$ ▷ Make predictions for validation set
    $D^{pred}_{valid} = f_{net}(D^x_{valid}; W_t)$ ▷ Calculate validation loss
    $L_{vt} = L(D^{obs}_{valid}, D^{pred}_{valid}; W_t)$ ▷ Check for convergence of validation loss
    if $|L_{vt} - L_{v(t-1)}| \leq \epsilon$ **then return** W_t
    end if
  end while
end procedure

2.12.6 Regularisation

According to Efron and Hastie [51]

"regularization describes almost any method that tamps down statistical variability in high dimensional estimation or prediction problems"

Many of these techniques involve the addition of a penalty term to the loss function. Two common penalty functions are the $\ell 1$ norm and the $\ell 2$ norm:

$$\ell 1 = \sum_{j}^{nW} |w_j| \tag{2.21}$$

$$\ell 2 = \sum_{j}^{nW} |w_j|^2 \tag{2.22}$$

The corresponding penalised loss functions L_1' and L_2' are:

$$L_1' = L(y^{pred}, y^{obs}) - \lambda \ell 1 \tag{2.23}$$

and

$$L_2' = L(y^{pred}, y^{obs}) - \lambda \ell 2 \tag{2.24}$$

The use of $\ell 1$ is known as *ridge regression* or the *LASSO* [51] while use of the $\ell 2$ penalty is also known as *Tikhonov regression*. The term λ is a hyperparameter. According to Efron and Hastie [51] there is at presemt no well-grounded theory for selecting λ, which is usually chosen on empirical grounds.

The addition of the penalty term ensures that weights stay low in magnitude, favouring simpler models over more complex ones, obeying Occam's razor. Regularisation is effectively a Bayesian approach because it confines the probability space to a region defined by the modeller.

2.12.7 Dropout

The use of *dropout* is another regularisation technique [161]. Dropout randomly removes neurons and their connections from the network, enabling the weights of neurons to change independently as training proceeds. During training a bundle of "thinned" networks with fewer weights is used. For testing and prediction the thinned networks are averaged and the final network is a full network that uses suitably weighted averaged values.

2.12.8 Augmentation

Another way of improving the accuracy of weights is to add extra cases to the training data. In the case of pathology images it is reasonable to assume that rotating and reflecting an image has no effect on its validity for training purposes. It

is straightforward to rotate by 90, 180 or 270 degrees and optionally add a reflection. This augments the data by at least a factor of four. In addition it is reasonable to stretch the image, to blur it, and to modify the colour range. Each image presented in a training epoch is randomly transformed using some augmentation technique, working markedly against overfitting.

2.13 Object Identification

Identifying a collection of objects in an image is a harder task than deciding if an image contains a single instance of an object.

General object modelling techniques include YOLO (You only look once) [143] in which the loss function combines the losses associated with detection and classification. Object identification is also provided by Fast R-CNN [68].

Both the 'Cell' algorithm and the 'Hovernet' algorithm are object identification models. Segmentation of an image automatically identifies objects, so the segmentations produced by 'Hovernet' define individual cells. 'Cell' finds individual cells by training a detection algorithm that can decide if a small patch contains a cell. To extend object recognition to a large image a grid is defined over the entire image and the detection algorithm is applied to each location defined by the grid. Thus, for each grid location an associated probability of an object being present in the patch around the grid point may be computed. This defines a probability function for which the peaks correspond to cells. The 'Cell' model estimates the locations of peaks using clustering [157]. This approach was used with Gaussian mixture models by [107] in their refinement of that detection algorithm.

2.14 Competitions in Cell Identification

The MICCAI 2018 conference held a satellite competition, MONUSEG, in which the aim was to segment cell nuclei in images from a variety of organs and TCGA sites (**?**]) . Training data was a set of cell-rich tiles that had been extracted from thirty TCGA images, including two images of colon carcinomas. Test data included tiles from fourteen TCGA images, one from the colon collection. Cell boundaries were hand marked for use in both training and testing.

The test metric was the Jacquard Index. The top value was 0.69, with 95%

confidence limit of (0.68, 0.70). Confidence intervals of next four entries were within these bounds.

Various forms of colour normalisation were used in preprocessing. The winning entries used data augmentation heavily: all of them used rotation and affine deformation. The winning entry added colour jitter to the augmentation techniques. U-Net was the most popular CNN in use. All competitors applied boundary separation. The top-rated entrants carried out boundary separation as a separate task with its own loss function, while others included boundary separation as a post-processing stage.

MoNuSAC 2020, a 'Multi-organ Nuclei Segmentation and Classification Challenge' Verma et al. [180] was similar to MONUSEG. Contestants were:

"provided with H&E stained tissue images of four organs with annotations of multiple cell-types including epithelial cells, lymphocytes, macrophages, and neutrophils. Participants used the annotated dataset to develop computer vision algorithms to recognize these cell-types from the tissue images of unseen patients released in the testing set of the challenge."
Many teams entered the competition which invited contestants to consider both the 'Cell' and 'Hovernet' models when devising their models. The Panoptic Quality was used as the evaluation metric (Kirillov et al. [103]).

Competitors in both competitions improved on existing results: digital pathology continues to develop.

Chapter 3

Deep Learning with Sampling

Whole-slide images are large and processing them is costly. Using a sample, rather than the entire WSI may speed up processing significantly. In addition, sampling can aid the analysis of features of interest: the spatial behaviour of such features is important in the analysis of whole-slide images.

This chapter, based on Shapcott et al. [154], describes experiments with TCGA whole slide images of colon cancer. In the experiments two sampling policies were applied to the data: *Random Sampling (RS)* and *Systematic Random Sampling (SRS)*. Two cell identification algorithms were used, the 'Cell' algorithm (Sirinukunwattana et al. [157]) and the 'Hovernet' algorithm (Graham et al. [71]). The Background chapter introduced these algorithms in Section 2.11: here they are dealt with in more depth.

The chapter is structured as follows. Section 3.1 describes the use of sampling in pathology images while Section 3.2 introduces the RS and SRS sampling policies. The operation of 'Cell' and the 'Hovernet' algorithms are described in detail in Section 3.3. Section 3.4 describes the workflow that results when these algorithms are applied using sampling. The output of cell identification is a list of detected points (defined by coordinates and cell types) and an aggregate (summary) function extracts a *profile* from the list. In this chapter the profile was defined to be a list of *counts* of the different types of cell. Various experiments using sampling were carried out using the sampling workflow (Section 3.5). Estimates of the error in the cell counts associated with sampling were calculated as were results for different types of cell (Section 3.7) and the RS and SRS sampling policies were compared for accuracy. Section 3.8 is an example application in which SRS was used with 'Cell' to examine associations between cell counts for colon cancer and various

clinical TCGA variables. Section 3.9 concludes this chapter with a general discussion.

3.1 Sampling in Histopathology

Sampling of regions within an image is a normal activity in manual pathology. Pathologists are accustomed to rapidly scanning tissue slides under the microscope and selecting interesting regions for intensive consideration. Kayser et al. [94] discuss how an equivalent procedure can be carried out using digital pathology. They propose an implementation using three stages. In the first stage a set of regions in the image is generated by automated sampling, in the second stage an information measure is calculated for each region, and in the final stage the most informative regions are selected for intensive consideration by the pathologist. The authors argue that this hybrid approach can achieve viewing times that are comparable with those achieved in manual pathology.

Automated sampling within an image is used in *stereology*, originally the analysis of three-dimensional structures, using two-dimensional sections. In stereology various statistical procedures are used to extract significant structural information. A typical approach is to lay a regular grid over the image, and to sample the image using the grid. Stereology has been applied using digital pathology by [95]. The authors found that the use of their automated sampling algorithm was 50% to 90% more-time efficient than conventional random sampling.

In another study sampling was employed in the analysis of cases of colon cancer where pathologists were asked to categorise the tissue type at three hundred randomly selected points in a dense region of tissue [182]. The study found that a low proportion of tumour cells was related to poor cancer-specific survival.

Regarding sampling applications in digital pathology, a description of the use of sampling in the detection of invasive breast cancer in histopathology images can be found in [39]. A trained CNN classifier accepted patches of fixed size as input. The pathology image was tiled and in the first sampling step the resulting patches were randomly sampled. Each patch in the sample set was classified as homogeneous or heterogeneous. Regions of interest were those where the classification was uncertain. The regions surrounding tiles of uncertain classification were searched by sampling them systematically, using the gradient of the uncertainty map to guide the search.

In histopathology applications the choice of a sampling policy is affected by *spatial dependency*, whereby characteristics at neighbouring locations tend to have similar values. Standard statistical sampling techniques that assume that observations are independent of each other do not take spatial dependencies into account and are not always the most appropriate methods. Sampling policies that do take account of spatial dependencies have been developed in geospatial statistics [45] and of these, systematic random sampling is an established technique [44].

3.2 Basic Random Sampling and Systematic Random Sampling

In the experiments described in this chapter the following approach was used: sampling of a set of fixed size square regions (*tiles*) followed by cell identification and profile generation.

For each sampling policy (i.e. RS or SRS), tiles in an image W were sampled, then a deep learning model was applied to each tile in the sample. For each tile t a *cell map* was obtained, a set of labelled cell locations, then a tile profile ϕ_t was extracted from the cell map. The profile for the WSI $\phi(W)$ was calculated by assuming that the tile profiles could be averaged to yield an estimate for $\phi(W)$. When n_T tiles were sampled the aggregate profile was:

$$\phi(W) = \frac{\sum_t^{n_T} \phi_t}{n_T} \tag{3.1}$$

In some situations, non-random sampling, such as uniform spacing may be adequate. Uniform spacing gives good coverage of the WSI but will fail if there are periodicities in the image, or if there are relationships that depend on distance that should be estimated from the sample.

The basic form of Random Sampling used in this chapter operates as follows. The image is assumed to be already segmented into foreground tiles, containing tissue, and background tiles which do not. If there are n_F foreground tiles, n_S integers are selected randomly from the integers between 1 and n_F (without replacement). The deep learning algorithm accepts a tile as input and outputs cell locations and labels. Random sampling is straightforward to implement but if spatial dependencies are present random sampling tends over-sample some areas and under-sample others ([44]).

Systematic Random Sampling overcomes the unbalanced sampling problem of RS. The whole slide image is overlaid with a grid of identical *sub-grids*, or *sample grids* and a tile is randomly sampled from each sub-grid. SRS may be viewed as a combination of non-random sampling (all sub-grids are used) and random sampling (tiles within a sub-grid are randomly sampled).

Note that sampling may be used in other ways. In *adaptive sampling*, information is derived from the samples already taken, and used to choose later samples. If elements of search are incorporated into the sampling process, then adaptive sampling may be appropriate. SRS and RS are non-adaptive sampling policies: all observations are extracted at once, according to the same rule.

This chapter reports on experiments with sampling polices, RS and SRS. Because there was no prior information to indicate that any specific feature in the morphological profile should be prioritized, the use of adaptive sampling was not considered. This does not rule out the use of adaptive sampling in future applications, for example, when it is necessary to concentrate on features that are uncommon when the sampling policy might be directed towards areas with such features. For example, if a tissue sample consists mainly of normal cells, but we wish to analyze the features of rare abnormal cells, it might be advisable to search near points already sampled that were found to contain abnormal cells.

3.3 Cell Identification Models

The two cell identification models 'Cell' and 'Hovernet' were introduced in Chapter 2. Here they are presented in more detail.

3.3.1 The 'Cell' model

'Cell' is an algorithm carried out in three stages: namely two convolution neural networks, linked by an intermediate step, applied in series. In Stage I the first CNN outputs an activation map containing probabilities of nuclear material at pixels. The intermediate stage II finds clusters in the activation map, and assigns them to the locations of nuclei. In the implementation of 'Cell' the CNN in Stage III is based on the 'cifar10' algorithm. It classifies a nucleus into one of the four categories defined in training, in practice cells are 'epithelial', 'inflammatory', 'fibroblasts' or 'other' cells.

Input to the 'Cell' algorithm is a tile, a H&E image of colorectal cancer. The size of the tile is 500×500 at 20X.

The 'Cell' architecture was used in [158]. WSIs of colorectal carcinomas were tiled and the 'Cell' algorithm was applied to each tile, predicting cell locations and cell types. The per-tile output results were aggregated into a map of cell locations labelled by cell type. Nearest neighbour networks were calculated from the map and used to extract motifs which were then used to create image profiles. The authors found that the profiles were predictive of distant metastasis.

3.3.2 The 'Hovernet' Model

As already remarked the 'Hovernet' algorithm segments an image into areas which contain cellular material. In training three main error components are optimised simultaneously. In the first place parameters used to calculate behaviour of the activation map with distance are optimised. In the second place "a novel loss function which calculates the mean squared error between horizontal and vertical gradients and the GT gradients of the horizontal and vertical maps respectively and the corresponding gradients of the (Ground Truth)".

The authors created a new dataset of 41 tiles from 16 WSIs of colorectal cancer, H&E diagnostic slides from 16 patients. The images included overlapping nuclei and artefacts such as ink markings. These tiles were used in training. Training used a cost function which included expressions for each of the three branches in the model, 'ND' for the accuracy of distance prediction, 'NP' for the quality of detection prediction and 'NC' for the quality of classification prediction.

3.4 Workflow: Using Cell Identification Algorithms with Whole-Slide Images

Figure 3.1 illustrates the stages used to create a profile from a whole-slide image. It is assumed that the cell identification algorithm has been trained using images of fixed size and resolution and that the image has been subdivided with tiles of the same size and resolution. (In the experiments described later in this chapter, the training image size was the same as the tile size, (500×500) pixels at a resolution of 20X, approximately 0.5 microns/pixel). For a given tile the cell identification algorithm detects cells and classifies them as one of the types used in training.

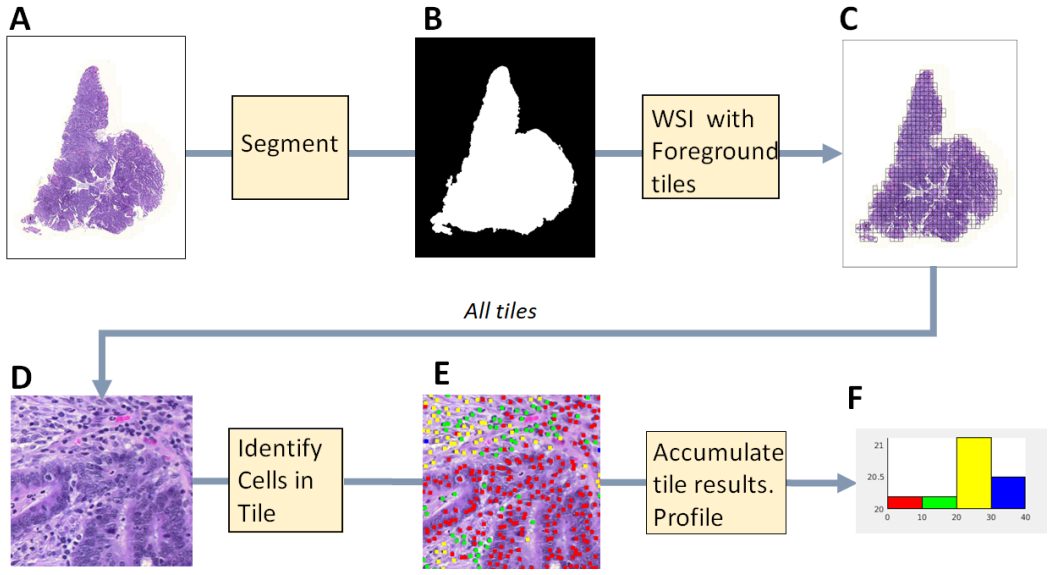

Figure 3.1: Workflow - From WSI to Profile

3.4.1 Foreground/Background Segmentation

In the first stage of the workflow, that of *segmentation*, the whole slide image (A) is separated into foreground and background regions, represented by a binary mask (B).

In the second stage the mask is divided into tiles which are the same size and resolution as those used to train the algorithm. Each tile then is categorised as foreground or background, depending on how many pixels in the intersection of the tile with the segmentation mask are white or black respectively. Figure 3.3 displays the resulting foreground tiles in the case of Patient AA-3543 in the TCGA COAD data set. For each foreground tile (D) the cell identification algorithm locates and classifies cells (E). The information concerning cell nuclei is summarised in a set of cell locations labelled with a cell type. The tile results are then stitched together, creating a map over the WSI containing the locations of labelled cells. The map can then be analysed, creating a profile of the WSI. In Figure 3.3 the frequencies of the four cell types in the entire image, are the output profile, displayed as a histogram (F).

In this study the cell profile was defined straightforwardly as the areal densities of the different types of cell. These features can be interpreted as measures of cellularity, the number of cells of a given type in the cancer tissue. Cellularity is described as "The degree, quality, or condition of cells that are present" [52].

3.5 Materials and Methods

3.5.1 Experimental Dataset

Diagnostic images were downloaded from the TCGA COAD data set, via the Genomic Data Commons Portal [72]. COAD contains 433 viable diagnostic images, stored in SVS format, which have a nominal resolution of 40X (0.25 microns/pixel). Each TCGA diagnostic image file contains both pixel intensity maps and meta-information such as the actual resolution and the name of the capturing device. In the case of the TCGA COAD files, the finest resolution is nominally 40X - 0.250 microns per pixel. In practice, images are stored at several resolutions within the file. As well as the maximum resolution image, coarser images are stored at various downsampling values. Typical downsampling values are 1, 4, 8, 32, 64, though the exact set used in the COAD diagostic images varies from image to image.

The experimental data set contained 142 diagnostic images from COAD, selected from a single site, the 'AA' site. These were images from patients where gene expression data was included in the patient data: gene expression data that is analysed in Chapter 5.

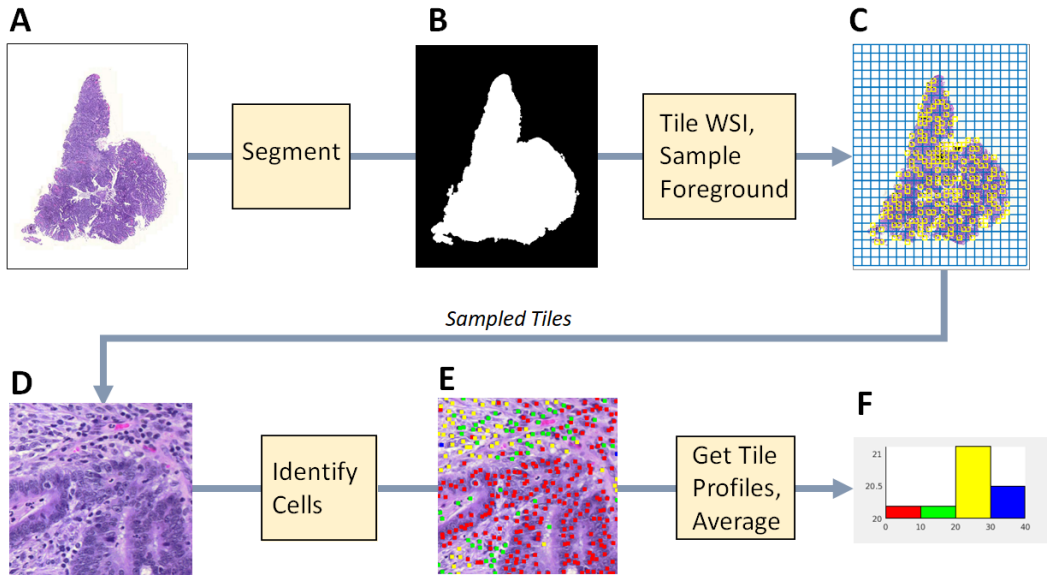

Figure 3.2: Workflow with Sampling - From WSI to Profile

Figure 3.2 illustrates the stages used to create a morphological profile from

a whole-slide image when sampling is used. Segmentation of foreground and background results in a mask. The mask is applied to a tiling of the image, resulting in a set of defined foreground tiles. The foreground tiles are sampled, and the cell identification algorithm is applied to each one. For each tile an aggregate function is applied to the resulting cell map, and a tile profile is output. The tile profiles are averaged to create a profile for the whole-slide image. In the following sub-sections the workflow stages are described in more detail.

3.5.2 Tiling and Image Segmentation

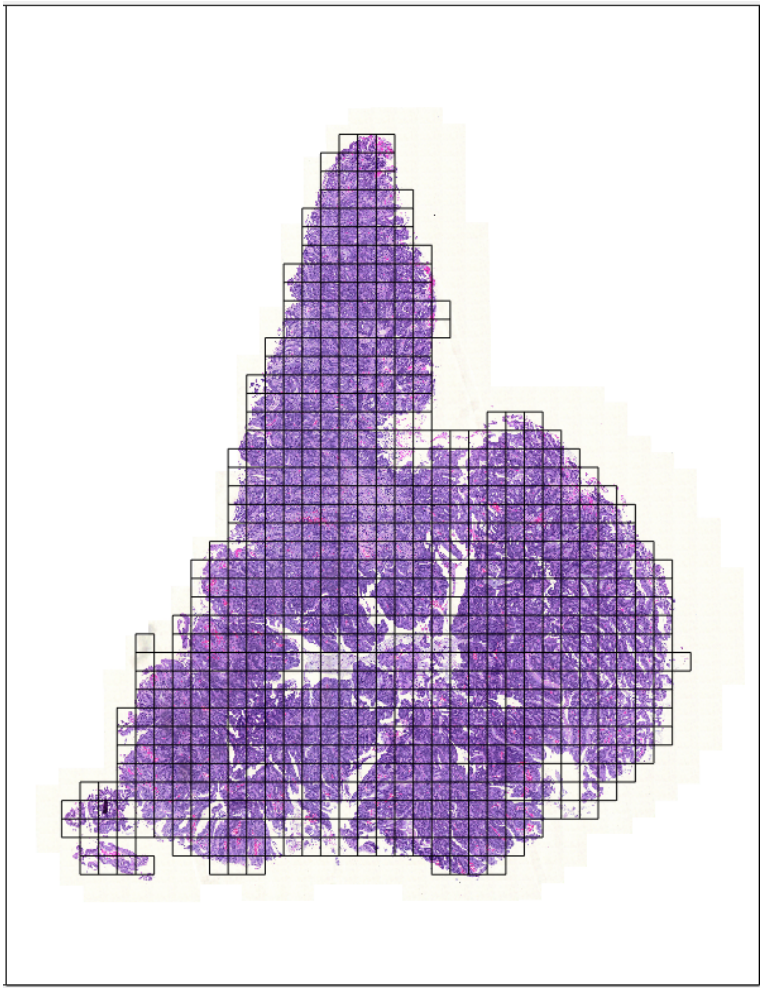

Figure 3.3: Foreground Tiles - TCGA COAD - AA-3543

The first stage in dealing with the whole slide image was to separate foreground from background. Working at a resolution of 8 microns per pixel the image was subdivided into patches, and the entropy-based measure due to Trahearn [171]

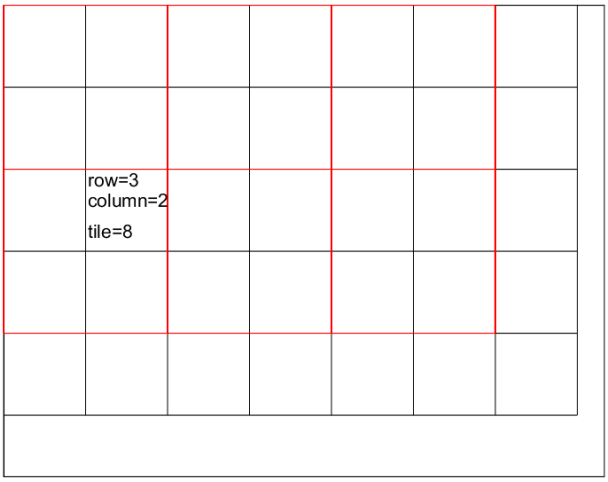

Figure 3.4: Schematic of Tiling: Outer rectangle represents perimeter of WSI, tiling is a grid of four rows and six columns

was used to decide if patches were foreground (tissue) or background (empty areas). Foreground patches, having more variations in intensity, had higher entropy values and background patches had lower values. A mask was created from the pattern of foreground and background patches. Tiles that overlapped with the foreground mask were denoted as foreground tiles and marked for possible processing with the detection algorithm. Figure 3.3 displays a WSI after segmentation, with foreground tiles outlined.

Figure 3.4 is a schematic diagram of a grid laid out in the enclosing rectangle that represents the perimeter of a WSI. The WSI can accommodate five rows and seven columns of tiles (the first tile has been laid with its top left corner placed on the top left corner of the enclosing rectangle). If the sampling policy is RS, then n_S tiles are sampled by selecting n_S values from the integers between 1 and 35 (without replacement).

Otherwise, if the sampling policy is SRS, then sample grids are defined. In Figure 3.4 sample grid is a square containing 4 tiles. In the diagram the sample grids are outlined in red. In the creation of the set of sample tiles each 2×2 sample grid is considered and an integer from the set [1, 2, 3, 4] is selected randomly. There are six sample grids, so the number of tiles sampled is also six.

We may identify tiles by their index in the grid: an index generated by the tile row and column. If a tile is in row i and column j of the grid then its tile index is:

$$k = (j-1) * n_R + i \tag{3.2}$$

where n_R is the number of rows in the tiling.

3.5.3 Artefact Handling

Artefacts were handled on the fly. Although no coloured ink markings were visible in the set of diagnostic images, there were artefacts caused by fixative that had not been wiped clean, and what appeared to be black ink splodges. Tiles were tested for the presence of such artefacts *after* sampling.

3.6 Training the 'Cell' Identification Algorithm

The two CNNs in the 'Cell' algorithm were trained as follows.

Training data for detection consisted of 852 hand-marked images, including the images described in [157]. The authors of that article marked locations of cell nuclei and each nucleus was tagged with its cell type. This could be done rapidly and approximately 30,000 nuclei from ten CRC patients were marked. For training the detection network, small patches were selected from the 852-tile set, those surrounding each tagged cell location and also background patches. The patches were fed to the detection CNN which assigned them a probablity of being nuclear tissue. In successive iterations of the training algorithm the predictions of the CNN were compared with ground truth values and used to adjust the weights in the CNN.

Specifically, input to the detection CNN was a tile of size 500×500 at 20X, and output was an *activation* map which mapped each pixel to the probablity that the pixel was nuclear material. The intermediate stage clustered the activation map, then assigned the cluster centres to detection points. Small patches (33×33 pixels at 20X) were extracted from the tile and input to the classification model.

The detection algorithm was trained using the method in [157]. The detection code used 'vlfeat' software implemented in Matconvnet Vedaldi and Lenc [179]. The same clustering algorithm as detailed in [157] was applied to the probability map

output by the convolutional neural network and generated the locations of cell nuclei.

The classification model was based on the Tensorflow 'cifar10' model [110], and was trained with Tensorflow [4], using the Pycharm IDE (Python 2 and Tensorflow 1.4). The layers of the classification CNN were the same as those defined in the 'cifar10' model [10], and the following hyperparameters were applied: (batch size = 128, moving average decay = 0.9999, number of epochs per decay = 350, learning rate decay factor = 0.1, initial learning rate = 0.1, maximum number of steps = 1,000,000).

Patches for training the classification network were generated by selecting (51×51) pixel images around hand marked points. There were 111,659 of these, from which smaller patches of size (33×33) pixels were extracted subject to random displacements that allowed for inaccuracies in location (an average of up to 5 pixels) and each which was augmented in training by extra images generated by rotation and reflection. All processing was done at 20X (0.5 microns/pixel). The average RGB intensities of the training patches were recorded for later use in standardisation. An accuracy of 84% was achieved in evaluation of classification using a hold-out set.

3.6.1 Identifying Nuclei with the 'Cell' Algorithm

Feature f_j is the number of cells of type j in the cell map (adjusted by a constant ρ to ensure an effective area of $100\mu M \times 100\mu M$).

$$f_j = \rho \sum_{p=1}^{n_M} (c_p == j) \tag{3.3}$$

Figure 3.5 displays a tile marked with the results of the cell identification algorithm. The algorithm has identified (i.e. detected and classified) a mixture of epithelial cells (red squares) and inflammatory cells (green squares), plus cells identified as fibroblasts (yellow squares). To compute the morphological profile of the tile, we simply count the numbers of different types of cell and multiply by ρ.

3.6.2 Implementing 'Hovernet'

In 'Hovernet' the tiles were 256×256 square pixels in size. To ensure comparability with 'Cell' each 500 by 500 'Cell' tile was subdivided into four regions. Hovernet tiles were created for each region and the four of them were offered to the Hovernet algorithm. The Hovernet output was stitched together (four tiles back into one) to

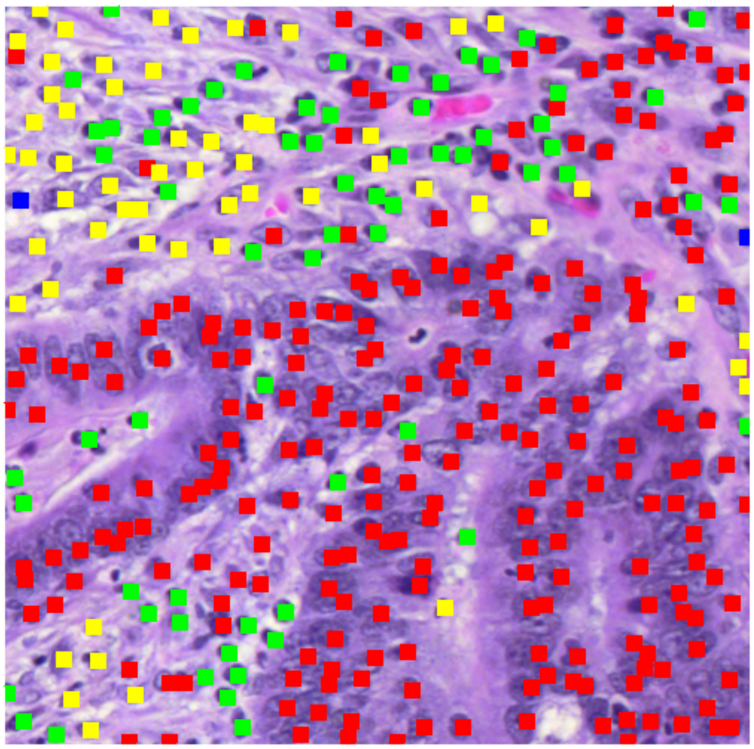

Figure 3.5: Tile with Identified Cells

fit the 500×500 'Cell' tile.

In equation 3.4 the cell identification model M_I accepts an image I and, together with the intermediate clustering step, creates a cell-map consisting of n_M cell nuclei, located at points $< x, y >$, each of which is labelled with the cell type c.

$$M_I(I) = \{< x_p, y_p, c_p >: 1 \leq p \leq n_M\} \tag{3.4}$$

The image's morphological profile is a set of J features.

$$\phi(I) = [f_1, ...f_j...f_J] \tag{3.5}$$

3.6.3 Sampling for Cell Identification - RS

In the case of RS and for each experimental run, n_T tiles were randomly sampled from the set of n_F foreground tiles. The cell detection algorithm was applied to each tile individually. The detection component calculated the haemotoxylin channel and supplied it to the detection CNN. The classification module extracted patches around each detected point, normalised them collectively, using the average intensities saved from the training stage, and applied the classification algorithm to each patch

individually.

If n_t cells are detected in image I_t we may denote the coordinates of cell p by (x_p, y_p). If the cell is classified as type c_p then the tile detection function g_t is:

$$g_t(I_t) = \{n_t, (x_p, y_p, c_p) : 1 \leq p \leq n_t\} \tag{3.6}$$

If there are n_T tiles altogether the detection function g that applies to I, the image in its entirety, is:

$$g(I) = \bigcup_{1 \leq t \leq n_T} g_t(I_t) \tag{3.7}$$

3.6.4 Calculating Profiles

We define the profile of image I as a set of n_J features, where each feature f_J is an aggregate function of I:

$$f_j(I) = f_j(g(I)) \tag{3.8}$$

In many cases these aggregate functions may be computed on a per-tile basis:

$$f_{jt}(I_t) = f_{jt}(g(I_t)) \tag{3.9}$$

And the per-image profiles may be aggregated in turn:

$$f_j(I) = \frac{\sum_{1 \leq t \leq n_T} f_{jt}(g(I_t))}{n_T} \tag{3.10}$$

Sampling can be used to calculate features in a straightforward way. If n_T tiles are randomly sampled then we may use the formula:

$$f'_j(I) = \frac{\sum_{1 \leq t' \leq n_T} f_{jt'}(g(I_t))}{n_T} \tag{3.11}$$

3.6.5 Implementing SRS

SRS was implemented as follows. A nominal sample size n_{NOM} was defined: the number of non-background, non-artefact tiles to be sampled. A coarse tiling of the WSI used sample grids, squares that each contained $(g \times g)$ tiles. The value of g was calculated using n_{NOM} and γ, an estimate of the fraction of tiles in the image that are *not* artefacts and n_F the number of foreground tiles:

$$g = \left\lfloor \frac{\gamma n_F}{n_{NOM}} \right\rfloor \tag{3.12}$$

Note that the proportion of artefacts, if not estimated in advance, was assumed to be zero. With SRS one patch was sampled randomly from each sample grid. If the tile was a foreground tile then the cell identification algorithm was applied to it and the resulting profile was added to a list of profiles associated with the WSI. Tiles judged to be artefacts were excluded from the calculations. Otherwise, if the tile was a background tile as assigned in the segmentation mask, it was ignored. The whole-slide profile was calculated by averaging the profiles in the list of included tiles.

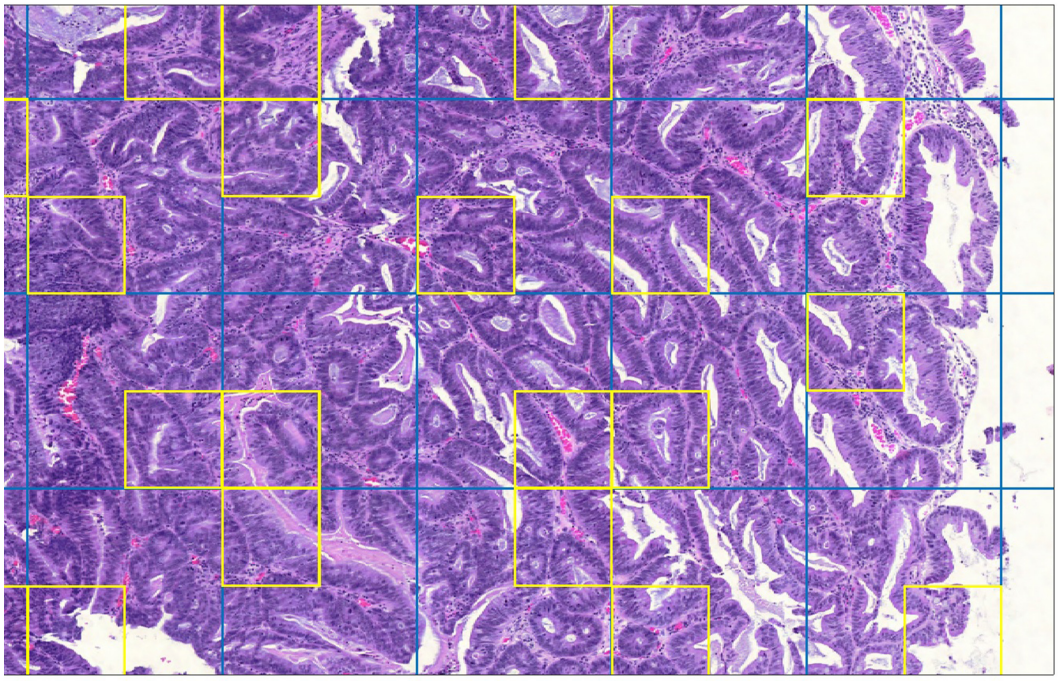

Figure 3.6: Region of H&E Image Displaying Sample Grids

Figure 3.6 is a detail of a WSI split into sample grids with divisions indicated by blue lines. Each sample grid contains four tiles in a (2×2) pattern. The tiles selected by SRS for processing by cell identification are outlined in yellow. Not shown here, but there may be cases where none of the tiles in a sample grid is outlined in yellow. This happens when a sample grid contains only background tiles or if a background tile is selected in sampling the sample grid.

3.7 Results

3.7.1 Evaluation of Cell Identification Using Hand Marking

In five of the TCGA diagnostic images 1,500 cells were hand-marked by a pathologist. Cells were classified as already noted except that epithelial cells were classified as normal cells or malignant cells. Patches containing hand-marked cells were run through the cell identification algorithm and detection and classification were both scored. (Note that the two types of epithelial cells were merged into one, because the cell identification algorithm did not distinguish them.) Both detection achieved 65% accuracy on average, while classification was 76% accurate on average. See Table 3.1.

Table 3.1: Detection and Classification Accuracy

Patient ID	Detection Accuracy	Classification Accuracy
AA-3543	0.85	0.66
AA-3845	0.68	0.76
AA-3864	0.62	0.81
AA-3986	0.61	0.90
AA-A02J	0.50	0.66
Average	0.65	0.76

These scores were lower than the scores achieved when predictions for a hold-out set were evaluated in training. This is not surprising: the TCGA data is from a data set independent of the training data set.

3.7.2 Comparing Batches - 'Cell' Algorithm

Experiments with sampling policies RS and SRS were conducted with the 'Cell' algorithm, using varying sample sizes: [25, 50, 100]. For each experiment two batch runs were executed. In each batch run the sampling policy was applied to the 142 whole slide images described in Subsection 3.5.1. The batch runs of RS were done after those for SRS using the actual sample sizes generated by SRS, ensuring that the runs could be compared for accuracy.

Figure 3.7 comprises four scatterplots. They compare two batch runs that apply SRS with a nominal sample size of 100 to the 142 whole slide images. There is one scatterplot for each type of cell. x values are profile features calculated in the first batch run and y values are the corresponding features output by the second batch

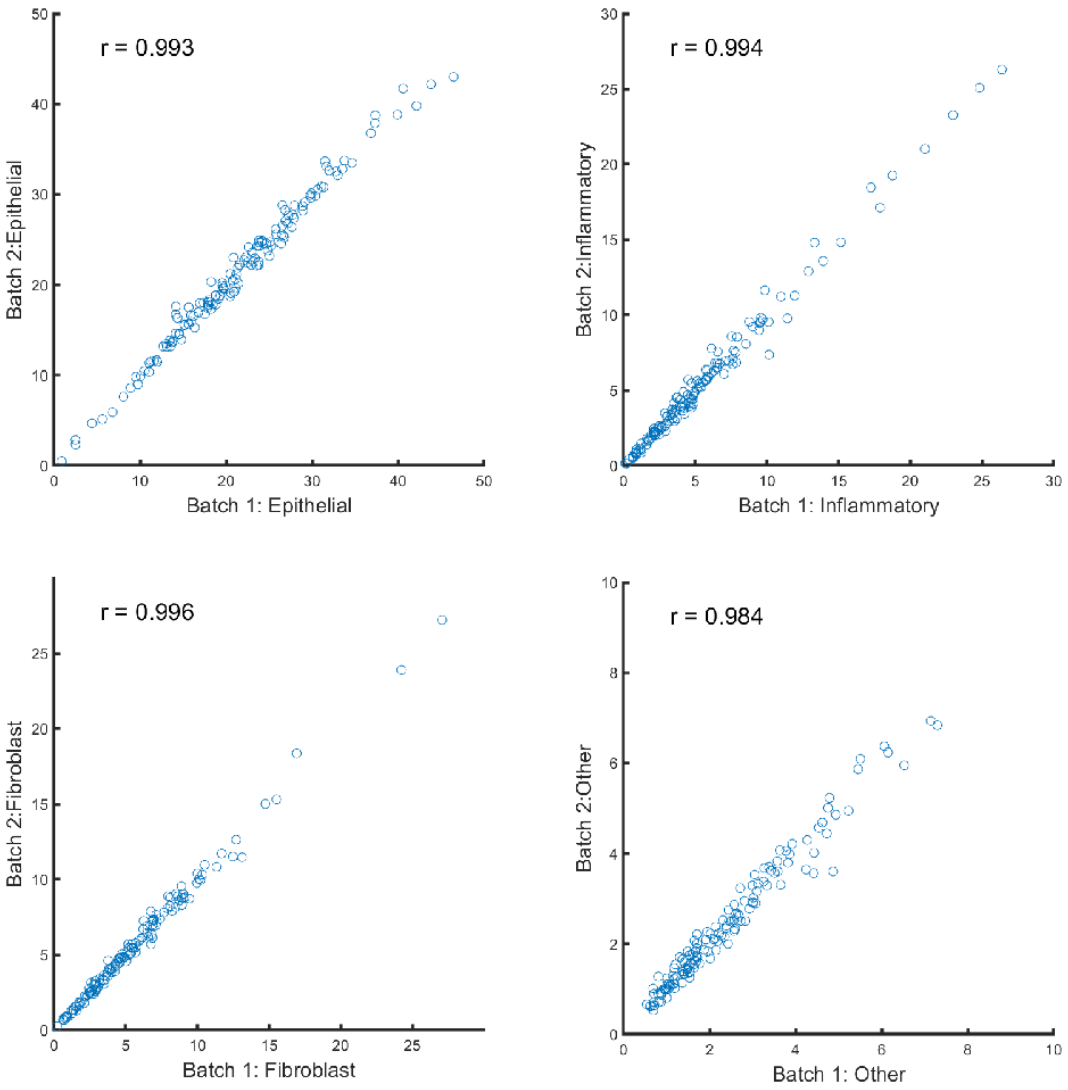

Figure 3.7: 'Cell' algorithm: Comparison of Batch Runs (SRS: sample size = 100)

run. The correlation between x and y is shown on each plot. Values of correlations are very high, indicating that this level of sampling is very satisfactory.

Table 3.2: Correlation matrix of cell counts

	Epithelial	Inflammatory	Fibroblast	Other
Epithelial	1	0.2	-0.59	-0.63
Inflammatory	0.20	1	-0.34	- 0.13
Fibroblast	-0.59	-0.34	1	0.56
Other	-0.63	-0.13	0.56	1

As well as correlations between batches it is also possible to calculate correlations between features in the profiles. Table 3.2 shows the marginal correlations

between counts of the four types of cells. In general, *partial correlations* are better indicators of multivariate relationships ([183]) and Table 3.3 displays partial correlation coefficients that have been extracted from the marginal coefficients.
A priori one might expect cell counts to be negatively correlated because cells are competing for space in the tissue. Ignoring the 'Other' cell category (there are small numbers of these cells) it can be seen that the graph associated with the partial correlations has links <fibroblast, epithelial>(strength -0.34) and <fibroblast, inflammatory>(strength -0.29), indicating that the number of fibroblasts is directly linked to the other two cell types, but that the <epithelial, inflammatory>link is small (strength 0.038).

Table 3.3: Partial Correlations Between Cell Counts

	Epithelial	Inflammatory	Fibroblast	Other
Epithelial	1	0.038	-0.34	-0.45
Inflammatory	0.038	1	-0.29	-0.086
Fibroblast	-0.34	-0.29	1	0.31
Other	-0.45	-0.086	0.31	1

3.7.3 Comparing RS and SRS using different sample sizes

For each sampling policy i.e. RS or SRS and each nominal sample size n_s the detection algorithm was run for each patient of interest. This was done in two batches. For each batch a sample was taken according to the current sampling policy. To explain how results were obtained we adopt the following notation. The profiles resulting from batch run b are denoted by Z_{isub} where i was the patient index, s was an index into the array of nominal sample sizes u denoted the cell type and b was the batch number. Table 3.4 compares SRS and RS for a range of sample sizes and cell types.

For each patient i, each s, the relative mean error for cell type u is defined in the equation below. (The indices 1 and 2 refer to batch numbers.)

$$e_{isu} = 2\frac{|Z_{isu1} - Z_{isu2}|}{|Z_{isu1} + Z_{isu2}|} \tag{3.13}$$

The mean error, averaged over all n_P patients is $\bar{e}_{su}$ where:

$$\bar{e}_{su} = \frac{\sum_i e_{isu}}{n_P} \tag{3.14}$$

It can be observed from Table 3.4 that the mean error decreases with increas-

ing sample size. For epithelial cells and the SRS sampling policy the mean error for the sampling size 100 is 3.5%.

As expected SRS performs better than RS in all cases. For example, for epithelial cells and various sample sizes the SRS average relative error compared with the RS error was as follows: 81% (9.20/11.3) for sample size = 25, 74% (6.0/8.1) for sample size 50 and 60% (3.5/5.8) for sample size 100.

Table 3.4: Error Values(%) - RS and SRS ('Cell')

Sample Size	25	50	100
(Nominal Number of Tiles)			
Epithelial Cells Global Average	21.3 cells		
RS	11.30	8.10	5.80
SRS	9.20	6.00	3.50
Inflammatory Cells Global Average:	5.52 cells		
RS	19.20	12.50	8.30
SRS	17.40	7.60	6.50
Fibroblasts Global Average:	5.64 cells		
RS	14.50	11.00	8.00
SRS	13.80	8.10	4.80
'Other' Cells Global Average:	2.43 cells		
RS	24.30	16.90	9.90
SRS	20.10	11.50	7.80

3.7.4 'Hovernet' - Sampling Experiments

The experiments previously described in Subsections 3.7.2 and 3.7.3 were repeated, the only difference being that the 'Hovernet' algorithm [71] was used as the cell locator instead of the 'Cell' algorithm. Note that 'Hovernet', previously introduced in Section 2.11, is also described in Subsections 3.3.2 and 3.6.2.

Figure 3.8 includes six scatterplots, one for each cell type. The sampling policy was SRS and the sample size was the same as for 'Cell', namely 100. Each scatterplot compares the results of sampling run Batch 2 with those of sampling run Batch 1. Correlation coefficients are shown on the scatterplots. The minimum correlation is 0.978: the results for the two batches are strongly related.

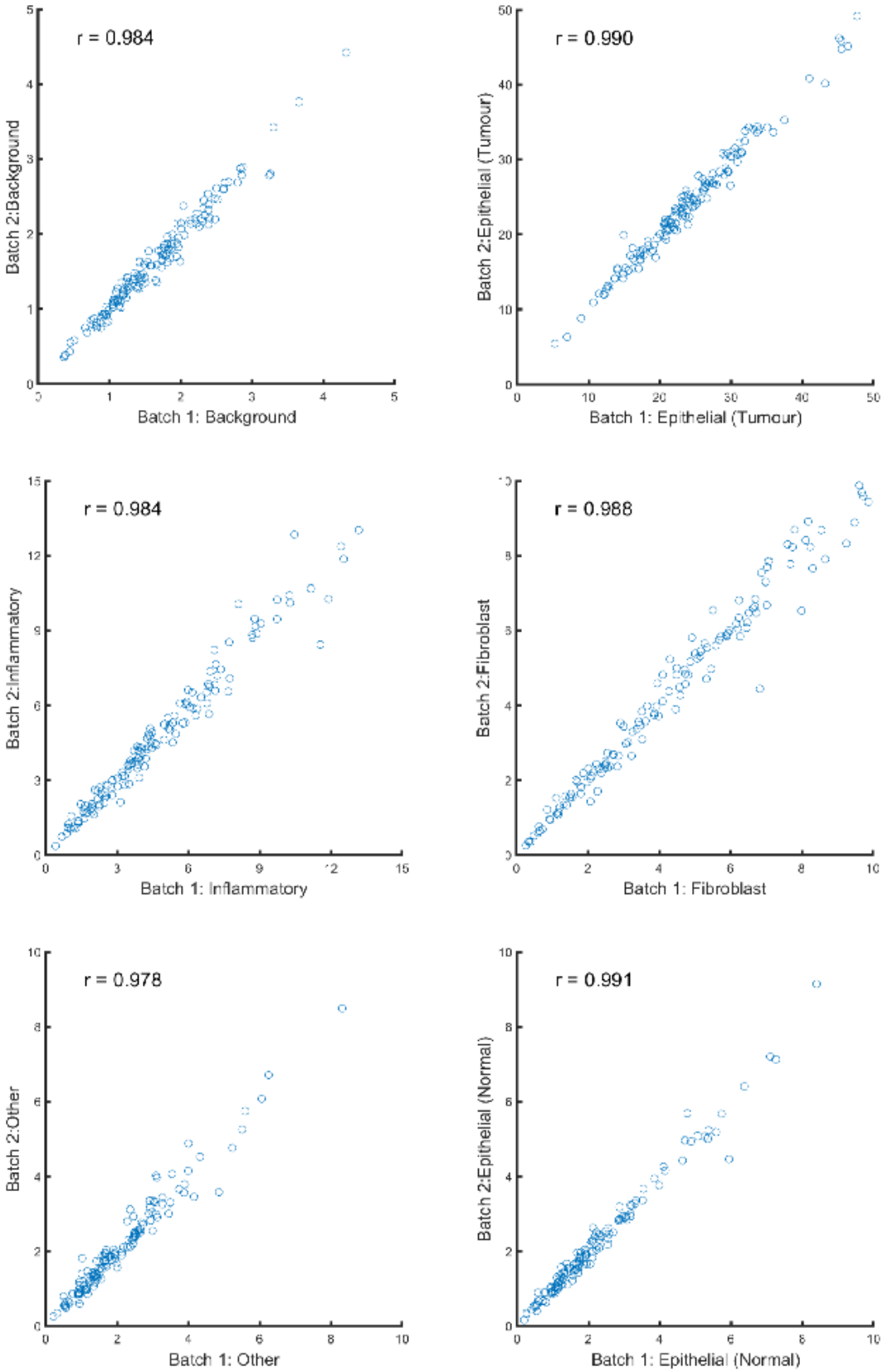

Figure 3.8: Hovernet - Comparison of Batch Runs (SRS: sample size=100)

Table 3.5: Error Values (%) - RS and SRS - 'Hovernet'

Sample Size	25	50	100
Background Cells Global Average	1.64 cells		
RS error	8.58	7.03	4.54
SRS error	7.47	4.47	2.84
Epithelial Cells Global Average	23.90 cells		
RS error	6.66	4.88	3.37
SRS error	5.28	3.21	1.86
Inflammatory Cells Global Average	4.78 cells		
RS error	13.59	10.75	7.10
SRS error	11.66	7.04	3.97
Fibroblasts Global Average	4.64 cells		
RS error	13.71	9.97	5.76
SRS error	10.68	6.86	3.45
Other Cells Global Average	2.07 cells		
RS error	17.18	10.90	8.41
SRS error	13.52	8.05	4.81
Normal Cells Global Average	2.34 cells		
RS error	11.18	9.36	6.52
SRS error	11.51	6.11	3.93

Table 3.5 displays the sampling errors associated with the 'Hovernet' algorithm. It follows a similar pattern to Table 3.4. The numbers of each type of cell detected are greater than those for the 'Cell' algorithm in Table 3.4 but the relative distributions are similar, roughly four times as many epithelial cells as inflammatory cells, and four times as many epithelial cells as fibroblasts. The average number of epithelial cells predicted by 'Cell' was 21.3, and correspondingly values for inflammatory cells were 5.5, for fibroblasts were 5.6 and for 'Other' cells were 2.4. The corresponding 'Hovernet' averages were 23.9 epithelial cells, 4.8 inflammatory cells, 4.6 fibroblasts and 2.1 'Other' cells in each $100\mu M \times 100\mu M$ square. For SRS with a sample size of 100 the average error is 1.86% indicating that sampling is accurate enough to compute profile values.

The error values calculated in the experiments carried out here refer to the errors due to sampling, not to errors in the cell identification process itself. Sampling accuracy could be high for both the 'Cell' algorithm and the 'Hovernet' algorithm, but those algorithms predictions of the average numbers of cells differed by 10% to 20%. Equivalently, it is possible that the sampling errors are low, but the cell identification algorithm itself has poor accuracy.

3.8 Application: Assocations between Profile Values and Clinical Variables

Preprocessing of the clinical data associated with the 142 images in the data set identified fourteen clinical variables of interest. (Variables with large numbers of missing values were excluded, as were variables with constant values.)

Table 3.6: Associations between cells counts and clinical variables

Clinical Variable	Group 1	Group 2	p-value	BH p-value	BH sig
Metastasis					
M0 (n=120) M1 (n=21)	M0	M1			
Epithelial	22.1	17.2	0.00152	0.0411	Y
Inflammatory	5.8	4.0	0.0372	0.0411	Y
Fibroblast	5.3	7.7	0.0156	0.0429	Y
Other	2.1	3.5	0.00506	0.0482	Y
Residual Tumor					
R0 (n=117) R2 (n=20)	R0		R1		
Epithelial	22.1	17.7	0.0130	0.0438	Y
Inflammatory	5.8	4.4	0.0506	0.0393	N
Fibroblast	5.3	7.8	0.0179	0.0420	Y
Other	2.2	3.3	0.0100	0.0464	Y
Vascular Invasion					
NO (n=64) YES (n=73)	NO	YES			
Fibroblast	4.6	6.4	0.00661	0.0473	Y
Venous Invasion					
NO (n=98) YES (n=30)	NO	YES			
Epithelial	22.9	19.4	0.0111	0.0455	Y
Fibroblast	4.8	6.4	0.0116	0.0446	Y
Mucinous Carcinoma					
NO (n=120) YES (n=20)	NO	YES			
Inflammatory	5.9	3.4	0.00361	0.0491	Y
Vital Status					
Alive (n=130) Dead (n=12)	Alive	Dead			
Inflammatory	5.70	3.80	0.0488	0.0402	N

Each variable was cross-tabulated against each of the four profile features, or correlation coefficients were calculated, or a MANOVA was performed. Where the clinical variable was a binary categorical variable, t-tests were used to compare the mean value of the profile variable by clinical group. For example, metastasis was grouped by value as 'M0' or 'M1' and it was natural to compare the average

numbers of different types of cells in the two groups.

Table 3.6 shows the six clinical variables for which the (uncorrected) t-test had a p-value less than or equal to 0.05 for at least one of the four cell types. The other clinical variables were also tested, but no significant associations were found and these results are not displayed. Table 3.6 shows the name of the clinical variable in the first column followed by the categories of interest and the number of patients in each category. In lines containing cell types, the average value of the cell count is shown for each category, followed by the p-value. The significance value of 0.05, appropriate to a single test has been adjusted using the Benjamini-Hochberg correction [83], [15] and is shown in the column labelled "BH p-value".

Differences between the two categories for metastasis had significant p-values for all cell types. Compared with M0, (colorectal cancer without evidence of distant metastasis), the category M1, where metastasis was present, had increased numbers of fibroblasts and 'Other' cells and fewer epithelial cells and inflammatory cells. The presence of residual tumor was also associated with more fibroblasts and 'Other' cells and fewer epithelial cells and inflammatory cells. Both vascular invasion and venous invasion were associated with increased numbers of fibroblasts. Venous invasion was associated with fewer epithelial cells.

Mucinous carcinomas were associated with fewer inflammatory cells than were non-mucinous carcinomas. Finally, the twelve patients who were recorded as dead when added to the TCGA repository were also likely to have fewer inflammatory cells detected than patients who were recorded as alive, although the associated p-values were not significant.

Note that the remaining clinical variables, for which no associations were found, were as follows: Gender, Age, T Stage and N Stage, History of colon polyps, History of other malignancy, Anatomic neoplasm subdivision (Tumour Location - left side versus right side) and CEA level.

3.9 Discussion

There were five clinical variables for which we found significant associations with morphological features. Four clinical variables had significant associations with fibroblast counts: in each case higher fibroblast counts were associated with poorer values of the clinical variable. This is not unexpected [82]. In a review of the role of cancer-associated fibroblasts in the tumour microenvironment, [92] refers to

fibroblasts as the 'cockroaches' of the human body and states that they play an important role in tumorigenesis and cancer progression.

Two clinical variables were associated with differences in inflammatory cell counts, namely metastasis, and mucinous carcinoma. Poor values of the clinical variables were associated with lower numbers of inflammatory cells, which might be expected, in the light of the positive role of tumour infiltrating lymphocytes in slowing down disease progression [136], [134].

Finally, metastasis, residual tumour and venous invasion were related to lower numbers of epithelial cells.

The morphological features extracted from the 142 diagnostic images from the COAD data set may be regarded as expressions of *cellularity*, the numbers, degree or quality of cells present in a tumour. Cellularity is a familiar concept in pathology: here each morphological feature corresponds to the spatial density of the corresponding cell type. Regarding the four different types of cell, deep learning generated morphological features that are indicators of cell density. Cellularity has been reported to be related to patient survival and other diagnostic and prognostic indicators, indicating that the features calculated here may be of general usefulness.

To train a CNN to recognise an object, images need only be large enough to include relevant information from the object's neighbourhood. For example, the models used in this thesis were trained on image patches of size (33×33) at 20X magnification. To apply a model to a whole-slide image the image was segmented into (500×500) at 20X) tiles and the cell identification procedure was applied on a per-tile basis. In this approach the procedure is applied to each tile independently, outputting a set of features that characterise that tile. The per-tile features were aggregated over the WSI to generate a collection of features which characterise the cellular characteristics of the WSI: a WSI profile.

However, such an approach is computationally costly: on average, each WSI in the data set used in this study contained about 900 tiles that had significant amounts of tissue. Computational costs were reduced by sampling a limited number of tiles, applying the identification algorithm to each, then averaging the per-tile features over the sample of tiles. In principle, if enough tiles are sampled, processing costs can be reduced without significant loss of accuracy.

Using sampling to obtain results can save processing costs. In this application the average number of tiles containing tissue in an image was approximately nine hundred, so processing a random sample of patches had the potential to greatly reduce computation costs. Two sampling policies were examined in this study. The first policy was random sampling which samples patches with uniform weighting. The second policy was systematic random sampling which takes spatial dependencies into account. Compared with the processing of complete whole slide images there was a seven-fold improvement in performance. When systematic random spatial sampling was used to select 100 tiles from the whole-slide image for processing there was very little loss of accuracy (approximately 4% on average).

The profiles being computed were particularly suitable for some form of random sampling because the features of interest all associated with 'cellularity' were additive over regions in the images. In applications where the regions of interest are sparse and spatially concentrated, adaptive sampling may be more appropriate. The two examples from the literature, discussed in the introduction, use random sampling to find regions of interest followed by adaptive sampling to narrow the search.

Further experiments remain to be done. For example, we have calculated profiles using quite large tiles, usually containing hundreds of cells. Sampling using smaller tiles, but more of them, might well prove effective. The performance is likely to be affected by both pure speed-ups and by latency costs associated with loading data onto the GPU, so some experimental work would be useful.

The work described here has been experimental: no explict statistical modelling of locational distributions has been used. For example, a first approximation would be to assume that the spatial distribution of cells has a Poisson distribution. This assumption allows standard errors and other statistics to be estimated using standard statistical machinery.

In addition, *multi-level* modelling could be considered. The theory of multilevel modelling applies to data in which statistics may be calculated within objects, then used as variables that describe those objects. For example, we may determine the distribution of test scores of pupils within each class in a school. Summary descriptors of the distribution, such as class test averages, may then be used as attributes when comparing the performance of different classes within the school [70]. The TCGA data is multilevel data. It consists of patient data: images containing

cells, clinical data and molecular data. We are calculating the distribution of cells within an image (the image profile), then we compare patients, using both profiles and clinical data. The use of the theory of multilevel data is a possible extension to the work done here.

In addition to the cellularity features studied here, other features may be calculated using deep learning. Such features, most of which have been discussed in the previous chapter, include tumour budding which is the presence of single tumour cells or small clusters of up to five cells in the stroma and which is associated with aggressive cancer ([43], [174] and [104]). In addition, [106] suggests that poorly differentiated clusters, perineural invasion, and desmoplastic reaction are also important in diagnosis. Another morphology of interest is that of serrated cancers in which the colonic glands are of distinctly serrated form [123], [61].

[91] classified colorectal cancers according to molecular features, observing that they are related to morphological features such as the number of tumour infiltrating lymphocytes, differentiation, presence of dirty necrosis, serration, tumour budding, mucinous/not mucinous and presence of an expanding invasive margin. [54] reported that serrated cancers have distinct molecular features. Deep learning has recently been used to predict diagnostic molecular features from morphology, e.g. for lung cancer [37], and breast cancer [38]. It is to be expected that future work with deep learning will enable morphological, clinical and molecular data to be linked.

The experimental results in this paper were obtained from a single TCGA site. The analysis could be extended to all sites in the TCGA colon cancer repository. In the experiments carried out here, standardisation was straightforward, using the pooled average intensities of a group of whole-slide images to normalise data. Unfortunately, there is no guarantee that this approach will always be successful. Techniques that cater for the many different originating sites in TCGA should be used. The next chapter addresses the colour variability found in the different TCGA sites.

3.10 Conclusions

The work here has shown experimentally that a cell identification algorithm using deep learning can uncover informative 'profiles' of diagnostic images and that systematic sampling of tissue regions can improve performance without losing accuracy.

In analysing a set of diagnostic images from TCGA, statistical sampling of tiles from whole-slides images proved to be worthwhile: significant improvements in classification performance were achieved with very little loss of accuracy. This finding applied both to the 'Cell' algorithm and to the 'Hovernet' algorithm.

For both 'Cell' and 'Hovernet' systematic random sampling (SRS) was markedly more accurate than straightforward random sampling (RS). For example, with a sample size of 100, and considering epithelial cell counts the batch difference error was 3.5% for systematic random sampling and 5.8% for basic random sampling (Table 3.4 above).

An application of sampling to the 'Cell' algorithm found statistically significant associations between morphology and various clinical variables. The TNM grading system used in cancer treatment considers tumour penetration, nodes and metastasis [88]. Of these three indicators significant associations were found for metastasis, for all four types of cell. In addition, associations between WSI profiles and 'Residual Tumour', 'Venous Invasion', 'Vascular Invasion', and 'Vital Status'. Mucnous carcinomas were found to be associated with fewer inflammatory cells.

Chapter 4

Colour Normalisation

In the preceding chapter sampling was used in cell identification. In that work, analysis was confined to a subset of TCGA patients from the AA contributing site, patients for whom gene expression data was stored,. A typical image, a tile from a diagnostic image is displayed in Figure 4.1.

This chapter considers the entire TCGA COAD (colon cancer) data store. The COAD repository contains 433 high-resolution diagnostic images which have been uploaded from twenty-four different sites and vary greatly in appearance, mainly in the intensities of the haemotoxilyn and eosin regions. The haemotoxilyn stain binds to the nuclei of epithelial cells and varies in colour from light mauve in Figure 4.1 to nearly black in Figure 4.2. The eosin stain binds to cytoplasm and stroma. Variations in stroma colour can be seen by comparing Figure 4.3 which has light staining of stroma with Figure 4.4 which has heavy staining of stroma. The lightly stained images are both from the contributing site labelled AA while the heavily stained images are from the A6 site and the D5 site.

The human observer is not particularly impeded by variability in colour intensity: a trained pathologist will have learned to cope. Even to the untrained eye the cells in the preceding figures are mostly quite easy to identify. Unfortunately, the typical automated system is not so versatile. When heavily stained images are presented to the 'Cell' classification model used in the preceding chapter, the model almost invariably predicts the cell type poorly. See Table 4.9 in Sub section 4.3.7 which summarises the results of the 'Cell' classification model when both unnormalised and normalised patches are input. The first column of results displays the per-site classification accuracies of 'raw' (unnormalised) patches. In general these accuracies are low and the average accuracy for unnormalised images across all sites

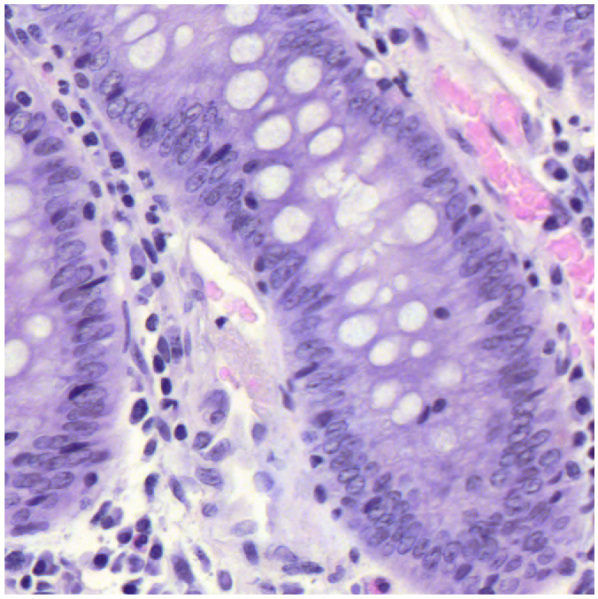

Figure 4.1: Tile with epithelial nuclei - light staining (Patient:AA-3845 Tile:1412)

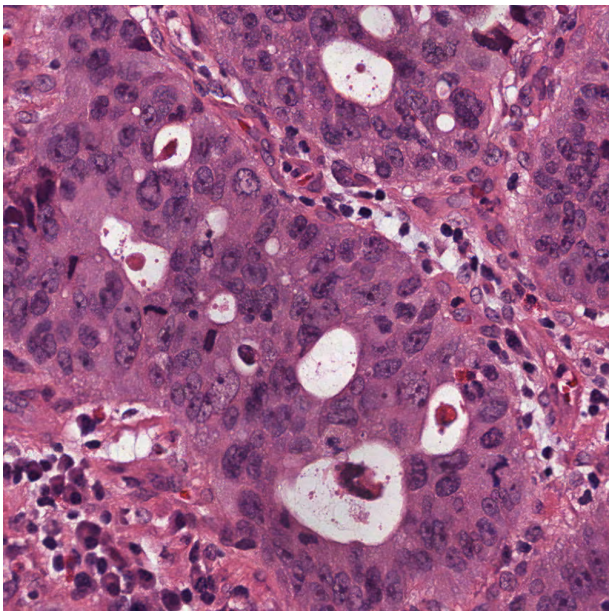

Figure 4.2: Tile with epithelial nuclei - heavy staining (Patient:A6-2686 Tile:6010)

is 35%.

Colour variability in TCGA COAD data is summarised in Table 4.1. Colour statistics of patches detected by the SRS detection algorithm were computed for all 433 viable WSIs in the COAD data set. Table 4.1 displays the average colour

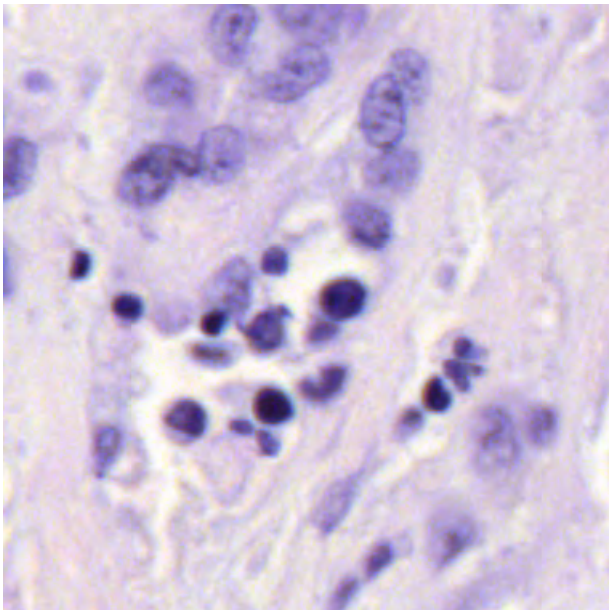

Figure 4.3: Lightly coloured stroma: (Patient:AA-3845 Tile:704)

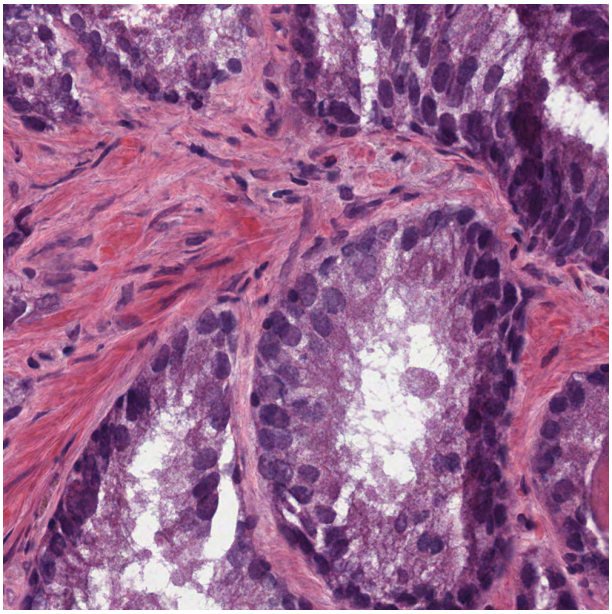

Figure 4.4: Heavily stained stroma: (Patient:D5-6928 Tile:1131)

intensities at the ten sites with the largest number of patients (omitting one site with images with many coloured pen markings). In addition, Figures 4.5, 4.6 and 4.7 plot Blue against Red, Green against Red and Green against Blue site averages. In each plot the corresponding point for the classification training patches is shown as a red dot (R=191, G= 158, B=208). In these plots the point associated with the AA site is the closest one to this red dot, suggesting that of all the sites, the AA

site, analysed in Chapter 4, contains data that is closest to the training data.

Table 4.1: Classification Patches - Colour Statistics

Site	Mean			StDev w.r.t global			Stdev w.r.t 10 means		
	Red	Green	Blue	Red	Green	Blue	Red	Green	Blue
AA	165.0	132.5	197.9	37.7	40.0	40.0	19.2	19.5	13.4
A6	131.8	79.2	122.6	47.5	40.0	40.0	14.2	15.1	12.9
CM	145.3	95.7	140.2	44.5	39.7	39.7	12.2	14.4	12.9
D5	179.4	131.9	164.9	42.1	44.9	44.9	12.9	18.3	15.2
G4	156.8	101.7	146.7	40.7	38.4	38.4	23.7	17.6	16.1
AZ	150.7	92.2	132.5	50.4	43.2	43.2	19.0	17.7	14.8
F4	159.0	93.9	139.0	54.2	39.4	39.4	29.8	19.1	21.9
CK	149.6	96.1	138.9	51.7	42.1	42.1	14.4	13.2	10.2
AD	156.8	101.7	146.7	40.7	38.4	38.4	23.7	17.6	16.1
AY	128.3	84.7	129.4	42.5	35.6	35.6	21.2	13.8	10.0

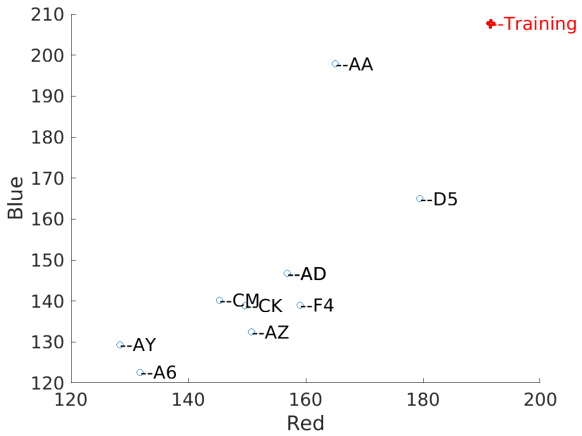

Figure 4.5: TCGA Sites: Mean Blue Intensity Plotted against Mean Red Intensity

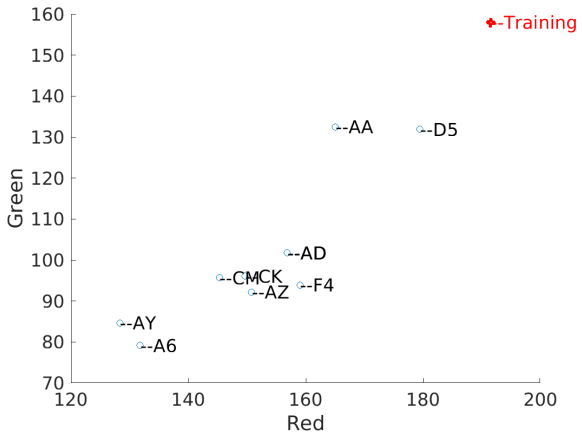

Figure 4.6: TCGA Sites: Mean Green Intensity Plotted against Mean Red Intensity

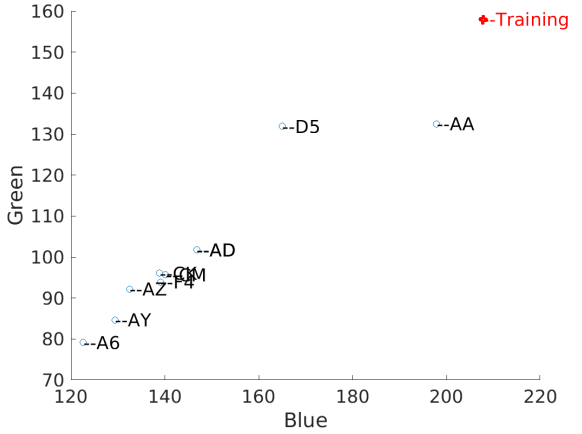

Figure 4.7: TCGA Sites: Mean Green Intensity Plotted against Mean Blue Intensity

Ideally, if standardised laboratory procedures are used to create a digital image from a tissue sample, the intensities in the digital H&E image will be independent of the laboratory. In practice, variations from the ideal occur for all sorts of reasons. Use of the microtome is a skilled operation: the depth of the tissue slices can vary, within a tissue section, from slice to slice, from machine to machine and from operator to operator. Variations in the time for which the tissue is left in the stain bath are also responsible for such *batch effects* [63]. The behaviour of the haemotoxilyn and eosin dyes can also vary. This is likely to be a batch effect: within-slide variation due to different dye behaviour is likely to be much less than differences resulting from the use of different dye manufacturers. In addition, calibration of the microscope can produce varying results.

There are several ways of dealing with such batch effects. The first approach, *colour normalisation*, transforms the image of interest to a new image that has statistics that match the statistics of reference images. Usually colour normalisation alters the colour intensity distributions of images so that they match those of a *reference* data set, usually the set of images that were used to train the model.

The second method, *augmentation*, extends the training space by applying random changes to the training data, thereby adding new cases which mimic possible variations in the experimental setup. For example, the effect of different dye manufactures can be mimicked by adding training images with colour intensities corresponding to dye varieties used at different sites. The augmented CNN model is then applied to images at these sites. CNN models have an abundance of trainable parameters, enough to deal with large image spaces, so this is a feasible strategy.

Finally, *generative adversarial networks, GANs*, deal with each new site separately. The training network uses both the labelled training images and unlabelled new site images as input. The main idea is to add a secondary network to the main network. The secondary network classifies images by site location: training site or new site. The aim is modify the parameters of the main network until the secondary network fails to discriminate between sites: the learnt parameters capture features that are invariant to the change of site. The term *adversarial* refers to the fact that the two networks operate in opposite ways: while the loss function of the main network is minimised, that of the secondary network is maximised. GANs are similar to augmentation because the parameters of the (hybrid) CNN contain parameters that embody information concerning features of the new site.

In the rest of this chapter we compare colour normalisation methods, applying them to diagnostic images from ten TCGA sites, the sites with the largest number of images. The effects of normalisation on both cell detection and on cell classification are examined. For each normalisation method we calculate appropriate statistics for the training data then apply normalisation to hand marked images. Appropriate metrics are used to evaluate the normalisation techniques on a per-site basis.

The chapter is organised as follows. Section 4.1 contains an introduction to colour normalisation and considers five colour normalisation methods. One method, 'Naive' colour normalisation, operates directly on colour intensities while the other four colour normalisation methods are based on colour separation: the extraction of the separate contributions of the dyes in operation. These methods are those of Ruifrok and Johnson [147], Khan et al. [100], Macenko et al. [117] and Vahadane et al. [177]. Section 4.2 describes the test harness used to compare these normalisation techniques in cell classification, a test harness in which cell patches were selected, hand marked, normalised, then subjected to the classification model of the 'Cell' algorithm. Results are presented in Section 4.3. Normalisaton techniques were also examined for effectiveness in detecting cells, using the detection model of the 'Cell' algorithm, in Section 4.4. Related work is considered in Section 4.5. Section 4.6 is a concluding discussion of results.

4.1 Colour Normalisation Methods

Normalisation uses two sets of statistics, one set for the training data, and the other for the prediction data. In this context the term *statistic* refers to a quantity that has been calculated from the features that define an object: in image processing this is usually the result of a calculation performed on (appropriate regions of) the image. The theory of exploratory data analysis [173] suggests that the first statistics to compute are measures of size, followed by measures of dispersion. Measures of size include means, modes and medians. Measures of dispersion include standard deviations and ranges, both raw and adjusted. In statistics these correspond to the first and second moments of a frequency distribution. The third and fourth moments of a distribution, skewness and kurtosis are occasionally used. For cases where order-preserving transformations are most appropriate, order statistics, such as quartiles and other percentiles are also used.

The general workflow in normalisation is displayed in Figure 4.8. Relevant statistics are extracted both from the training image (or images) and the whole slide

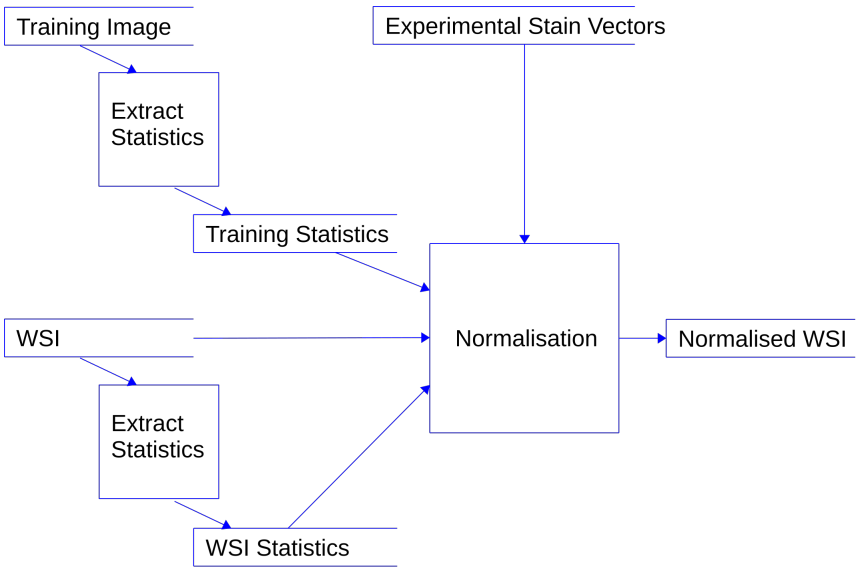

Figure 4.8: Workflows in Normalisation

image. The relevant regions of the image are loaded into memory and transformed so that the transformed image has the same statistics as the training image.

Note that in most cases of normalisation, background pixels in the image are identified and removed from consideration before statistics are extracted. This is because the background area of an image is a large proportion of the region that varies greatly in size, without containing useful colour information.

4.1.1 'Naive' Colour Normalisation

'Naive' colour normalisation adjusts RGB intensities to match training statistics. The 'Naive' algorithm used here applies standardisation so that the input image intensity maps have the same mean and standard deviations as those of the training data. A straightforward approach is to carry out standardisation as the combination of two transformations. The first transformation is a standard Z-transformation while the second transformation transforms points in Z-space to a space where their means and standard deviations are the same as the means and standard deviations of the reference (training) data.

In the implementation of 'Naive' normalisation it is assumed that there are three colour maps f_R, f_G, f_B and that each map function f_c returns the colour intensity of the pixel at a point belonging to the region of interest.

The mean intensity $\overline{f_c}$, and the standard deviation of the image intensity s_c are used to standardise the f_c. The transformed data z_c has zero mean and unit standard deviation:

$$z_c = \frac{(f_c - \overline{f_c})}{s_c} \tag{4.1}$$

We transform z_c to the normalised intensity f'_c using the mean intensity of the reference image(s) $\overline{f_c}^T$, and the corresponding standard deviation, s_c.

$$f'_c = z_c s_c^T + \overline{f_c^T} \tag{4.2}$$

Finally we ensure that the normalised intensity is in the allowable range: for example if the range is $0, 255$ by restricting the possible values. The component f''_c of the normalised image is:

$$f''_c = \max(\min(f'_c, 255), 0) \tag{4.3}$$

In 'Naive' colour normalisation two statistics are used: the mean colour intensity (over the region on which the image is defined), and the corresponding standard deviation. The statistics are calculated for both the reference images and for the current image. The normalisation transformation has the two lots of statistics as input, plus the image to be normalised.

4.1.2 Stain Separation and Stain Normalisation

When tissue is stained with a single stain such as the nucleus-staining haemotoxilyn, each pixel in the resulting image reflects the amount of stain present. The *Beer-Lambert transformation* [120] maps intensity values (f_R, f_G, f_B) to *optical density* values, denoted by (d_R, d_G, d_B). If the maximum intensity is 255, then the Beer-Lambert transformation of colour intensity f_c may be defined as:

$$d_c = -log(\frac{f_c + 1}{256}) \tag{4.4}$$

The set of all possible optical density values is referred to as *optical density (OD space)*. Points close to the origin in OD space are the brightest pixels, usually background pixels, whereas the darkest pixels are those farthest from the origin of OD space.

The optical density is linearly related to the quantity of stain. If a single stain is used, OD pixels correspond to areas of tissue where the stain has adhered: hence, in the case of H&E slides, the type of tissue of interest being nucleus or cytoplasm.

The linear relationship can be specified by a *stain vector*, a three-dimensional vector in optical density space. The stain vector for a given stain may be determined experimentally by applying the stain to tissue, preparing a digital image of the tissue, then fitting a line to the resulting points in OD space. Stain vectors for haemotoxilyn and eosin are generally available (Ruifrok et al. [146]). Figure 4.10 in Sub section 4.1.4 shows these stain vectors in OD space, together with pixels obtained from a particular WSI (patient A6-2686).

The (unit) stain vectors $\hat{h}$ and $\hat{e}$ may be composed into a stain matrix:

$$S = \begin{bmatrix} h_R & e_R \\ h_G & e_G \\ h_B & e_B \end{bmatrix} \tag{4.5}$$

For a particular image, the individual contributions of haemotoxilyn and eosin can be computed using *stain separation*, also known as *stain deconvolution*. They are estimated by transforming the image from RGB space to OD space, where a *stain deconvolution transformation* is applied.

4.1.3 Deconvolution in the Stain-Vector Plane

Figure 4.9 illustrates how the process takes place in the plane spanned by two unit stain vectors $\hat{h}$ and $\hat{e}$. We may project the optical density of a pixel in the image onto this plane, obtaining the 2D point (r_x, r_y). In the plane we set the X axis to lie in the same direction as $\hat{h}$. Let $\hat{n}$ be the unit normal to the plane and define the Y axis to lie in the direction of $\hat{j}$ where:

$$\hat{j} = \hat{n} \times \hat{h} \tag{4.6}$$

In Figure 4.9 the optical density vector (r_x, r_y) is decomposed into two vectors indicated by the sides of the parallelogram shown in the diagram. Let the sizes of these vectors be h and e respectively. The size of (r_x, r_y) is r:

$$r = \sqrt{x^2 + y^2} \tag{4.7}$$

Let (r_x, r_y) be at an angle α to the X axis, and let θ be the angle between the eosin stain vector and the X axis. Straightforward geometry allows us to calculate the stain intensities h and e as:

$$h = r \cos \alpha - \cot \theta r \sin \alpha \tag{4.8}$$

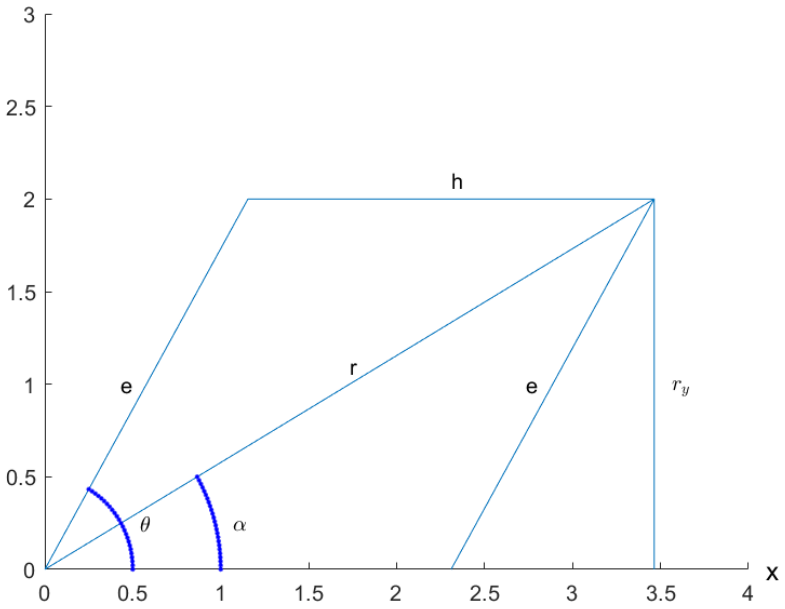

Figure 4.9: Workflows in Normalisation

and

$$e = r \csc\theta \sin\alpha \tag{4.9}$$

Which can be written in matrix form as:

$$\begin{bmatrix} h \\ e \end{bmatrix} = D \begin{bmatrix} r_x \\ r_y \end{bmatrix} \tag{4.10}$$

where D is:

$$D = \begin{bmatrix} 1 & -\cot\theta \\ 0 & \csc\theta \end{bmatrix} \tag{4.11}$$

Because h is a unit vector along the X axis, and e is a unit vector at angle θ to the X axis) the stain matrix is of the form:

$$S = \begin{bmatrix} 1 & \cos\theta \\ 0 & \sin\theta \end{bmatrix} \tag{4.12}$$

Inverting S we obtain the deconvolution matrix D.

$$D = S^{-1} \tag{4.13}$$

4.1.4 Stain Separation in Three-Dimensional Space

Stain separation in three dimensions is carried out as follows. The 3 by 2 stain matrix S_2 defined below is extended by adding the unit normal n to the plane that has h and e as basis vectors as specified in Equation 4.15.

$$S_2 = \begin{bmatrix} h_R & e_R \\ h_G & e_G \\ h_B & e_B \end{bmatrix} \tag{4.14}$$

$$S_3 = \begin{bmatrix} h_R & e_R & n_R \\ h_G & e_G & n_G \\ h_B & e_B & n_B \end{bmatrix} \tag{4.15}$$

Where:

$$\hat{n} = \frac{\hat{h} \times \hat{e}}{\left\| \hat{h} \times \hat{e} \right\|} \tag{4.16}$$

The 3-D deconvolution matrix D is:

$$D = S_3^{-1} \tag{4.17}$$

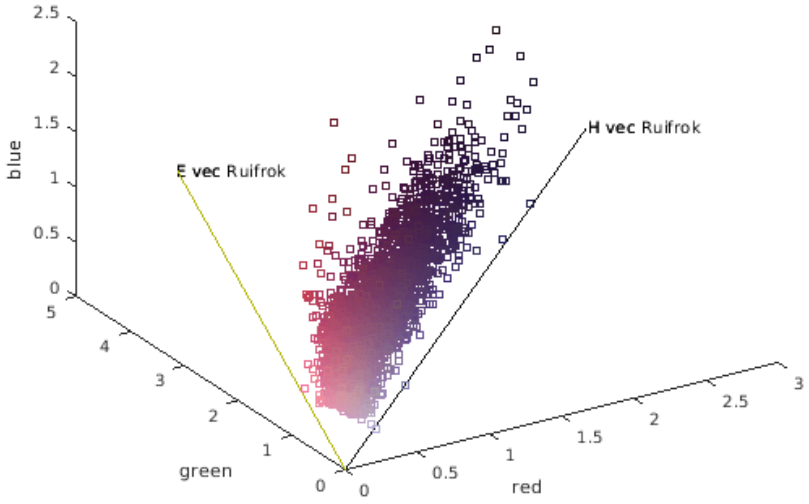

Figure 4.10: OD space showing stain vectors obtained by Ruifrok and Johnson (Patient: A6-2686)

To separate the stains we apply D to the OD values of the pixels in the image, resulting in two surfaces of OD density for the pixels considered in xy space. Application of the inverse of the Beer-Lambert transformation yields the corresponding RGB intensities. Figure 4.10 displays the distribution of pixels in OD space plus two lines that are projections of the stain vectors experimentally obtained by Ruifrok et al. [146]. The pixels tend to be clustered around these two lines in the 3D space. There is one line for H and one for E, both passing through the origin. Nucleic tissue, coloured red, appears in pixels close to the H line, while cytoplasmic tissue, coloured blue-grey, is represented by pixels near to the E line. Pixels representing high concentrations of a stain are further from the origin, while pixels representing low concentrations are closer to the origin. Stain vectors may be obtained experimentally (Ruifrok et al. [147]) or determined from current data by stain separation procedures. Well-known stain separation procedures are those developed by Khan et al. [100], Vahadane et al. [177] and Macenko et al. [117].

Stain normalisation is carried out by computing appropriate statistics for the two intensity distributions in OD space. The intensity distributions are normalised so that their statistics match the reference statistics, usually obtained from the training stage.

4.1.5 Ruifrok Normalisation

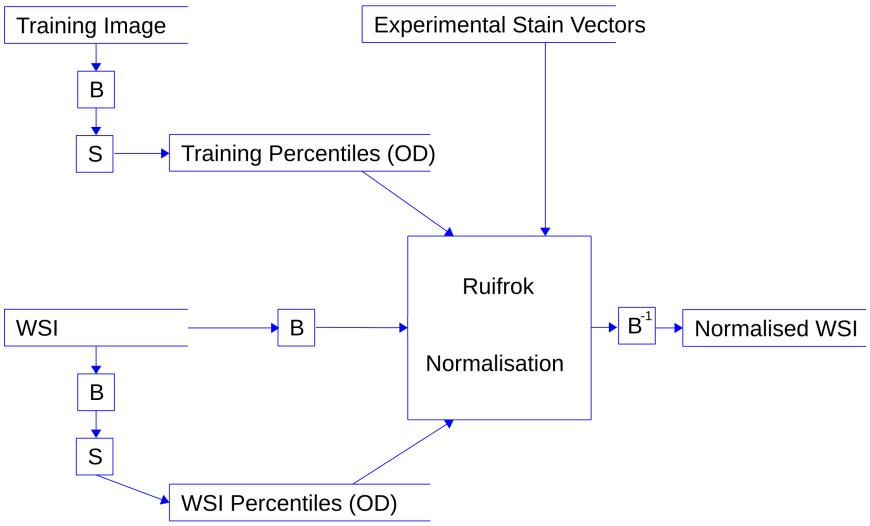

Figure 4.11: Stain normalisation using experimentally determined stain vectors

Figure 4.11 illustrates the workflow in the stain normalisation process when the experimentally determined stain vectors of Ruifrok and Johnson are used. The Beer-Lambert transformation is denoted by the square box containing B while the inverse Beer-Lambert transformation is denoted by B^{-1}. The image is transformed into OD space and the stain maps for H and E are calculated using the inverse stain matrix. The OD values are stretched so that they have the approximately the same maximum value as in the training data. The range is not a robust statistic, so the 99th percentile has been used instead.

4.1.6 Khan Normalisation

This normalisation technique [100] divides the three-dimensional colour space into labelled volumes: into a palette. The palette is used to label pixels as foreground, background or 'other'. The foreground pixels are also labelled as H or E. The h stain vector is the centroid of the H pixels scaled to a unit vector. The e vector is obtained in a similar fashion. The statistics computed in OD space are various percentiles of the stain maps, including the median and the 1% percentile. These percentiles are used to compute piecewise spline approximations of the E and H distributions: the WSI image is forced to the training image by matching the knots of the piecewise splines. The palette may be computed in advance or on the fly.

The workflow for Khan normalisation is shown in Figure 4.12. The vectors obtained by this algorithm for one WSI (A6-2686) are shown in Figure 4.13. In addition the standard Ruifrok vectors are shown. Observe that the Khan stain vector for H appears to be a better representative of the cloud of darker pixels than the Ruifrok vector for H.

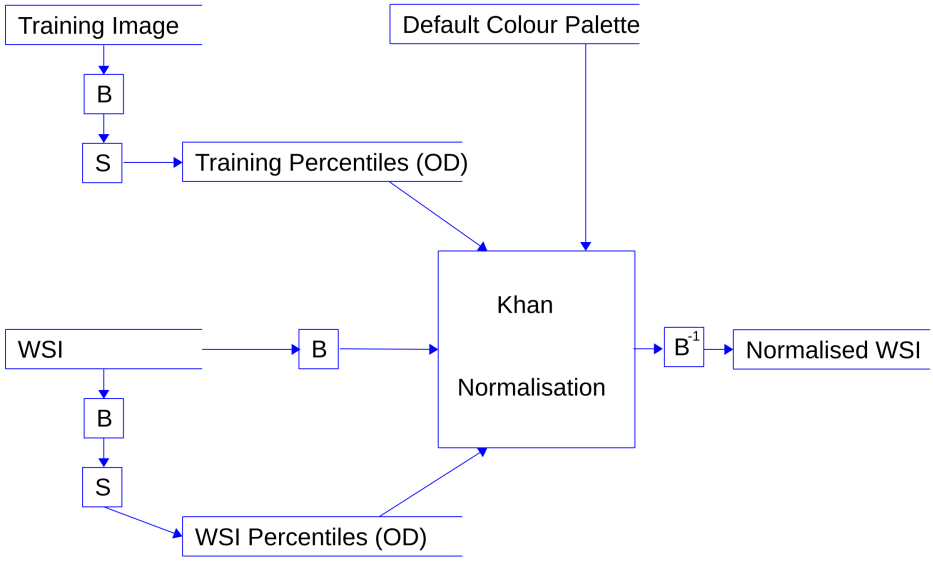

Figure 4.12: Workflow in Khan normalisation

4.1.7 Macenko Normalisation

The colour normalisation method of Macenko [117] computes stain vectors, carries out stain decomposition, then normalises pixels using robust maxima. Points in OD space are summarised by their covariance matrix and its eigenvalues and eigenvectors are extracted. The two stain vectors are assumed to lie in the plane spanned by the eigenvectors corresponding to the two largest eigenvalues and the projections of all points onto this plane are obtained. The region occupied by pixels in the plane is roughly cone-shaped: one edge of the cone is made up of nucleic pixels and the other edge contains cytoplasmic pixels. The edges are obtained by sweeping a clock-hand around the plane and counting pixels. The stain vectors are positions of the hand that have a small fraction of pixels on one side. Applications of Macenko algorithm described in this thesis used percentiles of 2% and 98% respectively. Once the stain vectors have been obtained stain normalisation is carried out using the method already described for Ruifrok normalisation.

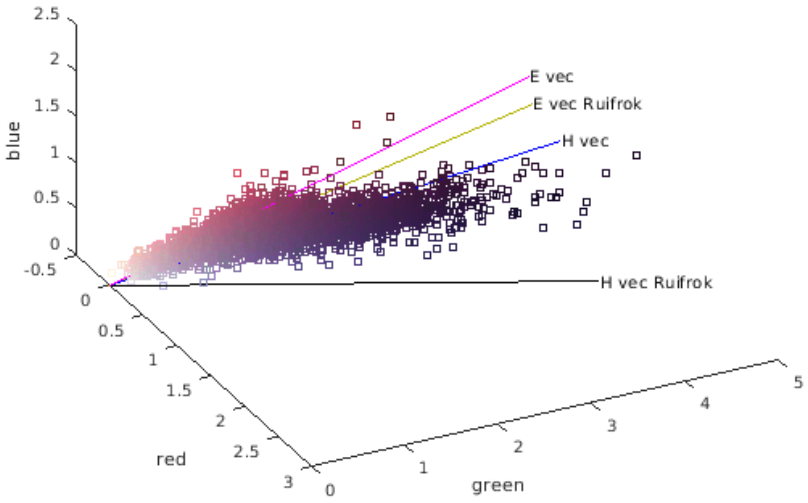

Figure 4.13: OD Space with stain vectors obtained by the Khan algorithm and by the Ruifrok algorithm(Patient: A6-2686)

4.1.8 Vahadane Normalisation

Vahadane's method [177] is another method that estimates stain vectors. The two stain vectors are non-orthogonal basis vectors for the data and may be estimated using non-negative matrix factorisation. The optimisation process that finds the stain vectors has sparsity constraints which are controlled by a sparsity parameter. Once again, stain normalisation is carried out using the same method as in Ruifrok normalisation.

4.2 Test Harness - Cell Classification

The test harness which applied the normalisation methods listed in Table 4.2 operated as follows. The ten COAD sites with the greatest numbers of diagnostic images were used. (Site DM has large numbers of WSIs with pen markings and has been excluded). Five WSIs were randomly sampled from each site. Three tiles of interest from each WSI were saved for hand marking. This was done as follows. The WSI was viewed with the display showing tiles from a previous SRS detection run (see Chapter 3). Figure 4.14 shows the tiles sampled in the 'Cell' detection algorithm, outlined in yellow. The user scanned the WSI (zooming and panning appropriately) and selecting at least three tiles containing examples of epithelial cells, inflammatory

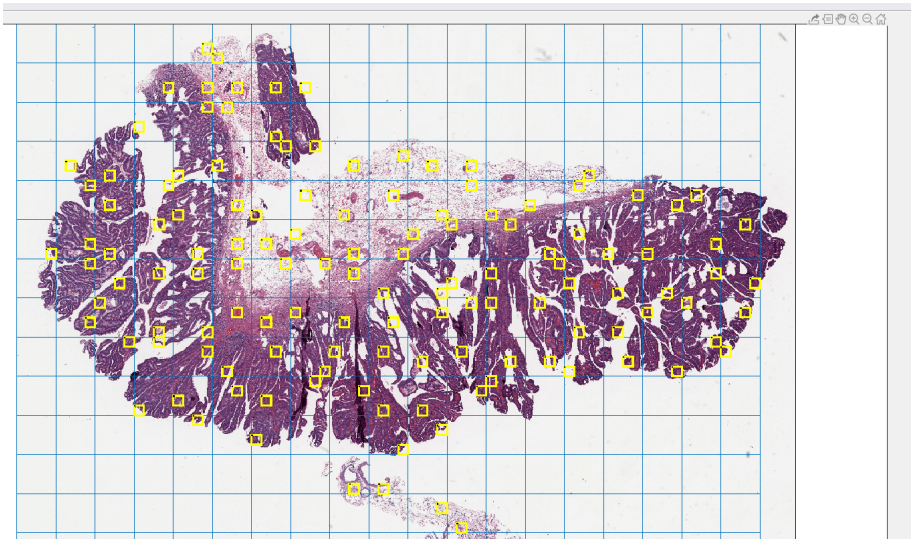

Figure 4.14: Whole slide image showing tiles sampled in detection

cells and fibroblasts. If possible the following structures were included in at least one of the three tiles: crypts containing normal cells, tumour cells, stromal material containing cells and regions containing inflammatory cells.

Table 4.2: Normalisation Methods

Normalisation
Raw
Ruifrok and Johnson
Khan
Macenko
Vahadane
'Naive'

4.2.1 Preprocessing: Calculation of Colour Statistics

For each normalisation method of interest various statistics were needed. These might include colour statistics for foreground and background pixels (average colour intensities), stain vectors and various percentiles. Statistics were calculated for both reference (training) data and for the whole slide images selected by the test harness. For the six types of colour normalisation only four sets of statistics needed to be calculated (Ruifrok, Khan, Macenko, Vahadane). This was because the 'raw' type does not use statistics at all, and the Naive algorithm can use the statistics generated for the Ruifrok algorithm.

Not all pixels in the WSIs were used in the generation of statistics. Instead, when hand marking, the user selected cell locations detected by the SRS algorithm as described in Subsection 4.2.2. Patches around the selected cell locations were used to compute the colour statistics to be used in normalisation. Small subpatches were randomly sampled from the patches and colour statistics of the sample set were computed.

4.2.2 Hand Marking for Classification

The selected tiles were hand marked for classification using the interface displayed in Figure 4.15. When hand marking, the user viewed the tile with previously detected cells shown in white-edged squares. The tile was displayed without the white-edged squares in a second figure: this helped the user to see the image without the distraction of the white edges. The user marked up to forty nuclear patches with their type - ten epithelial, ten inflammatory, ten fibroblasts and a few 'other' cells. The user marked only cells that that they could classify with confidence. Marked patches were saved for use in normalisation and prediction. In accuracy calculations the markings were referrred to as *ground truth*.

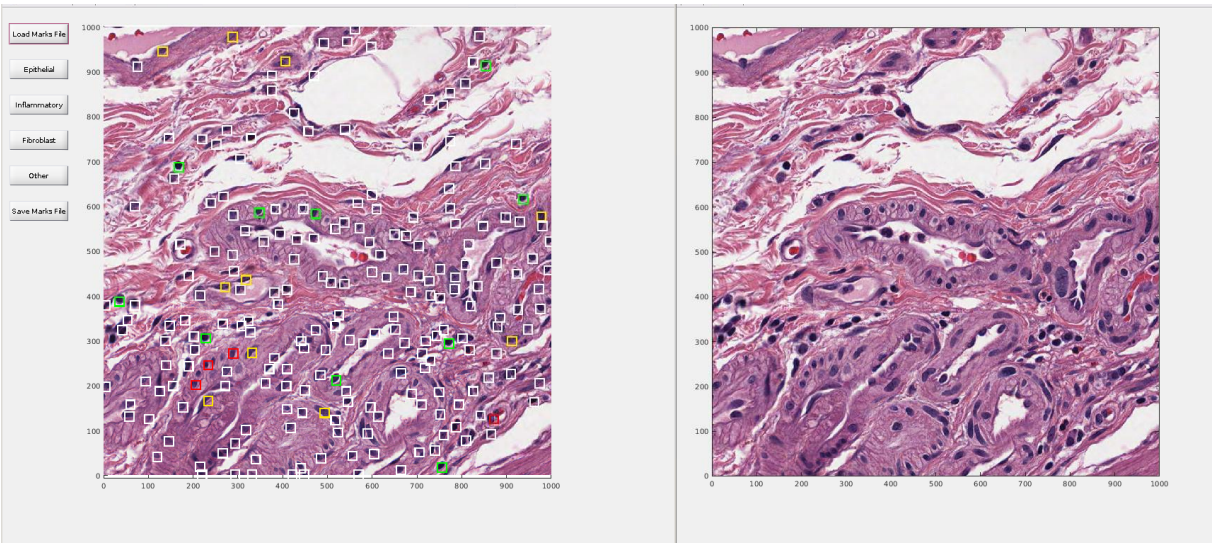

Figure 4.15: Selecting cells in a tile and hand marking them

4.2.3 Normalisation for Classification

For each normalisation technique the selected patches were normalised to the training statistics, using the WSI statistics applicable to the normalisation type. The normalised patches were saved for prediction. The trained classification CNN from

'Cell' predicted the classes of the selected patches. The patches' predicted cell types were compared with the hand marked classes. Confusion matrices were computed and accuracy values were extracted.

4.3 Results - Normalisation for Cell Classification

This section presents results for each normalisation technique.

4.3.1 Raw Data

Table 4.3 displays the confusion matrix applicable to raw data. The entries in the table add up to 100, so that if classification was perfect the entries on the leading diagonal would sum to 100. Rows show how 'true positives' have been classified and columns show how the predicted classification values correspond to the ground truth values. For example, for every 100 cells 16.9 of them are 'true positive' fibroblasts but have been classified incorrectly as 'Other' cells. The average accuracy over all cell types is 36.5%. The classification errors were particularly large for epithelial cells, most of which were classified as inflammatory cells. As previously observed, in most sites the diagnostic images are darker looking than the training data images. In the training data set inflammatory and 'Other' cells are dark coloured, so it is not too surprising that many cells are classified as inflammatory or 'Other'.

Table 4.3: Raw confusion matrix

	Epith	**Inflam**	**Fibro**	**Other**
Epith	8.7	25.3	2.2	1.8
Inflam	0.7	24.4	0.1	7.9
Fibro	1.2	5.4	1.8	16.9
Other	0	2.1	0	1.6

4.3.2 Ruifrok Normalisation

The Ruifrok normalisation accuracy was 69.7% - see Table 4.4. This is a marked improvement on the scores for raw images.

4.3.3 Khan Normalisation

Khan normalisation (Table 4.5) has an accuracy of 64.4%, similar to that of Ruifrok normalisation. The errors that do arise are due to about half the fibroblasts being classified as epithelial cells, and about a third of inflammatory cells being classed as 'Other' cells.

Table 4.4: Ruifrok algorithm - confusion matrix

	Epith	**Inflam**	**Fibro**	**Other**
Epith	33.5	0.8	0.6	3.2
Inflam	1.4	23.3	0.2	8.1
Fibro	5.4	1.1	11	7.7
Other	0.2	1.8	0.1	1.7

Table 4.5: Khan algorithm - confusion matrix

	Epith	**Inflam**	**Fibro**	**Other**
Epith	33.9	1.3	2.5	0.4
Inflam	4.9	17.6	1.2	9.3
Fibro	10.9	1.1	11.8	1.5
Other	1.1	1.2	0.2	1.1

4.3.4 Macenko Normalisation

Table 4.6 shows how Macenko normalisation performed. The accuracy, averaged over cell types was 83.7%. The three most common cell types had satisfactory behaviour. The small number of 'Other' cells tended to be classified poorly - the most common classification being as epithelial cells.

Table 4.6: Macenko algorithm - confusion matrix

	Epith	**Inflam**	**Fibro**	**Other**
Epith	34.9	0.5	2.6	0
Inflam	2.8	26.4	2.6	1.2
Fibro	0.8	0.8	21.8	0.4
Other	1.8	0.8	0.8	0.6

4.3.5 Vahadane Normalisation

Normalisation using the method described by Vahadane yielded the following results in the test environment (Table 4.7). The overall accuracy was 67.8%. The main contributor to inaccuracy was fibroblasts being predicted to be epithelial cells.

4.3.6 'Naive' Colour Normalisation

The confusion matrix for 'Naive' colour normalisation of images is displayed in Table 4.8. Results of 'Naive' colour normalisation were a big improvement on prediction using raw data. The average accuracy is 80.9%. This is a high value: the results of

Table 4.7: Vahadane algorithm - confusion matrix

	Epith	**Inflam**	**Fibro**	**Other**
Epith	31.9	1.2	4.6	0.4
Inflam	4.3	22.0	4.4	2.3
Fibro	10.5	1.1	13.1	0.6
Other	1.1	1.2	0.6	0.8

'Naive' colour normalisation are comparable to those of the Macenko normalisation, the best normalisation technique that used stain separation.

Table 4.8: 'Naive' algorithm - confusion matrix

	Epith	**Inflam**	**Fibro**	**Other**
Epith	34	0.3	3.3	0.1
Inflam	2.7	23	4.6	2.4
Fibro	1.1	0.2	23	1.3
Other	0.3	1.1	1.4	0.9

Table 4.9: Classification accuracy tabulated by site and normalisation method

Site	Raw	Ruifrok	Khan	Macenko	Vahadane	'Naive'	Site Avg.
AA	65	61	39	75	38	78	60
A6	23	72	45	84	76	79	63
CM	42	68	80	76	69	79	69
D5	40	75	63	86	80	84	72
G4	28	72	84	90	82	85	74
AZ	27	64	81	82	67	79	67
F4	17	54	78	80	46	73	58
CK	33	69	85	86	71	76	70
AD	35	69	41	75	66	77	61
AY	37	67	42	89	57	88	63
Avg.	35	67	64	82	65	80	65

4.3.7 Disaggregation by Site

Table 4.9 cross-tabulates classification accuracy by site and normalisation method. For all sites, apart from the AA site, classification accuracy for non-normalised patches is poor and varies markedly from site to site. The worst site for raw patches, F4, scores only 17%. In contrast, for the AA site that includes the images that were processed in the preceding chapter the 65% accuracy measure is nearly twice as good as the site average, 35%.

Macenko normalisation and Naive colour normalisation both perform well. They have similar ranges (75% to 90%) for Macenko normalisation and (73% to 88%) for Naive colour normalisation, although the Macenko average at 82% beats the average Naive colour score of 80%.

Most evaluations of stain normalisation and stain augmentation techniques consider only one or two sites. Authors should be careful when making claims about the specific improvements obtained in such situations. The percentage improvement in accuracy is site-dependent in this case, although ranking values are more stable. However, most of the differences between the stain normalisation techniques are consistent between sites and either the Macenko technique or 'Naive' colour normalisation is always a winner.

4.4 Test Harness - Detection

An experiment examining the effect of colour normalisation on the accuracy of the detection component of 'Cell' was undertaken. The experiment used the same set of tiles as had been sampled in the cell classification test harness. A tile typically contained hundreds of cells, potentially imposing a heavy burden on the person doing the hand marking. The effort involved was reduced by having a region within the tile selected for hand marking automatically. This was done on the basis that marking tens of cells rather than hundreds would still yield many hand marked cells per patient and that the overall accuracy of calculations was unlikely to be affected. Figure 4.16 displays the hand marking interface. The figure displays the region to be hand marked as a square marked in white. The user clicks on the centres of nuclei that are in the white square. In Figure 4.16 the clicked points are shown as small white squares. When the use is satisfied the coordinates of the markings are saved in an 'observations' file.

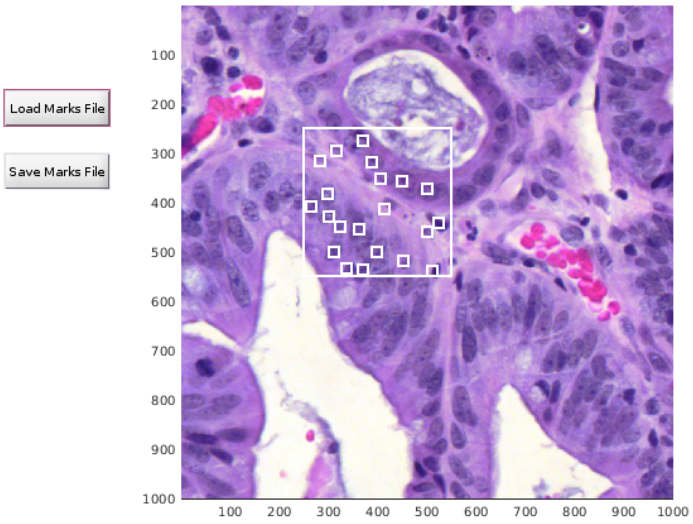

Figure 4.16: Hand marking of nuclei for detection (Patient:AA-3543, Tile:1328)

Tiles from the hand marked set were normalised, then fed to the 'Cell' detection algorithm. Coordinates of the predicted locations that were inside the hand marking square were saved.

Batch runs were carried out for five of the normalisation types listed in Table

4.2. (The Khan algorithm was not included because the 'Cell' detection algorithm already included Khan normalisation.)

A predicted location was scored as a *true positive* if a hand marking location was found close to the prediction, in this case within a distance of 5 μM. The Hungarian algorithm [112] was then used to assign predictions to observations and to decide if a particular prediction was correct.

The effectiveness of normalisation was then assessed by calculating values of *precision*, *recall* and *F1* using the numbers of true positives TP , false positives FP and false negatives FN. Precision is defined by:

$$precision = \frac{TP}{TP + FP} \tag{4.18}$$

Recall is defined by:

$$recall = \frac{TP}{TP + FN} \tag{4.19}$$

And the F1 score combines precision and recall:

$$F1 = \frac{2.precision.recall}{precision + recall} \tag{4.20}$$

Tables 4.10, 4.11 and 4.12 display precision, recall and F1 respectively, crosstabulated by site and normalisation type.

Table 4.10: Precision values displayed as percentages

Site	Raw	Ruifrok	Macenko	Vahadane	'Naive'	Average
AA	71	75	78	79	81	76.7
A6	67	76	69	75	77	73.1
CM	75	84	77	80	90	81.2
D5	88	87	85	85	90	87.2
G4	83	80	79	79	81	80.5
AZ	71	76	69	74	79	73.9
F4	68	72	70	72	72	70.8
CK	76	84	79	81	84	81.0
AD	75	80	79	74	82	78.0
AY	63	71	73	74	80	72.2
Average	73.9	78.5	75.9	77.3	81.7	

It can be seen that values of precision were improved by colour normalisation. Unfortunately, while precision was improved, recall was not.

Table 4.11: Recall Values Expressed as Percentages

Site	Raw	Ruifrok	Macenko	Vahadane	'Naive'	Average
AA	71	72	64	59	65	66.1
A6	82	71	80	74	55	72.5
CM	70	55	65	62	37	58.0
D5	48	43	52	44	38	45.0
G4	59	66	69	69	57	63.9
AZ	70	61	74	65	51	64.2
F4	73	66	74	70	58	68.2
CK	80	64	81	70	53	69.5
AD	64	57	64	40	42	53.5
AY	68	62	75	69	58	66.1
Average	68.6	61.7	69.8	62.3	51.3	

Recall values, displayed in Table 4.11 were worse for the Ruifrok, Vahadane and 'Naive' normalisation algorithms. The Macenko algorithm had approximately the same value as for 'Raw' data.

Table 4.12: Detection: F1 values expressed as percentages

Site	Raw	Ruifrok	Macenko	Vahadane	'Naive'	Average
AA	71%	73	70	68	72	70.8
A6	74	74	74	75	64	72.2
CM	73	66	71	70	53	66.5
D5	62	57	65	58	53	59.2
G4	69	72	74	74	67	71.1
AZ	71	68	71	69	62	68.2
F4	71	69	72	71	64	69.3
CK	78	73	80	75	65	74.2
AD	69	67	71	52	56	62.8
AY	65	66	74	71	67	68.7
Average	70.3	68.5	72.2	68.3	62.3	

The F1 metric combines precision and recall. Table 4.12 displays F1 crosstabulated by site and normalisation type. Values are given as percentages. Macenko normalisation beats unnormalised data but the difference is small and may be attributable to sampling effects.

There was no noticeable improvement between detection using raw data and using normalised data. This may be explained by the fact that in the 'Cell' detection algorithm stain separation is used to calculate the value of the haemotoxilyn channel which is then standardised before being input to the detection CNN.

4.5 Related Work

4.5.1 Stain Normalisation

Most stain normalisation techniques are similar to those presented here. They can vary in the way that stain vectors are calculated, or in the statistics used to pull the optical density maps towards the training maps.

[63] present a stain decomposition method that operates in OD space, but uses only the colour components of OD points to extract the stain vectors. A point in OD space is projected onto the Maxwellian Chromacity Plane, specifically a colour triangle. The presence of two stain components is indicated by a pattern of two clusters. The EM algorithm is used to identify the clusters, and thereby derive estimates for the stain vectors. Stain separation can be carried out in the usual way. The authors model the charge-couple sensor noise, thereby improving model accuracy. The method results in a set of optical density maps (over the region of interest), maps which may be quantified in various ways. The stain normalisation method developed by [14] operates in HSD colour space. It has been used by [35] in the classification of colorectal WSIs into nine different region types, such as tumour, stroma, lymphocytes, etc. Stain normalisation improved the accuracy from 50% to 75%.

4.5.2 Stain Augmentation

When a CNN is being trained the data space can be expanded by adding extra training points. The training points are generated by pushing existing points into new regions in the colour space, either by directly scaling RGB values or by doing *stain augmentation*. [165] trained a CNN to detect mitosis and used stain augmentation. They carried out stain deconvolution on sample images, then modified each channel individually with random stretches and biases before adding the OD channels and transforming the OD coordinates back to RGB. Stain augmentation contributed 0.4 to the best F1 score of 0.6 obtained with unseen data. In a later publication Tellez et al. [166] compared the effects of stain augmentation with those of stain normalisation. Similar results were obtained from both techniques, with the authors reporting that applying both techniques together improved results even further.

4.5.3 Adversarial Networks

Domain adversarial networks (DANNs) introduced by Ganin et al. [60] are forms of GANs, introduced at the start of this chapter. They use labelled data (the training data) and unlabelled data (the data from a new site for which predictions are wanted). The idea is to extract the statistics of the two sets of data and to use the information to improve accuracy. The main network has an extra classifier added to it, a classifier which is trained so that it does not discriminate between the data sources. As a result the predictions for the new site are improved. Lafarge et al. [113] use DANNs to train a mitotis-detecting CNN. The effects of using stain augmentation (SA), stain normalisation (SN) and a DANN were compared. All methods yielded improvements: baseline F1 = 0.33, SA F1 = 0.58, SN F1 =0.46, DANN = 0.55. SA and DANN were joint winners. Cycle-Consistent adversarial networks [189] were applied by [42] in renal histopathology, approximately doubling the accuracy obtained.

4.6 Discussion

In the case of cell classification the experiments with diagnostic images from ten different TCGA sites resulted in clear improvements resulting from all the normalisation techniques tested. The same improvements were not observed with cell detection where the detection algorithm executed colour normalisation internally.

Estimates of the haemoeotoxilyn vector tended to be better than estimates of the eosin vector. This may not always cause problems, but it would be useful to characterise cases that result in poor performance. In addition these results apply to cell identification algorithms and should be treated with caution if normalisation is being done for other objects.

Three of the algorithms that have been tested (Ruifrok, Macenko and Vahadane) use a robust upper range value R to determine the scaling factor that is used to transform optical density values in test space to training space. This approach uses the optical density of the darkest pixels: usually these pixels are in the nuclear regions of inflammatory cells. If, as occasionally happens, there are very few inflammatory cells in the tissue then R does not correspond to inflammatory cells and as a result the transformation may not be correct.

Similar arguments may apply to the use of central statistics, such as the mean and mode of optical density pixels. In some cases of high-grade cancer the epithelial cells which comprise the bulk of the cells in the image look very washed out, large

and distorted. Using their mean intensities in normalisation will tend to result in a normalised image that is darker than it should be. Central statistics are used in the Khan algorithm and in the (pure) colour normalisation algorithm, so they may be vulnerable to this effect.

Divergences from biological assumptions are challenging for normalisation algorithms. The problem is that these divergences are not always easy to identify. For example, if there are very few inflammatory cells in a tissue section then the robust range in the normalising transformation will be incorrect: it will be less than the value that would have been measured if inflammatory cells were present. Possibly there are recognisable features which are present in every WSI and whose underlying colour intensity is conserved. These could be used to calibrate each image. For example the colour intensities of fibroblasts or stroma might be effectively invariant and usable for normalisation. A workaround might be to include stain augmentation at training time. This might do the same work as stain normalisation without there being a need to specify the normalisation transformation explicitly.

It is an open question, as to which of the three general methods, stain normalisation, stain augmentation or adversarial methods yields the best results. Future work would be to repeat this exercise with both stain augmentation and with adversarial networks. We remark that although colour normalisation and colour augmentation are effective tools on their own, practical applications might benefit from the use of both methods in cell identification.

In this chapter various stain normalisation methods were systematically compared with respect to cell identification. The methods included three based on colour separation and one, a 'Naive' standardisation method which operated on the three colour channels directly. Input came from ten sites in the TCGA data repository. For cell classification the two best performing methods, Macenko standardisation and 'Naive' standardisation, performed well across all sites. The good performance of 'Naive' standardisation, a straightforward approach, suggests that it should be considered seriously due to its speed and simplicity.

It was found that the accuracy of cell classification varied markedly across all sites, indicating that the evaluation of colour normalisation techniques should include testing across a large and varied a range of data sets as possible.

Experiments were also carried out, evaluating the effect of performing colour normalisation before detection. Improvements, if any were marginal, indicating that the internal normalisation operations in the 'Cell' detection component were already

performing adequately.

Chapter 5

Molecular Expression: From Image Stacks to TCGA

This chapter analyses multiplexed images output by the Toponome Imaging System (TIS), a robotic system described in Schubert et al. [151]. The robot treats a tissue section to rounds of treatment with *antigens*, chemicals associated with complex molecules of interest, mainly proteins. Exposure to fluorescent light results in a grayscale image which reflects the concentration of the molecule of interest. The patterns made by *pairs* of antigens allow us quantify their degree of *colocalisation*, their tendency to be located in the same place (Dunn et al. [49]). Colocalisation is important because its presence suggests that the molecules are involved in the same chain of interaction.

The chapter has two parts. In the first part TIS data was analysed. Bivariate analysis associated with colocalisation was extended to multivariate colocalisation, using *probabilistic graphical models (PGMs)* (also known as Markov random fields). Clustering was applied to TIS images. In the second part of the chapter various clustering algorithms were applied to molecular data from the TCGA colorectal data sets. An algorithm was developed, *BHC-NW*, which is an extension to Bayesian Hierarchical Clustering [81].

5.1 TIS: The Imaging Robot, Tags and Stacks

The imaging robot takes a tissue section as input, applies various reagents to the section and outputs images that reflect the concentration of those reagents in the tissue section.

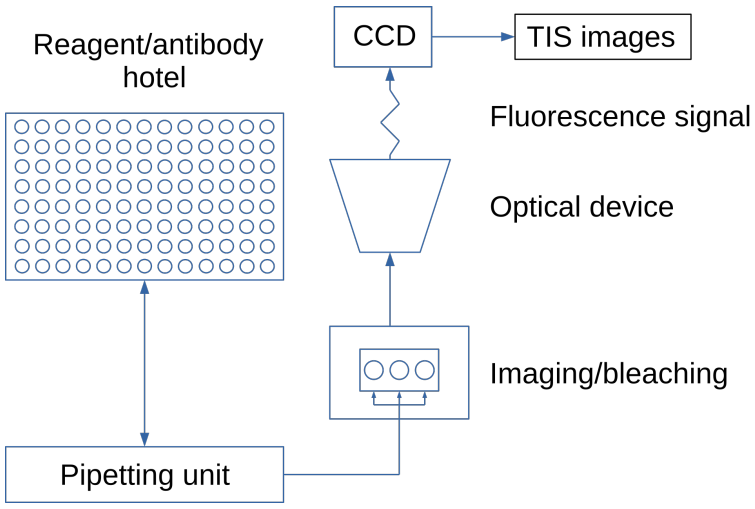

Figure 5.1: TIS Imaging Robot (After [56])

A detailed account of the operation of the TIS imaging robot may be found in Friedenberger et al. [56]. To operate the robot experimenters select a collection of reagents, *tags* of interest. Examples of tags include antigens associated with proteins which are known or suspected to participate in certain chemical processes. Some antigens are chosen because they can detect proteins which bind to specific cell organelles; others detect proteins which are cancer markers. Tags also include reagents such as DAPI which binds to nucleic material.

The selected chemical reagents are placed in the robot's 'tag hotel'. See Figure 5.1. A slice of tissue is excised from an existing tissue block and placed on a slide which is put into the robot. The robot then subjects the tissue to a list of processing cycles. Each cycle in the list is carried out as follows. A specific tag is selected from the tag library by a pipetting unit and applied to the tissue. The resulting fluorescent activity is registered by a CCD camera and the image is stored. Subsequent bleaching and washing of the tissue sample removes (most) fluorescent activity from the tag and allows the cycle to be followed by processing cycles that use the other tags in the 'tag hotel'. In an overview, Schubert et al. [152] state that the choice of 'soft bleaching' resulted from a feasibility study in the 1980s, being the key to 'high reproducibility of data'. In the 'soft bleaching' process the bleach removes fluorescence, but does not remove the other chemical bonds between molecular groups in the tissue, enabling the robot to perform repeated applications involving different tags but the same slice of tissue.

The final result of a robot run is a stack of image sets, one image set for each

tag application. Each image set contains two grayscale images, plus a phase-contrast image. The first grayscale image is the fluorescence pattern that the sample exhibits before the tag is applied and the second image is the fluorescence pattern produced by the tag and which reflects the spatial distribution of the protein or biological entity associated with the tag. The images created by a robot run are subject to pre-processing. An image registration algorithm is used to align images from different cycles - see [153]. In addition, some regions may contain artifacts and in such cases a mask may be created which identifies the affected areas in the image, allowing them to be excluded from calculations.

5.1.1 Image Stacks

Eleven stacks were analysed in this chapter. Stacks were labelled by a combination of patient identifier one of five (13, 15, 17, 18, or 20), a designation of cancer (a) or normal (b), followed by a sample number (1 or 2). For example, the first stack 13a2 was for patient 13, labelled 'a' for cancer and was sample 2. There were four possible stacks for each patient: a1, a2, b1, b2, but not all were available. Six stacks were labelled as normal tissue and five were labelled as cancer.

$$\text{Stacks} = \{13a2,\ 15a1,\ 15a2,\ 15b1,\ 15b2,\ 17b1,\ 17b2,\ 18a2,\ 20a2,\ 20b1,\ 20b2\ \}$$

The number of patients is small, but most analysis here involves measurements associated with individual nuclei. There are approximately 3,000 nuclei, allowing for analysis of general characteristics.

5.1.2 Tag Selection

Tags referred to in this chapter were selected by biologists and are listed in Table 5.1. Many antibodies were connected with stem cell proliferation, as well as general cancer markers, CEA and P53. Surface proteins and cyclins, plus the mucin protein MUC2 were included. DAPI was also used in the tag library, to segment nuclei.

Table 5.1: Tags used in Analysis of TIS Stacks

CD133	Cell transmembrane glycoprotein involved in regulation of stemness, associated with cancer local recurrence and survival. Stem cell protein. [188]
CD166	Cell adhesion molecule associated with the development of adenoma to carcinoma. [188]
CD24	Cell adhesion molecule [188], [181]
CD36	Scavenger receptor: a mechanism by which cells recognize, phagocytose and clear damage and debris through broad pattern recognition [75]. These receptors are well characterized on immune cells. They are also expressed by non-immune cells and are associated with lipid metabolism.
CD44	Cell surface glycoprotein involved in malignant progression, cell adhesion and migration. Associated with less sensitivity to apoptosis signals and more resistance to therapies. [188]
CD57	Human NK cells are lymphocytes with expression levels CD3-, CD56+, CD16(plus or minus). Lymphocytes with high CD57 expression are highly cytotoxic. Presence is generally beneficial.
CEA	Carcino Embryonic Antigen. Glycoprotein. Found in normal tissue of colon, in epithelial cells and goblet cells. Clinical marker of colon cancer [17].
Cyclin A	Cyclins govern cell proliferation, regulating transitions through key checkpoints of the cell cycle. Cyclin A controls the transition to mitosis [12].
Cyclin D	The transition from the G1 phase to the S phase of the cell cycle is regulated by Cyclin D1. It is overexpressed in many tumours [12].
Muc2	Forms protective layer of colon. Expression suppressed in non-mucinous adenocarcinomas. Associated with cancer development, including metastasis [84].
CK19	Keratin. One of the main cytoskeleton proteins of epithelial cells. Breast cancer cells release CK19 [9].
CK20	Cytoskeleton protein. Metastasis marker. Distinguishes between different tumour types [135].
EpCAM	"Cell adhesion molecule involved in Cadherin-Catenin and Wnt pathway, associated with lymph node metastasis, vascular invasion and distant metastasis ." [188].
P53	"Acts as a tumor suppressor in many tumor types; induces growth arrest or apoptosis depending on the physiological circumstances and cell type." [168].

5.2 Related Work

In the literature the analysis of stacks created with TIS has concentrated on creating summary features that characterise such multiplexed images. Motifs extracted from image stacks, *molecular coexpression patterns* (MCEPs), were identified in [13]. These motifs are similar to item-sets found when mining for association rules (Agrawal et al. [6]). A web-based visualisation tool that displays MCEPs is presented in [105].

Experimental use of the TIS system in the analysis of colon cancer tissue has been reported in various publications. Clustering was used to identify MCEPs, using two stacks, one of cancer tissue, and one of normal tissue [87]. The MCEPs divided naturally into two groups: one of cancer cells and one of normal cells. Clustering using a non-linear embedding [100] used three tissue stacks and found many clusters so the reported results are not directly comparable to those reported in this thesis. Cell phenotyping using TIS image stacks was carried out by [98].

Humayun et al. [87] describe the use of clustering in TIS data. They find a clear difference between the clusters assigned to normal tissue and those assigned to cancer tissue. However, the presence of *batch effects* cannot be ruled out here, and the technique needs to be applied to more stacks in order to be validated fully. Khan et al. [97] applied a locality-preserving dimensional reduction technique to data from three 12-tag stacks, two from normal tissue and one from cancer tissue. The nuclear regions in the images were identified and segmented. For each nucleus the average intensity for each tag was computed, resulting in a vector of intensities. The data were subject to dimensionality reduction and the reduced data were then subjected to clustering techniques. The resulting clusters discriminated well between cancer cells and normal cells. In an extension of this work Khan et al. [98] identified five phenotypes in cancerous cells and fifteen phenotypes in normal cells and identified the most significant tags.

Another clustering approach has been used by [105] as part of WHIDE, a Web-based Hyperbolic Image Data Explorer. The WHIDE software is designed to display and process image stacks and other forms of *multivariate bioimages (MIBs)*. WHIDE has four main functions. In the first place WHIDE gives an overview of the image using a pseudocolour visualisation. In the second place, WHIDE supports the identification and display of MCEPs. In the third place, the display technique enables users to apprehend differences and similarities between MCEPs. Finally, WHIDE enables the user to filter and zoom according to both tissue type and protein colocation.

Kovacheva et al. [109] have carried out an analysis of the interactions between proteins using the eleven image stacks already described. They identified protein pairs with significantly higher coexpression.

Note that more examples of related work are described in the rest of this chapter, in sections where the context is more fully discussed.

5.3 Colocalisation

Colocalisation is discussed in this section, starting with a brief survey of bivariate colocalisation, followed by a discussion of multivariate colocalisation techniques that use TIS data. The main approaches are the use of Pearson correlation and clustering techniques.

5.3.1 Bivariate Colocalisation

Most of the existing literature on colocalisation concerns the analysis of pairs of proteins (as opposed to sets of proteins) and bivariate analysis is used. Both [22] and [5] discuss the techniques that are in use.

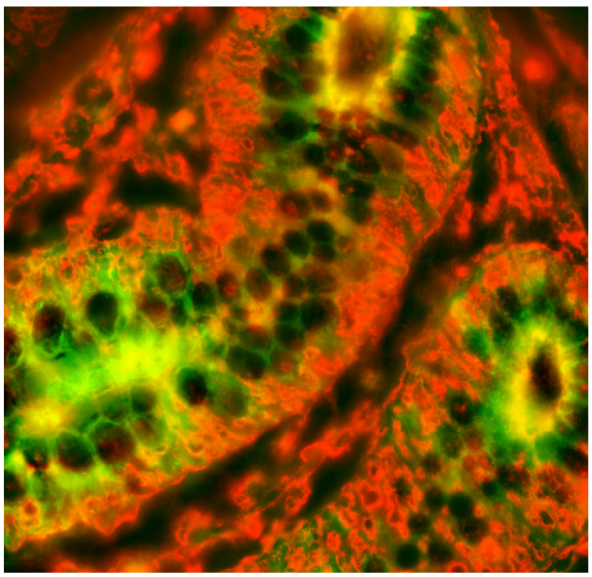

Figure 5.2: Overlay of Fluorescence Images of Normal Tissue:CD133 and CEA: Stack 15b1

The simultaneous application of two tags to a sample allows the calculation of bivariate colocalisation in fluorescence microscopy. The tag associated with the first protein fluoresces in one part of the spectrum and the tag for the second protein fluoresces in another: the procedure yields two grayscale images. Conventionally, one image is mapped to a red monochrome image and the other is mapped to green. In the combined image where the red and green images are overlaid, areas of high overlap are yellow, allowing the viewer to deduce that these are locations where the two proteins may interact.

Figure 5.2 shows such an overlay. The images belong to a stack created from a TIS processing run on a sample of normal tissue. Tag CD133 was used in the 'red' image, while the second 'green' image was produced by the tag CEA. Two crypts are clearly visible and there appears to be colocalisation at the *apical* (closed) ends of the crypts.

Qualitative analysis gives useful insights into protein-protein interactions but it has disadvantages. Simple image processing may be used to manipulate the profile of intensity levels of the red and green channels, increasing the chances that the image contains yellow regions that are not significant. In addition, experts may differ in their interpretation of the data. Objective measurements of colocalisation are therefore desirable. Several in common use are described below.

5.3.2 Pearson Correlation

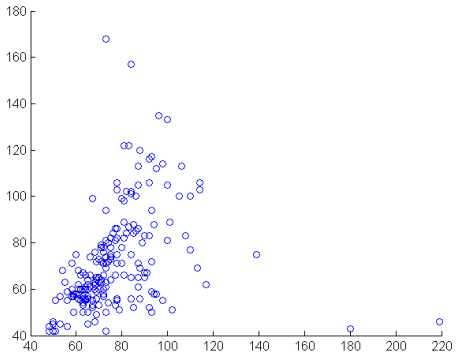

Figure 5.3: Scatterplot of Intensity of tag CD133 vs tag CEA in Cancer Stack 18a2

Figure 5.3 is a scatterplot created using two images of tumour tissue, images for proteins CD133 and CEA. Pixels have been sampled randomly from the aligned images and each point on the scattergram is associated with a sampled pixel, with

the x value being the grayscale intensity of a pixel from CD133 and the y value being the corresponding grayscale intensity of the matching pixel for CEA. It is clear that x and y values are associated. Scatterplots are useful in giving an intuitive visualisation of the relationship between two tags. Some of the metrics described in the following sections can be interpreted in terms of the properties of scatterplots. For example, the Pearson correlation coefficient, a frequently used metric, defines a straight line which provides a visual description of the relationship.

The Pearson correlation may be defined as follows. Consider two images, I_A and I_B with n_R rows and n_C columns. Assume that in I_A the intensity has the distribution $f(x, y)$, and image I_B has the distribution $g(x, y)$.

The mean intensity of f, denoted by $\overline{f}$ is:

$$\overline{f} = \frac{\sum_{x,y} f(x, y)}{n_R n_C} \tag{5.1}$$

The standard deviation of f is s_f defined as:

$$s_f = \sqrt{\frac{\sum_{x,y} (f(x, y) - \overline{f})^2}{n_R n_C - 1}} \tag{5.2}$$

For the spatial intensity distributions, $f(x, y)$ and $g(x, y)$ the Pearson correlation r between f and g is defined as follows:

$$r = \frac{\sum_{x,y} (f(x, y) - \overline{f})(g(x, y) - \overline{g})}{(n - 1) s_f s_g} \tag{5.3}$$

The Pearson correlation [155] is in fairly common use in fluorescence microscopy. Various authors describe modifications designed to reduce spurious effects. For example [36] have produced an algorithm in which low-level intensity values are regarded as background noise and excluded from calculations. An automatic method is used to compute the threshold used to decide if a pixel belongs to the background.

Note that the usual statistical theory dealing with the Pearson correlation assumes that the value of $f(x, y)$ is independent of values at nearby points. In practice biological images are heavily structured. We expect that if $q = (x + d, y + e)$ is a point near $p = (x, y)$, i.e. if d and e are small in absolute value, then $I(q)$ will be close to $I(p)$ i.e the two values will *not* be independent. When the Pearson correlation is applied to spatially correlated data it is desirable to calculate statistics in a way that takes spatial correlation into account.

5.3.3 Multivariate Colocalisation

Multivariate analysis is needed to treat TIS data, where stacks contain more than two tags. The modelling of multiway interactions requires many more parameters than two-way interactions and the accuracy of multivariate interaction models decreases as their complexity increases. In multivariate modelling the experimenter usually builds simpler models first and gradually increases the model complexity until some stopping rule is satisfied.

5.3.4 Combinatorial Molecular Pattern Technique (CMPs)

The inventors of the TIS machine [152] describe the *Combinatorial Molecular Pattern Technique* (CMP), a pixel-based technique. The method is designed to detect frequently occurring combinations of proteins. It is applied to a stack of images as follows. For each tag t in the stack a threshold T_t is chosen. A pixel which has intensity values $(f_1, f_2, ..., f_{nT})$ is assigned the binary vector $\boldsymbol{b}$. In $\boldsymbol{b}$ an element corresponding to tag t is 1 if the intensity of is greater than T_t: otherwise it is zero. The algorithm detects frequent patterns in the collection of vectors: a combinatorial molecular pattern (CMP) is a vector $\boldsymbol{p}$ of n_T elements in which the tth element is either 1, 0 or *(the 'don't care' or 'indifferent' word). Vector $\boldsymbol{b}$ matches a given CMP pattern $\boldsymbol{p}$ at element t if $\boldsymbol{p}[t]$ is *; otherwise it matches if both $\boldsymbol{b}[t]$ and $\boldsymbol{p}[t]$ are 1 or if they are both 0. If $\boldsymbol{b}$ matches $\boldsymbol{p}$ at all elements then it matches $\boldsymbol{p}$. The authors go on to group CMPs together to form *motifs* - motifs always share at least one lead protein. The authors define a *lead protein* to be a protein which has the value 1 for all CMPs in the motif. (That is, it is always overexpressed in the motif.)

[151] developed CMP motifs in different situations, using *toponome maps* to identify interesting functional regions in images of skin disease contrasted with images of normal skin. Toponome maps of TE671 rhabdomyosaroma cells taken during the migratory state showed the presence of significant CMPs. Another application, that of analysis of the murine hippocampus, is described in [20].

It is not clear how repeatable this method is, given that threshold selection is carried out manually by the user and given the variability in the data from one sample to the next. [151] addressed the issue by inverting the sequence of tags and having independent experts assign thresholds. They reported that closely similar

results were obtained.

The CMP approach has been applied to both cancerous tissue and histologically normal tissue from the colon in an exploratory study by [17]. In that study 6,813 CMPs were found in cancer tissue and 32,009 CMPs in normal tissue. The authors identified five potential cancer stem cells using specific CMP motifs with CD133, CD44, EpCAM and CD166 as lead proteins.

5.3.5 Pixel Protein Profiles

Similarity Mapping (SIM) is a technique composed of both automatic and manual methods [152]. Each image in a stack is assigned a colour, and the images are merged. Defining a pixel-protein profile (PPP) as the vector of tag intensities at a given pixel, different pixels may be compared for similarity, based on their PPPs. In addition, the merged image reflects the distribution of PPPs in the sample: usually cell structures are clearly visible. In the manual stage of SIM the user examines the merged image. Selecting a pixel with the mouse results in pixels with similar profiles being highlighted. The authors describe an example of the use of SIM in skin samples from patients with psoriatic disease. It was easy to identify PPPs whose presence distinguished between involved (diseased) and uninvolved (normal) regions of skin. The usage of PPPs is based on identification of interesting regions by the user, a subjective process. This subjectivity is a possible limit on the effectiveness of Similarity Mapping.

5.4 Probabilistic Graphical Models for Multivariate Colocalisation

Undirected probabilistic graphical models (also known as Markov random fields) [183] generalise the Pearson correlation statistic. In this section probabilistic graphical modelling is applied to TIS stack data.

A probabilistic graphical model contains both a graph (containing vertices and edges) and a set of probability mass functions. The vertices are the variables of interest and the presence of an edge between two variables indicates a direct interaction between the variables. On its own, without considering the probability mass functions, the graph can be regarded as a network which captures independence relations between the variables. The probability mass functions define numerical

probabilities that indicate the strengths of the interactions.

A simple example of a graphical model may be defined as follows. Assume that there are five variables, which stand for expression levels of a set of tags, say $CEA, EpCAM, Muc2, CK19, CD133$. We may capture the probabilistic behaviour of these five variables by the function $f()$ which maps the variables to the unit interval. By this we mean that if we run some hypothetical experiment the probability of observing the outcome $CEA, EpCAM, Muc2, CK19, CD133$ is:

$f(CEA, EpCAM, Muc2, CK19, CD133)$.

Figure 5.4 displays a possible independence graph for f. If this butterfly graph does apply to f then the following conclusions may be made:

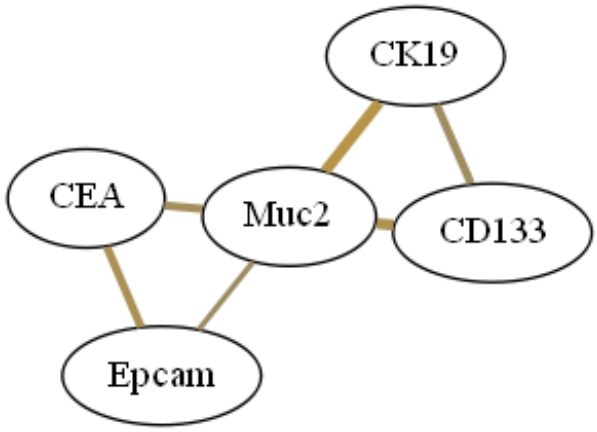

Figure 5.4: Independence Graph for Colocalisation of Five Tags

The function f is *separable*: it may be split into the product of two functions g and h which have only one variable ($Muc2$) in common:

$$f(CEA, EpCAM, Muc2, CK19, CD133) = g(CEA, EpCAM, Muc2)h(Muc2, CK19, CD133) \tag{5.4}$$

Given the value of $Muc2$ the value of CEA is independent of the values of $CK19$ and $CD133$. CEA is independent of $CK19$ (or $CD133$) conditional upon $Muc2$.

Graphical models are interesting in the analysis of real-world data, because the presence or absence of edges in an independence graph may indicate direct or indirect associations in the real world. For example, when we consider the variables

in Figure 5.4 the graph indicates that $Muc2$ is a good predictor of the levels of other tags. In addition the thickness of the links is related to the strength of association. The weakest link shown is that between CEA and $Muc2$ while strong links are associated with the set $\{Muc2, CK19, CD133\}$

In the case of multivariate data it is usual to assume that the data have been generated by sampling from some predefined multivariate distribution. It is usual to fit the data to a graphical model which includes as few parameters as are needed. The theory for multivariate normal (Gaussian) distributions is well-developed and can be seen as a natural generalisation of the theory concerning correlation. It is also related to linear regression which can be used when the aim is to predict one of the variables as a function of the others. (For example, we may wish to predict that a given pixel belongs to a cancer cell given the values of tag variables and we may use linear regression to compute this probability.)

Multivariate data have interesting properties compared with bivariate data. Perhaps the most important difference is that the correlation between variables needs to be defined carefully. There is an important difference between *marginal correlation* and *partial correlation*.

5.4.1 Multivariate Dependencies in Graphical Models

When a stack of TIS images is considered then each pixel is associated with a number of tags - up to around twenty. An obvious way to generalise the use of correlation coefficients is to simply create an array of the coefficients and to generate a network of linkages using those correlations above a threshold value. However, when more than two variables are involved, the value of an edge strength has a more subtle interpretation. In the multivariate case, there is a difference between the marginal correlation and the partial correlation between variables and it is the partial correlation which should be used to indicate edge strength.

Define A to be the set of pixels in the stack which have the combination of intensity values i_1, i_2, i_3. Assume that the number of pixels in each image is n.

$$A = \{x, y : I_1(x, y) = i_1, I_2(x, y) = i_2, I_3(x, y) = i_3\} \tag{5.5}$$

Then we may assert that the chance that an arbitrarily selected pixel has this combination of intensities is:

$$P(i_1, i_2, i_3) = \frac{\sum_{x,y} \mathbf{1}_A(x, y)}{n} \tag{5.6}$$

Let us consider the *conditional distribution* of two variables i_1 and i_2 given a third i_3:

$$P(i_1, i_2|i_3) = \frac{P(i_1, i_2, i_3)}{P(i_3)} \tag{5.7}$$

The *partial covariance* $cov(i_1, i_2|i_3)$ is the covariance of i_1 and i_2 for a given value of i_3. It is defined by:

$$cov(i_1, i_2|i_3) = cov(i_1, i_2) - cov(i_2, i_3)var(i_3)^{-1}cov(i_3, i_2) \tag{5.8}$$

The *partial correlation* of i_1 and i_2 with respect to i_3 is:

$$r_{1,2|3} = \frac{cov(i_1, i_2|i_3)}{\sqrt{var(i_1|i_3)var(i_2|i_3)}} \tag{5.9}$$

In the three-node graphical model which displays independence relationships between three variables an edge is only drawn between two variables if the partial correlation coefficient between them is not identically zero. Partial correlations are indicators of statistical independence in situations where there are more than three variables. In addition, similar arguments apply when the data are binary or ordinal in nature. Note that the theory may be extended to situations where the data refer to objects and interestingly to objects which are linked to each other in some way. [65] describe how heterogeneous objects may be modelled using link mining.

5.4.2 Graphical Models applied to TIS Data

Graphical models were applied to two forms of TIS data. In the first case, existing nucleic masks were applied to the raw image data, so that only pixels from nucleic material were included in the analysis. In the second case, DAPI, which associates with nucleic material, was applied to a tissue section and the resulting image was segmented into nuclei [99]. As a result each data point was a nucleus, accompanied by aggregate feature(s): in practice, the average R, G and B intensities in the nuclear region.

5.4.3 Graphical Models Based on Pixel-Level TIS Data

The eleven available stacks mentioned in the introductory chapter were analysed both separately (i.e a probabilistic graphical model was produced for each stack) and also as pooled data (pooled cancer data and pooled 'normal' data).

In addition, the effect of coarsening the pixel grid was examined. Coarsened patches of various sizes were used. For example, the image could be divided into patches of size (5×5) pixels, and the average intensity of each tag in a patch computed. A minimum number of pixels in the patch had to be included by the masking process for a patch to participate in the calculation of intensities.

The tags used are shown in Table 5.1, and are the same as used in analysis by Khan et al. [97].

Algorithm 5 Pixel-Based Graphical Model

procedure GM(T Images with Grayscale Intensity)
  for each tag t **do**
    $i = 0$
    for each pixel p in a nucleus **do**
      $i = i + 1$
      $f_{it} = Intensity_t(p)$
    end for
    $n_t = i$
    $\overline{f}_t = \frac{\sum_i f_{it}}{n_t}$
    $s_t^2 = \sum_i (f_{it} - \overline{f}_t)^2$
  end for
  for each tag t **do**
    for each tag u **do**
      $r_{tu} = \frac{\sum_i (f_{it} - \overline{f}_t)(f_{iu} - \overline{f}_u)}{s_t s_u}$
    end for
  end for
  $C = r^{-1}$
  r_{part} = normalised C
end procedure

5.4.4 Graphical Models for Individual Stacks

Graphical models were produced for individual stacks using the twelve tags listed above and for a variety of effective pixel sizes. Algorithm 5 outlines the code used to extract a PGM from pixel data. In order to handle spatial correlations the images were coarsened, producing models that varied by patch size. Models computed for the same stack but with differing patch sizes were strongly related to each other. See Table 5.2 below. The many correlation values near 1 indicate that in general the models are very similar. The models for single pixels are least like the models for the patch size of 20×20.

Table 5.2: Correlations between Models with Different Patch Sizes

	1×1	2×2	3×3	4×4	5×5	10×10	15×15	20×20
1×1	1	1.00	0.99	0.99	0.99	0.97	0.97	0.95
2×2	1.00	1	0.99	0.99	0.99	0.97	0.97	0.95
3×3	0.99	0.99	1	0.99	0.99	0.98	0.97	0.96
4×4	0.99	0.99	0.99	1	1.00	0.99	0.98	0.97
5×5	0.99	0.99	0.99	1.00	1	0.99	0.99	0.97
10×10	0.97	0.97	0.98	0.99	0.99	1	1.00	0.99
15×15	0.97	0.97	0.97	0.98	0.99	1.00	1	0.99
20×20	0.95	0.95	0.96	0.97	0.97	0.99	0.99	1
	1×1	2×2	3×3	4×4	5×5	10×10	15×15	20×20

Table 5.3 compares pairs of models by correlating their edge strengths. The correlations may be regarded as a measure of similarity. The average correlation between two models from differing stacks was 0.37 for this set of tags, indicating that there is some overall similarity between all the models. However there was no discernible grouping of cancer stacks or normal stacks. There was some indication of batching effects: the within-patient similarities for patients 15, 17 and 20 were relatively high.

Table 5.3: Correlations between Pixel-Level Graphical Models

	13a2	15a1	15a2	18a2	20a2	15b1	15b2	17b1	17b2	20b1	20b2
13a2	1	0.41	0.31	0.32	0.28	0.36	0.32	0.33	0.26	0.09	0.22
15a1	0.41	1	0.48	0.48	0.33	0.43	0.39	0.49	0.61	0.32	0.34
15a2	0.31	0.48	1	0.28	0.37	0.47	0.29	0.34	0.51	0.14	0.38
18a2	0.32	0.48	0.28	1	0.40	0.32	0.37	0.33	0.37	0.31	0.37
20a2	0.28	0.33	0.37	0.40	1	0.37	0.30	0.33	0.30	0.40	0.59
15b1	0.36	0.43	0.47	0.32	0.37	1	0.54	0.55	0.57	0.18	0.33
15b2	0.32	0.39	0.29	0.37	0.30	0.54	1	0.62	0.49	0.09	0.15
17b1	0.33	0.49	0.34	0.33	0.33	0.55	0.62	1	0.63	0.16	0.33
17b2	0.26	0.61	0.51	0.37	0.30	0.57	0.49	0.63	1	0.16	0.21
20b1	0.09	0.32	0.14	0.31	0.40	0.18	0.09	0.16	0.16	1	0.73
20b2	0.22	0.34	0.38	0.37	0.59	0.33	0.15	0.33	0.21	0.73	1
	13a2	15a1	15a2	18a2	20a2	15b1	15b2	17b1	17b2	20b1	20b2

5.4.5 Graphical Models Based on Nuclear Segmentation

Algorithm 6 describes the generation of a PGM from a set of ministacks. Each ministack contains a stack of grayscale images associated with a single nucleus, a mask being defined for the nucleus using the segmentation. The term I_{ut} refers to the image in ministack u and tag t. In Algorithm 6 the aggregate function f_t is applied to each image resulting in value g_{ut}. Values of g_{ut} averaged over u and corresponding variances are used to compute the correlation matrix which is inverted, normalised and its negative taken. The result is a matrix of partial correlations.

Algorithm 6 Nucleus-Based Graphical Model

procedure GM(n_U nuclei labelled by u, n_T tags labelled by t, n_U grayscale images I_{ut}, n_T aggregate functions f_t)
  for each tag t **do**
    for each nucleus u **do**
      $g_{ut} = f_t(I_{ut})$ ▷ Calculate aggregate function for image
    end for
    $\bar{g}_t = \frac{\sum_u g_{ut}}{n_U}$
    $s_t^2 = \sum_u (g_{ut} - \bar{g}_t)^2$
  end for
  for each tag t **do**
    for each tag v **do**
      $r_{tv} = \frac{\sum_i (g_{ut} - \bar{g}_t)(g_{uv} - \bar{g}_u)}{s_t s_v}$ ▷ Marginal correlation matrix
    end for
  end for
  $C = r^{-1}$ ▷ Invert correlation matrix
  r_{part} = - normalise(C) ▷ Partial correlations
  return r_{part}
end procedure

Nucleic regions were obtained from segmentations described by [97]. For each region the average intensity per tag was calculated. The resulting table of regions versus tags was used to extract a graphical model. A comparison of the models is shown in Table 5.4. In the table each cell displays the correlation between the edge strength of the models and thus can be taken as a measure of model similarity. The average value of the cells (excluding the diagonal) was 0.30, worse than for the pixel-based calculations, but showing some similarity between models. Models obtained for stacks which were from the same patient were often highly similar:

there appeared to be batch effects at work. For example the models for 17b1 and 17b2 (normal tissue from patient 17) had a similarity of 0.57. Models for 15b1 and 15b2 had a similarity of 0.42 and the models for 20b1 and 20b2 had a similarity of 0.67.

Table 5.4: Correlations between Region-Based Graphical Models

	13a2	15a1	15a2	18a2	20a2	15b1	15b2	17b1	17b2	20b1	20b2
13a2	1	0.29	0.20	0.27	0.31	0.37	0.23	0.29	0.15	0.09	0.26
15a1	0.29	1	0.33	0.37	0.33	0.35	0.28	0.41	0.53	0.29	0.3
15a2	0.20	0.33	1	0.18	0.28	0.46	0.14	0.36	0.5	0.12	0.28
18a2	0.27	0.37	0.18	1	0.35	0.33	0.25	0.25	0.32	0.25	0.30
20a2	0.31	0.33	0.28	0.35	1	0.24	0.29	0.31	0.14	0.4	0.41
15b1	0.37	0.35	0.46	0.33	0.24	1	0.42	0.47	0.54	0.21	0.29
15b2	0.23	0.28	0.14	0.25	0.29	0.42	1	0.56	0.44	0.04	0.04
17b1	0.29	0.41	0.36	0.25	0.31	0.47	0.56	1	0.57	0.14	0.18
17b2	0.15	0.53	0.5	0.32	0.14	0.54	0.44	0.57	1	0.18	0.26
20b1	0.09	0.29	0.12	0.25	0.4	0.21	0.04	0.14	0.18	1	0.67
20b2	0.26	0.3	0.28	0.30	0.41	0.29	0.04	0.18	0.26	0.67	1
	13a2	15a1	15a2	18a2	20c2	15b1	15b2	17b1	17b2	20b1	20b2

Table 5.5: Partial Correlations for Pooled Normal Nuclei

	CD133	CEA	Cyclin A	Muc2	CK19	CD166	CD36	CD44	CD57	CK20	Cyclin D	EpCAM
CD133	100	3	49	-18	10	-14	-33	15	14	32	-36	41
CEA	3	100	-15	-2	28	-15	15	-10	2	-32	13	64
Cyclin A	49	-15	100	23	4	26	1	-10	34	-1	14	-2
Muc2	-18	-2	23	100	81	-29	7	52	-0	0	-10	-11
CK19	10	28	4	81	100	16	-15	-33	6	8	16	-4
CD166	-14	-15	26	-29	16	100	-20	35	27	13	-20	20
CD36	-33	15	1	7	-15	-20	100	-16	84	13	-5	10
CD44	15	-10	-10	52	-33	35	-16	100	5	-11	32	6
CD57	14	2	34	-0	6	27	84	5	100	-22	14	-7
CK20	32	-32	-1	0	8	13	13	-11	-22	100	52	-6
Cyclin D	-36	13	14	-10	16	-20	-5	32	14	52	100	-3
EpCAM	41	64	-2	-11	-4	20	10	6	-7	-6	-3	100

Table 5.5 records the partial correlation values (shown as percentages) obtained when regions from normal stacks were pooled. Table 5.6 contains the cor-

Table 5.6: Partial Correlations for Pooled Cancer Nuclei

	CD133	CEA	Cyclin A	Muc2	CK19	CD166	CD36	CD44	CD57	CK20	Cyclin D	EpCAM
CD133	100	-14	14	31	-21	20	20	1	-6	-1	-42	12
CEA	-14	100	14	17	-20	-7	27	-32	-9	-30	3	82
Cyclin A	14	14	100	5	51	29	38	18	-15	-3	13	-10
Muc2	31	17	5	100	47	7	6	19	-3	34	-4	8
CK19	-21	-20	51	47	100	-38	2	12	-0	-12	14	23
CD166	20	-7	29	7	-38	100	8	0	-5	17	30	28
CD36	20	27	38	6	2	8	100	-16	44	17	11	-24
CD44	1	-32	18	19	12	0	-16	100	7	-18	12	-2
CD57	-6	-9	-15	-3	-0	-5	44	7	100	0	-1	9
CK20	-1	-30	-3	34	-12	17	17	-18	0	100	11	1
Cyclin D	-42	3	13	-4	14	30	11	12	-1	11	100	-7
EpCAM	12	82	-10	8	23	28	-24	-2	9	1	-7	100

responding results for cancer stacks. Visualisations of the graphical models are shown in Figure 5.5 and in Figure 5.6. In the graphs, a yellow link between two tags indicates a positive partial correlation; a blue link denotes a negative partial correlation. The width of the link scales with the absolute value of the partial correlation. Stronger yellow is used when the partial correlation is near +1, and when the partial correlation is near -1 stronger blue is used. Partial correlations r_{corr} are represented on the diagrams if their absolute values pass a threshold value of 20%.

In both the normal and the cancer graphs the edge $CEA - EpCAM$ between two cancer markers is one of the highest strength links (N=64%, C=82%). This result is to be expected, but if it had not been obtained questions concerning the usefulness of the PGM approach would have been raised. Links appear to be weakened in cancer tissue, compared with normal tissue. Let us consider the most positive links in the normal graph (excluding the $CEA-EpCAM$ link). In cases of strong positive links in normal tissue, the links are reduced in cancer tissue: for $CD57 - CD36$ the strengths are (N=84%, C=44%, for $Muc2 - CK19$ the strengths are (N=81%, C=47%), for $CD44 - Muc2$, (N=52%, C=19%) and for $CyclinD - CK20$, (N=52%, C=11%). Considering the four most negative links, two change only slightly: $CEA - CK20$ (N=-32%, C=-30%), $CD133 - CyclinD$ (N=-36%,C=-42%) and two links go from negative to weak positive. These are $CD133 - CD36$ (N=-33%, C=20%) and $CD44 - CK19$ (N=-33%,C=12%).

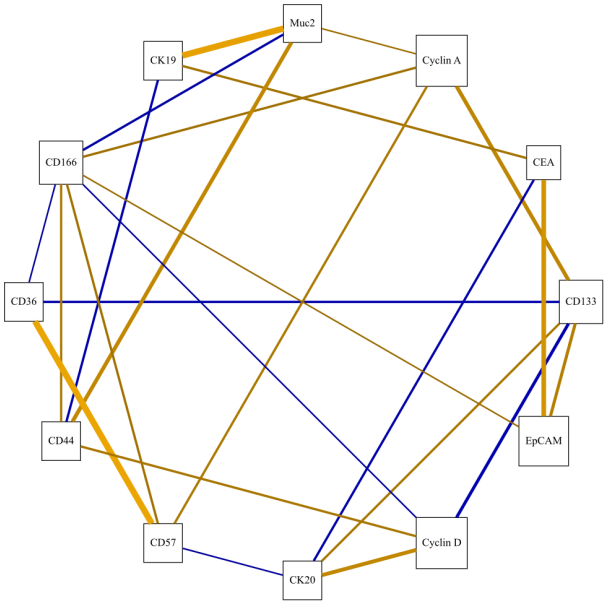

Figure 5.5: Graphical Model for Pooled Normal Data

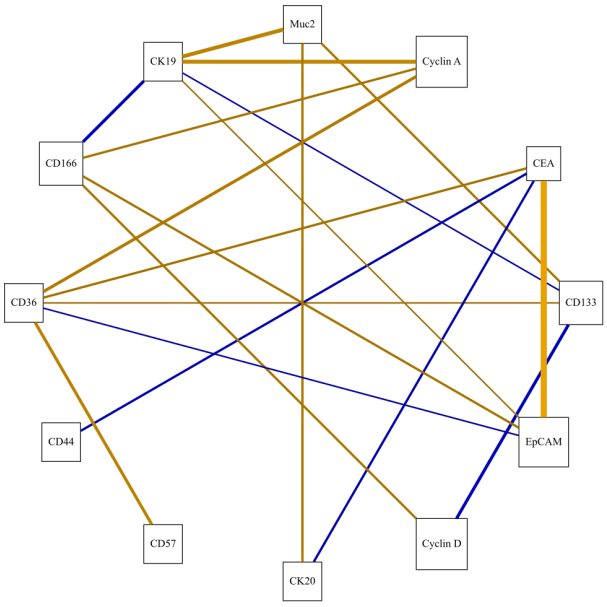

Figure 5.6: Graphical Model for Pooled Cancer Data

Kovacheva et al. [108] introduce a model of interactions between proteins. The model, *DISWOP*, extracts within-nucleus statistics, values calculated using only data from the ministack associated with each nucleus. The statistics reflect the degree of disorder in the nucleus so tag-tag values reflect how much the two tags are in concordance within a nucleus. In contrast, the parameters of the PGM model are calculated using image-wide statistics. The per-cell values are computed using the cell's role as part of the image. For a given tag and nucleus the tag value is the (average) intensity of the nucleus compared with the average tag value in the entire image.

The results presented in Kovacheva et al. [108] were compared with the results shown here. The $CEA - EpCAM$ link was in the top 10% of dependency scores found by DISWOP for both normal and cancer tissue, In addition, in the DISWOP results Muc2 is strongly linked to CD133 in cancer tissue. A link with strength 23% is shown on the cancer graph in Figure 5.6, a link in the top quarter of strength values. Due to differing calculation methods the results of PGM modelling are not directly comparable with the results in Kovacheva et al. [108], but similar results were obtained in some cases.

Due to the very small sample sizes, only six normal patients and five cancer patients, it is not possible to decide if these results would carry over to data sets with more members. However, PGMs are a natural generalisation of bivariate analysis which is grounded in experimental results, so it would be expected that the technique would be generally applicable. In the next section of this chapter clustering methods are used to analyse TIS data.

5.5 Clustering

This section describes the analysis of the TIS stack data set using clustering. In the clustering algorithm, each data point represented a cell nucleus with a set of features, each feature being associated with an antibody tag. For each tag, the same feature was used: the mean intensity of the tag in the region occupied by the nucleus. This was chosen for its simplicity although various texture features could also have been selected. As in the preceding subsection, the segmentation of nuclei described in [99] was used.

Clustering was applied to the data from each stack. EM mixture model clustering Dempster et al. [46] was used with a selection of 12 tags. Figure 5.7 illustrates the results obtained when the EM model (4 clusters) was applied to stacks individually. The six images in the top half of Figure 5.7 have been obtained from normal tissue, while the five images in the bottom half originate in cancer tissue. Four colours are used: red, brown, dark brown (for epithelium) and green (for stroma). In the images where crypts are clearly visible nuclei have been assigned to clusters which have been coloured red, brown or dark brown. Colours were assigned as follows. If only opithelium, those cells were coloured red; if two clusters were assigned to epithelium they were coloured red or brown: if epithelial cells were coloured, red, brown or dark brown, according to cluster assignment. Correspondingly stromal material was coloured green or dark green. The exception to these colour assignments was cancer stack 15a2 which has been displayed with some nuclei coloured blue: this is because it was not obvious whether these nuclei should be assigned to epithelium or stroma. The other four cancer stacks exhibit some crypt-like structures, particularly stack 15a1, so it was possible to assign colours to clusters. To summarise, in the case of normal tissue, clustering has been successful in separating stromal cells from the epithelial cells that comprise the crypt walls. For cancer tissue, however, where the crypt/stroma divide is not always clear, the clusters do not always decisively separate the cells along this divide.

To examine the generality of this approach, images in each stack were standardised (to mean zero, and unit standard deviation) and pooled. Clustering using the EM algorithm for Gaussian mixture models was employed. Figure 5.8 presents the results graphically where the number of clusters is set to four. Four colours were assigned to nuclei according to their cluster membership: red and brown for epithelial cells, green for stroma . Cells were coloured blue if they were assigned to the remaining cluster which appear to be mainly epithelial cells but did include cells located in stroma.

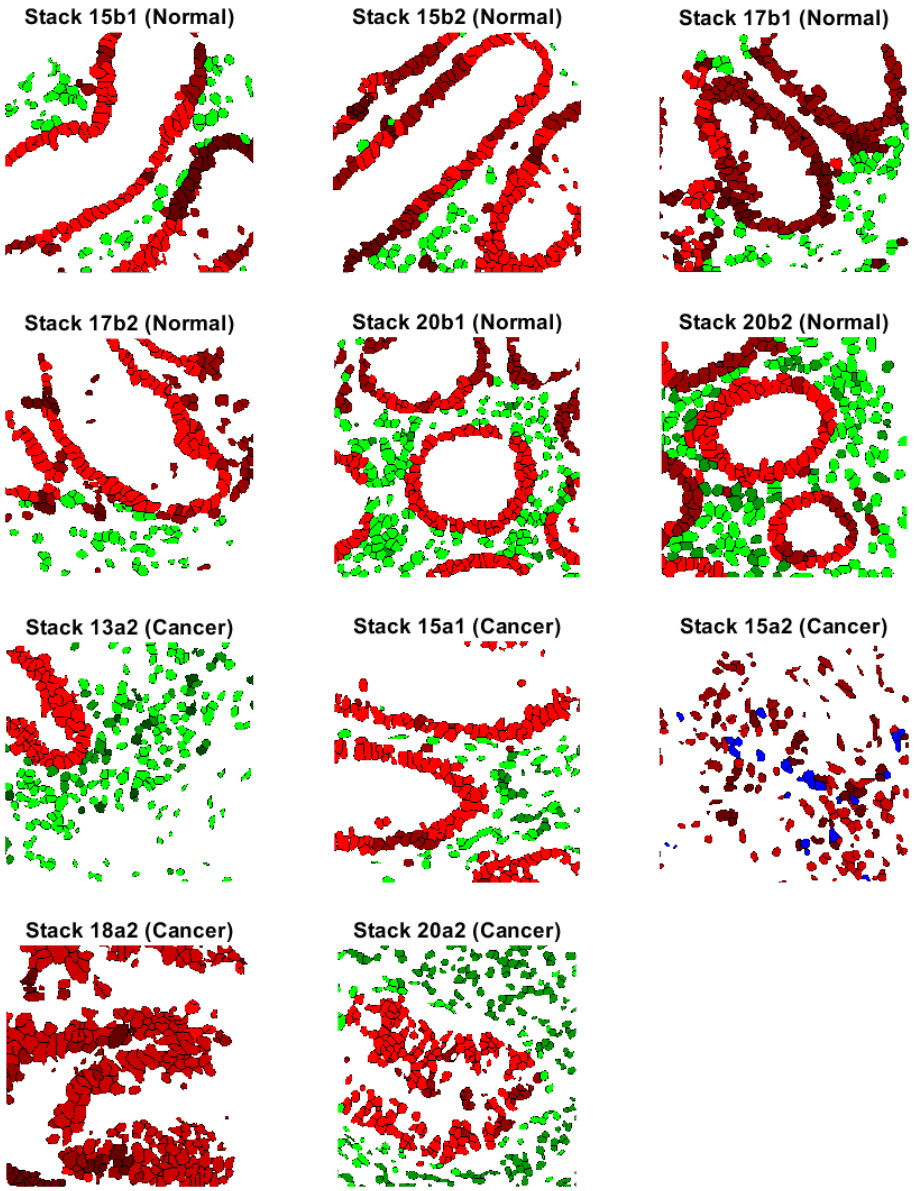

Figure 5.7: EM clustering of eleven individual stacks - 4 clusters

In conclusion the main effect of clustering TIS expression data was to separate epithelial nuclei from nuclei associated with stroma. When data from different stacks were pooled this separation was also observed, indicating that the clustering reflected the underlying biology. It was not possible to decide if a particular cell in the stromal region was an epithelial cell, an inflammatory cell or a fibroblast, so the use of markers to identify the cell type would aid in the analysis of multiplexed images.

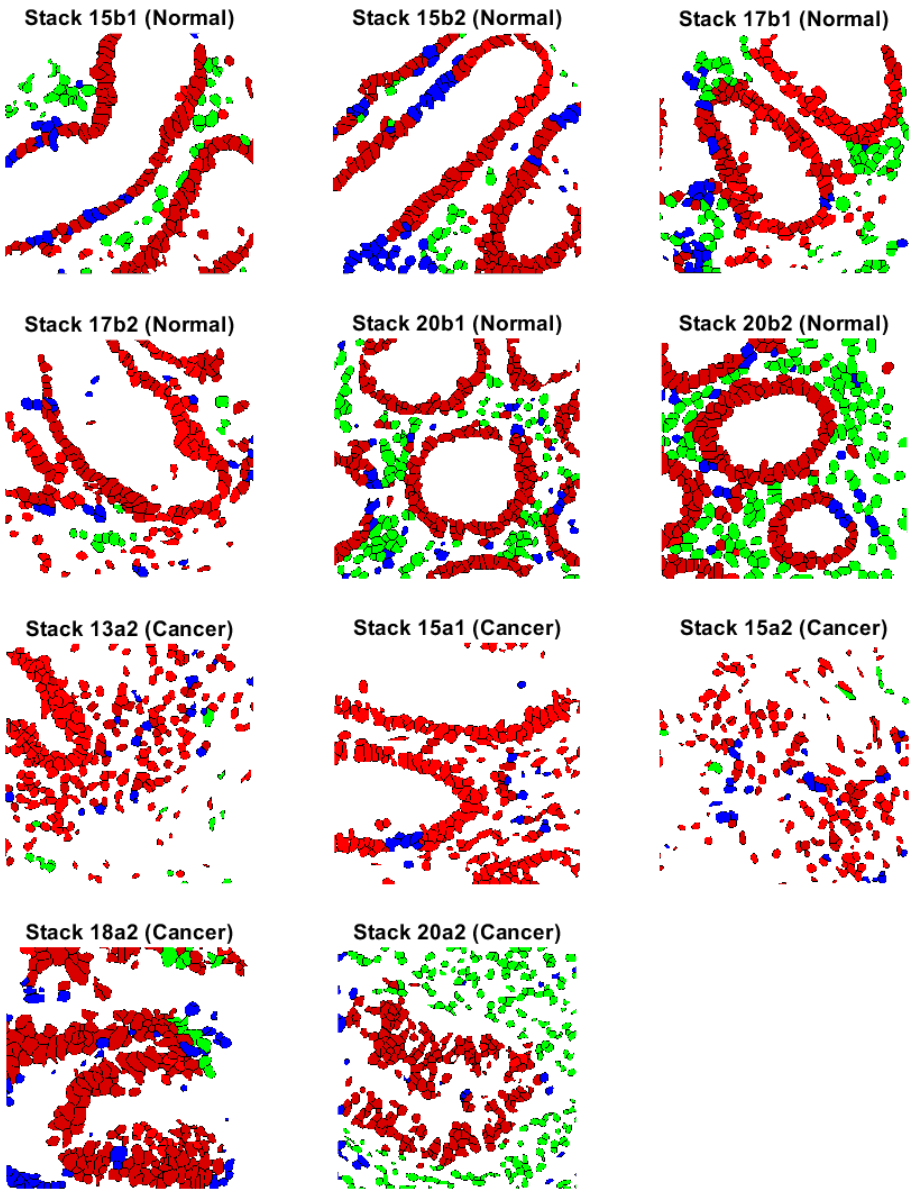

Figure 5.8: EM clustering of pooled data - 4 clusters

5.6 TIS Tags and TCGA Colorectal Cancer Data

This section explores the role of the TIS tags in colorectal cancer, using the TCGA COAD and READ data sets. The Cancer Genome Atlas (TCGA) was introduced in Chapter 1. TCGA is a repository of cancer-related data, obtained with many different advanced techniques, such as microarray technology and gene sequencing [127].

The work described in this section applied clustering techniques to TCGA gene expression data. Eleven genes of particular interest were employed, mapping to the tags used in the TIS analysis. Of the 461 colon cancer cases in COAD and 171 rectal cancer cases in READ gene expression data were available for 155 and 69 cases respectively. In addition, clinical data were available for all patients, as described previously. The COAD and READ data sets were primarily prospective in nature and for the majority of cases outcomes such as time to disease recurrence or time to death were not available and so have not been analysed here.

Variables in the TCGA clinical data set were examined for their relationship to clustering results and those variables having statistically significant associations with cluster membership have been reported. However, in the TCGA repository there are many variables, with large numbers of missing cases, so that it has not been possible to examine all potential relationships.

5.6.1 Methods and Results

All gene expression files in the COAD and READ data sets, named in accordance with patient bar-codes (one per patient), were downloaded and merged into two large tables, one for colon cancer and one for rectal cancer. The bar codes of participating patients, and the names of genes found in the gene expression files were also saved. Each table was filtered by confining the genes being expressed to the eleven genes of interest. Clinical data were also downloaded.

Tags used in the TIS project were matched to genes used in the COAD and READ gene expression files. In some cases the TIS tag corresponded to more than one gene. For example the marker CAM5.2, cytokeratin, is quoted as reacting to a cocktail of CK8, CK18 and CK19 low molecular weight proteins (KRT8, KRT18 and KRT19 in the TCGA gene expression data). It was found that expression levels for KRT8 and KRT18 were quite strongly correlated, ($r = 0.83$), and the decision was made to include only one of these genes in the analysis. The resulting list of genes,

Table 5.7: Tags Used in TIS Study and Corresponding Proteins Selected from TCGA

TIS	TCGA
CAM5.2	KRT18
CAM5.2	KRT19
Ki67	MKi67
P53	TP53
MLH1	MLH1
MSH2	MSH2
MSH6	MSH6
PMS2	PMS2
CDH1	CDH1
EpCAM	TACSTD1
CD133	PTEN

Table 5.8: Mean Log-Scores of Gene Expression Values

Gene	Score
KRT18	0.1838
KRT19	0.7964
Ki67	-0.8660
P53	-0.2128
MLH1	-1.1403
MSH2	-1.2558
MSH6	-1.4445
PMS2	-0.3113
CDH1	2.1145
EpCAM	3.6795
PTEN	0.4237

with corresponding TIS tags is shown in Table 5.7:.

Table 5.8 contains average expression values in the COAD data set for the selected genes. E-Cadherin(CDH1) and EpCAM(TACSTD1) have the highest mean log-scores in the data set as a whole.

5.6.2 EM Clustering of COAD and READ Expression Data

The EM algorithm was applied to the COAD data with varying values for the numbers of clusters k. In the three-dimensional scattergram in Figure 5.9 k is 2.

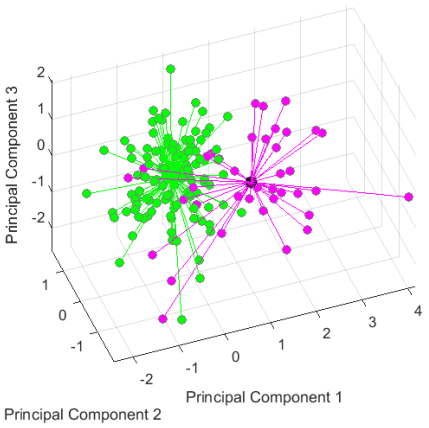

Figure 5.9: COAD - EM - Two clusters projected onto three principal components

Table 5.9: COAD - EM with k = 3 - Mean Log-Scores of Clusters

Gene	C1	C2	C3
KRT18	−0.021	0.115	0.845
KRT19	0.540	0.754	1.359
Ki67	−0.946	−0.879	−0.695
P53	0.093	−0.253	−0.342
MLH1	−2.071	−0.813	−1.961
MSH2	−1.220	−1.272	−1.205
MSH6	−1.542	−1.481	−1.108
PMS2	−0.455	−0.253	−0.484
CDH1	1.383	2.312	1.826
EpCAM	3.411	3.790	3.346
PTEN	−0.099	0.564	0.221

An advantage of the EM algorithm is that the number of clusters may be specified in advance, although this has implications for deciding on how many clusters there really are. Extending the EM algorithm to the case where there are three clusters we obtain the clusters, denoted here as C1 (23 pts), C2 (113 pts) and C3 (19 pts). Tables 5.9 and 5.10 contain means and Z-deviations.

Figure 5.10 displays the three clusters, using the three highest principal components of the data. Entries in Table 5.10 are the Z-deviations of the genes' mean cluster values. The largest cluster, Cluster 2, appears to be quite similar to the larger cluster for the two-cluster example and relatively high in MLH1, while Clusters 2 and 3 are both low in MLH1. Cluster 3 is high in KRT10 and KRT19 and

Table 5.10: COAD - EM with k=3 - Z-deviations of Clusters

Gene	C1	C2	C3
KRT18	−0.100	−0.034	0.322
KRT19	−0.125	−0.021	0.274
Ki67	−0.039	−0.006	0.084
P53	0.149	−0.020	−0.063
MLH1	0.454	0.160	−0.400
MSH2	0.017	0.008	0.025
MSH6	−0.047	−0.018	0.164
PMS2	−0.070	0.029	−0.084
CDH1	−0.357	0.096	−0.141
EpCAM	−0.131	0.054	−0.163
PTEN	−0.255	0.069	−0.099

low in MLH1. If Clusters 1 and 3 are amalgamated, his is in line with a commonly found division of gene expression values in CRC into two groups [91]. The smaller group is of interest because it is associated with suppression of the mismatch repair gene MLH1. In the next section of this chapter, Section 5.7, similar results are obtained using Bayesian Hierrachical Clustering.

Table 5.11: COAD - EM - k=3 - Counts of Mucinous Adenocarcinoma Cases in Clusters

Cluster	Colon Adenocarcinoma	Colon Mucinous Adenocarcinoma
C1	13	9
C2	102	10
C3	16	3

Table 5.12: COAD - EM - k=3 - Gender vs Cluster Assignment

Cluster	FEMALE	MALE
C1	15	8
C2	47	66
C3	14	5

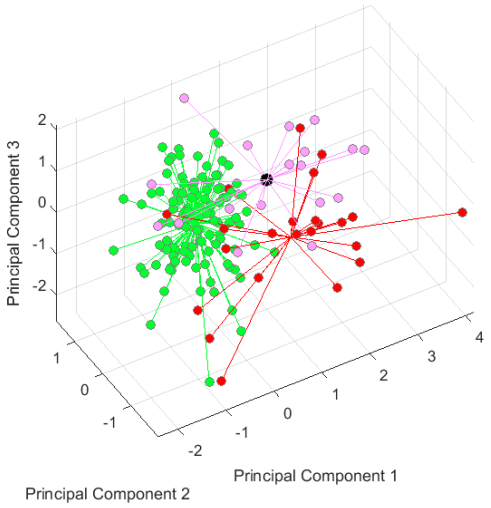

Figure 5.10: COAD - EM - Three Clusters Projected onto three Principal Components

Cluster 1 has a high frequency of patients with Mucinous Adenocarcinoma compared with the other two clusters. See Table 5.11. Both the chi-square test and Fisher's exact test [155] yielded highly significant results ($\chi^2 = 15.311$, df = 2, p-value = 0.0004735) (Fisher p= 0.0010). Both Cluster 1 and Cluster 3 have higher proportions of women than Cluster 2 (Table 5.12). The Chi-squared test statistics are ($\chi^2 = 9.5338$, df = 2) and Fisher's exact test yields a p-value = 0.0089.

No other clinical variables were found to have significant associations with cluster assignment. The effect of specifying k=3 for the EM algorithm appears to be that of splitting the largest cluster into two. Mucinous adenocarcinomas and

Table 5.13: Rectal Cancer - Mean Log-Scores of Gene Expression Values

Gene	Score
KRT18	0.007
KRT19	0.748
KI67	−0.991
P53	−0.131
MLH1	−0.699
MSH2	−1.339
MSH6	−1.511
PMS2	−0.250
CDH1	2.176
EpCAM	3.591
PTEN	0.507

Table 5.14: READ EM with Two Clusters - Counts of Mucinous Adenocarcinoma Cases in Clusters

Cluster	Rectal Adenocarcinoma	Rectal Mucinous Adenocarcinoma
C1	6	5
C2	52	2

females continued to be overrepresented in the two smallest groups, whereas they were underrepresented in the two larger groups.

5.6.3 Rectal Cancer Data

Similarly to the gene expression data for colon (COAD) tumours, the rectal gene expression values shown in Table 5.13 are very high for CDH1 and EpCAM. The EM clustering algorithm was used to cluster rectal cancer data. Running the EM algorithm against READ expression data, with $k = 2$ clusters, yielded a cluster with 12 members and one with 57 members. Figure 5.11 is a scattergram of the two clusters projected onto the three principal components of the gene expression data.

Similarly to colon cancer, the clustering of the selected gene expression profiles for rectal cancer that the frequency of mucinous carcinomas in the two clusters was different. Table 5.14 contains counts for the two clusters. Note that in this table only 11 cases are shown in Cluster 1, and 54 in Cluster 2. This is because there are some missing values for cancer type and these have been omitted from the analysis. No other clinical variables were significantly related to cluster predictions. However, the

Table 5.15: READ - EM with Two Clusters - Mean Log-Scores of Clusters

Gene	C1	C2
KRT18	−0.021	0.115
KRT19	0.540	0.754
KI67	−0.946	−0.879
P53	0.093	−0.253
MLH1	−2.071	−0.813
MSH2	−1.220	−1.272
MSH6	−1.542	−1.481
PMS2	−0.455	−0.253
CDH1	1.383	2.312
EpCAM	3.411	3.790
PTEN	−0.099	0.564

Table 5.16: READ - EM with Two Clusters - Z-deviations of Clusters

Gene	C1	C2
KRT18	−0.158	0.033
KRT19	−0.170	0.036
KI67	−0.204	0.043
P53	0.295	−0.062
MLH1	0.122	−0.026
MSH2	−0.103	0.022
MSH6	−0.073	0.015
PMS2	−0.089	0.019
CDH1	−0.155	0.033
EpCAM	−0.144	0.030
PTEN	−0.165	0.035

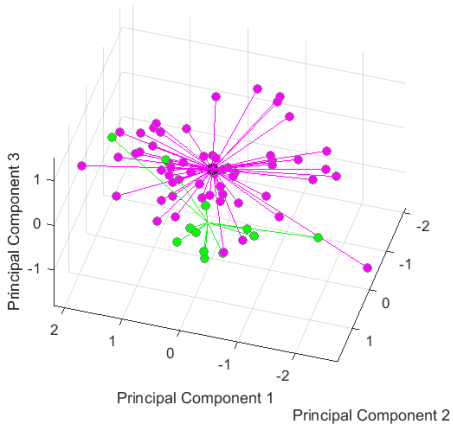

Figure 5.11: READ - EM k=2 - Clusters Projected onto Three Principal Components

numbers are very small, so there may be relationships that are not strong enough to detect. The EM algorithm was also applied to the rectal data, with k=3, but no significant relationships with clinical data were detected.

5.6.4 Pooling COAD and RECT

The TCGA report [167] concluded that there are great resemblances between colon and rectal cancers, so with this conclusion in mind the colon and rectal cancer gene expression datasets were pooled to form one large data set. Application of the EM algorithm to this data set using k=2 resulted in a cluster containing colon cancers, many of which were mucinous, and a larger cluster containing non-mucinous colon cancers and all the rectal cancers. When the number of clusters was increased the mucinous rectal cancers continued to be included with the cluster containing non-mucinous cancers. It was decided that for the current set of tags it was appropriate to cluster the colon and rectal expression datasets separately.

5.7 BHC - Bayesian Hierarchical Clustering

The previous section applied the K-Means algorithm and the EM algorithm to TCGA data. In this section we concentrate on Bayesian Hierarchical Clustering (BHC) originally presented in [81]. Bayesian Hierarchical Clustering developed by [81] and [156] is a form of agglomerative hierarchical clustering.

BHC is a bottom-up clusterer which builds a binary tree in which the leaves are data points. Each node of the tree is potentially a cluster. BHC identifies clusters by computing the odds of merging child nodes versus keeping them as separate clusters. This thesis describes an extension by the author to an existing version of BHC. The existing version assumes that within a given cluster, variables are uncorrelated [156]. In this extension, which is termed BHC-NW, the Normal-Wishart version of BHC, the algorithm can deal with clusters that have correlated features. We apply the algorithm to data pertaining to colorectal cancer, both data obtained from immunofluorescence images and also data extracted from TCGA [127].

Frequentist methods for estimating model parameters estimate the values of θ that maximise the *likelihood* of the data set $D = \{x^{(i)}\}$:

$$P(D|\theta) = \prod_i f(x^{(i)}|\theta) \tag{5.10}$$

The application of a clustering algorithm results in a model M which may be used for prediction. Given a new data point x we predict $M(x)$ to be the cluster (or clusters) most likely to be associated with x. Note that if clusters overlap a lot then two or more clusters may have non-zero probabilities for a given point. Various objective functions may be used to measure how well the model fits the data, including both distance-based criteria such as sums-of-squares, and probabilistic functions such as the log-likelihood.

There are various difficulties associated with clustering. One problem is that of finding a global minimum for the objective function. The solution space is very large, with many local maxima, and the problem is NP-Hard. Clustering algorithms with feasible performance are greedy, finding local minima rather than the global minimum. The problem may be ameliorated by running a local optimisation algorithm repeatedly, using different random starting points. For example, K-means [118] can use randomly chosen points as starting points for cluster centres, being run repeatedly. Similarly, the *Expectation Maximisation* (EM) algorithm [46] can use starting points derived from the outputs of different runs of K-Means. Another

issue, common to many machine-learning algorithms, is that of *overfitting* [79]. If the number of clusters n_k is allowed to vary as part of the modelling process, then, in general, for given training data, the objective function continues to improve as n_k increases. However, if the objective function is measured against independent test data, this improvement will not in general be observed. Overfitting may be tackled by adjusting for high values of n_k with penalty terms such as the Akaike Information Criterion, AIC [79] or the Bayesian Information Criterion, BIC [79]. Alternatively, n-fold cross-validation may be used to establish how well the model performs with test data. Both penalty terms and cross-validation enable the number of clusters n_k to be estimated.

Bayesian clustering may be used to avoid various problems associated with traditional frequentist algorithms. For example, variational Bayes clustering [18] generalises the EM algorithm by using prior values for model parameters, values which smooth the development of the algorithm and prevent the development of singularities in the solution.

BHC is *Bayesian* because various assumptions are made about the nature of the populations which are assumed to generate the data. These assumptions are captured by the values of various *hyperparameters*. For example, in the course of BHC, when estimating the population mean at a node of the hierarchy, we use both the sample mean of the data points in the sub-tree defined by the node and a prior mean specified by a hyperparameter ξ. The BHC algorithm described in [156] assumes that the underlying sub-populations are multivariate Gaussian with constraints on their structure . The constraints are as follows: for each subpopulation, in the associated covariance matrix, all off-diagonal terms are zero.

In the work described here a model was developed which relaxed this requirement. The model, BHC-NW, allowed the off-diagonal terms in the covariance matrix to be non-zero. This meant that correlations between the coordinates of data points in the same cluster could be represented. In addition, the hyperparameters of BHC-NW were optimised in hyperspace, using non-linear optimisation techniques. Two versions of optimisation have been developed. In the first version, BHC-NW-TREE, gradients of the optimisation function are computed numerically. In the second version, BHC-NW-GRAD, gradients are computed using closed-form versions of the partial derivatives. These closed-form gradients apply only to fixed tree structures, so optimisation with respect to closed-form gradients must alternate with steps in which the tree is constructed afresh.

5.7.1 Dirichlet Process Model

The BHC algorithm is based on the *Dirichlet Process Model* (DPM), an approach in which clusters are built up as data points are added to a data set [57]. DPMs have various advantages, such as automatically generating the number of clusters (subject to the value of a hyperparameter known as the *concentration parameter*).

We may express the marginal likelihood $P(D_k|T_k)$ of the data D_k in tree T_k as the sum of two terms, corresponding to two hypotheses. The first hypothesis H_k^1 assumes that the data at node k comes from a single phenotype, and the second hypothesis H_k^2 assumes that there are separate phenotypes (i.e. cluster groups) corresponding to child trees T_i and T_j. Both H_k^1 and H_k^2 are assigned prior probabilities which depend only on the sizes of the trees T_i, T_j, T_k and the hyperparameter α (the concentration parameter).

$$P(D_k) = \Pi_k P(D_k|H_k^1) + (1 - \Pi_k)P(D_i|T_i)P(D_j|T_j) \tag{5.11}$$

We have denoted the probability of hypothesis H_k^1 by Π_k and that of the alternative hypothesis H_k^2 by $(1 - \Pi_k)$.

The probability r that the data at T_k belongs to a single phenotype is:

$$r = \frac{\Pi_k P(D_k|H_k^1)}{P(D_k|T_k)} \tag{5.12}$$

Note that if $r \geq 0.5$ then we mark the node as a *merging* node.

Algorithm 7 BHC Algorithm

procedure BHC(data, α)
    $c = n$
    $k = n + 1$
    for each data point i **do**
        $D_i = \{x^{(i)}\}$
        Mark i as active
    end for
    while $c > 1$ **do**
        for All active pairs of nodes $m = \{i, j\}$ **do**
            Π_m = COMPUTEPI(m, α)
            Compute probability r_m of merged hypothesis
        end for
        Select pair $\{i, j\}$ which maximises r_m
        Create new node k with children i and j
        $D_k = D_i \cup D_j$
        Mark i and j as inactive
        $c = c - 1$
        $k = k + 1$
    end while
end procedure

Algorithm 8 Computation of prior values Π_k

procedure COMPUTEPI(m, α)
    for each node $k \in$ m (as it is generated by the main BHC algorithm) **do**
        $d_k = \alpha * \Gamma(n_k) + d_{left} d_{right}$
        $\Pi_k = \frac{\alpha \Gamma(n_k)}{d_k}$
    end for
end procedure

Execution of the BHC algorithm proceeds from the leaves of the binary tree, inwards as the tree is built up. Initially all nodes are leaf nodes and are marked as *active*. In each iteration all pairs of active nodes i and j are considered, and the probability $r(i, j)$ that they should be merged into a single node k is computed. Next the pair (i, j) which maximises r is selected and the tree is augmented with a node k which is designated as the parent of i and j. The node k is marked as active, and the nodes i and j are marked as *inactive*. Iterations continue until there is only one active node left (the root node). Algorithm 7 contains pseudocode for the main

BHC process, while Algorithm 8 describes how the hypothesis probability values Π_k are calculated from child values as execution proceeds.

The completed tree is examined for clusters by finding nodes for which the r value is less than 0.5, that is, where the associated log-odds ratio is less than zero. These are nodes whose children should remain as separate clusters. The final model is based on a set of nodes where mixing proportions are determined by the number of leaves under each node, divided by the size of the data set, and the other parameters are determined from the data.

We now discuss the role of the Bayesian approach in BHC. From a Bayesian perspective we assume that not all parameter values θ are equally likely, but instead we specify that the probability of θ follows a distribution $h(\theta|\xi)$ where ξ is a *hyperparameter* which determines the precise shape of h. Then, the probability of the observed data point $x^{(i)}$ is given by:

$$P(x^{(i)}) = g(x|\theta)h(\theta|\xi) \tag{5.13}$$

A natural way of defining $h()$ and ξ is to use *conjugate priors*. This approach assumes that a set of virtual data has already been sampled, and that this set has been sampled from the same form of probability distribution as the actual data. We assume that data set D_k consisting of data points that have been randomly sampled from the sub-population k. We are interested in sufficient statistics for D_k which in the case of the multivariate Gaussian distribution are the sample mean and sample variance. We assume that we have already taken pseudo-samples which can be used to weight the actual observations. The pseudo-samples have a specified prior mean and a specified prior variance, which act as hyperparameters in our calculations. The prior mean μ_0 is assumed to be calculated from a data set of size κ_0 and the prior sum of squared deviations T_0 is calculated from a virtual data set of with degrees of freedom ν_0.

A suitable prior distribution for multivariate Gaussian data is the *Normal-Wishart prior* [18] which has precisely the properties specified in the preceding paragraph. The Normal-Wishart prior is the product of a multivariate Normal (Gaussian) distribution and a Wishart distribution. The Wishart distribution models the sample inverse variance matrix Λ for a multivariate normal (Gaussian) distribution with population variance T_0 and with ν_0 degrees of freedom. The Normal component models the sampling distribution of the mean of κ_0 data points sampled from a multivariate normal distribution with mean μ_0 and inverse variance Λ.

$$h(\mu, \Lambda) = N(\mu|(\mu_0, \Lambda)^{-1})Wi_\nu(\Lambda|T_0) \tag{5.14}$$

To be explicit, the list of hyperparameters ξ can be written as:

$$\xi = (\mu_0, \kappa_0, T_0, \nu_0) \tag{5.15}$$

In the subsequent discussion we denote data point i in D_k by $x_k^{(i)}$. We assume that the dimensionality of the data is d, so $x_k^{(i)}$ is a vector composed of d real numbers. Denoting the number of points in D_k by n_k we may express the mean value $\bar{x}_k$ as follows:

$$\bar{x}_k = \frac{\sum_{i=1}^{n_k} x_k^{(i)}}{n_k} \tag{5.16}$$

Another statistic S_k the total sum of squared deviations may be expressed as follows:

$$S_k = \sum_{i=1}^{n_k} (x_k^{(i)} - \bar{x}_k)(x_k^{(i)} - \bar{x}_k)^T \tag{5.17}$$

Next we create updated hyperparameters as follows (note that for clarity we have dropped the subscript k):

$$\mu_n = \frac{\kappa_0 + n\bar{x}}{\kappa_0 + n} \tag{5.18}$$

$$\kappa_n = \kappa_0 + n \tag{5.19}$$

$$\nu_n = \nu_0 + n \tag{5.20}$$

$$T_n = T_0 + S + \frac{n\kappa_0}{\kappa_0 + n}(\mu_0 - \bar{x})(\mu_0 - \bar{x})^T \tag{5.21}$$

And the marginal likelihood of D is:

$$P(D|\xi) = \frac{1}{\pi^{\frac{nd}{2}}} \frac{\Gamma_d(\frac{\nu_n}{2})}{\Gamma_d(\frac{\nu_0}{2})} \frac{|T_0|^{\frac{\nu_0}{2}}}{|T_n|^{\frac{\nu_n}{2}}} (\frac{\kappa_0}{\kappa_n})^{\frac{d}{2}} \tag{5.22}$$

In the equation above terms of the form $\Gamma_d(x)$ denote the multivariate Gamma function of order d. A definition is:

$$\Gamma_d(x) = \pi^{\frac{d(d-1)}{4}} \prod_{j=1}^{d} \Gamma(x + (1-j)/2) \tag{5.23}$$

We must also consider how Π_k is generated. In practice values of Π_k are calculated as the tree is built up by the BHC algorithm, moving inwards from the leaves.

5.7.2 BHC - Optimisation over Hyperparameters

The discussion above has described the basic operation of BHC-NW. One output of BHC-NW is the marginal log-likelihood of the data at the root of the tree. This log-likelihood ϕ varies with hyperparameter values, and optimisation of ϕ with respect to the hyperparameter values, may be used to obtain the optimal binary tree. In practice not all of the hyperparameter space has been considered in the optimisation process, and furthermore all hyperparameter values have been considered equally likely (flat priors). Two algorithms which implement hyperparameter optimisation have been developed.

The two algorithms differ in their calculations of the partial derivatives of ϕ with respect to ξ. The first algorithm, which we call BHC-NW-TREE uses non-linear optimisation where gradients are calculated numerically by a non-linear optimiser (in practice this is the MATLAB optimiser $fmincon()$). The second algorithm, BHC-NW-G, calculates gradients using analytic formulae, as the binary hierarchy is formed. The reader is referred to Appendix A for the details of the analytic formulae for the partial derivatives of $P(D|T_j)$ and to Appendix B for the method of accumulating the partial derivatives of the marginal likelihood $P(D)$ as the tree is built.

Algorithm 9 BHC-NW-TREE

1: **procedure** BHC-NW-TREE(DATA, ξ_0, BOUNDSξ)
2: Normalise(Data)
3: Set function Pointer fp to BHC(Data, ξ)
4: **end procedure**

Table 5.17: COAD - BHC-NW - Mean Log-Scores of Clusters

Gene	C1	C2
KRT18	0.121	0.533
KRT19	0.780	0.888
Ki67	−0.920	−0.597
P53	−0.292	0.173
MLH1	−0.836	−2.911
MSH2	−1.256	−1.275
MSH6	−1.453	−1.294
PMS2	−0.266	−0.538
CDH1	2.238	1.419
EpCAM	3.763	3.205
PTEN	0.478	0.181

Algorithm 10 BHC-NW-G

1: **procedure** BHC-NW-G(DATA, ξ_0, BOUNDSξ)
2: Normalise(Data)g
3: Set function Pointer fp to BHC(Data)
4: Set gradient Pointer gp to BHCGrad(Data)
5: NonLinearOptimiser(fp, gp, ξ_0, Boundsξ)
6: **end procedure**

5.7.3 Bayesian Hierarchical Clustering - TCGA Colon Cancer Data

The BHC-NW algorithm was used to cluster the 155 COAD cases that had gene expression data. BHC-NW found three clusters. One cluster that contained only one member was regarded as trivial and excluded from further calculations. Two non-trivial clusters were found, containing 131 and 23 data points and are referred to as C1 and C2. Mean log-scores of points in the two non-trivial clusters found by BHC-NW are shown in Table 5.17.

Contrasts between clusters have been computed for genes using Bioconductor software [86]. The association may be computed for each gene and each cluster by subtracting the mean log-score of the gene from its cluster mean, then dividing by the standard deviation of the log-score for the gene. Table 5.18 displays the Z-deviations for the two non-trivial clusters.

The highest contrast is for EpCAM, the cancer marker gene. Genes MLH1,

Table 5.18: COAD - BHC-NW - Contrasts Between Clusters - p-values

Gene	p-value
KRT18	1.080×10^{-4}
KRT19	7.450×10^{-28}
Ki67	1.530×10^{-32}
P53	1.220×10^{-3}
MLH1	5.140×10^{-89}
MSH2	5.520×10^{-83}
MSH6	6.700×10^{-87}
PMS2	1.800×10^{-24}
CDH1	1.050×10^{-97}
EpCAM	1.370×10^{-150}
PTEN	3.600×10^{-21}

MSH2, MSH6 and PMS2 are associated with the DNA mismatch repair pathway. We note that they are all suppressed in Cluster 2, the smaller cluster. For example, MLH1, is described as follows:

"The MLH1 gene provides instructions for making a protein that plays an essential role in DNA repair. This protein helps fix mistakes that are made when DNA is copied (DNA replication) in preparation for cell division." [176]

The heatmap in Figure 5.12 displays the genetic expression profiles of the patients, together with the binary tree ouput by BHC-NW. Underexpression is indicated by red, and overexpression by green.

In Figure 5.13 the data points have been projected onto the first three principal components of the restricted (11-gene) COAD data set. The blue data points in the larger cluster are clearly separated from the magenta points in the smaller cluster.

5.7.4 BHC-NW - Evaluation Metrics Using TCGA Data

Evaluation was carried out for the BHC-NW and EM algorithms when applied to the TCGA colon data (COAD) used in the previous section. Table 5.19 displays the values of various metrics for both the BHC-NW algorithm and also for the EM algorithm where the number of clusters is two, three and four, respectively. The Adjusted Rand Index [142] uses an independent test set to compute the effectiveness of clustering and finds BHC-NW to be a clear winner. The BHC-NW algorithm is also the best algorithm for the Davies Bouldin metric [40] and the Silhouette metric

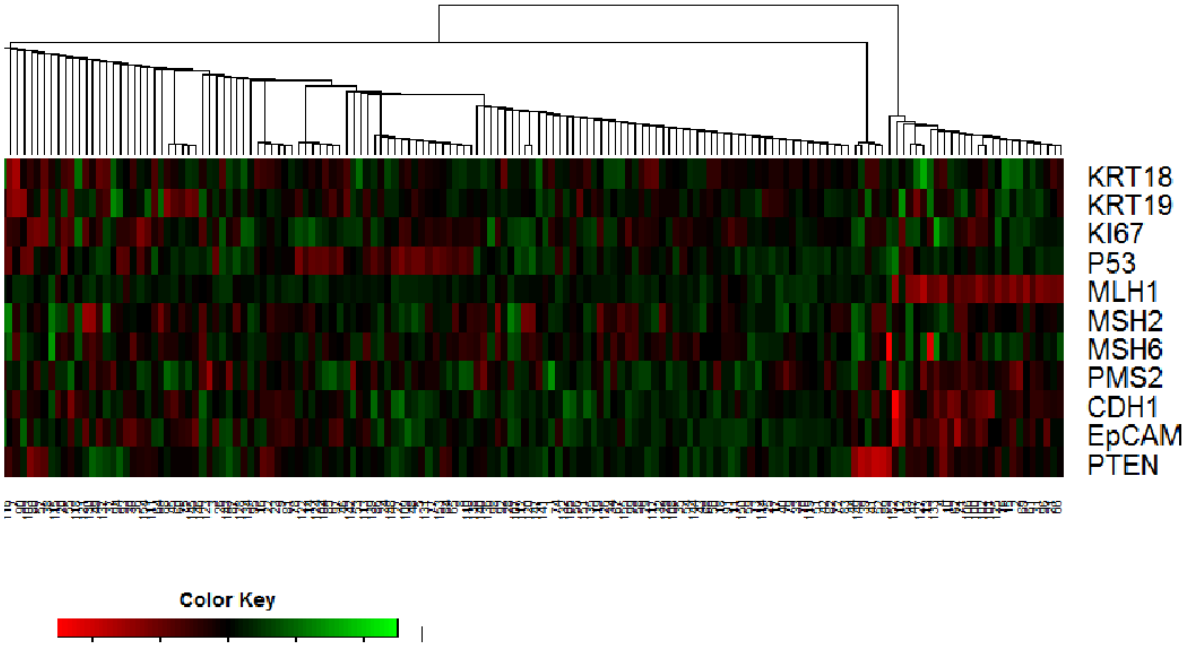

Figure 5.12: COAD - BHC-NW - Heatmap

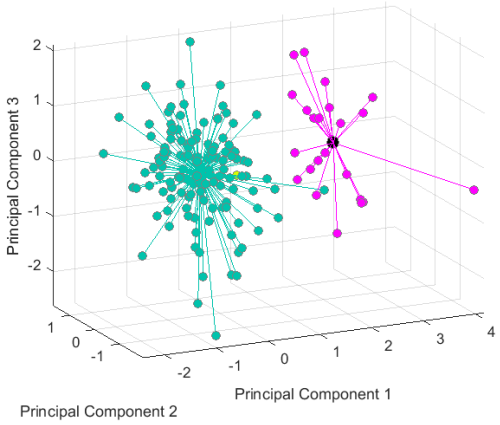

Figure 5.13: COAD - BHC-NW - Clusters Projected onto three Principal Components

[145] . BHC-NW does not perform so well in the Variance Ratio test [27]. EM with three clusters has a variance ratio of 10.7 compared with the BHC-NW variance ratio of 19.31. However the Variance Ratio metric decreases monotonically with cluster number and preferably should be used for the comparison of clusterers that have the same number of clusters, in this case two. For EM k=2 the variance ratio is 20.65 and very similar to the variance ratio of BHC-NW.

Table 5.19: Results of Clustering TCGA COAD data

Metric	BHC-NW	EM k=2	EM k=3	EM k=4
Adjusted Rand Index [142]	0.23	0.07	0.06	0.07
Purity [101]	0.85	0.85	0.85	0.87
Variance Ratio [27]	19.31	20.65	27.26	10.7
Davies Bouldin [40]	7.96	5.49	3.92	5.13
Silhouette [145]	0.45	0.31	0.22	0.06
Log Likelihood [183]	na	-1,080	-985	-891
Akaike Information Criterion [8]	na	2,480	2,440	2,400
Bayesian Information Criterion [66]	na	2,950	3,140	3,350

Table 5.20 displays the results of ten-fold cross-validation when used with the EM algorithm. The highest value of the test log-likelihood is for $k = 2$, for two clusters. This result indicates that the EM algorithm should be used with two clusters.

Table 5.20: 10-Fold Cross-Validation

Metric	EM k=2	EM k=3	EM k=4
LLTest	-140	-155	-184

5.7.5 BHC-NW - TCGA COAD Clusters and Clinical Variables

In each application of clustering the predicted cluster assignments were saved, and matched with the corresponding clinical data. A script written in the R statistical programming language examined all clinical variables and their behaviour with respect to a given set of predictions. In practice, many of the clinical variables had nearly all missing values in the colorectal clinical data and only significant results have been reported. Data were examined for relationships between cluster membership and clinical variables.

Table 5.21: COAD - BHC-NW - Counts of Mucinous Adenocarcinoma Cases in Clusters

Cluster	Colon Adenocarcinoma	Colon Mucinous Adenocarcinoma
C1	116	13
C2	14	9

Table 5.22: COAD - BHC-NW - Gender vs Cluster Assignment

Cluster	FEMALE	MALE
C1	58	73
C2	18	5

Cancers in the smaller cluster are significantly more likely to be mucinous as recorded in the patient's histologic diagnosis - see Table 5.21. Pearson's Chi-squared test with Yates' continuity correction yielded values of $\chi^2 = 11.066$, p-value = 0.0009, and Fisher's exact test for count data had a p-value = 0.001.

Patients in the smaller cluster are far more likely to be female than male. Table 5.22 shows counts for male and female patients, for each cluster. ($\chi^2 = 7.7$, df = 1, p-value = 0.005, Fisher's Exact Test p-value = 0.003)

Table 5.23: BHC-NW COAD - Anatomic Neoplasm Subdivision vs Cluster
Asc.=Ascending, Desc.=Descending, Trans.=Transverse

Cluster	Discre-pancy	Asc. Colon	Cecum	Desc. Colon	Hepatic Flexure	Sigmoid Colon	Splenic Flexure	Trans. Colon
C1	0	19	22	6	4	66	2	11
C2	1	9	6	0	5	0	0	2

Table 5.24: COAD - BHC-NW - Ajcc Pathologic Tumour Stage vs Cluster Assignment

Cluster	I	II	IIA	IIB	III	IIIA	IIIB	IIIC	IV	IVA
C1	25	10	37	3	5	3	8	16	22	1
C2	4	1	9	2	3	0	4	0	0	0

The location of the tumour also varied significantly. None of the tumours associated with the smaller cluster were in the colon proper, but were almost all in the ascending colon, the cecum or the hepatic flexure (Table 5.23). Statistics were ($\chi^2 = 37.631$, p-value = 3.6e-06) and a p-value = 3.1e-07 for Fisher's exact test.

Patients in the smaller cluster were significantly more likely to have early stage tumours. See table 5.24 showing "Ajcc Pathologic Tumour Stage" against cluster number. Significance levels were $\chi^2 = 17.3$, df = 9, p-value = 0.04 (Pearson's Chi-squared test) and p-value = 0.019 (Fisher's exact test).

5.8 Conclusions

Probabilistic graphical models generalise the Pearson correlation, a standard measure of colocalisation to multiplexed images. PGMs that could handle to stacks of TIS images were developed. Both pixel-level data and segmented nuclei were modelled. Multivariate colocalisation was applied to both 'normal' and 'cancer' stacks. Strong relationships characterised certain pairs of tags in normal tissue, while appearing weaker in cancer.

Clustering was also applied to TIS stacks. Clustering nuclei according to tag values segmented them into two groups, one containing epithelial cells and the other containing stromal cells.

Bayesian Hierarchical Clustering was extended in this chapter. The BHC-NW algorithm catered for clusters in which the underlying Normal distribution allowed off-diagonal terms in the covariance matrix. This allowed many more data sets to be modelled accurately. In addition, by using the Wishart distribution for prior values, hyperparameter optimisation for BHC-NW was implemented.

The BHC-NW algorithm was applied to TCGA gene expression data, to eleven proteins matched to TIS tags. Clustering results were compared with those of the EM algorithm and it was found that BHC-NW outperformed EM with regard to most metrics.

Clusters resulting from BHC-NW were examined with respect to available TCGA clinical data. For both colon (COAD) and rectal (READ) cancers, the BHC clustering algorithm found two significant clusters in TCGA protein expression levels, the smaller cluster being associated with lowered levels of the DNA mismatch repair protein MLH1. This result matches the main finding of the TCGA paper, where mutational data separates patients into in two groups, one of which has suppression of the mismatch repair protein.

Chapter 6

Conclusions

6.1 Deep Learning with Sampling

Chapter 3 demonstrated that sampling regions from whole-slide diagnostic images and applying cell identification models to these selected regions was effective in identifying cells. Furthermore, Systematic Random sampling performed appreciably better than Random Sampling. Two different cell identification algorithms, 'Cell' and 'Hovernet' were used and sampling performed well with both of them.

As an example, the 'Cell' identification results were used to look for associations between the spatial densities of different types of cell and clinical variables. A range of significant associations was found.

Time constraints have prevented the examination of the behaviour of sampling at varying levels of resolution and region (tile size). The experiments used a fixed image resolution of 20X ($0.5\mu M$) and the tile size was also fixed, at (500×500) pixels. It would be useful to estimate the effect of varying image resolution and/or tile size. There are various latencies, associated with loading the GPU with input data, so that many small patches might be slower to process than a few large ones. Performance modelling would entail measuring these latencies as well as computation times for on-GPU processes once they have been initiated.

In principle, sampling could be used as a preprocessing step, applicable to many deep learning models. In addition, sampling captures the spatial distribution of cells and other objects, which is of interest to pathologists in modelling the characteristics of tissue, both normal and cancer.

6.2 Colour Normalisation

Colour normalisation was studied in Chapter 4 which describes the systematic comparison of several colour normalisation methods when applied to cell classification.

The algorithms used in Chapter 3 did not use explicit colour handling (such as normalisation or augmentation). Although good results were obtained with high-quality TCGA images from the AA site, it was desirable to carry out cell identification using colour normalisation for the remaining sites in the TCGA COAD dataset. Although all colour normalisation methods gave considerably better results than doing nothing, there were two clear winners, namely 'Naive' colour normalisation and Macenko normalisation.

Chapter 4 used a test harness to compare normalisation algorithms as well as considering the effects of site differences. Sites varied greatly in the level of staining and as a result accuracy scores for cell classification at different sites also varied, particularly for unnormalised images. For the most part, the rankings of the different normalisation algorithms were preserved when going from site to site, but even for the best algorithms the accuracy varied between sites: from 73% to 89% for naive colour normalisation and from 79% to 89% for Macenko normalisation. This result indicates that researchers should take care when evaluating new normalisation algorithms: a well-accepted algorithm should always be applied to the data for comparison purposes.

The effect of colour normalisation on the accuracy of 'Cell' detection (as opposed to cell classification) was examined. No systematic improvement was observed, possibly because Khan normalisation was carried in the detection phase.

Although various colour normalisation algorithms were compared, stain normalisation with stain augmentation or with adversarial algorithms were not evaluated in the work described here. These results suggest that it would be profitable to carry out experiments with augmentation and adversarial algorithms, using the hand marking set created for this study.

6.3 TIS Stacks and TCGA Expression Data

Several areas of interest were explored in Chapter 5, "Molecular Expression: From Image Stacks to TCGA".

The first research area was the analysis of image stacks, multiplexed fluorescence images created using the Toponome Imaging System. Probabilistic graphical models were applied to image stacks of tissue from patients with colorectal cancer. In this application the nodes represented antigens, while the edges represented dependencies between the nodes. For a pair of images the Pearson correlation coefficient is a standard measure of colocalisation. For multiplexed images, the corresponding measure is the set of partial correlations extracted from the graphical model. The graphical models produced for TIS data had various interesting linkages between antigens.

The second part of Chapter 5 also considered multiplexed fluorescence images from TIS data. When clustering was applied to the image stacks it was found that the clusters separated epithelial cells from cells in the stroma.

Consideration of the literature suggests that the algorithms in parts 1 and 2 of Chapter 5 could be improved in various ways. In the TIS data there are two images associated with each application of an antigen: 'before' and 'after' images. The analysis undertaken in Chapter 3 considers the 'after' image only, possibly decreasing the accuracy of the graphical model. Improvements might be obtained by operating on the difference between the 'before' and 'after' images. In addition, various authors have taken background effects into account. Future work would examine the effect of including background effects. Regarding future work, several directions are possible. In the first place, the analysis applies to multiplexed fluorescence images but could be easily extended to multi-stain images in the visible spectrum. In the second place only regions containing nuclear material have been considered. It would be useful to include areas of cytoplasm as well.

The third area of interest in Chapter 5 was Bayesian Hierarchical Clustering. This algorithm was extended to a new version, BHC-NW, which allowed the covariance matrices of clusters to contain off-diagonal entries. BHC-NW performed well with respect to various metrics when compared with Gaussian mixture models. When BHC-NW was applied to protein expression data from TCGA (The Cancer Genome Atlas) it was found that cluster membership was associated with various molecular and clinical features. Of the two highest-level clusters the smaller cluster had properties corresponding to the group identified by the TCGA project as having high mutation rates and suppression of mismatch repair proteins.

Extending BHC-NW to deal with high dimensional data would allow the

algorithm to be applied to a greater range of data. Most protein expression data sets have very large numbers of proteins and cannot be handled by low-dimensional clustering algorithms. A direction for future work is the modification of BHC-NW to a high-dimensional version.

6.4 Concluding Remarks

The work described in this thesis has found links between features that are associated with histopathology, and genomic and clinical features. Only a small subset of all possible features has been considered. Many other features could have been included in the analysis - clinical features such as height and weight, identifiable histologic features such as cell complexes and glands, stroma and inflammation, topological features such as edges, textures and regions, and genomic features. Many studies have uncovered links between different types of feature, but these links often relate to an isolated set of features. The task of modelling relationships in a single model that links all features remains incomplete.

Appendix A

Bayesian Hierarchical Clustering

As described in Chapter 5, Bayesian Hierarchical Clustering is a bottom-up hierarchical clustering method. In the version of the BHC algorithm described in [156] the data in each cluster are modelled using a specialised form of the multivariate normal distribution, in which the off-diagonal elements of the covariance matrix are zero. The aim of the analysis in this appendix is to model the general multivariate normal case in which the covariance matrix has off-diagonal components. Specifically, for each cluster corresponding to a data set D containing n data points, we aim to compute the likelihood of D, given a prior distribution for the mean and variance.

The existing BHC algorithm requires modification in two main places. The first part of the algorithm to be changed is the calculation of the statistics that specify a cluster. It is necessary to generalise the term which specifies the intra-cluster covariance by including off-diagonal terms in the calculations. The second modification required is to generalise the hyperparameter optimisation procedure. We use a term for the marginal log-likelihood $P(D)$ that includes correlated features. Partial derivatives of the log-likelihood may subsequently be used in a numerical optimisation procedure. The appendix presents details of both modifications.

In a particular cluster we assume that the d-dimensional data point $\mathbf{x}$ has been generated from a multivariate normal distribution with (unknown) population mean μ and concentration matrix Λ, the inverse of the *covariance matrix* Σ. Note that the case where the d variables comprising $\mathbf{x}$ are uncorrelated then the off-diagonal elements of the concentration matrix are zero. We may write:

$$x \sim \mathcal{N}(\mu, \Lambda) \tag{A.1}$$

where the probability density function (pdf) of the multivariate normal distribution is:

$$(\frac{1}{2\pi})^{d/2}(det\Lambda)^{\frac{1}{2}}exp(\frac{1}{2}(x-\mu)\Lambda(x-\mu)) \tag{A.2}$$

In the Bayesian modelling approach data point $\mathbf{x}$ may be regarded as a sample from a model with parameters θ (in this case, the mean and the concentration matrix). In turn, the parameters θ depend on a set of hyperparameters ξ, for which we assume we have a suitable form for the *prior* $P(\theta|\xi)$ which specifies the dependency of θ on ξ. The hyperparameters may be interpreted as variables which capture the degree of belief in initial values of the model parameters which contribute, together with the data, to the estimated values of the model parameters.

$$P(x|\theta,\xi) = P(x|\theta)P(\theta|\xi) \tag{A.3}$$

A.1 Conjugate Prior - The Normal-Wishart prior

The Normal-Wishart prior specifies the probabilistic dependence of parameters μ and Λ on hyperparameters κ_0, μ_0, ν_0 and T_0 when data are sampled from the multivariate normal distribution. It is a *conjugate prior*, having a particular form with a natural interpretation (in terms of pseudo-sampling) which makes computations more tractable than an arbitrary prior.

Pseudo-sampling may be regarded as a process which samples quantities of (imaginary) prior data which have the appropriate mean μ_0 and variance T_0/ν_0. We interpret μ_0 as resulting from prior pseudo-sampling of a dataset of κ_0 points. Similarly we may assume that some prior pseudo-dataset of ν_0 points has a scale matrix T_0.

To define the Normal-Wishart prior we proceed as follows. (See **??** for details.)

First we define the Wishart function. Let Λ be a d-dimensional symmetric positive-definite matrix. Let T_0 be a *scale matrix* (also symmetric and positive-definite). Assume that the number of degrees of freedom is ν_0. Let Γ_d be the *generalised Gamma function*:

$$\Gamma_d(\alpha) = \pi^{\frac{d(d-1)}{4}} \prod_{i=1}^{d} \Gamma(\frac{2\alpha+1-i}{2}) \tag{A.4}$$

Define a normalising factor Z:

$$Z = 2^{d\nu_0/2}\Gamma_d(\nu_0/2)(detT_0)^{\nu_0/2} \tag{A.5}$$

The Wishart function W is:

$$W(\Lambda|T_0,\nu_0) = \frac{1}{Z}(det\Lambda)^{(\nu_0-d-1)/2}exp(-\frac{1}{2}tr(T_0^{-1}\Lambda)) \tag{A.6}$$

The pdf of the Normal-Wishart prior is:

$$N(\mu|\mu_0,(\kappa_0\Lambda)^{-1})Wi(\Lambda|T_0,\nu_0) \tag{A.7}$$

Where the pdf for the Normal factor is:

$$N(\mu|\mu_0,(\kappa_0\Lambda)^{-1}) = \frac{\kappa_0^{\frac{d}{2}}(det\Lambda)^{\frac{1}{2}}}{(2\pi)^{\frac{d}{2}}}exp(-\frac{\kappa_0}{2}(\mu-\mu_0)^T\Lambda((\mu-\mu_0)) \tag{A.8}$$

And the pdf for the Wishart factor is:

$$Wi(\Lambda|T) = \frac{(det(\Lambda))^{\frac{\nu_0-d-1}{2}}exp(-tr(T_0^{-1}\Lambda)/2)}{2^{\frac{\nu_0 d}{2}}(det(T_0))^{\frac{\nu_0}{2}}\Gamma_d(\frac{\nu_0}{2})} \tag{A.9}$$

For the special case where $d = 1$ the probability density function reduces to that of a Normal-Gamma prior: (See Equation 7 of [156]):

$$NG(\mu,\sigma|\mu_0,\kappa_0,\lambda_0,\beta_0) = \frac{\beta_0^{\lambda_0}}{\Gamma(\lambda_0)}(\frac{\kappa_0}{2\pi})^{\frac{1}{2}}\sigma^{-2(\lambda_0-\frac{1}{2})}exp(-\frac{1}{2\sigma^2}(\kappa_0(\mu-\mu_0)^2+2\beta_0)) \tag{A.10}$$

The Normal-Wishart prior reduces to this expression when we substitute $d = 1$, $\sigma^{-2} = \Lambda$, $\lambda_0 = \nu_0/2$ and $\beta_0 = T_0^2$.

A.2 Updating Hyperparameters for use in Estimation

In the presence of n independent random data samples we may update the hyperparameters, which, because we have selected a conjugate prior, are also estimates of the parameters of interest. That is, μ_n estimates μ and T_0/ν_n estimates Λ. Assume that the sample mean is $\bar{x}$ and the sample sum of squares S is:

$$S = \sum_{i=1}^{n}(x_i-\bar{x})(x_i-\bar{x}) \tag{A.11}$$

We update the hyperparameters as follows:

$$\mu_n = \frac{\kappa_0\mu_0+n\bar{x}}{\kappa_0+n} \tag{A.12}$$

$$T_n = T_0 + S + \frac{\kappa_0 n}{\kappa_0 + n}(\mu_0 - \bar{x})(\mu_0 - \bar{x})^T \tag{A.13}$$

$$\nu_n = \nu_0 + n \tag{A.14}$$

$$\kappa_n = \kappa_0 + n \tag{A.15}$$

A.3 Marginal Likelhood

The marginal likelihood is used during tree construction. The BHC algorithm examine the current pool of subtrees and combines those which optimise an expression including the likelihood. In addition, hyperparameter optimisation maximises the log-likelihood of the root node of the tree.

The marginal likelihood in terms of the updated hyperparameters is (See equation 234 of [124]):

$$P(D) = (\pi)^{\frac{-nd}{2}} \frac{\Gamma_d(\nu_n/2)}{\Gamma_d(\nu_0/2)} \frac{(det(T_0))^{\frac{\nu_0}{2}}}{(det(T_n))^{\frac{\nu_n}{2}}} (\frac{\kappa_0}{\kappa_n})^{\frac{d}{2}} \tag{A.16}$$

We recast this equation by substituting the expression $\nu_0/2$ with α_0 and $\nu_n/2$ with α_n to obtain:

$$P(D) = (\pi)^{\frac{-nd}{2}} \frac{\Gamma_d(\alpha_0)}{\Gamma_d(\alpha_0/2)} \frac{(det(T_0))^{\alpha_n}}{(det(T_n))^{\alpha_0}} (\frac{\kappa_0}{\kappa_n})^{\frac{d}{2}} \tag{A.17}$$

This substitution simplifies the computation of partial derivatives.

A.4 Partial Derivatives

In order to optimise the log-likelihood w.r.t. hyperparameters numerically we supply a gradient-descent algorithm [67] with expressions for partial derivatives of the log-likelihood with respect to these hyperparameters.

$$logP(D) = log(\Gamma_d(\alpha_n)) - log(\Gamma_d(\alpha_0)) + \alpha_0 log(det(T_0)) - \alpha_n log(det(T_n)) + \frac{d}{2} log(\kappa_0) - \frac{d}{2} log(\kappa_n) \tag{A.18}$$

We now present formulae for the partial derivatives of log(P(D)) w.r.t. κ_0, α_0, μ_0 and T_0.

A.4.1 Derivative of $logP(D)$ w.r.t. κ_0

Terms in $logP(D)$ which contain κ_0 are

$$Y_1 = -\alpha_n log(det(T_n)) \tag{A.19}$$

$$Y_2 = \frac{d}{2} log(\kappa_0) \tag{A.20}$$

$$Y_3 = -\frac{d}{2} log(\kappa_n) \tag{A.21}$$

$$\frac{\partial Y_1}{\partial \kappa_0} = -\alpha_n tr(T_n^{-1} \frac{\partial T_n}{\partial \kappa_0}) \tag{A.22}$$

Substituting for T_n where

$$T_n = T_0 + S + \frac{\kappa_0 n}{\kappa_0 + n}(\mu_0 - \bar{x})(\mu_0 - \bar{x})^T \tag{A.23}$$

we obtain:

$$\frac{\partial T_n}{\partial \kappa_0} = \frac{n^2}{(\kappa_0 + n)^2}(\bar{x} - \mu_0)(\bar{x} - \mu_0)^T \tag{A.24}$$

And:

$$\frac{\partial Y_1}{\partial \kappa_0} = -\alpha_n tr(T_n^{-1} \frac{n^2}{(\kappa_0 + n)^2}(\bar{x} - \mu_0)(\bar{x} - \mu_0)^T) \tag{A.25}$$

$$\frac{\partial Y_2}{\partial \kappa_0} = \frac{d}{2\kappa_0} \tag{A.26}$$

$$\frac{\partial Y_3}{\partial \kappa_0} = -\frac{d}{2(n + \kappa_0)} \tag{A.27}$$

Hence:

$$\frac{\partial log(D)}{\partial \kappa_0} = -\alpha_n tr(T_n^{-1} \frac{n^2}{(\kappa_0 + n)^2}(\bar{x} - \mu_0)(\bar{x} - \mu_0)^T) + \frac{nd}{2\kappa_0(n + \kappa_0)} \tag{A.28}$$

A.4.2 Derivative of $log(D)$ w.r.t. α_0

Terms in $log(P(D))$ which contain α_0 are as follows:

$$Z_1 = log(\Gamma_d(\alpha_n)) = log(\Gamma_d(\alpha_0 + n)) \tag{A.29}$$

$$Z_2 = -log(\Gamma_d(\alpha_0)) \tag{A.30}$$

$$Z_3 = \alpha_0 log(det(T_0)) \tag{A.31}$$

$$Z_4 = -\alpha_n log(det(T_n)) = -(\alpha_0 + n) log(det(T_n)) \tag{A.32}$$

$$\frac{\partial Z_1}{\partial \alpha_0} = \psi^d(\alpha_0 + n) \tag{A.33}$$

where ψ^d is a generalisation of the digamma function ψ. See Appendix B for the definition.

$$\frac{\partial Z_2}{\partial \alpha_0} = -\psi^d(\alpha_0) \tag{A.34}$$

$$\frac{\partial Z_3}{\partial \alpha_0} = log(det(T_0)) \tag{A.35}$$

$$\frac{\partial Z_4}{\partial \alpha_0} = -log(det(T_n)) \tag{A.36}$$

A.4.3 Derivative of $log(D)$ w.r.t. μ_0

The term below is the only one in the expression for $log(D)$ which contains the vector μ_{0j} where the subscript j refers to the data attributes:

$$W = -\frac{\alpha_n}{2} log(det(T_n)) \tag{A.37}$$

And we have:

$$\frac{\partial W}{\partial \mu_{0j}} = -\frac{\alpha_n}{2} tr(T_n^{-1} \frac{\partial T_n}{\partial \mu_{0j}}) \tag{A.38}$$

The partial derivative of T_n w.r.t. μ_{0j} is the partial derivative of the term:

$$\frac{\kappa_0 n}{\kappa_0 + n}(\mu_0 - \bar{x})(\mu_0 - \bar{x})^T \tag{A.39}$$

Defining:

$$V = (\mu_0 - \bar{x})(\mu_0 - \bar{x})^T \tag{A.40}$$

We may express V in subscript notation as:

$$V_{ij} = (\mu_{0i} - \bar{x}_i)(\mu_{0j} - \bar{x}_j) \tag{A.41}$$

If $i \neq j$ then:

$$\frac{\partial V}{\partial \mu_{0j}} = (\mu_{0i} - \bar{x}_i) \tag{A.42}$$

Otherwise if $i = j$:

$$\frac{\partial V}{\partial \mu_{0j}} = \frac{\partial(\mu_{0j}^2 - \bar{x}_j\mu_{0j} + \mu_{0j}^2)}{\partial \mu_{0j}} = 2(\mu_{0j} - \bar{x}_j) \tag{A.43}$$

A.4.4 Derivative of $log(D)$ w.r.t. T_0

The terms containing T_0 are:

$$V_1 = -\alpha_0 log(det(T_0)) \tag{A.44}$$

and:

$$V_2 = -\alpha_n log(det(T_n)) \tag{A.45}$$

Using the formula in Appendix B for the derivative of $log(det(T))$ we obtain:

$$\frac{\partial V_1}{\partial T_{0j}} = \alpha_0 tr(T_0^{-1}) \tag{A.46}$$

and:

$$\frac{\partial V_2}{\partial T_{0j}} = -\alpha_n tr(T_n^{-1}) \tag{A.47}$$

And

$$\frac{\partial log(D)}{\partial T_{0j}} = \alpha_0 tr(T_0^{-1}) - \alpha_n tr(T_n^{-1}) \tag{A.48}$$

Appendix B

Useful Formulae

B.1 First Derivatives of the Determinant

Let B be a square matrix with *adjugate* $Adj(B)$. The derivative of the determinant according to [119] is :

$$\frac{\partial det(B)}{\partial x} = tr(Adj(B)\frac{\partial B}{\partial x}) \tag{B.1}$$

Or:

$$\frac{\partial det(B)}{\partial x} = det(B)tr(B^{-1}\frac{\partial B}{\partial x}) \tag{B.2}$$

The derivative of the logarithm of the determinant is:

$$\frac{\partial log(det(B))}{\partial x} = tr(B^{-1}\frac{\partial B}{\partial x}) \tag{B.3}$$

(See The Matrix Cookbook: equations 41, 42, 43.)

B.2 Generalised Gamma Function

The generalised Gamma function $\Gamma_d(x)$ has the definition:

$$\Gamma_d(x) = \pi^{d(d-1)/4}\prod_{j=1}^{d}\Gamma(x+(1-j)/2) \tag{B.4}$$

The derivative of $log(\Gamma(x))$ is the *digamma function* denoted by $\psi(x)$ where:

$$\psi(x) = \frac{d}{dx}log(\Gamma(x)) \tag{B.5}$$

And we denote the derivative of $log(\Gamma_d(x))$ by $\psi^d(x)$ which satisfies:

$$\psi^d(x) = \sum_{j=1}^{d} \psi(x + (1-j)/2) \tag{B.6}$$

Note that the use of the notation $\psi^d(x)$ is intended to distinguish this term from the *polygamma* function $\psi_d(x)$ in which the subscript denotes the $(d-1)th$ order derivative of the logGamma function.

Bibliography

[1] Colon polyps. https://www.mayoclinic.org/diseases-conditions/colon-polyps/symptoms-causes/syc-20352875.

[2] The cancer digital slide archive, 2020. URL http://cancer.digitalslidearchive.net.

[3] nvidia technology - geforce products, 2020. URL https://www.nvidia.com/en-gb/geforce/technologies/cuda/technology/.

[4] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, et al. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *arXiv preprint arXiv:1603.04467*, 2016.

[5] Jeremy Adler and Ingela Parmryd. Colocalization analysis in fluorescence microscopy. In *Cell Imaging Techniques*, pages 97–109. Springer, 2013.

[6] Rakesh Agrawal, Tomasz Imieliński, and Arun Swami. Mining association rules between sets of items in large databases. In *Proceedings of the 1993 ACM SIGMOD international conference on Management of data*, pages 207–216, 1993.

[7] Nita Ahuja. Colorectal cancer stem cells–hype or real?: Comment on" combined cd133+/cd44+ expression as a prognostic indicator of disease-free survival in patients with colorectal cancer". *Archives of Surgery*, 147(1):24, 2012.

[8] Hirotugu Akaike. Use of an information theoretic quantity for statistical model identification. In *Proc. 5th Hawaii Int. Conf. System Sciences*, pages 249–250, 1972.

[9] Catherine Alix-Panabières, Jean-Pierre Vendrell, Monique Slijper, Olivier Pellé, Eric Barbotte, Grégoire Mercier, William Jacot, Michel Fabbro, and Klaus Pantel. Full-length cytokeratin-19 is released by human tumor cells: a potential

role in metastatic progression of breast cancer. *Breast Cancer Research*, 11(3): R39, 2009.

[10] The TensorFlow Authors. Definition of cifar10 model, 2018. URL `https://github.com/tensorflow/models/blob/master/tutorials/image/cifar10/cifar10.py`.

[11] Ayesha Azam. (This pathologist from University Hospitals Coventry and Warwickshre kindly outlined the structure of a post-operative pathology report concerning colon cancer). Private Communication, 2020.

[12] Abeer A Bahnassy, Abdel-Rahman N Zekri, Soumaya El-Houssini, Amal MR El-Shehaby, Moustafa Raafat Mahmoud, Samira Abdallah, and Mostafa El-Serafi. Cyclin a and cyclin d1 as significant prognostic markers in colorectal cancer patients. *BMC gastroenterology*, 4(1):22, 2004.

[13] A. Barysenka, A. W. Dress, and W. Schubert. An information theoretic thresholding method for detecting protein colocalizations in stacks of fluorescence images. *J Biotechnology*, Jan 2010.

[14] Babak Ehteshami Bejnordi, Geert Litjens, Nadya Timofeeva, Irene Otte-Höller, André Homeyer, Nico Karssemeijer, and Jeroen AWM van der Laak. Stain specific standardization of whole-slide histopathological images. *IEEE transactions on medical imaging*, 35(2):404–415, 2015.

[15] Yoav Benjamini and Yosef Hochberg. Controlling the false discovery rate: a practical and powerful approach to multiple testing. *Journal of the Royal statistical society: series B (Methodological)*, 57(1):289–300, 1995.

[16] Kaustav Bera, Kurt A Schalper, David L Rimm, Vamsidhar Velcheti, and Anant Madabhushi. Artificial intelligence in digital pathology—new tools for diagnosis and precision oncology. *Nature Reviews Clinical Oncology*, 16(11): 703–715, 2019.

[17] Sayantan Bhattacharya, George Mathew, Ernie Ruban, David BA Epstein, Andreas Krusche, Reyk Hillert, Walter Schubert, and Michael Khan. Toponome imaging system: in situ protein network mapping in normal and cancerous colon from the same patient reveals more than five-thousand cancer specific protein clusters and their subcellular annotation by using a three symbol code. *Journal of Proteome Research*, 9(12):6112–6125, 2010.

[18] Christopher M Bishop. *Pattern recognition and machine learning*. springer, 2006.

[19] PDQ Cancer Genetics Editorial Board. Cancer genetics risk assessment and counseling. 2016.

[20] M. Bode, M. Irmler, M. Friedenberger, C. May, K. Jung, C. Stephan, H. E. Meyer, C. Lach, R. Hillert, A. Krusche, J. Beckers, K. Marcus, and W. Schubert. Interlocking transcriptomics, proteomics and toponomics technologies for brain tissue analysis in murine hippocampus. *Proteomics*, 8:1170–1178, Mar 2008.

[21] Michael V Boland and Robert F Murphy. A neural network classifier capable of recognizing the patterns of all major subcellular structures in fluorescence microscope images of hela cells. *Bioinformatics*, 17(12):1213–1223, 2001.

[22] S Bolte and FP Cordelieres. A guided tour into subcellular colocalization analysis in light microscopy. *Journal of microscopy*, 224(3):213–232, 2006.

[23] Fred T Bosman, Fatima Carneiro, Ralph H Hruban, Neil D Theise, et al. *WHO classification of tumours of the digestive system.* Number Ed. 4. World Health Organization, 2010.

[24] Scarlet Fiona Brockmoeller and Nicholas Paul West. Predicting systemic spread in early colorectal cancer: Can we do better? *World Journal of Gastroenterology*, 25(23):2887, 2019.

[25] Taraz E Buck, Jieyue Li, Gustavo K Rohde, and Robert F Murphy. Toward the virtual cell: Automated approaches to building models of subcellular organization "learned" from microscopy images. *Bioessays*, 34(9):791–799, 2012.

[26] Dmitrii Bychkov, Nina Linder, Riku Turkki, Stig Nordling, Panu E Kovanen, Clare Verrill, Margarita Walliander, Mikael Lundin, Caj Haglund, and Johan Lundin. Deep learning based tissue analysis predicts outcome in colorectal cancer. *Scientific reports*, 8(1):1–11, 2018.

[27] Tadeusz Caliński and Jerzy Harabasz. A dendrite method for cluster analysis. *Communications in Statistics-theory and Methods*, 3(1):1–27, 1974.

[28] Cancer Research UK. Bowel cancer statistics, 2020. URL `https://www.cancerresearchuk.org/health-professional/cancer-statistics/statistics-by-cancer-type/bowel-cancer#headingZero`. [Online; accessed 4-October-2020].

[29] Centers for Disease Control and Prevention. Colorectal cancer statistics, 2020. URL `https://www.cdc.gov/cancer/colorectal/statistics/index.htm`. [Online; accessed 4-October-2020].

[30] IP Chandler and RS Houlston. Interobserver agreement in grading of colorectal cancers—findings from a nationwide web-based survey of histopathologists. *Histopathology*, 52(4):494–499, 2008.

[31] GJ Chang. Ajcc cancer staging 8th edition. `https://cancerstaging.org/CSE/Physician/Documents/AJCC 8th Edition Colorectal Webinar Secured.pdf`, 2018 (Accessed February 2020).

[32] Jianxu Chen and Chukka Srinivas. Automatic lymphocyte detection in h and e images with deep neural networks. *arXiv preprint arXiv:1612.03217*, 2016.

[33] S. C. Chen and R. F. Murphy. A graphical model approach to automated classification of protein subcellular location patterns in multi-cell images. *BMC Bioinformatics*, 7:90, 2006.

[34] Xueman Chen and Erwei Song. Turning foes to friends: targeting cancer-associated fibroblasts. *Nature reviews Drug discovery*, 18(2):99–115, 2019.

[35] Francesco Ciompi, Oscar Geessink, Babak Ehteshami Bejnordi, Gabriel Silva De Souza, Alexi Baidoshvili, Geert Litjens, Bram Van Ginneken, Iris Nagtegaal, and Jeroen Van Der Laak. The importance of stain normalization in colorectal tissue classification with convolutional networks. In *2017 IEEE 14th International Symposium on Biomedical Imaging (ISBI 2017)*, pages 160–163. IEEE, 2017.

[36] Sylvain V Costes, Dirk Daelemans, Edward H Cho, Zachary Dobbin, George Pavlakis, and Stephen Lockett. Automatic and quantitative measurement of protein-protein colocalization in live cells. *Biophysical journal*, 86(6):3993–4003, 2004.

[37] Nicolas Coudray, Paolo Santiago Ocampo, Theodore Sakellaropoulos, Navneet Narula, Matija Snuderl, David Fenyö, Andre L Moreira, Narges Razavian, and Aristotelis Tsirigos. Classification and mutation prediction from non–small cell lung cancer histopathology images using deep learning. *Nature medicine*, 24 (10):1559–1567, 2018.

[38] Heather D Couture, Lindsay A Williams, Joseph Geradts, Sarah J Nyante, Ebonee N Butler, JS Marron, Charles M Perou, Melissa A Troester, and Marc Niethammer. Image analysis with deep learning to predict breast cancer grade, er status, histologic subtype, and intrinsic subtype. *NPJ breast cancer*, 4(1): 1–8, 2018.

[39] Angel Cruz-Roa, Hannah Gilmore, Ajay Basavanhally, Michael Feldman, Shridar Ganesan, Natalie Shih, John Tomaszewski, Anant Madabhushi, and Fabio González. High-throughput adaptive sampling for whole-slide histopathology image analysis (hashi) via convolutional neural networks: Application to invasive breast cancer detection. *PloS one*, 13(5), 2018.

[40] DL Davies and DW Bouldin. A cluster separation measure. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1(2):95–104, 1979.

[41] Heather Dawson, Richard Kirsch, David K Driman, David E Messenger, Naziheh Assarzadegan, and Robert H Riddell. Optimizing the detection of venous invasion in colorectal cancer: the ontario, canada, experience and beyond. *Frontiers in oncology*, 4:354, 2015.

[42] Thomas de Bel, Meyke Hermsen, Jesper Kers, Jeroen van der Laak, Geert Litjens, et al. Stain-transforming cycle-consistent generative adversarial networks for improved segmentation of renal histopathology. In *Proceedings of the 2nd International Conference on Medical Imaging with Deep Learning; Proceedings of Machine Learning Research*, pages 151–163, 2019.

[43] Linde De Smedt, Sofie Palmans, and Xavier Sagaert. Tumour budding in colorectal cancer: what do we know and what can we do? *Virchows Archiv*, 468(4):397–408, 2016.

[44] Michael John De Smith, Michael F Goodchild, and Paul Longley. *Geospatial analysis: a comprehensive guide to principles, techniques and software tools*. Troubador Publishing Ltd, 2007.

[45] E Delemelle. Spatial sampling. In A Stewart Fotheringham and Peter A Rogerson, editors, *The SAGE handbook of spatial analysis*. SAGE Los Angeles, 2009.

[46] Arthur P Dempster, Nan M Laird, and Donald B Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the royal statistical society. Series B (methodological)*, pages 1–38, 1977.

[47] Carsten Denkert, Gunter von Minckwitz, Silvia Darb-Esfahani, Bianca Lederer, Barbara I Heppner, Karsten E Weber, Jan Budczies, Jens Huober, Frederick Klauschen, Jenny Furlanetto, et al. Tumour-infiltrating lymphocytes and prognosis in different subtypes of breast cancer: a pooled analysis of 3771 patients treated with neoadjuvant therapy. *The lancet oncology*, 19(1):40–50, 2018.

[48] John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of machine learning research*, 12(7), 2011.

[49] Kenneth W Dunn, Malgorzata M Kamocka, and John H McDonald. A practical guide to evaluating colocalization in biological microscopy. *American Journal of Physiology-Cell Physiology*, 300(4):C723–C742, 2011.

[50] Ed Uthman. Normal colonic mucosa. `https://flickr.com/photos/euthman/2802708709`, 2008. [Online; accessed 2020-03-08; Copyright 2008 Ed Uthman; licence CC BY 2.0 https://creativecommons.org/licenses/by/2.0/legalcode].

[51] Bradley Efron and Trevor Hastie. *Computer age statistical inference*, volume 5. Cambridge University Press, 2016.

[52] Partner Farlex. Medical dictionary. *Saunders comprehensive veterinary Dictionary*, 2012.

[53] Eric R Fearon. Molecular genetics of colorectal cancer. *Annual Review of Pathology: Mechanisms of Disease*, 6:479–507, 2011.

[54] E Melo Felipe De Sousa, Xin Wang, Marnix Jansen, Evelyn Fessler, Anne Trinh, Laura PMH De Rooij, Joan H De Jong, Onno J De Boer, Ronald Van Leersum, Maarten F Bijlsma, et al. Poor-prognosis colon cancer is defined by a molecularly distinct subtype and develops from serrated precursor lesions. *Nature medicine*, 19(5):614, 2013.

[55] Matthew Fleming, Sreelakshmi Ravula, Sergei F Tatishchev, and Hanlin L Wang. Colorectal carcinoma: Pathologic aspects. *Journal of gastrointestinal oncology*, 3(3):153, 2012.

[56] M. Friedenberger, M. Bode, A. Krusche, and W. Schubert. Fluorescence detection of protein clusters in individual cells and tissue sections by using toponome imaging system: sample preparation and measuring procedures. *Nat Protoc*, 2:2285–2294, 2007.

[57] Bela A Frigyik, Amol Kapila, and Maya R Gupta. Introduction to the dirichlet distribution and related processes. *Department of Electrical Engineering, University of Washignton, UWEETR-2010-0006*, 2010.

[58] Jérôme Galon and Anastasia Lanzi. Immunoscore and its introduction in clinical practice. *The Quarterly Journal of Nuclear Medicine and Molecular Imaging: Official Publication of the Italian Association of Nuclear Medicine*

(AIMN)[and] the International Association of Radiopharmacology (IAR),[and] Section of the Society of..., 2020.

[59] Jérôme Galon, Anne Costes, Fatima Sanchez-Cabo, Amos Kirilovsky, Bernhard Mlecnik, Christine Lagorce-Pagès, Marie Tosolini, Matthieu Camus, Anne Berger, Philippe Wind, et al. Type, density, and location of immune cells within human colorectal tumors predict clinical outcome. *Science*, 313(5795): 1960–1964, 2006.

[60] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *The Journal of Machine Learning Research*, 17(1):2096–2030, 2016.

[61] J García-Solano, P Conesa-Zamora, P Carbonell, J Trujillo-Santos, D Torres-Moreno D, I Pagán-Gómez, E Rodríguez-Braun, and M Pérez-Guillermo. Colorectal serrated adenocarcinoma shows a different profile of oncogene mutations, msi status and dna repair protein expression compared to conventional and sporadic msi-h carcinomas. *International journal of cancer*, 131(8):1790–1799, 2012.

[62] L. Gartner and L. Hiatt. The color textbook of histology, second edition, 2001.

[63] Milan Gavrilovic, Jimmy C Azar, Joakim Lindblad, Carolina Wählby, Ewert Bengtsson, Christer Busch, and Ingrid B Carlbom. Blind color decomposition of histological images. *IEEE transactions on medical imaging*, 32(6):983–994, 2013.

[64] GDC Support Team. Gdc data release, 2020. [e-mail; accessed 4-October-2020].

[65] Lise Getoor and Christopher P Diehl. Link mining: a survey. *ACM SIGKDD Explorations Newsletter*, 7(2):3–12, 2005.

[66] Schwarz Gideon et al. Estimating the dimension of a model. *The annals of statistics*, 6(2):461–464, 1978.

[67] Jean Charles Gilbert and Jorge Nocedal. Global convergence properties of conjugate gradient methods for optimization. *SIAM Journal on optimization*, 2(1):21–42, 1992.

[68] Ross Girshick. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015.

[69] Xavier Glorot, Antoine Bordes, and Yoshua Bengio. Deep sparse rectifier neural networks. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pages 315–323, 2011.

[70] Harvey Goldstein. Multilevel models in education and social research. 2011.

[71] Simon Graham, Quoc Dang Vu, Shan E Ahmed Raza, Ayesha Azam, Yee Wah Tsang, Jin Tae Kwak, and Nasir Rajpoot. Hover-net: Simultaneous segmentation and classification of nuclei in multi-tissue histology images. *Medical Image Analysis*, 58:101563, 2019.

[72] Robert L Grossman, Allison P Heath, Vincent Ferretti, Harold E Varmus, Douglas R Lowy, Warren A Kibbe, and Louis M Staudt. Toward a shared vision for cancer genomic data. *New England Journal of Medicine*, 375(12): 1109–1112, 2016.

[73] Justin Guinney, Rodrigo Dienstmann, Xin Wang, Aurélien De Reyniès, Andreas Schlicker, Charlotte Soneson, Laetitia Marisa, Paul Roepman, Gift Nyamundanda, Paolo Angelino, et al. The consensus molecular subtypes of colorectal cancer. *Nature medicine*, 21(11):1350–1356, 2015.

[74] David A Gutman, Jake Cobb, Dhananjaya Somanna, Yuna Park, Fusheng Wang, Tahsin Kurc, Joel H Saltz, Daniel J Brat, Lee AD Cooper, and Jun Kong. Cancer digital slide archive: an informatics resource to support integrated in silico analysis of tcga pathology data. *Journal of the American Medical Informatics Association*, 20(6):1091–1098, 2013.

[75] James S Hale, Balint Otvos, Maksim Sinyuk, Alvaro G Alvarado, Masahiro Hitomi, Kevin Stoltz, Qiulian Wu, William Flavahan, Bruce Levison, Mette L Johansen, et al. Cancer stem cell-specific scavenger receptor cd36 drives glioblastoma progression. *Stem cells*, 32(7):1746–1758, 2014.

[76] Chencheng Han, Tongyan Liu, and Rong Yin. Biomarkers for cancer-associated fibroblasts. *Biomarker Research*, 8(1):1–8, 2020.

[77] Robert M Haralick. Statistical and structural approaches to texture. *Proceedings of the IEEE*, 67(5):786–804, 1979.

[78] C Hassan, A Gimeno-García, M Kalager, Cristiano Spada, A Zullo, Guido Costamagna, C Senore, DK Rex, and E Quintero. Systematic review with meta-analysis: the incidence of advanced neoplasia after polypectomy in patients with and without low-risk adenomas. *Alimentary pharmacology & therapeutics*, 39(9):905–912, 2014.

[79] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The elements of statistical learning: data mining, inference, and prediction*. Springer Science & Business Media, 2009.

[80] National Health. Bowel polyps. `https://www.nhs.uk/conditions/bowel-polyps`.

[81] Katherine A Heller and Zoubin Ghahramani. Bayesian hierarchical clustering. In *Proceedings of the 22nd international conference on Machine learning*, pages 297–304. ACM, 2005.

[82] Robert E Hewitt, Desmond G Powe, G Ian Carter, and David R Turner. Desmoplasia and its relevance to colorectal tumour invasion. *International journal of cancer*, 53(1):62–69, 1993.

[83] Yosef Hochberg and Yoav Benjamini. More powerful procedures for multiple significance testing. *Statistics in medicine*, 9(7):811–818, 1990.

[84] Hui-Ping Hsu, Ming-Derg Lai, Jenq-Chang Lee, Meng-Chi Yen, Tzu-Yang Weng, Wei-Ching Chen, Jung-Hua Fang, and Yi-Ling Chen. Mucin 2 silencing promotes colon cancer metastasis through interleukin-6 signaling. *Scientific reports*, 7(1):1–14, 2017.

[85] K. Huang and R. F. Murphy. Automated classification of subcellular patterns in multicell images without segmentation into single cells. In *Proceedings of the 2004 IEEE International Symposium on Biomedical Imaging*, 2004.

[86] Wolfgang Huber, Vincent J Carey, Robert Gentleman, Simon Anders, Marc Carlson, Benilton S Carvalho, Hector Corrada Bravo, Sean Davis, Laurent Gatto, Thomas Girke, et al. Orchestrating high-throughput genomic analysis with bioconductor. *Nature methods*, 12(2):115, 2015.

[87] Ahmad Humayun, Shan-e-Ahmed Raza, C Waddington, Sylvie Abouna, Michael Khan, and Nasir Mahmood Rajpoot. A novel framework for molecular co-expression pattern analysis in multi-channel toponome fluorescence images. *Proceedings of the 2011 Microscopic Image Analysis with Applications in Biology*, 2011.

[88] National Cancer Institute. Tumor grade factsheet, 2018. URL `https://www.cancer.gov/about-cancer/diagnosis-staging/prognosis/tumor-grade-fact-sheet`. [Online; accessed November 22nd 2018].

[89] National Cancer Institute. The cancer genome atlas program. https://www.cancer.gov/about-nci/organization/ccg/research/structural-genomics/tcga, 2020. Accessed March, 2020.

[90] Andrew Janowczyk and Anant Madabhushi. Deep learning for digital pathology image analysis: A comprehensive tutorial with selected use cases. *Journal of pathology informatics*, 7, 2016.

[91] JR Jass. Classification of colorectal cancer based on correlation of clinical, morphological and molecular features. *Histopathology*, 50(1):113–130, 2007.

[92] Raghu Kalluri. The biology and function of fibroblasts in cancer. *Nature Reviews Cancer*, 16(9):582, 2016.

[93] Klaus Kayser. *Travels on conferences: Evolution of Digital Pathology 1992–2018*. Lehmanns Media, 2019.

[94] Klaus Kayser, Holger Schultz, Torsten Goldmann, Jürgen Görtler, Gian Kayser, and Ekkehard Vollmer. Theory of sampling and its application in tissue based diagnosis. *Diagnostic Pathology*, 4(1):6, 2009.

[95] Kresten Krarup Keller, Ina Trolle Andersen, Johnnie Bremholm Andersen, Ute Hahn, Kristian STENGAARD-PEDERSEN, E-M Hauge, and Jens Randel Nyengaard. Improving efficiency in stereology: a study applying the proportionator and the autodisector on virtual slides. *Journal of microscopy*, 251(1): 68–76, 2013.

[96] Feras J Abdul Khalek, G Ian Gallicano, and Lopa Mishra. Colon cancer stem cells. *Gastrointestinal cancer research: GCR*, (Supplement 1):S16, 2010.

[97] A A Khan, Mujahid Humayun, SeA Raza, Michael Khan, and Nasir M. Rajpoot. A novel paradigm for mining cell phenotypes in multi-tag bioimages using a locality preserving nonlinear embedding. In *Neural Information Processing:Lecture Notes in Computer Science Volume 7666*, pages 575–583, 2012.

[98] Adnan M Khan, Shan-e-Ahmed Raza, Michael Khan, and Nasir M Rajpoot. Cell phenotyping in multi-tag fluorescent bioimages. *Neurocomputing*, 134: 254–261, 2014.

[99] Adnan Mujahid Khan, Ahmad Humayun, Michael Khan, Nasir M Rajpoot, et al. A novel paradigm for mining cell phenotypes in multi-tag bioimages using a locality preserving nonlinear embedding. In *International Conference on Neural Information Processing*, pages 575–583. Springer, 2012.

[100] Adnan Mujahid Khan, Nasir Rajpoot, Darren Treanor, and Derek Magee. A nonlinear mapping approach to stain normalization in digital histopathology images using image-specific color deconvolution. *IEEE Transactions on Biomedical Engineering*, 61(6):1729–1738, 2014.

[101] Hyunsoo Kim and Haesun Park. Sparse non-negative matrix factorizations via alternating non-negativity-constrained least squares for microarray data analysis. *Bioinformatics*, 23(12):1495–1502, 2007.

[102] Diederik P Kingma and Jimmy Lei Ba. Adam: A method for stochastic gradient descent. In *ICLR: International Conference on Learning Representations*, 2015.

[103] Alexander Kirillov, Kaiming He, Ross Girshick, Carsten Rother, and Piotr Dollár. Panoptic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 9404–9413, 2019.

[104] Viktor H Koelzer, Inti Zlobec, and Alessandro Lugli. Tumor budding in colorectal cancer—ready for diagnostic practice? *Human pathology*, 47(1):4–19, 2016.

[105] Jan Kölling, Daniel Langenkämper, Sylvie Abouna, Michael Khan, and Tim W Nattkemper. Whide - a web tool for visual data mining colocation patterns in multivariate bioimages. *Bioinformatics*, 28(8):1143–1150, 2012.

[106] Tsuyoshi Konishi, Yoshifumi Shimada, Lik Hang Lee, Marcela S Cavalcanti, Meier Hsu, J Joshua Smith, Garrett M Nash, Larissa K Temple, José G Guillem, Philip B Paty, et al. Poorly differentiated clusters predict colon cancer recurrence: an in-depth comparative analysis of invasive-front prognostic markers. *The American journal of surgical pathology*, 42(6):705, 2018.

[107] Navid Alemi Koohababni, Mostafa Jahanifar, Ali Gooya, and Nasir Rajpoot. Nuclei detection using mixture density networks. In *International Workshop on Machine Learning in Medical Imaging*, pages 241–248. Springer, 2018.

[108] Violeta N Kovacheva, Adnan M Khan, David Epstein, Michael Khan, and Nasir M Rajpoot. Diswop: A novel measure for cell-level protein network analysis in localised proteomics image data. 2013.

[109] Violeta N Kovacheva, Adnan M Khan, Michael Khan, David BA Epstein, and Nasir M Rajpoot. Diswop: a novel measure for cell-level protein network analysis in localized proteomics image data. *Bioinformatics*, 30(3):420–427, 2014.

[110] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

[111] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[112] Harold W Kuhn. Variants of the hungarian method for assignment problems. *Naval research logistics quarterly*, 3(4):253–258, 1956.

[113] Maxime W Lafarge, Josien PW Pluim, Koen AJ Eppenhof, Pim Moeskops, and Mitko Veta. Domain-adversarial neural networks to address the appearance variability of histopathology images. In *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*, pages 83–91. Springer, 2017.

[114] Corinna Lang-Schwarz, Balint Melcher, Franziska Haumaier, Anna Schneider-Fuchs, Klaus Lang-Schwarz, Jens Krugmann, Michael Vieth, and William Sterlacci. Budding, tumor-infiltrating lymphocytes, gland formation: scoring leads to new prognostic groups in world health organization low-grade colorectal cancer with impact on survival. *Human pathology*, 89:81–89, 2019.

[115] Yann LeCun, Bernhard Boser, John S Denker, Donnie Henderson, Richard E Howard, Wayne Hubbard, and Lawrence D Jackel. Backpropagation applied to handwritten zip code recognition. *Neural computation*, 1(4):541–551, 1989.

[116] Libre Pathology. Tils, 2020. URL `https://librepathology.org/wiki/File:Tumour-infiltrating_lymphocytes_-_2_--_very_high_mag.jpg`. [Online; accessed 14-October-2020].

[117] Marc Macenko, Marc Niethammer, James S Marron, David Borland, John T Woosley, Xiaojun Guan, Charles Schmitt, and Nancy E Thomas. A method for normalizing histology slides for quantitative analysis. In *2009 IEEE International Symposium on Biomedical Imaging: From Nano to Macro*, pages 1107–1110. IEEE, 2009.

[118] David JC MacKay and David JC Mac Kay. *Information theory, inference and learning algorithms*. Cambridge university press, 2003.

[119] Jan R. Magnus and H. Neudecker. Wiley, 1999.

[120] Thomas G Mayerhöfer, Susanne Pahlow, and Jürgen Popp. The bouguer-beer-lambert law: Shining light on the obscure. *ChemPhysChem*, 21(18):2029, 2020.

[121] Bojana Mitrovic, David F Schaeffer, Robert H Riddell, and Richard Kirsch. Tumor budding in colorectal carcinoma: time to take notice. *Modern Pathology*, 25(10):1315, 2012.

[122] Matthew J Munro, Susrutha K Wickremesekera, Lifeng Peng, Swee T Tan, and Tinte Itinteang. Cancer stem cells in colorectal cancer: a review. *Journal of Clinical Pathology*, 71(2):110–116, 2018.

[123] Oscar Murcia, Miriam Juárez, Eva Hernández-Illán, Cecilia Egoavil, Mar Giner-Calabuig, María Rodríguez-Soler, and Rodrigo Jover. Serrated colorectal cancer: Molecular classification, prognosis, and response to chemotherapy. *World journal of gastroenterology*, 22(13):3516, 2016.

[124] K. Murphy. Conjugate bayesian analysis of the gaussian distribution. `http://http://www-devel.cs.ubc.ca/~murphyk/Papers/bayesGauss.pdf`, 2007. Accessed Aug 30, 2014.

[125] Jawad Nagi, Frederick Ducatelle, Gianni A Di Caro, Dan Cireşan, Ueli Meier, Alessandro Giusti, Farrukh Nagi, Jürgen Schmidhuber, and Luca Maria Gambardella. Max-pooling convolutional neural networks for vision-based hand gesture recognition. In *2011 IEEE International Conference on Signal and Image Processing Applications (ICSIPA)*, pages 342–347. IEEE, 2011.

[126] Iris D Nagtegaal, Robert D Odze, David Klimstra, Valerie Paradis, Massimo Rugge, Peter Schirmacher, Kay M Washington, Fatima Carneiro, Ian A Cree, and WHO Classification of Tumours Editorial Board. The 2019 who classification of tumours of the digestive system. *Histopathology*, 76(2):182–188, 2020.

[127] National Cancer, Institute. The Cancer Genome Atlas. `http://cancergenome.nih.gov/`, 2016. `http://cancergenome.nih.gov/`.

[128] National Cancer Institute. lymphocyte, 2020. URL `https://www.cancer.gov/publications/dictionaries/cancer-terms/def/lymphocyte`. [Online; accessed 4-October-2020].

[129] National Cancer Institute, 2020. URL `https://www.cancer.gov/types/colorectal/hp/colorectal-genetics-pdq#cit/section_2.28`. Online; accessed 2020-08-02.

[130] National Health Service. Treatment ovarian cyst, 2020. URL `https://www.nhs.uk/conditions/ovarian-cyst/treatment/`. [Online; accessed 2020].

[131] National Institutes of Health: National Cancer Institute. Cenomic data commons, 2020. URL `https://gdc.cancer.gov`. [Online;accessed 4-October-2020].

[132] Justin Newberg, Juchang Hua, and Robert F Murphy. Location proteomics: systematic determination of protein subcellular location. In *Systems Biology*, pages 313–332. Springer, 2009.

[133] Justin Y Newberg, Jieyue Li, Arvind Rao, Pontén, Fredrik, Uhlén, Mathias, Emma Lundberg, and Robert F Murphy. Automated analysis of human protein atlas immunofluorescence images. In *Biomedical Imaging: From Nano to Macro, 2009. ISBI'09. IEEE International Symposium on*, pages 1023–1026. IEEE, 2009.

[134] C Lo Nigro, A Comino, D Vivenza, C Granetto, M Ferrero, L Lattanzio, C Varamo, V Ricci, and MC Merlano. Tumor-infiltrating lymphocytes (tils) density as prognostic determinant in stage ii colorectal cancer, 2016.

[135] Oddmund Nordgård, Satu Oltedal, Hartwig Kørner, Ole Gunnar Aasprong, Kjersti Tjensvoll, Bjørnar Gilje, and Reino Heikkilä. The potential of cytokeratin 20 and mucin 2 mrna as metastasis markers in regional lymph nodes of colon cancer patients investigated by quantitative rt-pcr. *International journal of colorectal disease*, 24(3):261–268, 2009.

[136] Katsuhiko Nosho, Yoshifumi Baba, Noriko Tanaka, Kaori Shima, Marika Hayashi, Jeffrey A Meyerhardt, Edward Giovannucci, Glenn Dranoff, Charles S Fuchs, and Shuji Ogino. Tumour-infiltrating t-cell subsets, molecular changes in colorectal cancer, and prognosis: cohort study and literature review. *The Journal of pathology*, 222(4):350–366, 2010.

[137] Jong Seob Park, Jung Wook Huh, Yoon Ah Park, Yong Beom Cho, Seong Hyeon Yun, Hee Cheol Kim, Woo Yong Lee, and Ho-Kyung Chun. Prognostic comparison between mucinous and nonmucinous adenocarcinoma in colorectal cancer. *Medicine*, 94(15), 2015.

[138] Bernhard Preim and Charl P Botha. *Visual computing for medicine: theory, algorithms, and applications*. Newnes, 2013.

[139] Judith MS Prewitt and Mortimer L Mendelsohn. The analysis of cell images. *Annals of the New York Academy of Sciences*, 128(3):1035–1053, 1966.

[140] Jing Qian, Kaja Tikk, Korbinian Weigl, Yesilda Balavarca, and Hermann Brenner. Fibroblast growth factor 21 as a circulating biomarker at various

stages of colorectal carcinogenesis. *British journal of cancer*, 119(11):1374–1382, 2018.

[141] JA Ramos-Vara and MA Miller. When tissue antigens and antibodies get along: revisiting the technical aspects of immunohistochemistry—the red, brown, and blue technique. *Veterinary pathology*, 51(1):42–87, 2014.

[142] WM Rand. Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical Association. American Statistical Association*, 66 (336):846–885, 1971.

[143] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016.

[144] Lucia Ricci-Vitiani, Eros Fabrizi, Elisabetta Palio, and Ruggero De Maria. Colon cancer stem cells. *Journal of Molecular Medicine*, 87(11):1097–1104, 2009.

[145] Peter J Rousseeuw. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics*, 20:53–65, 1987.

[146] Arnout C Ruifrok, Dennis A Johnston, et al. Quantification of histochemical staining by color deconvolution. *Analytical and quantitative cytology and histology*, 23(4):291–299, 2001.

[147] Arnout C Ruifrok, Dennis A Johnston, et al. Quantification of histochemical staining by color deconvolution. *Analytical and quantitative cytology and histology*, 23(4):291–299, 2001.

[148] Eugenio Sangiorgi and Mario R Capecchi. Bmi1 is expressed in vivo in intestinal stem cells. *Nature genetics*, 40(7):915–920, 2008.

[149] Emma M Schatoff, Benjamin I Leach, and Lukas E Dow. Wnt signaling and colorectal cancer. *Current colorectal cancer reports*, 13(2):101–110, 2017.

[150] Dominik Scherer, Andreas Müller, and Sven Behnke. Evaluation of pooling operations in convolutional architectures for object recognition. In *International conference on artificial neural networks*, pages 92–101. Springer, 2010.

[151] W. Schubert, B. Bonnekoh, A. J. Pommer, L. Philipsen, R. Bockelmann, Y. Malykh, H. Gollnick, M. Friedenberger, M. Bode, and A. W. Dress. Ana-

lyzing proteome topology and function by automated multidimensional fluorescence microscopy. *Nat. Biotechnol.*, 24:1270–1278, Oct 2006.

[152] Walter Schubert, Anne Gieseler, Andreas Krusche, Peter Serocka, and Reyk Hillert. Next-generation biomarkers based on 100-parameter functional super-resolution microscopy tis. *New biotechnology*, 29(5):599–610, 2012.

[153] Ahmad Humayun Shan-e Ahmed Raza, Sylvie Abouna, Tim W Nattkemper, David BA Epstein, Michael Khan, and Nasir M Rajpoot. Ramtab: robust alignment of multi-tag bioimages. *PloS one*, 7(2), 2012.

[154] Caroline Mary Shapcott, Nasir Rajpoot, and Katherine Hewitt. Deep learning with sampling for colon cancer histology images. *Frontiers in Bioengineering and Biotechnology*, 7:52, 2019.

[155] S. Siegel. *Nonparametric statistics for the behavioral sciences*. McGraw-Hill series in psychology. McGraw-Hill, 1956. URL `https://books.google.co.uk/books?id=6t9fAAAAIAAJ`.

[156] Korsuk Sirinukunwattana, Richard S Savage, Muhammad F Bari, David RJ Snead, and Nasir M Rajpoot. Bayesian hierarchical clustering for studying cancer gene expression data with unknown statistics. *PloS one*, 8(10):e75748, 2013.

[157] Korsuk Sirinukunwattana, Shan E Ahmed Raza, Yee-Wah Tsang, David RJ Snead, Ian A Cree, and Nasir M Rajpoot. 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, 2016.

[158] Korsuk Sirinukunwattana, David Snead, David Epstein, Zia Aftab, Imaad Mujeeb, Yee Wah Tsang, Ian Cree, and Nasir Rajpoot. Novel digital signatures of tissue phenotypes for predicting distant metastasis in colorectal cancer. *Scientific reports*, 8(1):1–13, 2018.

[159] David RJ Snead, Yee-Wah Tsang, Aisha Meskiri, Peter K Kimani, Richard Crossman, Nasir M Rajpoot, Elaine Blessing, Klaus Chen, Kishore Gopalakrishnan, Paul Matthews, et al. Validation of digital pathology imaging for primary histopathological diagnosis. *Histopathology*, 68(7):1063–1072, 2016.

[160] American Cancer Society. Survival rates for colorectal cancer, 2020. URL `https://www.cancer.org/cancer/colon-rectal-cancer/detection-diagnosis-staging/survival-rates.html`.

[161] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958, 2014.

[162] Blausen staff. The colon. `https://librepathology.org/wiki/File:Blausen_0603_LargeIntestine_Anatomy.png`, 2014. [Online accessed 2021-04-01;This file is licensed under the Creative Commons Attribution 3.0 Unported license.].

[163] Daniel A Symonds and Austin L Vickery Jr. Mucinous carcinoma of the colon and rectum. *Cancer*, 37(4):1891–1900, 1976.

[164] Yasser M Tabana, Saad S Dahham, Amin M Shah, and Abdul Majid. Major signaling pathways of colorectal carcinogenesis. *Recent Adv Colon Cancer*, 1: 1–2, 2016.

[165] David Tellez, Maschenka Balkenhol, Nico Karssemeijer, Geert Litjens, Jeroen van der Laak, and Francesco Ciompi. H and e stain augmentation improves generalization of convolutional networks for histopathological mitosis detection. In *Medical Imaging 2018: Digital Pathology*, volume 10581, page 105810Z. International Society for Optics and Photonics, 2018.

[166] David Tellez, Geert Litjens, Péter Bándi, Wouter Bulten, John-Melle Bokhorst, Francesco Ciompi, and Jeroen van der Laak. Quantifying the effects of data augmentation and stain color normalization in convolutional neural networks for computational pathology. *Medical image analysis*, 58:101544, 2019.

[167] The Cancer Genome Network et al. Comprehensive molecular characterization of human colon and rectal cancer. *Nature*, 487(7407):330–337, 2012.

[168] The Human Protein Atlas. Tp53, 2020. URL `https://www.proteinatlas.org/ENSG00000141510-TP53/`.

[169] Eu-Wing Toh, Peter Brown, Eva Morris, Ian Botterill, and Philip Quirke. Area of submucosal invasion and width of invasion predicts lymph node metastasis in pt1 colorectal cancers. *Diseases of the Colon & Rectum*, 58(4):393–400, 2015.

[170] Joke Tommelein, Laurine Verset, Tom Boterberg, Pieter Demetter, Marc Bracke, and Olivier De Wever. Cancer-associated fibroblasts connect metastasis-promoting communication in colorectal cancer. *Frontiers in oncology*, 5:63, 2015.

[171] Nicholas Andrew Trahearn. *Registration and multi-immunohistochemical analysis of whole slide images of serial tissue sections*. University of Warwick, 2017.

[172] Tadashi Tsujino, Iwao Seshimo, Hirofumi Yamamoto, Chew Yee Ngan, Koji Ezumi, Ichiro Takemasa, Masataka Ikeda, Mitsugu Sekimoto, Nariaki Matsuura, and Morito Monden. Stromal myofibroblasts predict disease recurrence for colorectal cancer. *Clinical cancer research*, 13(7):2082–2090, 2007.

[173] John W Tukey. *Exploratory data analysis*, volume 2. Reading, Mass., 1977.

[174] H Ueno, J Murphy, JR Jass, H Mochizuki, and IC Talbot. Tumour 'budding' as an index to estimate the potential of aggressiveness in rectal cancer. *Histopathology*, 40(2):127–132, 2002.

[175] Hideki Ueno, Kazuo Hase, Yojiro Hashiguchi, Hideyuki Shimazaki, Shinji Yoshii, Shin-ei Kudo, Masafumi Tanaka, Yoshito Akagi, Takeshi Suto, Shinji Nagata, et al. Novel risk factors for lymph node metastasis in early invasive colorectal cancer: a multi-institution pathology review. *Journal of gastroenterology*, 49 (9):1314–1323, 2014.

[176] U.S. National Library of Medicine. Genetics Home Reference - MLH1. `https://ghr.nlm.nih.gov/gene/MLH1`, 2016. `https://ghr.nlm.nih.gov/gene/MLH1`.

[177] Abhishek Vahadane, Tingying Peng, Amit Sethi, Shadi Albarqouni, Lichao Wang, Maximilian Baust, Katja Steiger, Anna Melissa Schlitter, Irene Esposito, and Nassir Navab. Structure-preserving color normalization and sparse stain separation for histological images. *IEEE transactions on medical imaging*, 35 (8):1962–1971, 2016.

[178] Valerie Healey. (Valerie kindly provided photgraphs of a microtome and a digital microscope). Private Communication, 2019.

[179] Andrea Vedaldi and Karel Lenc. Matconvnet: Convolutional neural networks for matlab. In *Proceedings of the 23rd ACM international conference on Multimedia*, pages 689–692, 2015.

[180] Ruchika Verma, Neeraj Kumar, Abhijeet Patil, Nikhil Kurian, Swapnil Rane, and Amit Sethi. Multi-organ nuclei segmentation and classification challenge 2020, 02 2020. URL `http://rgdoi.net/10.13140/RG.2.2.12290.02244/1`.

[181] Wilko Weichert, Carsten Denkert, Mick Burkhardt, Tserenchunt Gansukh, Joachim Bellach, Peter Altevogt, Manfred Dietel, and Glen Kristiansen. Cyto-

plasmic cd24 expression in colorectal cancer independently correlates with shortened patient survival. *Clinical Cancer Research*, 11(18):6574–6581, 2005.

[182] NP West, M Dattani, P McShane, G Hutchins, J Grabsch, W Mueller, D Treanor, P Quirke, and H Grabsch. The proportion of tumour cells is an independent predictor for survival in colorectal cancer patients. *British journal of cancer*, 102(10):1519–1523, 2010.

[183] J. Whittaker. *Graphical Models in Applied Multivariate Statistics*. Wiley: Chichester., 1990.

[184] Wikimedia user Nephron. Sessile serrated adenoma 2 high mag, 2010. URL `https://commons.wikimedia.org/wiki/File:Sessile_serrated_adenoma_2_high_mag.jpg`. [Online; accessed 2020-03-05; Copyright 2010 Michael Bonert; licence CC BY-SA 3.0 https://creativecommons.org/licenses/by-sa/3.0/legalcode].

[185] Wikipedia contributors. Large intestine — Wikipedia, the free encyclopedia, 2019. URL `https://en.wikipedia.org/wiki/Large_intestine`. [Online; accessed 27-July-2020].

[186] Yan Xu, Zhipeng Jia, Liang-Bo Wang, Yuqing Ai, Fang Zhang, Maode Lai, I Eric, and Chao Chang. Large scale tissue histopathology image classification, segmentation, and visualization via deep convolutional activation features. *BMC bioinformatics*, 18(1):1–17, 2017.

[187] Yinyin Yuan. Modelling the spatial heterogeneity and molecular correlates of lymphocytic infiltration in triple-negative breast cancer. *Journal of The Royal Society Interface*, 12(103):20141153, 2015.

[188] Yujuan Zhou, Longzheng Xia, Heran Wang, Linda Oyang, Min Su, Qiang Liu, Jingguan Lin, Shiming Tan, Yutong Tian, Qianjin Liao, et al. Cancer stem cells in progression of colorectal cancer. *Oncotarget*, 9(70):33403, 2018.

[189] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2223–2232, 2017.

[190] Timothy L Zisman and David T Rubin. Colorectal cancer and dysplasia in inflammatory bowel disease. *World journal of gastroenterology: WJG*, 14(17): 2662, 2008.

[191] Inti Zlobec, Martin D Berger, and Alessandro Lugli. Tumour budding and its clinical implications in gastrointestinal cancers. *British journal of cancer*, 123 (5):700–708, 2020.