Classifying non-small cell lung carcinoma in histological images using a convolutional neural network

Veera Timonen

University of Tampere

Faculty of Medicine and Health Technology

Master’s Thesis

June 2019

ABBREVIATIONS

IHC = Immunohistochemistry

WSI = Whole Slide Image

CNN = Convolutional Neural Network

PD-L1 = Programmed Death-Ligand 1

WSI = Whole Slide Image

BCE = Binary Cross-Entropy

AUC = Area Under Curve

IoU = Intersection over Union

NSCLC = Non-Small Cell Lung Cancer

Keywords: CNN, convolutional neural network, digital pathology, immunohistochemical fluoro-chromogenic stain, lung cancer, machine learning, PD-L1, U-net, whole slide images

ABSTRACT

UNIVERSITY OF TAMPERE

Master's Degree Programme in Bioinformatics

TIMONEN, VEERA: **Classifying non-small cell lung carcinoma in histological images using a convolutional neural network**

Master of Science Thesis, 46 pages

May 2019

Major: Bioinformatics

Examiner: Prof. Matti Nykter

Keywords: CNN, convolutional neural network, deep learning, digital pathology, immunohistochemical fluoro-chromogenic stain, lung cancer, machine learning, PD-L1, U-net, whole slide images

The purpose of this Master's thesis was to teach a convolutional neural network to recognize non-small cell lung cancer from whole slide images (WSI) and to separate regions of interest from other tissue. IHC fluoro-chromogenically stained whole slide images under brightfield illumination were used as target images, and the same WSIs with cytokeratin masks applied under fluorescent illumination were used as input images.

An immunohistochemical fluoro-chromogenic dye is done when PD-L1-expressing tumor regions and PD1-expressing alveolar macrophages need to be distinguished. Cytokeratin-positive carcinoma regions show clearly in brightfield images. It is important to separate these regions especially when considering immunotherapy as treatment, because there exist antibody based medications against both PD1- and PD-L1 expressing tumor- and lymphocyte cells, and the areas surrounding cancer may cause false positives leading to immunotherapy being poorly targeted.

The method is based on U-net architecture in a convolutional neural network. A CNN is capable of achieving excellent results in tasks including image recognition, and U-net has been specifically designed for medical image analysis tasks.

The results show that the neural network used is capable of distinguishing cancer regions from other tissue with good accuracy (AUC = 0.96).

TIIVISTELMÄ

TAMPEREEN YLIOPISTO

Bioteknologian koulutusohjelma, Bioinformatiikan maisteriohjelma

TIMONEN, VEERA: **Ei-pienisoluisen keuhkosyövän luokittelu histologisissa kuvissa käyttäen konvoluutioneuroverkkoa**

Pro gradu, 46 sivua

Toukokuu 2019

Pääaine: Bioinformatiikka

Tarkastaja: Prof. Matti Nykter

Avainsanat: CNN, konvoluutioneuroverkko, digitaalipatologia, immunohistokemiallinen fluoro-kromogeeninen värjäys, keuhkosyöpä, koneoppiminen, syväoppiminen, PD-L1, U-net, kokoleikekuva

Tämän pro gradun tavoitteena oli opettaa konvoluutioneuroverkko tunnistamaan ei-pienisoluista keuhkosyöpää kudosleikkeistä otetuista kuvista ja erottamaan terveen kudoksen ja syövän alueet. Materiaalina käytettiin immunohistokemiallisesti ja fluoro-kromogeenisesti värjättyjä leikkeitä; alkuperäiset leikkeet syötekuvina ja leikkeet sytokeratiinimaskin kanssa tuloskuvina.

Immunohistokemiallinen fluoro-kromogeeninen värjäys tehdään, kun halutaan erottaa PD-L1-ekspressoivat tuumorialueet ja PD1-ekspressoivat alveolaariset makrofagit toisistaan. Immunoterapiaa ajatellen on tärkeää, että nämä alueet erotetaan toisistaan, sillä vasta-ainepohjaisia lääkkeitä on olemassa sekä PD1- että PD-L1-ekspressoivia syöpä- ja lymfosyyttisoluja vastaan, ja syöpää ympäröivät alueet voivat ilmentää väärää positiivisuutta. Esimerkiksi TAM-makrofagit ja kuolleet syöpäsolut voivat aiheuttaa vääriä positiivisia tuloksia, ja immunoterapia on vaikeampi kohdentaa.

Menetelmä perustuu U-net-arkkitehtuuriin konvoluutioneuroverkossa. Konvoluutioneuroverkko kykenee saavuttamaan hyviä tuloksia kuviin liittyvissä tehtävissä, ja U-net on suunniteltu erityisesti lääketieteellistä kuva-analyysiä varten.

Tulokset osoittivat, että neuroverkko kykenee erottamaan syöpäalueita muusta kudoksesta hyvällä tarkkuudella (AUC = 0.96).

PREFACE

This thesis study was conducted at Faculty of Medicine and Health Technology, Tampere University with the Bioimage Informatics group.

I would like to thank my thesis supervisor Pekka Ruusuvuori for the opportunity to work on an interesting study topic with this group. Pekka and my colleagues in the Bioimage Informatics group offered unwavering support and advice that made this thesis possible on schedule, so a big “thank you” goes to all of them as well. I’d also like to thank Jorma Isola, Satu Luhtala and Teppo Haapaniemi for providing the data and valuable information on staining techniques and other biological parts of the study.

I’d also like to express my endless gratitude towards my amazing parents and family for supporting me through my studies that took me on a unique and winding path from genetics to bioinformatics and finally machine learning.

Veera Timonen,

29. April 2019

CONTENTS

1. INTRODUCTION

This thesis study was conducted working for Pekka Ruusuvuori's Bioimage Informatics research group at Faculty of Medicine and Health Technology, Tampere University. In this chapter the work's purpose will be presented, along with how non-small cell lung cancer operates and a small summary of the used data and methods. The second chapter will be an overview on the theory behind used methods, third chapter will present the implementation of the pipeline, the dataset and other materials used. Fourth chapter will present the results, and discussion will take place in chapter five. Conclusions of the study will be in chapter six. The approach used was a combined perspective of cellular biology, signal processing and neural network medical image analysis.

Non-small cell lung cancer is the most common type of lung cancer found in humans, and its severity is measured in tumor stages I-IV with IV being the most advanced stage of cancer (**Table 1**). The treatment for early stages of lung cancer is commonly surgery, and chemotherapy for patients whose lung cancer tumors have already metastasized. Gefitinib is one of the licenced treatment methods. Carcinoma *in situ* stands for the precursor to lung cancer, when an actual tumor is not detected but cancerous cells are. There is a difference in the severity of non-small and small cell lung cancer: small cell lung cancer tends to be found at a more advanced stage, has a faster growth rate and thus may have a worse prognosis when first discovered. Despite this, the survival rate for all types of lung cancer is alarmingly low at 15%. Smoking is the cause of the majority of lung cancers, and lung cancer is not common for non-smokers, although the risk of lung cancer is increased when a non-smoker inhales secondhand smoke. Air pollution along with certain processed foods have been connected to lung cancer as well, but in lesser numbers. The best way to protect lungs along with not smoking or quitting smoking is to exercise, which has been linked to a lower risk of developing lung cancer. (Molina *et al.* 2008)

Table 1. Staging of non-small cell lung cancer. (“Types and Staging of Lung Cancer.”, *Lungcancer.org*)

Stage	***Severity***	***Subtypes***	***Spread***
0	Not severe	-	Small local carcinoma in situ, not spread
I	Local, not advanced	-	Only lung/lungs
II	Local, but increasing in severity	-	Lung and lymph nodes
III	Local, but advanced	IIIA and IIIB	Lung and chest lymph nodes
IV	Very advanced, possibly not local anymore	-	Both lungs and other parts of the body, metastatic. Survival rate low

Non-small cell lung cancer is classified into five subtypes, different by histology: solid, acinar, papillary, micropapillary, and lepidic. The treatment options for each of these depend on the tumor stage of the cancer, with e.g. solid tumor tissue patterns being associated with a poor prognosis – not to mention the different subtypes of lung cancer can be mixed together as histologically heterogeneous tumors, making diagnosis difficult (Wei *et al.* 2019).

Slide scanners produce whole slide images that can be viewed and analyzed utilizing digital pathology software, much like a modern version of using the microscope to view cells that are too small for the eye to see. This is obviously a remarkable benefit to the field of pathology, since high resolution images of slides of tissue can be viewed outside the laboratory on any computer screen instead of being preserved in glass only available in person, losing its quality over time (Al-Janabi *et al.* 2012). Images of these slides can be fed into for example a neural network to automatize the analyzing process and save time instead of a human viewing each slide individually, and in the case of this study whole slide images of lung cancer were used as data.

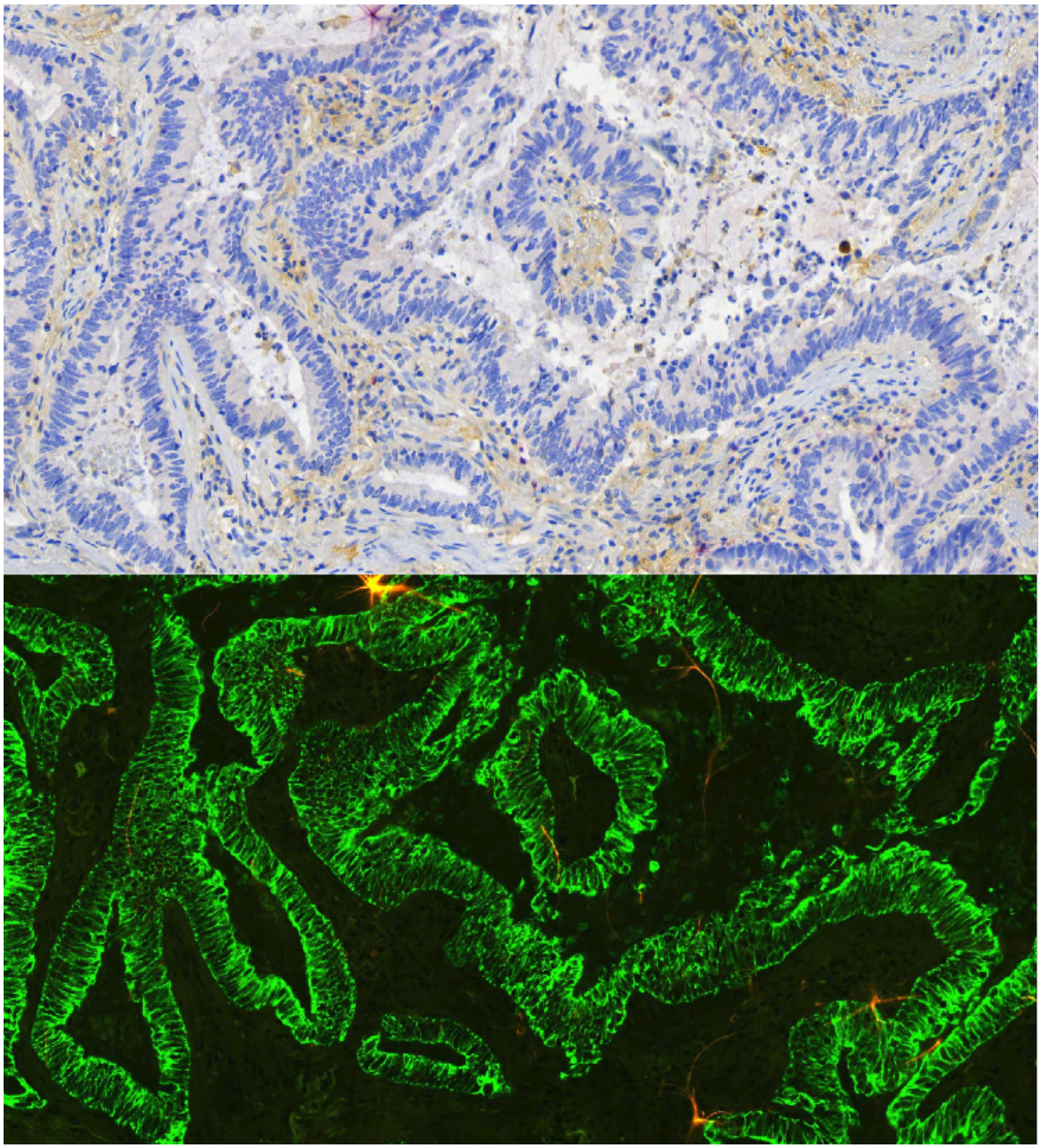

Figure 1. Immunohistochemical fluoro-chromogenic staining on cropped whole slide images, showing clear patterns of carcinoma (bright green areas in fluorescent image).

Immunohistochemical fluoro-chromogenic staining is helpful in discovering true positives when looking for PD-L1 and PD1 activated areas that are not alveolar macrophages. The pipeline constructed in this study aims to identify the regions where true cancerous tissue lies instead of false positives that may be generated by other types of tissue, which will be discussed further. The data used are whole slide images of non-small cell lung cancer stained with IHC fluoro-chromogenic

staining and cytokeratin mask. Input images are viewed under brightfield illumination and target images are viewed under fluorescent illumination.

Machine learning can be supervised or unsupervised – labels or ground truth can be fed to a model during training or the model is not fed ground truth data and finds areas or other information without it. In this study the binary image masks used as ground truth images were generated by thresholding IHC fluoro-chromogenic stained images. The whole slide images used in this study are from Prof. Jorma Isola, Satu Luhtala and Teppo Haapaniemi. PD-L1 images are used as input and target data for the network: with immunohistochemical fluoro-chromogenic staining, under brightfield and fluorescent illumination (**Figure 1.**). The images used as ground truth are processed from IHC fluoro-chromogenic stain images of non-small cell lung carcinoma (NSCLC) under fluorescent illumination. In the IHC fluoro-chromogenic staining for the lung cancer images a technique is used that blocks signaling between the programmed death ligand 1 and programmed cell death protein 1. This cytokeratin mask step used with fluorescent illumination in itself decreases the risk of misinterpreting cancer regions from healthy tissue, especially in uncertain and difficult to interpret cases. (Haapaniemi *et al.* 2017) The programmed death ligand 1, or PD-L1, is a transmembrane protein that under normal circumstances suppresses the immune system when needed. The secretion of PD-L1 is an immunological event by cancer cells in order to evade and distinguish T-cell attacks and the function of CD4 and CD8 cells (D'Arcangelo *et al.* 2019). The appearance of PD-L1 in lung cancer can be a sign of a progressed tumor grade, and it can be used as a biomarker, but a study by D'Arcangelo *et al.* 2019 concluded that it is not a completely reliable prognostic factor in at least early stage non-small cell lung cancer. In any case, it can serve as a useful tool in selecting patients for immunotherapy due to easier tumor grade assessment. An earlier study by Mu *et al.* 2011 had concluded PD-L1 to be linked to overall survival after non-small cell lung cancer surgery.

Digital pathology is a growing field, helping in analysis such as helping in grading cancer by doing mitotic count in whole slide images (Wang *et al*. 2014). Machine learning has become an important part in image analysis: especially deep learning has achieved astonishing results and allowed some networks to rival the accuracy of human experts. A learned network is more generalizable than a manual network and can save time learning by itself how to address other types of data instead of having to manually create a different network for each different kind of sample (Criminisi 2016).

The network used in this thesis study was U-Net, which in biomedical image analysis is a state-of-the-art method of analysing biomedical images. The architecture of U-net, created by Ronneberger *et al.* (2015), can be visualized in the shape of the letter U with images entering and exiting through a symmetric convolutional path. Multiple modified implementations of U-net exist on Github and they perform well on different types of segmentation and image classification tasks especially in medical analysis (Liimatainen *et al.* 2019), so it was chosen as network architecture. Other convolutional neural networks like AlexNet, Inception v3, recurrent networks or VGG19 could also have been used, since they are popular choices in medical image analysis. Some popular networks will be discussed in **Chapter 2.5.**

The underlying purpose of this pipeline is to bring value to the field of digital pathology. There have been numerous developments in the field of digital pathology to lessen the time pathologists spend on viewing real life samples and easing their workload whilst improving the accuracy of diagnosis. A tool to help in diagnosis making could prove to be useful when considering PD-L1/PD1 immunotherapy treatment options for patients.

2. THEORY OF USED METHODS

In this chapter the theory of used methods will be presented and further explained. The used machine learning model is explained along with other state-of-the-art solutions to similar problems and data preprocessing.

2.1. PD-L1 marker and lung cancer

Tumor cells and tumor-associated macrophages can be difficult to distinguish from each other, but immunohistochemical fluoro-chromogenic staining has been developed as a tool for it. The value of IHC fluoro-chromogenic staining lies in immunotherapy: when the interaction between programmed death receptor-1 and programmed death ligand-1 (PD-1 and PD-L1) is suppressed by antibodies, T-cells start attacking tumor cells. However the distinction between actual tumor cells and the surrounding area is important as tumor cells express PD-L1 and tumor-associated macrophages (TAM), along with dead tumor cells, express PD1. This type of immunotherapy medicine is based either on PD-L1 or PD1. Lymphocytes near the tumor can express PD1, but also the unwanted macrophages may express it. This can lead to false positives in searching for PD-L1 expression, because targeted immunotherapy needs to be very precisely applied to the tumor area, not to TAMs or necrotic tumor cells. (Haapaniemi *et al.* 2018) Fluoro-chromogenic labelling has also been used with e.g. breast cancer exhibiting Minichromosome Maintenance Protein 2 (MCM2) expression (Luhtala *et al.* 2018). Immunohistochemical fluoro-chromogenic staining combined with cytokeratin mask is an effective method of making regions of interest clearly visible under fluorescent light. Studies concerning both PD-L1 and deep learning are scarce at the moment.

There have been reproducibility problems with pathologists estimating PD-L1 and PD1 areas, and an automated method would be a benefit to the field. An automated quantitative score system Optra has been validated in a study by Taylor *et al.* 2019 with an AUC score of 0.73 for immune cells and 0.87 for tumor cells. The medicine Pembrolizumab, for example, requires an over 50% tumor proportion score of PD-L1 expression to achieve good performance. Identifying PD-L1 expressing areas challenges pathologists without IHC fluoro-chromogenically stained WSIs to help in confirming the decision areas. (Taylor *et al.* 2019)

2.2. Machine learning in image analysis

Machine learning is taking an increasingly large role in modern society whilst achieving impressive results in medical applications. Especially convolutional neural networks have been on the rise and keep setting the bar higher and higher, even outperforming pathologists at their tasks (Nirschl *et al.* 2018).

It is known that machines generally excel at specific, repetitive tasks whereas humans are better at tasks that require combining information and making complex or intuitive decisions. Machine vision may feel like a daunting task at first glance, but in reality it only consists of many very specific tasks stacked together. Neural networks are based on biological neural networks present in the mammalian brain, and in the same fashion consist of layers of neurons, in which individual neurons are able to transmit a signal onto other neurons in the following layer. In simplicity, a neural network consists of nodes, edges and weights, and layers of nodes are commonly called input, hidden and output layers. A neural network is commonly called deep if the number of hidden layers is high. The network is fed input and target data, and predictions (generally confidence maps) are evaluated against the target data. Bias can also be present in a network.

A good machine learning model is generalizable yet accurate without overfitting. Overfitting means improving in accuracy for one type of data at the cost of being able to classify a different type of data in the validation phase of estimating model performance. Underfitting, on the other hand, stands for poor overall performance of a model in classifying any of the data. Metrics to examine how well a model is doing also have to be chosen carefully because of different types of data used. For example in a binary classification case where the data are images that only have 1% of target class and the rest is 99% background, the basic accuracy metric would be very misleading. The model could predict only background for all images and still yield a 99% accuracy.

Computer vision is an efficient tool for image analysis. The extent of the capabilities of computer vision is ever-increasing: self-driving cars, algorithms able to assist in analyzing medical imagery, facial recognition, augmented reality, handwriting recognition and more (Danuser, 2011). For example virtual or augmented reality can be used to assist surgeons during surgery and training for surgery (Bernhardt *et al*. 2017). Computer vision does not necessarily replace the human eye, but is able to assist in attentive or difficult tasks. There are benefits to using computer vision instead of the human eye; the analysis can be automated and it can access features in the data that would be invisible to a human, for example. Humans, on the other hand, base their analysis of the image on

previously learned interpretation of similar images, and can potentially miss nuances in the data. When using computer vision, bias can be combatted. The ability of a human brain to associate seen signals with previous information and effortlessly find patterns can present a challenge to compute; this is called the association paradigm. However, association can lead to overlooking information that a machine could pick out – this is called the integrator paradigm. (Danuser, 2011) The human eye may find patterns where there are none, and draw false conclusions. A computer, thus, can outperform human vision in delicate tasks such as seeing tiny differences in data and producing reliable information without bias that would change from person to person.

Convolutional neural networks can also be used to output a result image the exact same size as the input – these are called fully convolutional networks. A study by Long *et al.* 2015 used popular convolutional networks to apply full convolutionality with transfer learning and end-to-end training. This sort of approach can be useful in e.g. semantic segmentation, where we want to label areas in the image as named objects like internal organs in CT scan images. Layers in a convolutional network are three-dimensional arrays, consisting of image height, width and the number of colour channels. Operations are performed on these layers, such as max pooling, convolution and activation functions yielding outputs that are fed to following layers. Upsampling, simply put, means backwards convolution or in layman's terms deconvolution. This enables a network such as U-net to learn by "expanding" feature maps and concatenating them.

2.3. Convolutional neural network

Deep learning has been a rising star in the field of machine learning for a while now – it entails neural networks with more layers than in previous systems to allow for more accurate predictions. The emergence of GPUs has also improved the processing times of neural networks, allowing the networks to get deeper and more reliable. For example a convolutional neural network usually consists of combinations of convolution, data reduction and pooling layers. Medical image interpretation by humans can be subjective as it is affected by the variance between interpreters and even their fatigue levels, so automatizing analysis and analysis support systems is beneficial. (Greenspan *et al.* 2016)

Convolutional neural network, ConvNet or CNN stands for a neural network that is composed of layers of perceptrons and is often used for image classification or segmentation tasks, such as for medical imagery (LeCun *et al.* 2015). A great benefit of CNNs is that they don't require as much preprocessing as some other types of machine learning algorithms. Convolutional neurons enable the network to operate on images with less neurons than would be required for e.g. a fully connected feedforward neural network, yet still keeping the depth of the architecture. Parameters of layers are the amount of elements that a filter can learn, and the layers that have parameters in a neural network are convolutional and fully connected. Pooling layers reduce the dimensions of neural networks and dropout layers reduce the amount of activated neurons per iteration, neither having parameters. Filter shape is usually 5x5 or 3x3, dropout is commonly 0.5 and max pooling layers tend to be 2x2.

CNNs have been making a significant foothold in the field of machine learning since the 70's, most importantly in computer vision and image analysis. CNNs are the most successful type of deep learning algorithm for medical image analysis to date. They consist of layers which use convolution filters to perform linear transformations on the input image, and can be applied to 1D, 2D and 3D data. For example 3D CNN applications could be used to analyze voxels in 3D medical scans. Sets of kernels are used to undergo convolution operations on each layer. One application of CNNs is U-net, which is a widely used U-shaped fully convolutional neural network for medical image analysis and performs well on e.g. segmentation tasks (Ronneberger *et al.* 2015). A fully convolutional network requires input images to be the same size and has a receptive field in the last layer. U-net uses upsampling to increase image size and downsampling to return to the input size, outputting an image of approximately the same size. In transfer learning a pre-trained network or pre-trained

weights are used, and it can be used to avoid having to train a network on a large amount of training data. The best results have been achieved when a pre-trained network is tuned to fit the data in question. (Litjens *et al.* 2017) Transfer learning and fine-tuning the networks are modern ways of battling the requirement for large amounts of training data, memory usage and overfitting. Overrepresentation of the normal class in medical imagery has been a challenge, and properly labelled data can be a scarce resource in some cases. (Greenspan *et al.* 2016)

CNNs use different types of activation functions to map the predicted values in between the wanted scale. A popular choice of activation function for convolutional layers is ReLU, which avoids the Vanishing Gradient Problem. ReLU stands for Rectified Linear Unit, and operates on the f(x) = max(0, x) principle to increase speed in training. ReLUs also do not need input normalization in order to not saturate. Neural networks function well but also face issues if trained too much or too little. Underfitting happens when the network does not learn as much as it could and overfitting when it starts to memorize data instead of learning the trends behind it. Overfitting shows as a high training accuracy and significantly lower validation accuracy. A way of reducing overfitting and saving computational memory is doing data augmentation, in which data is modified in such a way that the model cannot tell it is not the same data, for example by flipping images and adding noise to them while keeping the original image as well. This is a good way to increase the size of training data, if the volume of training data is not enough to combat overfitting. Another way of reducing overfitting is applying dropout to the data (**Figure 4.**), in which some neurons are randomly turned off, forcing the model to generalize. This will most likely extend the time a network takes to converge, but the model will be less likely to overfit due to neurons not being able to rely on other neurons each iteration of the network, thus reducing complexity of the solutions of the network in regard to data. An overly complex model will only memorize a specific dataset instead of learning to apply solutions to different types of data. Models with different architectures can also be combined in order to avoid trusting a single model's output too much, but this can be very memory-consuming considering that networks today tend to be large and require plenty of training time. (Krizhevsky *et al.* 2012)

The extent of convolutional neural networks' abilities seems almost limitless. CNNs can even potentially be used to identify skin cancer from cell phone images, increasing awareness for dermatological diseases (Esteva *et al.* 2017). CNNs have been shown to operate with identifying fine-grained objects such as skin lesions thus aiding in diagnosis of skin cancer, and to reach pathologist level accuracy in detecting cancerous lesions. Melanoma classification, melanoma

classification using dermoscopy and carcinoma classification all yielded good results for the constructed AI in a study by Esteva *et al*. The network used was a GoogleNet Inception v3 CNN, pre-trained with 1.28 million images, and trained with the new dataset using transfer learning. This sort of identification tool would be useful for the public because an AI application on a phone could detect lesions that need to be checked by a professional. However, smartphone pictures fed to a classifier can vary in contrast, focus, angles and other factors which makes it crucial for a model to be generalizable. In the study, this was combatted by feeding the classifier copious amounts of training data.

2.4. U-net

The choice of network architecture in this study was U-net. The specialty of ISBI 2012-challenge winning U-net is using skip-connections along with up- and downsampling layers that increase and decrease image size. The up- and downsampling layers are symmetric and enable the user to use less data than other options may require, due to the architecture's ability to understand context and localize features with good accuracy, especially in medical image segmentation tasks. Localization of features is especially relevant for medical image analysis, since many tasks involving medical data require segmentation of regions of interest (ROI).

U-net's up- and downsampling parts are also called convolutional encoding and decoding units. The output of the network is not a generated image, but a pixel-by-pixel confidence map showing the class predictions of each pixel in the original image. This prediction map is then compared to the ground truth, which in the case of this thesis study is a binary image created by thresholding the ground truth images. As U-net trains, the loss function is minimized and usually the best performing timepoint is when the validation loss is lowest. Training can be stopped by using an early stopping callback. Skip connections are created by concatenating layers and they pass information from downsampling layers to their corresponding upsampling layers (**Figure 3.**). U-net does this to combine local and contextual information to improve the precision of classification. The benefit of skip connections lies in downsampling paths losing information during training – this information can be fed to the upsampling path. Skip connections can be either short or long, and have been shown to improve the accuracy of fully convolutional networks as well as helping the model converge faster (Drozdzal *et al.* 2016).

Deformations (warped images) can be an issue in biomedical data, as networks may not generalize well to them. In the original U-net publication this was solved by applying elastic deformations along with regular data augmentation to training images. Another challenge in medical image segmentation is separating regions of interest from less interesting regions. U-net solves this by assigning higher weights to the pixels separating these two areas thus more efficiently telling them apart. The number of feature channels used by layers is doubled with each convolutional layer. Convolutional layers are unpadded, so the output image is smaller in size than the corresponding input – which is why a small batch size and large images are preferred for this network. (Ronneberger *et al.* 2015)

Many machine learning advances have been made for CT images in lung cancer in the field of Computer Aided Diagnostic (CAD). U-net is being used successfully for image segmentation from medical imagery such as Magnetic Resonance Imaging (MRI) when used as fully connected (FCN) or recurrent (RCNN) convolutional neural networks. Semantic segmentation is slow to do by hand and time is of the essence in a medical setting when many patients await for diagnosis, which is why CAD has become a popular choice. With the emergence of very deep neural networks there has been a problem with a vanishing gradient, but this can be fixed by using ReLU as activation function. ReLU's slope plateaus in only one direction, making it possible to avoid or reduce vanishing gradient problem. Vanishing gradient problem for recurrent and feed-forward network models is a sign of the weights not being able to update as training progresses thus stopping performance improvement. (Alom *et al.* 2018)

In many cases it pays off to use pre-trained networks that already have initialized weights from training with other datasets. This way the network already has some idea on how to classify objects and can undergo training for a new specific dataset to learn a new skill. U-net can use e.g. VGG11-initialized weights trained on Imagenet for fine tuning, but can perform well with a small dataset without pre-trained weights. In the case of smaller datasets, however, overfitting can become an issue. (Iglovikov & Shvets 2018)

2.5. Well-known networks

A deep neural network has a large number of filters in a large number of layers. VGGNet by Visual Geometry Group was one of the first deep networks released, different versions consisting of 11, 16 and 19 layers. VGGNet was published in 2014 and had great success in an ImageNet challenge. VGGNet became state-of-the-art when it became public as more layers meant higher accuracy. VGGNets use ReLU, convolutional layers and also fully connected layers with softmax end activation. (Simonyan & Zisserman 2014)

VGG-19 is at the small end of deep networks, however – a 2015 implementation of ResNet contained an astonishing amount of 1202 layers (Huang *et al.* 2016). CNNs represent supervised learning, as labels are provided with the data so the network can learn what it should find. Different uses of deep learning can be object detection, localization, explaining what is happening in a video or picture, recognizing speech and cancer recognition in images. Some networks after a succifient amount of training surpass human accuracy. An excellent feature of deep learning is to be able to utilize transfer learning in teaching the network to recognize something from a new problem domain. This expands the extent of data the network is able to use and opens possibilities of networks that can perform with an intense amount of data.

The first proposed CNN was LeNet-5 by the creator of convolutional networks, Yann LeCun (LeCun *et al.* 1998). LeNet-5 used convolutional, sub-sampling and fully connected layers and instead of the popular choices of today, used a Gaussian connection for the final layer. (Alom *et al.* 2018) After LeNet came AlexNet. AlexNet was released in 2012 to respond to the growing size of available labeled datasets, and has shown success in classification tasks. It was one of the first CNNs running on GPU to win an image classification contest. As the amount of training images is climbing over hundreds of thousands, the used networks need to up their learning capacity in accordance. AlexNet consists of five convolutional and three fully-connected layers and uses Rectified Linear Units (ReLUs) instead of saturating nonlinearities. Dropout is also used as a regularization method, which was a new approach at the time of AlexNet release. (Krizhevsky *et al.* 2012)

New applications of deep networks are being developed all the time and for new purposes. Computing power and possibilities set the limit for deep learning, but they are constantly changing and growing to enable more and more.

2.6. CNNs and deep learning in medical image analysis

Convolutional neural networks have been used in histological tissue analysis successfully to e.g. outline cancerous tissue in whole slide images. In a Bejnordi *et al.* (2017) study based on a coding challenge called the CAMELYON16, the best algorithm represented a better accuracy at detecting cancerous regions than pathologists at 0.994 vs. 0.884. Lymph node metastases were detected from whole-slide images, and it was found however good the result was, the algorithms were trained to only discriminate between normal and cancer tissue and could miss other pathologies, such as infections occurring at the same time in the same tissue. Things that improved accuracy of the classifiers seemed to be standardization and adjusting to class imbalance in the data. In uncertain identification cases pathologists may use extra steps such as cytokeratin masking in the process to confirm the result as accurate, and these uncertain cases can happen with especially hematoxylin- and eosin-stained samples.

In a study by Valkonen *et al.* (2017) good performance (0.97 block-wise AUC) was achieved by using a dual convolutional neural network (dCNN), which consists of two CNN's tied together by fully connected layers. In order to extract multi-scale features the constructed network was designed in a similar way to a "virtual pathologist" - zooming in to interesting regions after viewing the whole picture in general. This kind of approach is a fitting continuity of mammalian brain-mimicking neural networks, only the mimicked mammal is a pathologist and their workflow. Despite the good results these kinds of classifiers can achieve, this study also recommended it be used as a decision support system instead of accepting the results without evaluation of the imagery with human eye. Even when using a neural network as a decision support system instead of relying on it as a diagnostic tool it still reduces the workload of pathologists, which is also the aim of this study. Another study by Valkonen *et al.* (2018) used a pre-trained VGG-16 network to automate epithelial cell detection. The work was conducted using breast cancer cell slides stained with sequential hematoxylin-IHC and fluoro-chromogenic cytokeratin-Ki67 double staining, and 52 images were reclassified to high proliferation using this network after pathologists originally classified them as low proliferation. This type of convolutional neural network can be a valuable tool in diagnosis, drawing pathologists' attention to uncertain WSIs for a second assessment. The human eye can and does err, and it is crucial especially in the case of cancer to achieve as reliable results as possible – and neural networks can help with exactly that.

On top of histological images CT scans, MRI images, PET images and many others can be used with neural networks. Machine learning applications can be used in e.g. computer tomography, mammography, X-ray or any medical imaging technique. CNNs are able to analyze 2D and 3D image data by design instead of only reading in vectorized data like some other machine learning models. (Shen *et al.* 2017)

A study by Hou *et al.* 2016 found that patch-based convolutional neural networks can outperform CNNs that are based on images, and were the first to combine patch-level CNNs with supervised decision fusion. Whole slide images are too large to feed to a network as they are, so the images need to be divided into patches. An approach called MIL (Multiple Instance Learning) has also been utilized with whole slide images and CNNs, in which unlabeled patches are used in order to predict labels.

Deep neural networks have been shown to reach pathologist-level classification success with multiple tissue type WSIs, including lung adenocarcinoma which is a type of non-small cell lung cancer. A convolutional network was used in a Wei *et al.* 2019 study in order to find neoplastic cell regions and possibly mixed heterogeneous areas of cancer with a ResNet Patch Classifier and sliding window approach. Its Cohen kappa scores rivaled those of pathologists', but were not accurate enough to be used as a diagnostic tool, especially since these types of tasks can be difficult for humans as well. The benefits of using a neural network model on uncertain data (for example WSIs possibly containing different cancer subtypes) are substantial when thinking of diagnosis from a patient's view – any help in uncertain diagnosis making is an advantage. In the Wei *et al.* study data augmentation and modification was used to improve the generalizability of the model. Data augmentation can include adjusting brightness, contrast, axis or colour of the image and thus create more data to train the network on.

Different convolutional network architectures (AlexNet, GoogleNet, VGGNet-16 and ResNet) have been tested in finding associations between tumor cell morphology and subtypes, with all achieving over 90% Area Under Curve (AUC) scores, and all models perform better than previous methods that were feature-based (Yu *et al.* 2019). The overall conclusion was that simpler machine learning models perform well on more general tasks such as identifying tumors and more complex models are suitable for distinguishing between for example multiple tumor types.

CNN has been used with soft-voting in order to evaluate lung adenocarcinoma tumor growth patterns in whole slide images, with results as high as 89% accuracy. Soft-voting stands for

choosing the class with the highest percentage of votes given by the classifier. Lung adenocarcinoma comes in 6 different types, which have different patterns of growth and can differ in aggressivity, or different types of tumours can be mixed together. This can impact the time cancer treatment takes or even different methods that should be utilized to treat the cancer. CNNs have become a convenient state-of-the-art tool in digital pathology aiding pathologists in their workflow and decision-making, which can be a difficult task for the human eye, especially with heterogeneous tumors. (Gertych *et al.* 2019)

3. MATERIALS AND METHODS

3.1. The dataset

The dataset used to train the model consists of whole slide images from Jorma Isola, Teppo Haapaniemi and Satu Teppo taken from 34 patient samples. The slides of non-small cell lung carcinoma were stained using immunohistochemical fluoro-chromogenic and cytokeratin techniques and viewed under brightfield and fluorescent illumination, and corresponding WSIs were taken. Images in the final dataset were cropped from WSIs so that all cropped images contained PD-L1 positive regions. The fluoro-chromogenic images serve as a template for ground truth and the immunohistochemical images are the images from which the classifier needs to find regions of interest. The used staining technique blocks the signaling between PD-L1 and PD-L1 protein, resulting in PD-L1 positive regions becoming clearer to see with the eye.

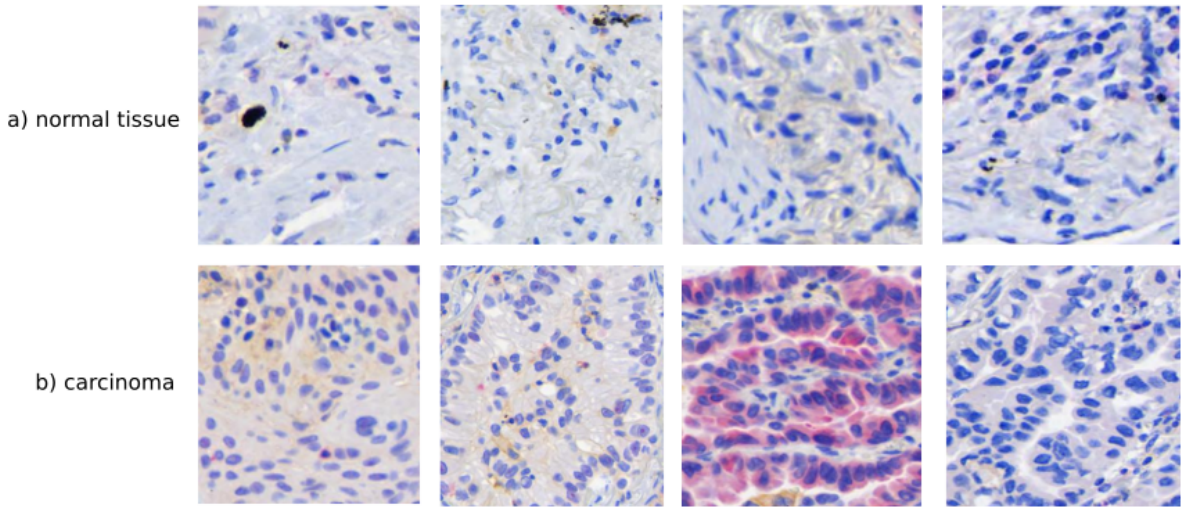

Figure 2. Comparison of regular tissue and carcinoma regions. The intensity of the PD-L1 stain differs between some slides.

This type of data can present a challenge to a neural network because the images are very similar. All of them contain cells, similar colours and nuclei, and it can even be difficult for pathologists, not to mention other humans, to tell apart tumorous areas. The strength and colour of the dye changed between samples, presenting another challenge. In addition, a part of the images contained orange-coloured stain representing PD1 marker. These were removed to the extent that was possible by only keeping the green (fluorescent) channel of the images that binary mask images were produced from.

The dataset was divided into three individual parts: training, validation and test datasets. The training set is used to train the model and the validation set is used to monitor under- or overfitting of the model during training in order to stop the process when validation loss stops decreasing. The test set is completely separate and used to test the performance of the model after it is done training. There was no overlap in the three different datasets due to them being manually selected to be representative and separate, and tiling with stride being done after dividing the dataset into three separate folders. This is to ensure there is no positive bias observed and that the model does not see the validation or testing images during training and thus cannot memorize them.

3.2. Preprocessing and tiling WSIs

The first step in creating an image preprocessing pipeline was cropping the whole slide images to equal sizes of 2048x1024 in order to achieve identical dimensions in each image. Python Image Processing (PIL) (Clark 2015) package was used to load images and the immunohistochemical fluoro-chromogenic WSIs were separated based on illumination into separate locations to represent the input and target images. After the initial preprocessing the equal-sized images were fed into a pipeline that created the ground truth binary masks from fluorescent images by adjusting brightness, changing the image into greyscale, blurring the image with a Gaussian blur filter, and extracting a threshold value with Otsu thresholding from the greyscale image (Otsu 1979). This threshold value was individual for each image and was used to create a binary mask from the greyscale image. Small objects were removed from binary masks to satisfy the requirements of pathologist-level outlining of tumor regions, and the resulting images were tiled to create the target dataset. Corresponding immunohistologically dyed images were also tiled but not processed otherwise to create a set of input images. During tiling, a custom padding function was used to pad the image edges to produce equal sized tiles, and a stride of image width / 2 was used to create more training

data and a better understanding of the tumor regions instead of separating tumors. The tiling followed a simple 2-fold rule: 128, 256, 512 etc.

3.3. Model implementation

A good practice to begin implementing a model is to get it to overfit first, then try to generalize it. If the model starts at underfitting, it is more difficult to find the problem behind a poor learning rate.

This model was run on CPU and GPU to reduce the computational load. A generator was also used to feed data directly from disk to the network instead of saving all images as NumPy arrays to memory. The U-net used was from Github (Zhixuhao 2019), with a few things modified: learning rate, optimizer and some metrics for training history were changed. The architecture of the model can be seen in **Figure 3**.

Training data was shuffled before feeding into a data generator to ensure the batches consisted of different types of images from different parts of the dataset. Without data shuffling the performance of a classifier may suffer as it may learn that the images always come in a certain order. ”Early stopping”- and ”checkpoint”-callbacks were used with the model, which ensured that the best weights corresponding with the best model presenting the lowest validation loss was saved. This was especially important because the model started overfitting quickly after reaching peak performance.

The U-net used has 24 convolutional layers, 4 max pooling layers, two dropout layers and is designed in a U-shape consisting of up- and downsampling layers. No pre-trained weights were used, and a finished model with weights was saved for evaluation with an independent test set. The amount of dropout applied was 0.5 for each dropout layer, and the amount of neurons followed a two-fold rule: 64, 128, 256, 512, 1024 and reversed for the downsampling, eventually decreasing to 1 neuron due to the binary classification nature of the data. The size of input images was 256x256, and validation loss, accuracy and IoU metrics were examined for each iteration. The similarity and diversity measure IoU (Intersection over Union) is also called the Jaccard index, explained in **Chapter 3.4.** along with other validation methods. The amount of total and trainable parameters was 31,032,837, without any non-trainable parameters.

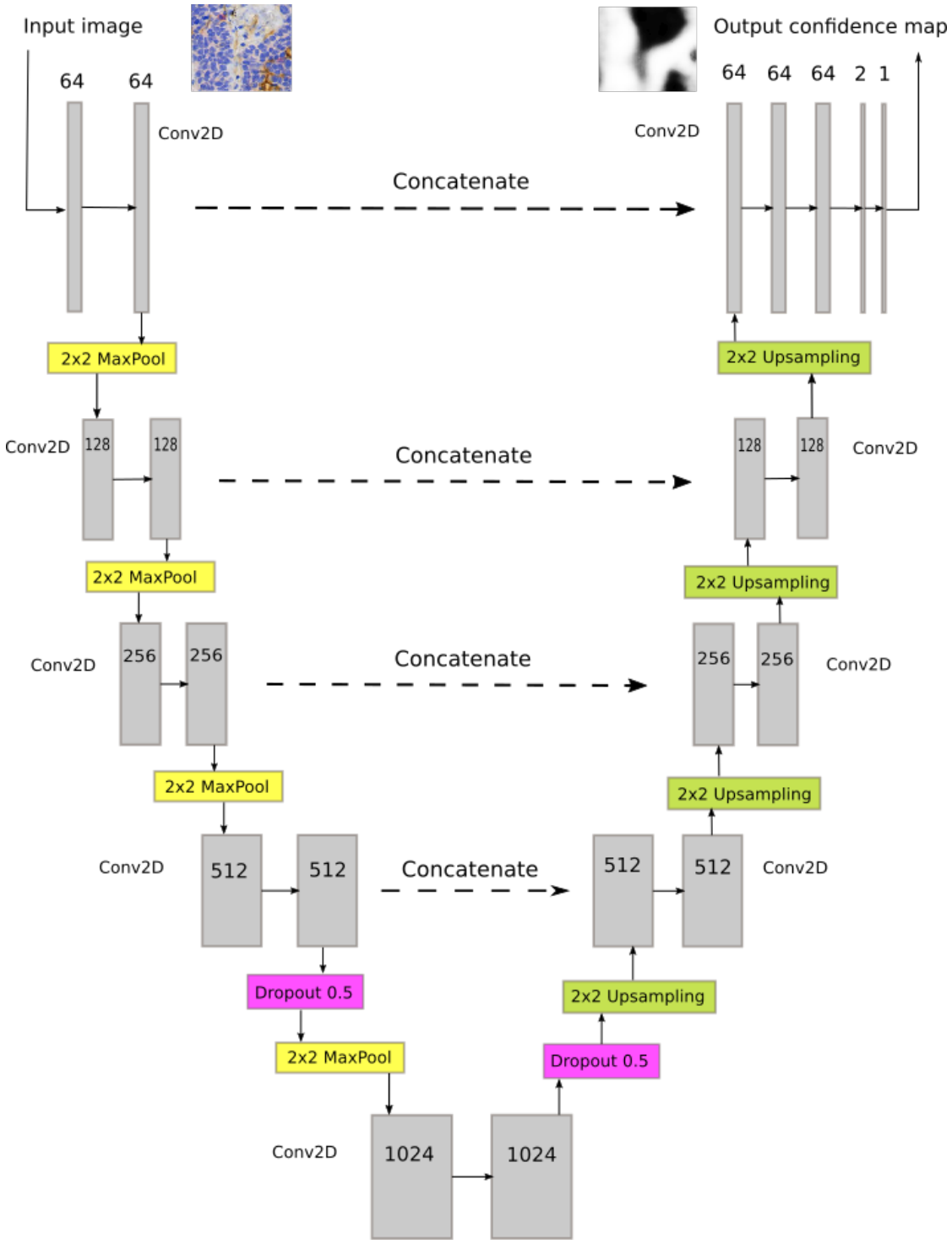

Figure 3. Architecture of U-net used in this thesis study. Concatenation arrows stand for skip connections, which transfer information from downsampling layers to upsampling layers. Conv2D stands for 2D convolutional layers.

A data generator function was created in order to feed data to the model directly from the disk instead of saving images as NumPy arrays into a variable requiring large amounts of memory. This significantly speeds up the process. The pipeline was created with Python (version 3.5.4). Packages NumPy 1.15.2 (Oliphant 2006), Python Image Processing 5.0.0 (Clark 2015), Matplotlib 2.1.1 (Hunter 2007), Keras 2.2.4 (Chollet 2015), Scipy 1.0.0 (Jones *et al.* 2001), Scikit-image 0.14.2 (van der Walt *et al*. 2014), os (Miscellaneous operating system interfaces, posix), re (Regular expression operations, 2.2.1) and functions from Bioimage Informatics-group's Github were used in the implementation of tiling, preprocessing, creating a generator function and the actual model. The backend used with Keras was Tensorflow.

Dropout was used to regularize the model. Dropout simply means that with each iteration of the data a part of the neurons are dropped out of training, not participating in the classification tasks (**Figure 4.**). With adequately shuffled training data, this will happen again with each epoch and lead to better generalization of the model. This way the neurons cannot solely memorize data instead of learning. In this pipeline methods like dropout and early stopping are taken advantage of to avoid overfitting. Early stopping means monitoring the validation loss after each epoch and stopping training when validation loss stops decreasing. Interestingly dropout is, in a way, modeled after living things like neural networks are – the idea is based on sexual reproduction, in which the combinations of genes that are submitted to offspring is random (Srivastava *et al.* 2014). In this case the activated and non-activated combinations are of neurons. The ultimate goal of a convolutional neural network is to be robust and generalizable with a good predictive accuracy.

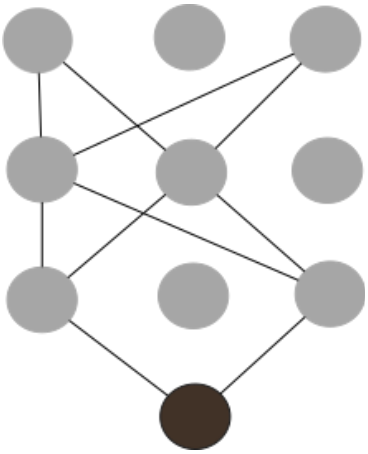

Figure 4. A single run of a network with dropout applied, withholding a random selection of 1/3 of neurons from taking part in the training and outputting a single class prediction at the end. The direction of the network is downwards. The withheld neurons may be used in the following iterations of the network.

Binary cross-entropy (BCE) **(1)** was used as the loss function because the image segmentation task was binary with two classes to classify pixels to: regular tissue and PD-L1 positive tissue, also called background (numerically 0) and target (numerically 1), so binary cross-entropy was a natural choice for a binary classification problem such as this, as defined by Drozdzal *et al.* (2016) below:

$$L_{BCE} = \sum_i y_i \cdot \log(o_i) + (1 - y_i) \cdot \log(1 - o_i) \quad 1)$$

Sigmoid activation function **(2)** was used for the final layer activation and ReLU otherwise for each convolutional layer. Sigmoid activation function is a commonly used two-class activation function that outputs results between the range of 0 and 1, useful for predicting probabilities such as the probability for a pixel to be classified into 0 or 1. In this study the last layer of U-net has a sigmoid activation function instead of ReLU in all other layers. The basic sigmoidal function from Han & Moraga (1995) is presented below:

$$f(h) \frac{1}{1 + \exp(-2\beta h)} \quad 2)$$

Adam, or Adaptive Moment Estimation, is a robust adaptive optimization method especially efficient with training neural networks and introduced by Kingma & Ba (2017), but it has been found by Keskar & Socher (2017) that in some cases adaptive optimizers do not generalize as well as the stochastic gradient descent method. Stochastic gradient method originates from stochastic approximation (Robbins & Monro 1951). The difference between stochastic gradient descent **(3)** and adaptive optimizer functions is that SGD does not limit how it scales the gradient, which has been fixed for adaptive functions such as Adam. Adaptive functions also correct bias. SGD uses a scalar learning rate, and adaptive functions use a vector of multiple learning rates which evolve and change as the training of the model goes on, creating one learning rate value for each parameter.

$$w := w - \eta \nabla Q_i(w) \quad \text{for i = iteration} \quad 3)$$

However in this study optimizers Adam **(4)** and Adadelta **(5)** were used for comparison instead of stochastic gradient descent due to their known good results with this type of data and network.

$$w_k = w_{k-1} - \alpha_{k-1} \cdot \frac{\sqrt{1 - \beta_2^k}}{1 - \beta_1^k} \cdot \frac{m_{k-1}}{\sqrt{v_{k-1}}} + \epsilon \quad 4)$$

Adadelta (Zeiler, 2012) is a robust optimizer derived from ADA-GRAD designed to require little concern to adjusting the learning rate due to smart adapting. The learning rate used by Adadelta is

dynamic, because it uses solely first-order information and is computed on a per-dimension basis. Learning rate decay is used to avoid getting stuck in local minima, which can happen if learning rate remains too high throughout training.

$$\Delta x_t = -\frac{RMS[\Delta x]_{t-1}}{RMS[g]_t} \cdot g_t \quad 5)$$

In this study Adam was used with learning rates of 0.0001 and 0.00001. Adadelta was used with a standard learning rate of 1.0. Adam optimizer and using dropout in order to prevent overfitting has been shown to be a good combination and produce good convergence by Kingma & Ba (2017). Two layers of dropout with the value of 0.5 were used in the pipeline.

There exist different types of rectified activation functions introduced by Hahnloser *et al.* 2000, such as standard rectified linear unit (ReLU) and leaky rectified linear unit (Leaky ReLU). The difference between ReLU **(6)** and leaky ReLU is that the latter does not drop the negative part, but allows a tiny gradient for it. ReLU, on the other hand, remains linear for positive values but becomes zero for negative values, which makes it a good choice for many machine learning problems. ReLU functions as a linear function that prunes the negative part of a piece to zero and keeps the positive part of the piece. This can also help the model to converge faster, which means decreasing training loss to an acceptable level while training. However by not accepting any negative values regular ReLU can create dead neurons, which are stuck outputting zero and are essentially useless. Other types of ReLUs exist as well, such as RReLU and PReLU. The benefits of using a non-saturated activation function such as the ReLU are to help the model converge faster and avoid the vanishing gradient issue. (Xu *et al.* 2015)

An important part of ReLU is being sparsely activated, much like actual neurons in a mammalian brain. Not all neurons fire at the same time, and there is a benefit to this not happening inside a machine learning model, either. For example overfitting and noise can be reduced by doing this.

$$f(x) = \begin{cases} x \, if \, x > 0 \\ 0 \, otherwise \end{cases} \quad 6)$$

Training set batch size can have an effect on network performance. Greater batch size can lead to much more accurate results, because the network learns differences between training samples better when there are more examples available each iteration (Radiuk 2017). It has been debated that the optimal amount of samples per batch lies between 64 and 512, but such large amounts of data per

iteration aren't possible for computationally heavy networks such as U-net. The batch size used with this study was 20. The network model looped through each pixel of the image tiles, classifying it a 0 or 1, with 0 being regular tissue and 1 being carcinoma tissue (or other PD-L1 activated tissue).

3.4. Validation methods

In this chapter the equations, metrics and other validation methods that were used are explained. Metrics like IoU, AUC and others should be used along with accuracy to give correct information on how well a segmentation has achieved its goals without giving in to bias present in the data due to class imbalance (Valkonen *et al.* 2018). In many machine learning problems the data is assumed to be i.i.d., identically and independently distributed (Sokolova *et al*. 2006). Metrics used to assess the performance of models were IoU, F-score, AUC-score, ROC curve and accuracy. The following equations for all but Jaccard index are from the Sokolova *et al.* 2006 publication.

Intersection Over Union (IoU) **(7),** also called the Jaccard index (Jaccard 1901), measures the amount of overlap between the predicted image and the ground truth image. In order to calculate it, a threshold value has to be chosen and the predicted images need to be binarized. In the case of this study the chosen threshold was 0.5.

$$IoU = \frac{Target \cap Prediction}{Target \cup Prediction} \quad 7)$$

Specificity is calculated by dividing the amount of true negatives with the combined amount of true positives and false negatives **(8)**. False Positive Rate or FPR is calculated by subtracting specificity from 1. AUC scores **(12)** were calculated and ROC curves **(13)** plotted based on False Positive Rate **(8)** and True Positive Rate **(9)**.

$$Specificity = \frac{True\ negatives}{True\ negatives + False\ positives} \quad 8)$$

$$FPR = 1 - Specificity$$

Sensitivity is calculated by dividing the number of true positives with the combined number of true positives and false negatives **(9)**. Sensitivity is also called the True Positive Rate or TPR.

$$Sensitivity = \frac{True\ positives}{True\ positives + False\ negatives} \quad 9)$$

F-score **(10)** is a commonly used binary classification evaluation metric for a more reliable accuracy metric, calculated as follows. Precision stands for the amount of true positives divided by the sum of true positives and false positives: all samples labelled positive that are actually positive.

$$F_1 = 2 \cdot \frac{precision \cdot sensitivity}{precision + sensitivity} \quad 10)$$

Basic accuracy **(11)** calculates the amount of true negatives and true positives over the whole set, but doesn't give information on the number of correct labels. This is the reason in some medical image classification problems accuracy should not be given too much importance in the final evaluation as an evaluator of performance.

$$accuracy = \frac{true\ positives + true\ negatives}{true\ positives + false\ positives + false\ negatives + true\ negatives} \quad 11)$$

Area Under Curve or AUC score **(12)** can be thought of as balanced accuracy and can be calculated as follows:

$$AUC = \frac{sensitivity + specificity}{2} \quad 12)$$

ROC curve is plotted based on the following equation:

$$ROC = \frac{P(x|positive)}{P(x|negative)} \quad 13)$$

4. RESULTS

The analysis phase consisted of two parts: choosing a good optimizer and cross-validating the network with the best chosen optimizer. The number of samples used was 34, out of which 3 samples were reserved for validation and 3 for testing. Both CPU and GPU were used. The results show a good average area under curve score (AUC=0.96) after phase two of the analysis in classifying IHC fluoro-chromogenic stained non-small cell carcinoma images (**Figure 5.**).

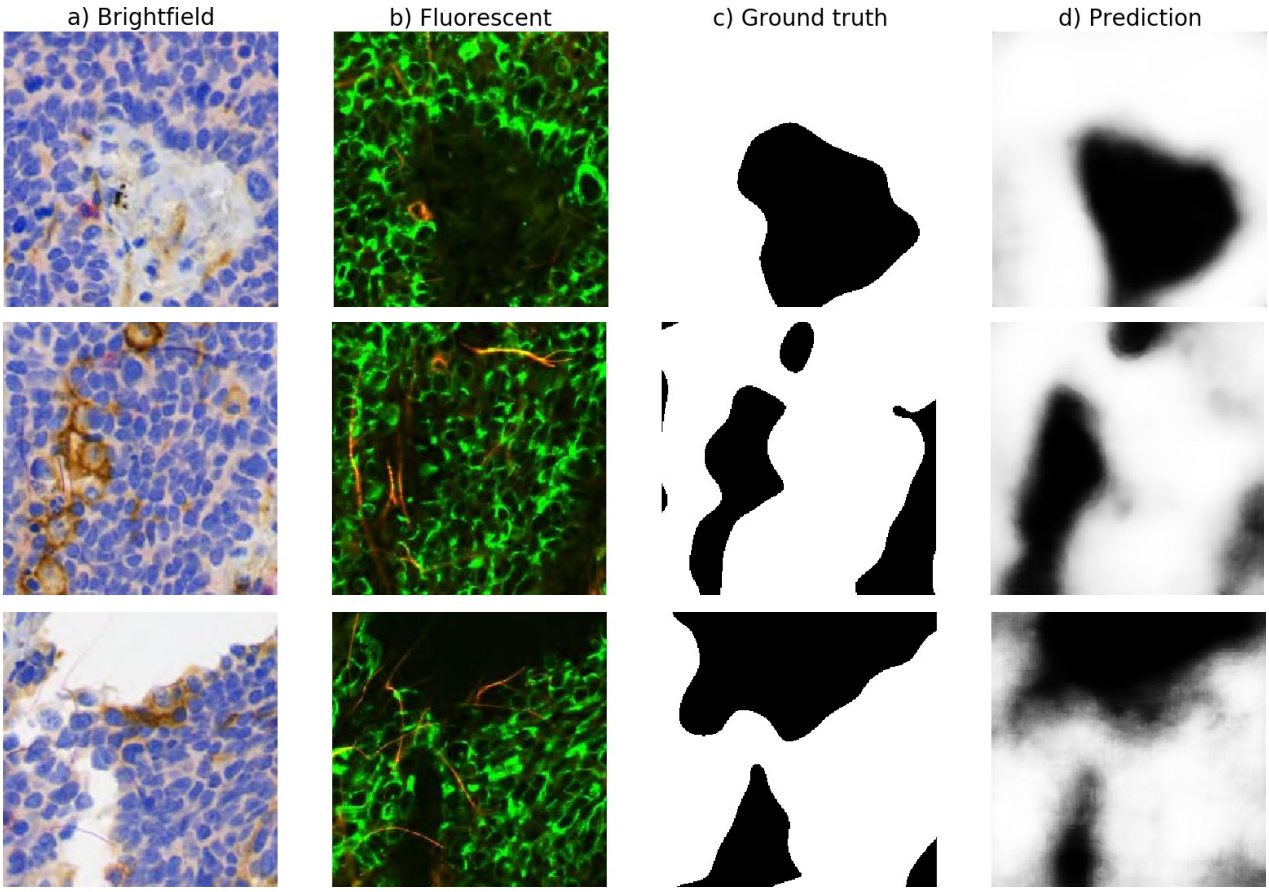

Figure 5. a) Blocks with immunohistochemical fluoro-chromogenic dye, under brightfield. b) Blocks with immunohistochemical fluoro-chromogenic dye, under fluorescent illumination. c) Ground truth produced from the fluorescent images. d) Prediction generated by model. First row represents model with Adam optimizer's learning rate set to 0.00001, second row Adam with learning rate of 0.0001, and third row model with Adadelta optimizer.

It is evident the performance of this model does not reach perfection (**Figure 6.**): for some blocks the binary mask automatically generated from fluorescent images is not an ideal fit to be used as ground truth, and for some blocks the PD-L1 positive regions don't have enough colour to be visible in brightfield illumination, making classification difficult.

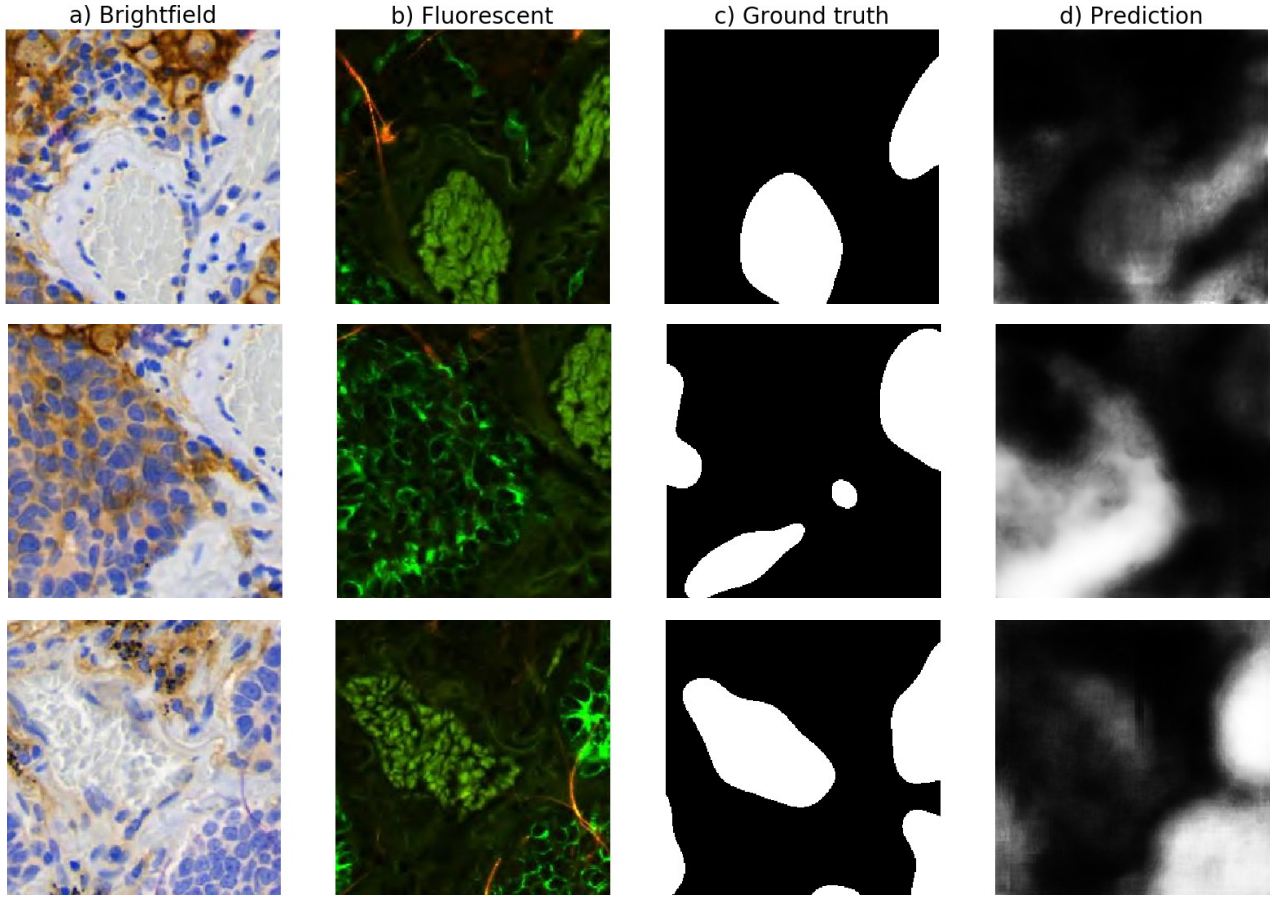

Figure 6. Examples of image blocks that the classifiers had trouble with. First row represents model with Adam optimizer's learning rate set to 0.00001, second row Adam with learning rate of 0.0001, and third row model with Adadelta optimizer. In these cases the binary mask is either not accurate enough or the staining isn't revealing ROI under brightfield, only under fluorescent illumination.

The results show that U-net performs well at recognizing cancerous areas in non-small cell lung carcinoma images: an AUC score of up to 0.934 was achieved during part one of testing (**Figure 7.**) for 2688 blocks of test data. The best chosen model achieves an AUC score of 0.960 after manual k-fold cross validation (**Table 3.**). Accuracy and loss were monitored through training and it was

found that the model reaches a high accuracy early in the training, after less than ten epochs in all tested cases (**Figures 8, 9, 10**). After the best performance is achieved the model quickly starts to overfit, so different dropout values were tested. The possible reasons of overfitting are discussed in **Chapter 5**. There were no significant changes to model performance from different parameters or batch size, but batch size 20 was found to yield the best results so it was chosen as batch size for all runs of the model. Epochs were set to 100 with early stopping and patience limit of 10-20 epochs, the loss used was binary cross-entropy and the last convolutional layer used sigmoid activation function to create prediction results in between 0 and 1. Patience limit means the amount of epochs the program waits for validation loss to decrease before using early stopping to save the best model and end the training. Target images were binary and input images were normalized in between -1 and 1. Adadelta and Adam optimizers with different learning rates were tested: Adadelta with 1.0 and Adam with 0.0001 and 0.00001 learning rate. Decay was set to 0.0 in all models.

A 2688-block validation data set for choosing the optimizer was hand picked to be as heterogeneous as possible. There was no overlap *between* training, validation and test set blocks. There was overlap in the blocks *inside* each dataset due to stride during tiling phase. There is no significant amount of class imbalance in the data, which is why in the case of this dataset and model the ROC-curve (**Figure 7.**) is a better measure of performance than precision-recall curve. All models performed well and on par with each other, with no significant differences. Class imbalance can be a challenge when using medical imagery as data, since the majority of the images tend to be normal and only a small part contain a tumor or lesion, leading to class imbalance problems.

The first step was to choose the best optimizer and learning rate, which is shown in the following figures **7, 8, 9** and **10**. After choosing the most promising optimizer manual k-fold cross-validation was repeated 5 times with independent test sets of 3500-4000 blocks.

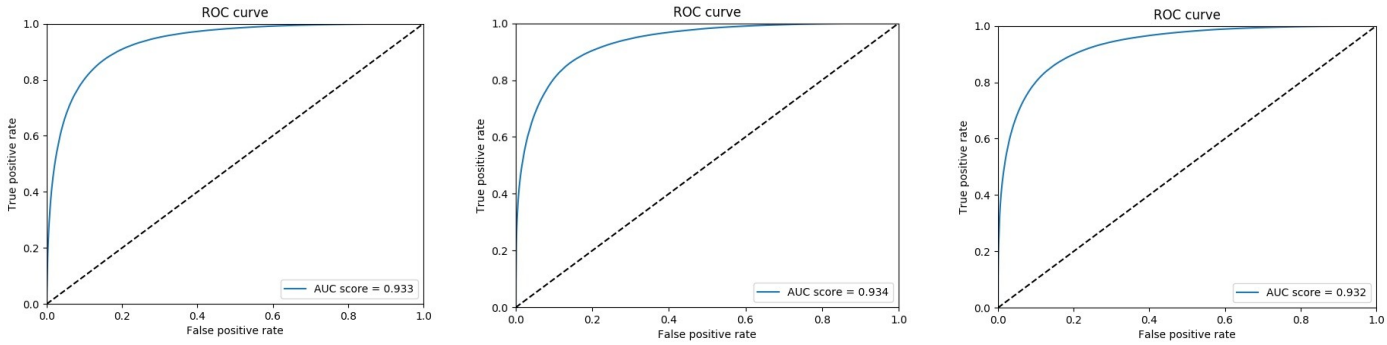

Figure 7. ROC curves and AUC scores for all models with different optimizers in the first phase. From left to right: Adam(lr=1e-5) with AUC=0.933, Adadelta(lr=1.0) with the best performance of AUC=0.934 and Adam(1e-4) with AUC=0.932.

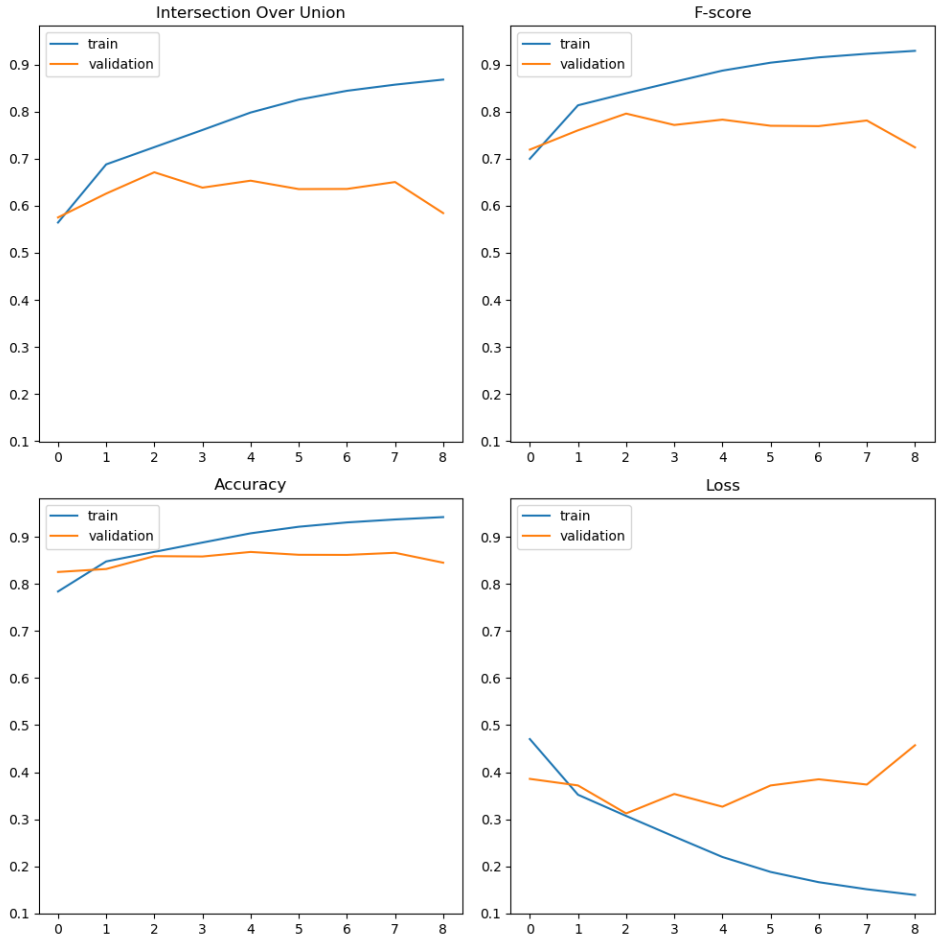

Figure 8. Monitored metrics for Adam optimizer with a 0.0001 learning rate, during training. This is the default learning rate for U-net. The model with the lowest validation loss (0.33) was chosen to be the best one for predictions. X-axis represents the number of epochs and y-axis the used metric scale. The best model has a loss of ~0.331 and accuracy of ~0.858. The axis scales are different for each model due to early stopping.

It can be seen that the version of the model with Adam optimizer (learning rate = 0.0001) reaches its peak fast and already presents good results after one epoch – this could be related to the size of the dataset being fed in its entirety to the network at once. Other possible reasons are that the data is quite similar, all images being tissue images with similar dye. This version of the model showed the smoothest learning curve, the others oscillating more. (**Figure 8.**)

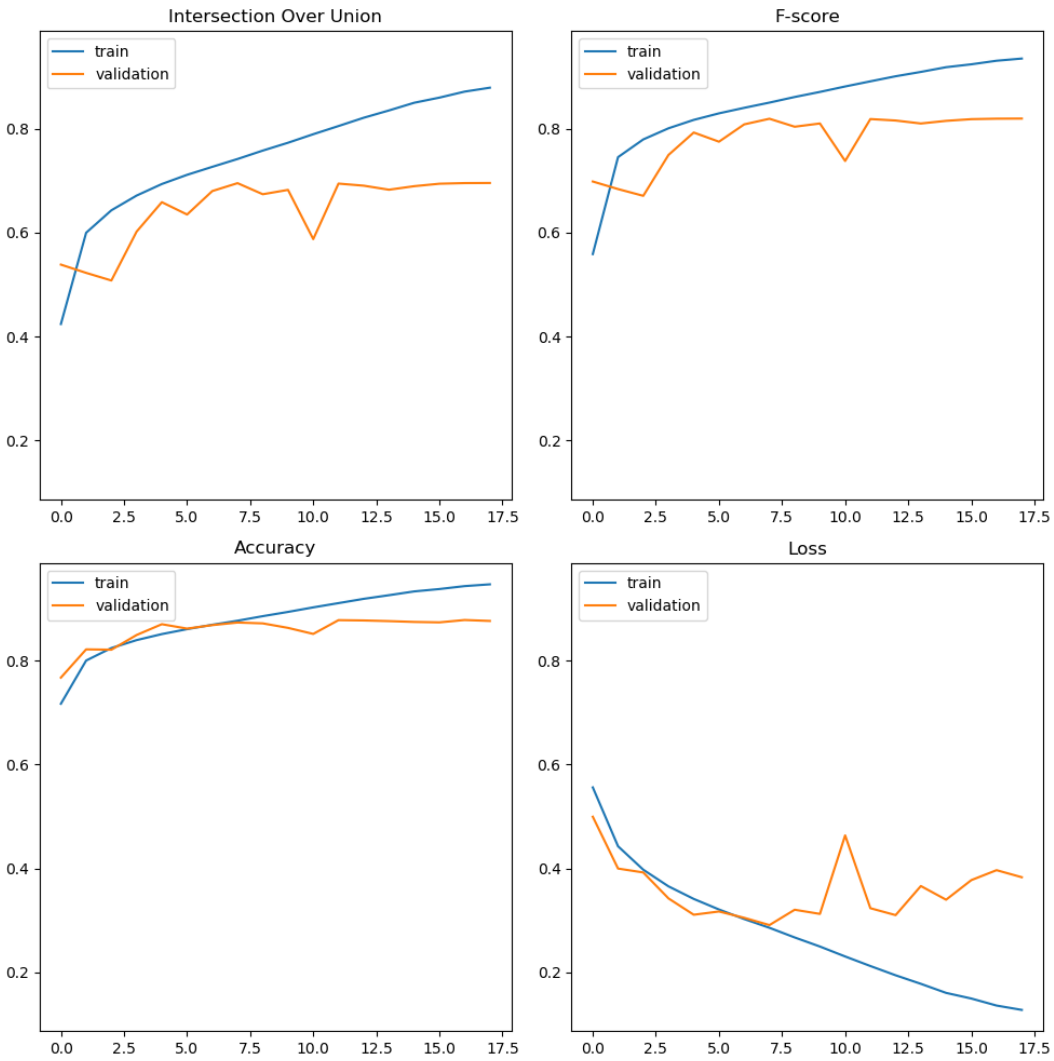

Figure 9. Monitored training metrics for Adadelta optimizer with a learning rate of 1.0. Default values were used: learning rate of 1.0, rho of 0.95, no epsilon and decay of 0.0. The best weights had a loss of ~0.321 and an accuracy of ~0.861.

Adadelta optimizer with default values started showing overfitting patterns after epoch 7.5 (**Figure 9.**): validation loss started increasing and validation accuracy started decreasing. Early stopping was

used to save the best weights at the peak of the model. Adadelta reached a loss of 0.321 and an accuracy of 0.861.

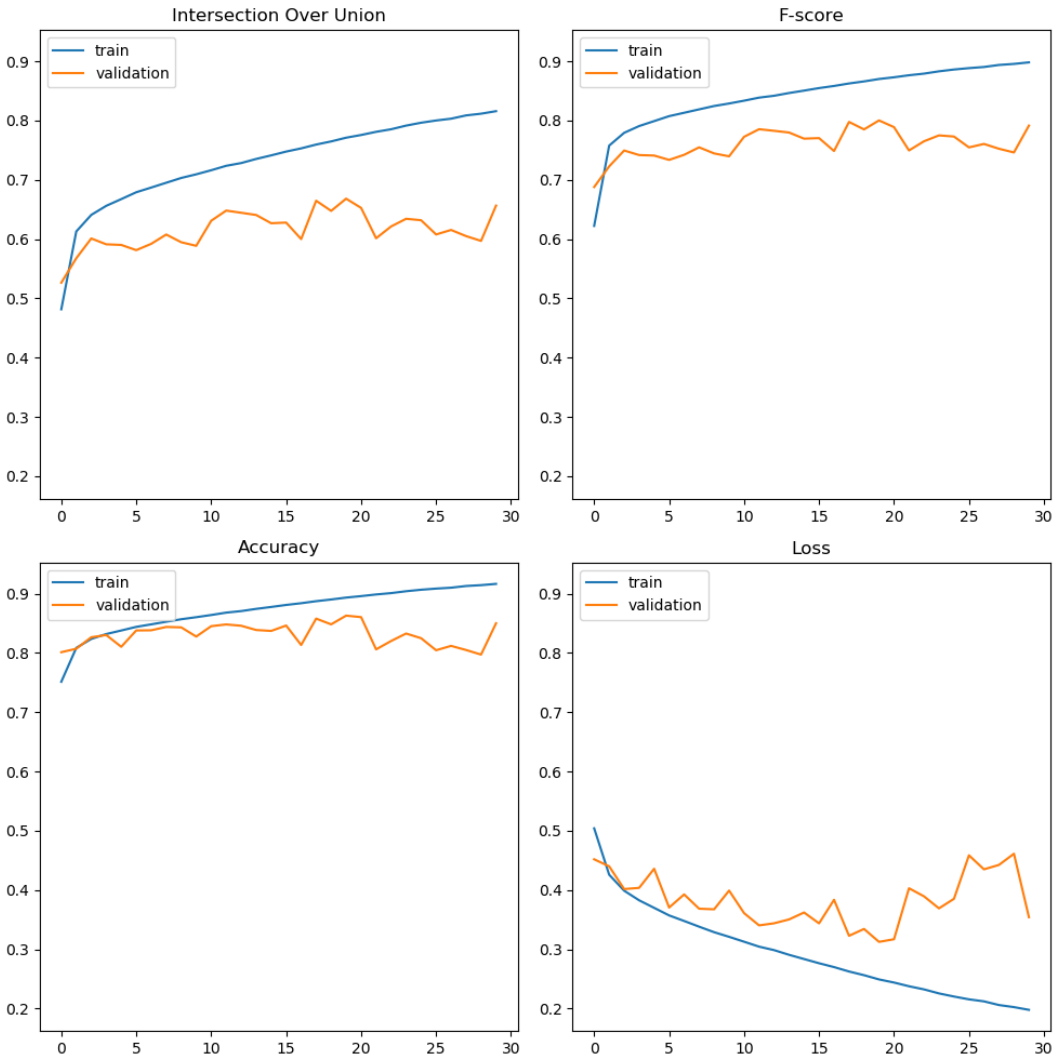

Figure 10. Monitored training metrics for Adam optimizer with a learning rate of 0.00001. The validation set accuracy has more fluctuation, but better accuracy and loss scores for the best weights. The best weights had a loss of ~0.329 and an accuracy of ~0.862.

Adam optimizer with a learning rate of 0.00001 (**Figure 10.**) showed oscillation in the accuracy and loss curves of the validation set. However the best accuracy 0.862 was achieved with this optimizer and learning rate, so it was chosen for cross-validation and final assessment. It is generally a good idea to reduce learning rate when overfitting happens.

Table 2. Metrics calculated for the same test set for the same model, only varying optimizers.

Metric	*Adadelta*	*Adam(lr=1e-4)*	*Adam(lr=1e-5)*
Accuracy	0.86	0.86	0.86
Loss	0.32	0.33	0.33
AUC	0.93	0.92	0.93
IOU for 0.5 threshold	0.7	0.7	0.7
Precision-recall score	0.91	0.91	0.9
F-score, weighted mean	0.86	0.86	0.86

ROC-curve was plotted for each run of the network and the AUC score each version yields is a good result of over 0.92 for each model (**Figure 7.**). Accuracy for each version of the model was ~85-86%. Accuracy was only used to monitor model training due to it being a less representative metric for this type of image segmentation than for example AUC score and IOU score (intersection over union) or otherwise known as Jaccard index. After testing different optimizers the best performing one was chosen and used for final cross-validation of the network. The best performing model was Adam with a learning rate of 0.00001, which was chosen for further validation.

The final dataset used had 28 samples for the training set, 3 samples for the testing set and 3 samples for the validation set. Some samples had differing amounts of images, so the results may have a small bias. The amount of 256x256 training blocks for each run of the network was approximately 31 500 and test and validation sets 3500-4000 blocks each.

The average AUC score for the best chosen model after manual k-fold cross-validation was **0.96± 0.02**, average precision-recall score **0.94±0.02** and average IOU score was **0.74±0.06.** The average F-score was **0.87±0.02**. (**Table 3.**)

Table 3. Results for cross-validation test sets for Adam 0.00001, rounded to .2 digits. The size of validation and test sets was 3456 blocks each. Std stands for standard deviation.

Metric	*Run 1*	*Run 2*	*Run 3*	*Run 4*	*Run 5*	*Final std*	*Final average*
Accuracy	0.87	0.86	0.89	0.91	0.85	**0.02**	**0.88**
Loss	0.29	0.34	0.25	0.22	0.32	**0.04**	**0.28**
AUC	0.96	0.93	0.97	0.97	0.95	**0.02**	**0.96**
IOU for 0.5 threshold	0.76	0.65	0.77	0.82	0.71	**0.06**	**0.74**
Precision-recall score	0.95	0.9	0.96	0.97	0.94	**0.02**	**0.94**
F-score, weighted mean	0.87	0.85	0.89	0.91	0.85	**0.02**	**0.87**

The U-net model used for cross-validation had the following parameters: Adam optimizer with 0.00001 learning rate, binary cross-entropy, two dropout layers of 0.5, ReLU activations for convolutional layers and sigmoid activation for the final convolutional layer. The cross-validation results show there is no great bias in how the training, validation and testing datasets are divided.

5. DISCUSSION

Deep networks like U-net are susceptible to overfitting, but are able to model data fast. This was the case for this study as well – overfitting starts after a few epochs but the results gained at the peak of the model are good and loss decreases fast. This study proves that non-small cell lung cancer data with immunohistochemical fluoro-chromogenic stained whole slide mages is trainable for U-net and U-net is capable of recognizing PD-L1 activated regions with good accuracy. The final average results from manual k-fold cross validation for the best model Adam with learning rate of 0.00001 were AUC of **0.96**, accuracy of **0.88**, binary cross-entropy loss of **0.28,** IoU of **0.74**, precision-recall score of **0.94** and F-score of **0.87**. Jaccard score (IoU) ranged between 0.699 and 0.704. (**Table 2, Table 3.**) It was also found that the difference between Adam- and Adadelta-optimizers is small when it comes to results with non-small cell lung cancer data and U-net.

As can be seen in **Figure 11**, there is room for improvement in the classification accuracy. There exist different methods that could be used to improve the model. Ground truth images could have been created by a pathologist by going through cancer areas pixelwise and by hand thus creating near perfect target images. In this case the ground truth images are good but not perfect due to them being automatically generated by thresholding. Data augmentation could be done to increase the size of the dataset from thousands to millions of images, improving the generalizability of the model and leading to increasing accuracy when trained for longer periods. With this dataset the model starts to overfit if it is trained for a longer time, so the accuracy also remains lower than it possibly could be with further optimization.

The test set could also be larger. Due to there being only 34 samples, during manual k-fold cross validation 3 samples were used for the test set and 3 samples were used for the validation set. Cross-validation was used to reduce the bias in validating the model, so the results are more reliable than they would be with only one run of the network. If the same testset was used for validation each time, the results would not show the real performance of the model.

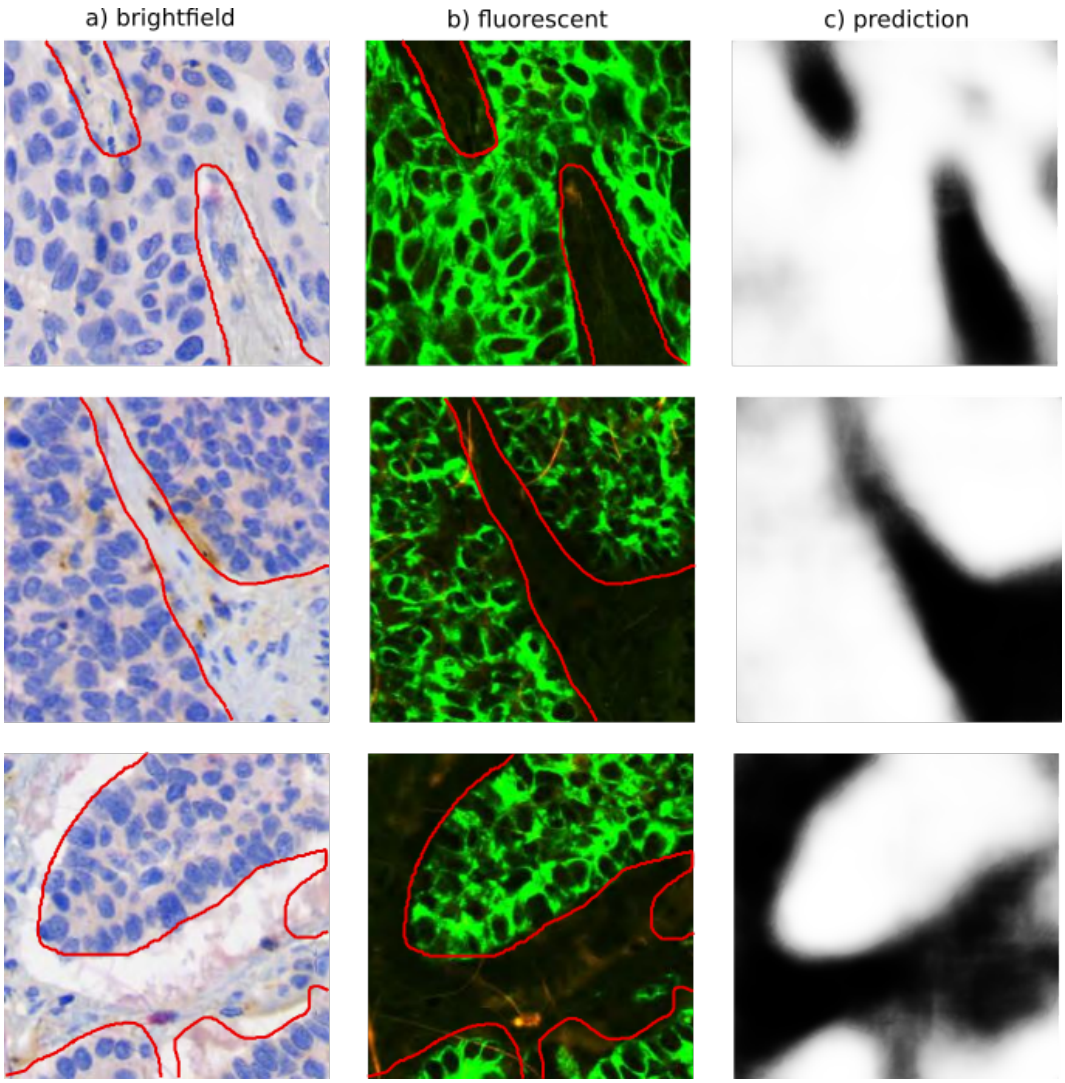

Figure 11. An example of prediction contours overlayed on image blocks with a) brightfield b) fluorescent illumination and c) predicted confidence maps. The boundaries of predicted areas are approximate as they are illustrated by hand.

In the future deep learning networks could be trained to identify which part of the body an image is coming from, but currently that still presents a challenge as most medical images need to be tiled in order to not exceed memory limitations when training the model. (Litjens *et al*. 2017)

This model otherwise is able to tell regions of interest well (**Figure 6.**), but could be more certain of the predictions. It also has trouble finding cancerous areas when they do not exhibit sufficient levels of staining (**Figure 11.**), which was expected. With this type of data, U-net is not yet at its fullest potential. The model could be more generalizable and data could be augmented in order to possibly improve the result. Data augmentation means increasing the size of the dataset by modifying existing data and ”fooling” the network into thinking the modified images have not been introduced

to it before. Especially image orientation and colour differences could be beneficial for stained tissue sample WSIs. Classification of images must be unrelated to orientation. Data augmentation such as rotating and flipping the data has been proven successful in generalizing a classifier (Wei *et al*. 2019). Plenty of convolutional neural networks reach their peak after being trained with millions of images, and in this case the amount of training data stayed in the thousands. The best action would be to ensure ground truth data is excellent by having a pathologist draw bounding boxes instead of doing automatic thresholding to get binary masks. Validation by a pathologist is also a method of getting a more reliable understanding on how the model performs.

6. CONCLUSIONS

In this thesis study a pipeline was created that preprocessed IHC fluoro-chromogenic histological images and trained a U-net on them. The results show that the convolutional neural network used was able to identify cancerous regions in non-small cell lung carcinoma WSIs with an average AUC score of 0.960. The study was performed working for Bioimage Informatics group at Faculty of Medicine and Health Technology, Tampere University.

This convolutional neural network model could be used as a decision making tool in classifying PD-L1 and PD1- activated regions in non-small cell lung cancer whole slide images. The benefit of a system like this pipeline being used by pathologists is that it could save time in diagnosis and possibly reduce costs of diagnosis by replacing an extra staining step. It could in the future be developed into usable software and further optimized for the use of pathologists in real-world scenarios. Optimization could include adding image augmentation, decreasing loss and improving generalization with different non-small cell lung carcinoma datasets. The results show it could be used as a decision support tool in classification of WSIs. There currently is no similar system being widely used in diagnosis of non-small cell lung cancer, and studies on non-small cell lung cancer, PD-L1 and deep learning combined are scarce.

The results of this study may be improved with data augmentation or a pathologist creating the target images by hand instead of automatically thresholding the binary masks from fluorescent images. Different parameters and changes to the network architecture may be examined, or different networks altogether. However, the performance of this model is good considering the data type can be complex to learn. Further validation could be performed by consulting a pathologist on the performance of the classifier.

REFERENCES

"Types and Staging of Lung Cancer." ***Lung Cancer 101*** | *Lungcancer.org*, www.lungcancer.org/find_information/publications/163-lung_cancer_101/268-types_and_staging.

Al-Janabi, S., Huisman, A., & Van Diest, P. J. (2012). **Digital pathology: current status and future perspectives.** *Histopathology*, *61*(1), 1-9.

Alom, M. Z., Hasan, M., Yakopcic, C., Taha, T. M., & Asari, V. K. (2018). **Recurrent residual convolutional neural network based on u-net (r2u-net) for medical image segmentation.** *arXiv preprint arXiv:1802.06955*.

Alom, M. Z., Taha, T. M., Yakopcic, C., Westberg, S., Sidike, P., Nasrin, M. S., ... & Asari, V. K. (2018). **The history began from AlexNet: a comprehensive survey on deep learning approaches.** *arXiv preprint arXiv:1803.01164*.

Clark, A. (2015). **Pillow (PIL Fork) Documentation.**

Criminisi A. (2016) **Machine learning for medical images analysis**. *Medical Image Analysis*, Volume 33, October 2016, Pages 91-93.

Danuser, G. (2011). **Computer vision in cell biology.** *Cell*, *147*(5), 973-978

D'Arcangelo, M., D'Incecco, A., Ligorio, C., Damiani, S., Puccetti, M., Bravaccini, S., ... & Landi, L. (2019). **Programmed death ligand 1 expression in early stage, resectable non-small cell lung cancer.** *Oncotarget*, *10*(5), 561.

Drozdzal, M., Vorontsov, E., Chartrand, G., Kadoury, S., & Pal, C. (2016). **The importance of skip connections in biomedical image segmentation.** In *Deep Learning and Data Labeling for Medical Applications* (pp. 179-187). Springer, Cham.

Ehteshami Bejnordi B, Veta M, Johannes van Diest P, et al. (2017) **Diagnostic Assessment of Deep Learning Algorithms for Detection of Lymph Node Metastases in Women With Breast Cancer.** *JAMA.* ;318(22):2199–2210. doi:10.1001/jama.2017.14585

Esteva, A., Kuprel, B., Novoa, R. A., Ko, J., Swetter, S. M., Blau, H. M., & Thrun, S. (2017). **Dermatologist-level classification of skin cancer with deep neural networks.** *Nature*, *542*(7639), 115.

Gertych, A., Swiderska-Chadaj, Z., Ma, Z., Ing, N., Markiewicz, T., Cierniak, S., ... & Knudsen, B. S. (2019). **Convolutional neural networks can accurately distinguish four histologic growth patterns of lung adenocarcinoma in digital slides.** *Scientific reports*, *9*(1), 1483.

Greenspan, H., Van Ginneken, B., & Summers, R. M. (2016). **Guest editorial deep learning in medical imaging: Overview and future promise of an exciting new technique.** *IEEE Transactions on Medical Imaging*, *35*(5), 1153-1159.

Haapaniemi T., Luhtala S., Ylinen O., Muhonen V., Tani T. & Isola, J. (**2017**) **Immunohistochemical fluoro-chromogenic double staining and digital image analysis for accurate detection of PD-L1 in cytokeratin-positive non-small cell lung cancer cells.** Presented in the 14th European Congress on Digital Pathology and the 5th Nordic Symposium on Digital Pathology, 29th May-1st June 2018, Helsinki, Finland.

Hahnloser, R. H., Sarpeshkar, R., Mahowald, M. A., Douglas, R. J., & Seung, H. S. (2000). **Digital selection and analogue amplification coexist in a cortex-inspired silicon circuit.** *Nature*, *405*(6789), 947.

Han, J., & Moraga, C. (1995, June). **The influence of the sigmoid function parameters on the speed of backpropagation learning.** In *International Workshop on Artificial Neural Networks* (pp. 195-201). Springer, Berlin, Heidelberg.

Hou, L., Samaras, D., Kurc, T. M., Gao, Y., Davis, J. E., & Saltz, J. H. (2016). **Patch-based convolutional neural network for whole slide tissue image classification.** In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 2424-2433).

Huang, G., Sun, Y., Liu, Z., Sedra, D., & Weinberger, K. Q. (2016, October). **Deep networks with stochastic depth**. In *European conference on computer vision* (pp. 646-661). Springer, Cham

Hunter, John D. **Matplotlib: A 2D Graphics Environment**, Computing in Science & Engineering, **9**, 90-95 (2007), DOI:10.1109/MCSE.2007.55

Iglovikov, V., & Shvets, A. (2018). **Ternausnet: U-net with vgg11 encoder pre-trained on imagenet for image segmentation.** *arXiv preprint arXiv:1801.05746*.

Jaccard, P. (1901). **Distribution de la flore alpine dans le bassin des Dranses et dans quelques régions voisines.** *Bull Soc Vaudoise Sci Nat*, *37*, 241-272.

Jones, E., Oliphant, T., & Peterson, P. (2016). **SciPy: Open source scientific tools for Python**, 2001.

Keskar, N. S., & Socher, R. (2017). **Improving generalization performance by switching from adam to sgd.** *arXiv preprint arXiv:1712.07628*.

Kingma, D. P., & Ba, J. (2014). **Adam: A method for stochastic optimization**. *arXiv preprint arXiv:1412.6980*.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). **Imagenet classification with deep convolutional neural networks.** In *Advances in neural information processing systems* (pp. 1097-1105).

LeCun, Y., Bengio, Y., & Hinton, G. (2015). **Deep learning.** *nature*, *521*(7553), 436.

LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). **Gradient-based learning applied to document recognition.** *Proceedings of the IEEE*, *86*(11), 2278-2324.

Liimatainen, K., Kananen, L., Latonen, L., & Ruusuvuori, P. (2019). **Iterative unsupervised domain adaptation for generalized cell detection from brightfield z-stacks.** *BMC bioinformatics*, *20*(1), 80.

Litjens, G., Kooi, T., Bejnordi, B. E., Setio, A. A. A., Ciompi, F., Ghafoorian, M., ... & Sánchez, C. I. (2017). **A survey on deep learning in medical image analysis.** *Medical image analysis*, *42*, 60-88.

Long, J., Shelhamer, E., & Darrell, T. (2015). **Fully convolutional networks for semantic segmentation.** In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 3431-3440).

Molina, J. R., Yang, P., Cassivi, S. D., Schild, S. E., & Adjei, A. A. (2008, May). **Non-small cell lung cancer: epidemiology, risk factors, treatment, and survivorship.** In *Mayo Clinic Proceedings* (Vol. 83, No. 5, pp. 584-594). Elsevier.

Mu, C. Y., Huang, J. A., Chen, Y., Chen, C., & Zhang, X. G. (2011). **High expression of PD-L1 in lung cancer may contribute to poor prognosis and tumor cells immune escape through suppressing tumor infiltrating dendritic cells maturation.** *Medical oncology*, *28*(3), 682-688.

Nirschl, J. J., Janowczyk, A., Peyster, E. G., Frank, R., Margulies, K. B., Feldman, M. D., & Madabhushi, A. (2018). **A deep-learning classifier identifies patients with clinical heart failure using whole-slide images of H&E tissue.** *PloS one*, *13*(4), e0192726.

Otsu, N. (1979). **A threshold selection method from gray-level histograms.** *IEEE transactions on systems, man, and cybernetics*, *9*(1), 62-66.

Robbins, H., & Monro, S. (1951). **A stochastic approximation method**. *The annals of mathematical statistics*, 400-407.

Ronneberger, O., Fischer, P., & Brox, T. (2015, October). **U-net: Convolutional networks for biomedical image segmentation.** In *International Conference on Medical image computing and computer-assisted intervention* (pp. 234-241). Springer, Cham.

Radiuk, P. M. (2017). **Impact of Training Set Batch Size on the Performance of Convolutional Neural Networks for Diverse Datasets.** *Information Technology and Management Science*, *20*(1), 20-24.
Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

Sokolova, M., Japkowicz, N., & Szpakowicz, S. (2006, December). **Beyond accuracy, F-score and ROC: a family of discriminant measures for performance evaluation.** In *Australasian joint conference on artificial intelligence* (pp. 1015-1021). Springer, Berlin, Heidelberg.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). **Dropout: a simple way to prevent neural networks from overfitting.** *The Journal of Machine Learning Research*, *15*(1), 1929-1958.

Taylor, C. R., Jadhav, A. P., Gholap, A., Kamble, G., Huang, J., Gown, A., ... & Rimm, D. L. (2019). **A Multi-Institutional Study to Evaluate Automated Whole Slide Scoring of Immunohistochemistry for Assessment of Programmed Death-Ligand 1 (PD-L1) Expression in Non–Small Cell Lung Cancer.** *Applied Immunohistochemistry & Molecular Morphology*, *27*(4), 263-269.

Valkonen, M., Kartasalo, K., Liimatainen, K., Nykter, M., Latonen, L., & Ruusuvuori, P. (2017). **Dual structured convolutional neural network with feature augmentation for quantitative characterization of tissue histology.** In *Proceedings of the IEEE International Conference on Computer Vision* (pp. 27-35).

Valkonen M., Isola J., Ylinen O., Muhonen V., Saxlin A., Tolonen T., Nykter M., & Ruusuvuori, P. (2018) **Cytokeratin-supervised deep learning for automatic recognition of epithelial cells in breast cancers stained for ER, PR, and Ki-67.** MANUSCRIPT

Van der Walt S, Colbert C, Varoquaux G. 2011. **The NumPy array: a structure for efficient numerical computation.** Computing in Science & Engineering **13(**2):22-30

Van der Walt S., Schönberger J., Nunez-Iglesias J., Boulogne F., Warner J., Yager N., Gouillart E., Yu T. and the scikit-image contributors. **scikit-image: Image processing in Python**, PeerJ 2:e453 (2014)

Wang, H., Roa, A. C., Basavanhally, A. N., Gilmore, H. L., Shih, N., Feldman, M., ... & Madabhushi, A. (2014). **Mitosis detection in breast cancer pathology images by combining handcrafted and convolutional neural network features.** *Journal of Medical Imaging*, *1*(3), 034003.

Wei, J. W., Tafe, L. J., Linnik, Y. A., Vaickus, L. J., Tomita, N., & Hassanpour, S. (2019). **Pathologist-level classification of histologic patterns on resected lung adenocarcinoma slides with deep neural networks.** *arXiv preprint arXiv:1901.11489*.

Xu, B., Wang, N., Chen, T., & Li, M. (2015). **Empirical evaluation of rectified activations in convolutional network.** *arXiv preprint arXiv:1505.00853*.

Yu, K. H., Wang, F., Berry, G. J., Re, C., Altman, R. B., Snyder, M., & Kohane, I. S. (2019). **Classifying Non-Small Cell Lung Cancer Histopathology Types and Transcriptomic Subtypes using Convolutional Neural Networks.** *bioRxiv*, 530360.

Zeiler, M. D. (2012). **ADADELTA: an adaptive learning rate method.** *arXiv preprint arXiv:1212.5701*.

Zhixuhao (2019) **U-net**, GitHub repository, https://github.com/zhixuhao/unet