UNIVERSITY OF TARTU

Faculty of Science and Technology

Institute of Technology

Ekaterina Sedykh

Automated segmentation of various features of glioblastoma in histopathological images

Bachelor's thesis (12 ECTS)
Curriculum Science and Technology

Supervisors:

Sarthak Pati, M.Sc.
Prof. Gholamreza Anbarjafari

Tartu 2022

Resümee/Abstract

Glioblastoomi erinevate tunnuste automaatne segmenteerimine histopatoloogilistel piltidel

Glioblastoom ehk neljanda astme glioom on agressiivne ajuvähi vorm, millel on üldiselt halb prognoos ja suur suremus. Täpseks kliiniliseks diagnoosiks ja raviks on oluline hinnata haiguse seisundit ja selle mikroskoopilisi struktuurseid omadusi, mis mõjutaksid positiivselt patsientide elulemust. Neid omadusi saab uurida uudse suure eraldusvõimega digitaalse patoloogia tehnika abil, mida nimetatakse kogu slaidi piltideks (whole slide image - WSI). Kasvaja anatoomia heterogeensuse tõttu on aga piltide annoteerimine väga aeganõudev ja tüütu ka kogenud patoloogidele. Arengud raalnägemise valdkonnas saavad võimaldada automaatset annoteerimist. WSI-de suuruse tõttu on tuleb eeltöötlusena teha suur pilt väiksemateks tükkideks. Selles töös hinnatakse mitmeid kaasaegseid süvaõppe meetodeid Ivy Glioblastoma Atlas Project andmebaasi piltide ühe - ja mitmeklassilises semantilises segmenteerimises. Eksperimendid hõlmasid erinevate pildi suuruste, neurovõrkude ja nende konfiguratsioonide võrdlemist, kusjuures kõige kõrgem dice'i koefitsent üheklassilisel treenimisel 100 WSI-ga on valideerimisandmetel üle 0,65. Esialgne uuring on mudel, mida saab laiendada suuremale andmestikule. Uuring pakub võrdlusalust olemasolevate süvaõppevõrkudega ja pakub uurimisalust tööks Ivy GAP histopatoloogia andmetega, näiteks erinevate slaiditasemete või piltisuurustega katsetamiseks. Töö viitab glioblastoomi struktuuride koetaseme automaatse segmenteerimise võimalikkusele ning selliste tehnikate edasiarendused võivad osutuda kasulikuks kliinilistes rakendustes.

CERCS: P176 Tehisintellekt; T111 Pilditehnika

Märksõnad: süvaõpe, glioblastoom, digitaalne patoloogia, kogu slaidi pildid, segmenteerimine

Automated segmentation of various features of glioblastoma in histopathological images

Glioblastoma or grade IV glioma is an aggressive cancer formed in the brain, with an initially poor prognosis and high mortality. For detailed clinical diagnosis and treatment, it is essential to assess the state of the disease and its microscopic structural features, which would positively impact patients' survival. These features can be examined using advances in digital pathology operating at high resolution, known as whole slide images (WSIs). However, due to the inherent heterogeneity of the tumor anatomy, manual annotation is time-consuming and tedious, even for experienced pathologists. Recent advances in computer vision can be leveraged towards generating computer-assisted annotations. WSIs create a computational bottleneck due to their large size, requiring preprocessing techniques like patch extraction. In this work, several state-of-the-art deep learning approaches are evaluated for single- and multi-class semantic segmentation tasks on patches extracted from WSIs from the Ivy Glioblastoma Atlas Project database, which has been categorically developed for glioblastoma research. The experimentation included working with different patch sizes, networks, and configurations, with the highest dice score from single-label training on just 100 WSIs being more than 0.65 for validation cases. The initial study serves as a blueprint to be broadened on a more extensive dataset. This study provides a benchmark with already existing deep learning networks and provides an exploration ground for working with Ivy GAP histopathology data, for example, trying different slide levels or patch sizes. It shows potential in tissue level automated segmentation of glioblastoma structural features and can potentially be used in clinical applications with further improvements.

CERCS: P176 Artificial Intelligence; T111 Imaging, image processing

Keywords: deep learning, glioblastoma, digital pathology, whole slide images, segmentation

Contents

List of Figures

List of Tables

Terms, abbreviations and notations

ANN - Artificial neural network

CAD - Computer-aided diagnosis

CNN - Convolutional neural network

CT - Cellular Tumor

DL - Deep learning

FN - False negative

FP - False positive

GBM - Glioblastoma

GT - Ground truth

HBV - Hyperplastic Blood Vessels

IoU - Intersection over Union

IT - Infiltrating Tumor

LE - Leading Edge

ML - Machine learning

MLP - Multi-Layer Perceptron

MSA - Multihead Self-Attention

MVP - Microvascular Proliferation

NE - Necrosis

PAN - Pseudopalisading Cells around Necrosis

PNZ - Perinecrotic zone

SGD - Stochastic gradient descent

TP - True positive

WHO - World Health Organization

WSI - Whole slide image

Introduction

Cancer causes anomalous cell division or inhibition of proper control on the system, hence, increases chances of mutations. Different treatment exists including surgery to remove diseased tissue, chemotherapy and radiation therapy to kill cancer cells and much more. Diagnosis process can include physical assessment of symptoms, laboratory test, imaging test and biopsy. Last one is needed to test possible tumor sample and get detailed pathology information before any therapeutical action can occur. Whilst the specimen can be examined under a microscope by pathologists to search for abnormalities, the manual process is subjective, time-consuming, and not scalable. Digital microscopy images with high resolution (also known as whole slide images (WSI)) provide detailed tissue information. Computationally-assisted processing of WSIs can help speed up the process of identifying diseased cells and help reduce clinician burnout.

Computer-assisted diagnosis (CAD) is a technology designed to assist medical professionals in interpreting medical images and making a diagnosis based on computer generated outputs. It is widely utilized to make processes more automated and to reduce the amount of time a clinical professional takes to process a single patient's scan. There are different tasks in tissue images analysis, including tissue classification, cancer grades classification, semantic segmentation [1, 2, 3]. Developing efficient methods for tissue image segmentation is more complex than those using computer vision datasets due to the tumor heterogeneity and complexity of tissue morphology. Deep learning methods have successfully spread out for large amount of various tasks, including image segmentation. They are a promising part of CAD, considering their performance in other fields. Technological advances applied in medical field may improve overall performance of clinical pipeline [4].

1 Literature review

1.1 Glioblastoma

Glioma is the most common malignant primary brain tumor, accounting for between 30 and 40 percent of all intracranial tumors [5], and they are among the most malignant tumors. This disease can cause a wide range of neurological symptoms, typically reflecting the physiological process controlled by the region of the brain most affected. It can cause cognitive issues, personality changes as well as movement dysfunction. Complete remission is extremely rare, and treatment aims to reduce symptoms and slow down its development. Glioblastoma (GBM) is a grade IV glioma, and is the most common malignant and aggressive type of brain tumor. This cancer forms from star-shaped astrocytes cells developing from weaker tumor [6].

Conducting a biopsy is one of the techniques used to get a definitive diagnosis, and it involves taking tissue sample from a brain for examination [7, 8]. GBMs contain a mixture of different cells, including blood vessels and areas of necrosis - dead cells, usually starts in cerebrum [9]. It grows fast and can spread quickly, infiltrating surrounding tissues, which makes it hard to treat.The tumor heterogeneity poses a challenge in diagnosis and treatment. For most patients with glioblastoma there is no positive prognosis, as even with standard treatment, median survival is expected to be $12-15$ months and 5 year mortality is >95 percent [10, 11, 12, 13].

Several studies showed that there is a correlation between expressed genes, hence molecular classification, and prognostic value [14, 15]. According to the Verhaak classification [16], GBM has four molecular subtypes based on gene expression-based:

- Neural subtype is characterized by expression of several neural markers: NEFL, GABRA1, SYT1, SLC12A5.
- Proneural subtype is typified by frequent point mutation in IDH1 and high PDGFRA gene expression.
- Classical shows chromosome 7 amplification and chromosome 10 loss, higher expression rates in EGFR, neural precursor and stem cell marker NES, Notch and Sonic hedgehog signalling pathways.

- Mesenchymal is the hardest subtype and can be characterized by having extensive necrosis, deletion or lower expression of tumor suppressor genes, inflammatory infiltrates.

Analysis of GBM histopathology may improve effectiveness of therapeutic treatment as well as provide potential prognosis information [17]. To understand the composition, tumor resection is performed, and all extracted parts need to be tested to its inherent heterogeneity. The features that can clearly characterize GBM on tissue level are microvascular hyperplasia and necrosis with surrounding pseudopalisades [18]. *Pseudopalisades* are neoplastic cells migrating from hypoxic areas, i.e., areas that do not have enough oxygen supply due to rapid tumor growth and insufficient blood supply. Therefore, they are located around *necrosis*, which are the dead cells, and move towards blood vessels. *Microvascular hyperplasia* is the increased production endothelial cells sprouting blood vessels which add to blood supply and fast development of malignant cells. Those factors have a high contribution to tumor hypoxia-driven growth starting from necrosis center [19, 20]. Other regions like cellular and infiltrating tumor can describe GBM state and give more definitive prognosis. *Cellular tumor* is the region mostly consisting of cancer cells, while *infiltrating tumor* is consisted of initially healthy brain tissue but with malignant invasion, hence it has lower cancer to normal cell ratio compared to cellular tumor. Infiltrating areas increase the chances of recurrence as it is difficult to safely remove all tumor mass during resection. Leaving these tissues damages healthy regions, and induces the formation of secondary tumors, while leading to lower survival [21]. GBM features and their definitions used in this study are described in Section 3.1.

1.2 Whole slide imaging

The main goal of whole slide imaging is to digitalize the entire glass slide with tissue into a high-resolution image, it is also known as virtual microscopy. Digital histopathological scans have been evolving with new technology developments, improving clinicians workflow as this microscopic specimens data can be stored, accessed and analyzed later. Whole slide images (WSIs) can be examined at high magnifications, allowing to see more details. These images can be viewed the same as glass slide under microscope. Usage of WSI has grown significantly and it has been approved for primary diagnosis [22, 23].

A scanner is practically a microscope, but with computer control. WSI scanners vary in resolution, scan speed, imaging mode (bright field and fluorescent), slide capacity, magnification and digital slide format [22]. Typical WSI scanners contains a camera, a light source, a microscope with magnifying lenses, a computer with needed software and robotics parts for glass movement. Modern systems use light-emitting diode-based strobe for light source, which adds to quality improvement, as well as other advances in image acquisition algorithms. Scanning speed per slide can vary from a couple dozens of seconds to several minutes, while being able to scan several slides at one load.

1.3 Machine Learning

The term ”machine learning” was firstly defined over 60 years ago - in 1959 by Arthur Samuel [24]. Machine learning (ML) is a branch of artificial intelligence in which machines learn without being explicitly programmed, enabled to automatically arrive at optimal solutions [25]. ML has been progressing during last couple of decades in fields of computer vision (CV), natural language processing (NLP), speech recognition, and others. Latest applications range from self-driving cars and smart assistants to email spam filters and search engines. Recently, it started to be applied in a broad range of applications in healthcare [26, 27, 28, 29], providing decision support to medical professionals. ML algorithms are designed to cater to specific applications, depending on the input type and desired output. Mainly ML can be divided into 3 categories [30], which is also illustrated on Figure 1.1:

1. Supervised learning
2. Unsupervised learning
3. Reinforcement learning

For model training and evaluation process data can be split into training, validation, and test sets, also known as discovery, replication and hold-out cohorts in clinical cases. The discovery cohort is used for model training and then is tested on the replication cohort to choose best weights and fine-tune parameters. Final model's unbiased evaluation is done with hold-out cohort. Both replication and hold-out set are held back from learning process. However, replication cohort helps to get higher metric in evaluation as it will give unbiased results rather than biased evaluation on training data [31].

Supervised learning

The most widely used type of ML is supervised learning [30, 32]. An algorithm is trained on discovery cohort with labels, mapping input data to desired output type. Then, the trained model is used to predict these labels on replication cohort. During training process, predicted output is compared with target output. The input variable x is a feature of a dataset, y is label or output that we want to predict. Algorithms can use one single feature or multiple features. The mapping of features and labels by a system is done by applying a function f: $y = f(x)$. Hence, $f(x)$ function is learned from (x, y) pairs. A feature is usually an n-dimensional vector containing a series of numbers that represents an object, e.g., an image can be represented by a 2D matrix. Overall, supervised learning is task-driven, with goal of finding an approximation of input-output mapping f.

Unsupervised learning

In unsupervised learning algorithms, unlabeled data is used [30, 32]. In this case, the algorithm receives the unlabeled discovery cohort x without y targets to predict. Model tries to find unknown patters in a dataset by searching for common aspects without the use of labels. It relies on data and its properties, hence, it's a data-driven type of learning. Unsupervised learning methods can be subcategorized to dimensionality reduction, such as principal component and linear discriminant analyses [33], and clustering, with an example of K-means method [34].

Reinforcement learning

The third type of ML is reinforcement learning that takes a set of actions to maximize the reward [30, 32]. Similar to unsupervised learning, the data given to the algorithm is unlabeled. The model interacts with environment and observes responses to actions. System needs to find optimal solution based on experience, using trial and error method [35]. Reinforcement learning is employed in a variety of areas, such as control theory [36] and game optimization [37].

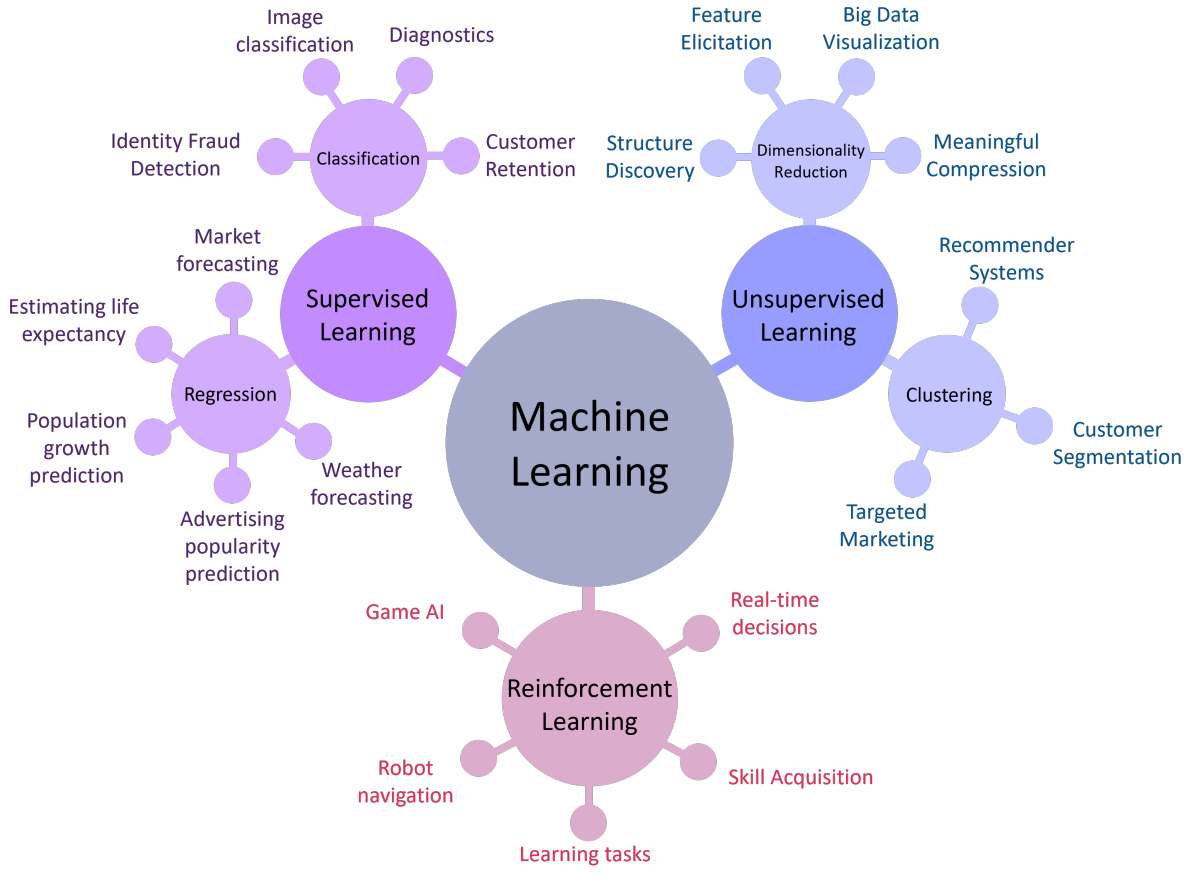

Figure 1.1: Machine learning categories. Classification of different learning algorithms in machine learning. ML can be categorized into 3 main categories: supervised, unsupervised, reinforcement. The choice of appropriate learning should be done considering application and available data. Image adapted from [38].

1.4 Deep Learning

Deep learning (DL) refers to a subset of ML techniques that is built from multi-layered neural network algorithms, computing parametrized function of observed data. There has been an increased interest in DL, since it seems to outperform previous state-of-the-art machine learning techniques in several fields [39, 40, 41], as well as in the presence of large amounts of data from different sources [42, 43, 44, 45]. In the medical field, this technique has proven to be extremely accurate in establishing diagnosis and predicting prognosis. In most cases, DL have major improvements in performance over other types of algorithms, although it requires more data to learn. Fortunately, the current availability of huge data for training and significant advances in computing hardware and software (PyTorch, [46], TensorFlow [47], Caffe [48]) have helped overcome most of the deep neural network's initial flaws and limitations.

The major inspiration for DL algorithms has been the human brain, which consists of hundreds of billions interconnected neurons that assist in recognizing patterns, learning new languages, and comprehending visual cues. Brain neuron functions can be imitated to some extent by machines. Artificial neural networks (ANN) are biologically inspired, showing similarities with the human neural system. Networks are densely interconnected neurons represented by nodes with connections aiming to mimic the brain's nature. Each connection has a specific numerical value – weight – based on its capacity of providing the desired outcome that is learned throughout the training phase. One node can take several inputs, adding them using considering respective weights. Each layer can be composed from different number of nodes. The depth of layers controls the model's ability to learn more complex features, as simple features can be found from the network's early layers, while more complex features are learnt deeper in the network. Neuroscience has been an important source of inspiration for DL, however, in recent years the shift from has been made with development of new algorithms.

In the case of simple ANN, a fully connected neural network is employed in which usually information is fed in forward direction - "feedforward" neural network, meaning it is passed from each node in the previous layer to each node in the next layer. Hence, the outputs of the neurons of one layer serve as the input for the neurons of the following layer.

An ANN usually comprises of an input layer, hidden layer(s), and an output layer, and Figure 1.2 illustrates these constituents. Input layer passes incoming data to neural network, hidden layers are used for nonlinear transformation of previous layer's output, output layer provides predicted output from the network.

They aim to recognize specific patterns in incoming data, then extract features, learn different characteristics of the dataset to produce results. Feature extraction is a process of transforming raw incoming data into some predictive variables; and in DL this is done automatically in the

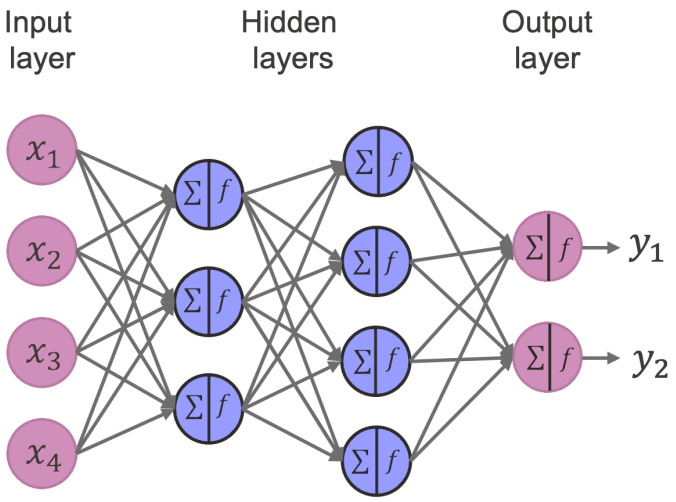

Figure 1.2: ANN structure. Neural network example, consisting of an input, 2 hidden, an output layers. In ANN dimensions of each layer can vary, as well as the number of hidden layers. In hidden layers each neuron (node) takes a set of weighted inputs from previous layer and produces an output using an activation function. Weights for each input connection are set during training process in which loss function has to be minimized. Image modified from [49].

network. This is the difference in DL and ML workflow (Figure 1.3). Learned representations usually result in better performance rather than human-design ones.

The basis of an ANN is the perceptron - a simple linear ML classifier algorithm, splitting data points into classes. Set of input features is transformed with the help of activation function. An activation function determines whether a neuron should be activated or not. By using mathematical operations, it determines whether the neuron's input to the network is important or not in the process of prediction. Commonly, activation functions are nonlinear as it helps to adapt to a variety of data [50].

The dominant approach to train feedforward neural networks is via "backpropagation", which is basically backward propagation of errors through the network. Backpropagation utilizes the generalized delta rule to compute the gradients for the network's parameters. The network's weights are fine-tuned through backpropagation so that the error between the desired output and the output predicted by ANN is minimized using an error function. After several iteration of training data, weights are adjusted and the network learns how to do close mapping function between input x and target y [51, 52].

Deep learning systems use iterative gradient-based optimization algorithms such as gradient descent in order to minimize an error function (also known as "cost function") and adjust parameters based on that error. During gradient descent, the model's parameters are gradually corrected in the steepest descent direction until output error reaches its minimum, using backpropagation gradients for neural network's parameters. A learning rate determines the step

algorithm need to take in each iteration to reach the global minimum [50]. This process repeats itself till the optimal solution is reached, or if the model cannot learn further.

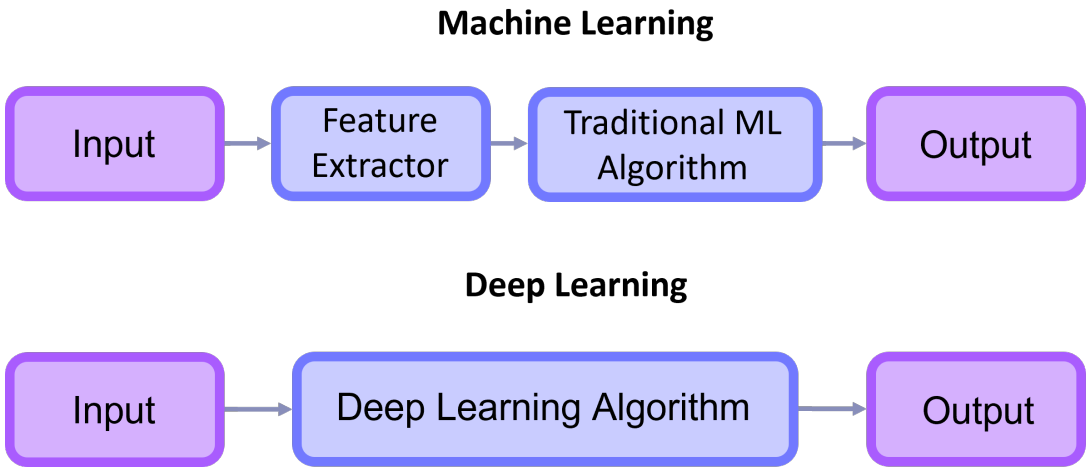

Figure 1.3: Machine learning and deep learning flow. Feature extraction is the process of converting raw data into numerical entities that quantify image characteristics. Deep learning methods aim to automatically extract features that will be more representative for learning process catering to the specific application. These learned features can enhance performance as they tend to be more meaningful than manually chosen ones.

1.5 Image Segmentation

Image segmentation methods classify pixels to a certain object or category, hence, there are different types of segmentation: semantic, instance, and panoptic [53]. A category is a region of identical texture, while an object is a countable entity like a single cell or nuclei inside a category. Semantic segmentation categorizes each image pixel to a class label - category. If each individual object inside a class is distinguished, then it is instance segmentation. Combining those two tasks, panoptic segmentation assigns every pixel a category and identifies separate objects of interest in each category. Therefore, unlike segmentation of instances, each pixel in panoptic case can have only one label corresponding to instance, forbidding overlapping segments, which are permitted in semantic segmentation - class labels need to be mutually exclusive. Ground truth labels for semantic segmentation do not contain objects information, only the category to which each pixel is classified [53].

1.6 Deep Learning in Image Segmentation

More biomedical imaging data has been produced in recent years as more advanced and fast techniques have been developed [54, 55]. Segmentation is an essential first step for biomedical analysis and CAD, which could aid researchers in gaining additional diagnostic information,

additionally speeding up drug research and diagnosis. Depending on the type of computation involved, image segmentation can be divided into manual, semi-automated and automated approaches [56, 57, 58]. It is a fundamental biomedical data analysis technique requiring specialists with knowledge and experience [59, 42]. Manual segmentation can be done using annotation tools with functionalities of drawing regions for multiple labels like CaPTk [58, 60] and ITK-SNAP [61]. However, manual segmentation is challenging, and time-consuming, especially for large number or size of images. As a result, there is a demand for segmentation to be done automatically by machines. It also forms the basis for further downstream computational analyses [62, 13, 63], such prediction of survival and recurrence.

Deep learning has been already in use for pathological images for several years, for example for nuclei segmentation [64, 65, 66, 67]. With more progress in DL, it can be applied to more complex tasks that have clinical significance. DL-based segmentation techniques have become an important part of WSI analysis of diseased tissues and in particular for tumors [68, 67]. The ability to differentiate between different tissue components is one of the main targets in digital histopathology. Tissue sample can have a variety of components, each having structural, functional, and chemical features. Automated segmentation is not simply needed for separating these tissue parts, but it can give an insight into pathological processes happening in a body that are linked to disease progression [69, 70]. Cellular complexity poses a problem on high accuracy of DL algorithms, making clinical deployability a big concern. Recently, convolutional neural networks based on encoder–decoder architecture have had an increased interest in usage on biomedical data. These networks have shown great promise in the field of radiology [71, 72, 73], thus making them a prime candidate for successful use in histology.

1.6.1 Purely convolutional methods

Deep learning has achieved state-of-the-art performance in computer vision, including segmentation for biomedical data [74]. Researchers focus on producing networks more specific for a task and data type, as medical image datasets differ significantly from those in the computer vision domain. DL has helped relieve the stress faced by clinicians when trying to process large quantities of data [75].

UNet

UNet [76] is a commonly used fully convolutional neural network specifically developed for precise biomedical image segmentation. The central concept lies in the contracting path - encoder and expansion path - decoder (Figure 1.4). The encoder captures the context on the image and is a sequence of convolutions followed by ReLu activation and max-pooling layers. The decoder learns localization features using a combination of convolution and upsampling layers. At each stage of the expansion path, the upsampled feature map is concatenated with the cropped feature map from the corresponding layer in the contracting path. These long skip connec-

tions between the downsampling and upsampling layers allow transferring information among encoder-decoder giving better information recovery. In the encoder part, number of channels increases stage by stage, while it decreases in the decoder. Additionally, there is a bottleneck mediating between contraction and expansion layers. Bottleneck layer contains fewer nodes then previous and later layers, hence reducing dimensionality of features. Thus, the architecture is almost symmetrical, resulting in a U-like shape. The important benefit of UNet is its ability to create detailed segmentations using limited samples in training, which is usually the case with medical data.

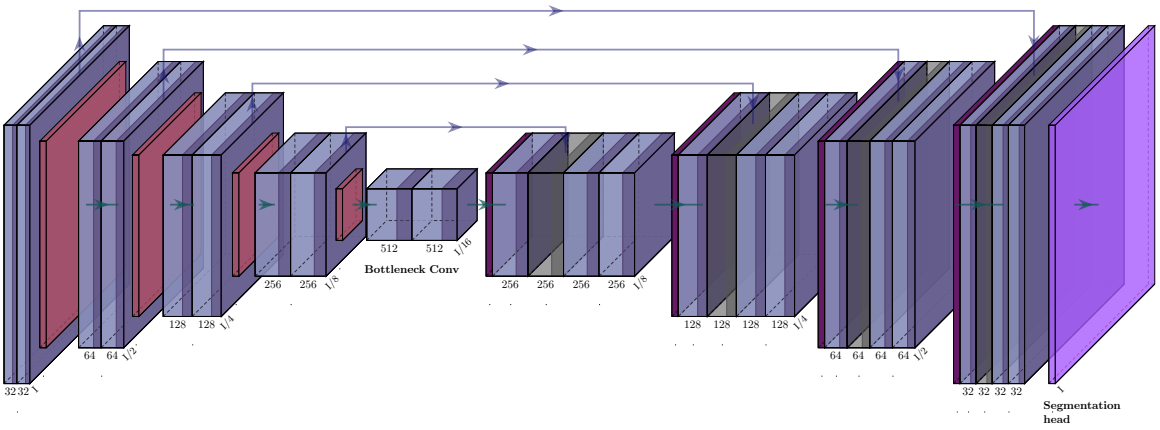

Figure 1.4: UNet architecture. Diagram of UNet architecture consisting of contracting and expansive paths which are commonly referred as encoder and decoder, accordingly. An input image is passed to the network, transformed by a series of convolutional layers with an activation function and maxpooling layers reducing data size. Then in decoding part there are also similar patterns of convolutional layer, but instead of maxpooling layers there are upsampling ones. Importantly, there are skip connections for preserving data, also helping with faster model convergence. Illustration made using [77].

ResUNet

Predictions tend to lack fine detail when using solely a UNet architecture, hence, cross or skip connections between network blocks can be introduced [78, 79]. There are variations on skip locations, and one of the more popular variants is including them in every convolutional layer, giving information from the contraction to the expansion path. Residual blocks allow to skip a section if needed, hence, more effectively finding global minimum and avoiding vanishing gradient problem [78]. Thus, the connections can utilize information from previous layer, improving backpropagation and as a consequence network's performance.

1.6.2 Transformer-based methods

Transformer is a neural network architecture based on attention mechanism to get global dependencies between input and output [80]. Transformer-based algorithms started to be heavily

used in NLP tasks [81]. Using transformers in computer vision became a new direction in object detection, segmentation and classification areas.

TransUNet

The TransUNet architecture (Figure 1.5) combines a transformer block with a CNN based on the UNet encoder-decoder architecture [82]. Self-attention mechanisms is introduced into the encoder part via the usage of Transformers. In the network, CNN-Transformer Hybrid is directly applied for encoding feature representations from image patches. First step is CNN which extract features, generating a feature map for the input instead of using a raw image. Then patch embedding is applied to 1×1 patches extracted from CNN feature map. Those embedded patches are fed into transformer part of encoder. A cascaded upsampler is used as a decoder consisting of upsampling steps for decoding hidden features to produce segmentation output. Additionally, to facilitate finding a better global minima, this architecture also has skip connections like UNet.

For the Transformer-based encoder, Vision Transformer (ViT) [83] was implemented with 12 Transformer layers. Each layer consists of Multihead Self-Attention (MSA) and Multi-Layer Perceptron (MLP) blocks, with LayerNorm applied before each block [80]. Multi-head Attention is an attention mechanism module that repeats computations with an attention mechanism multiple times in parallel. Input data is split and passed into separate heads. After that, the independent attention outputs are concatenated and linearly transformed into the expected output dimension to produce final attention score. Then after layer normalization, attention output is given to multi-layer perceptron.

1.6.3 Evaluation metrics

The choice of a suitable performance measure for a DL model depends on the application.

Intersection over Union

Intersection over Union (IoU) (1.1) is a commonly used evaluation metric in object detection field. It can be calculated by dividing the area of overlap between the ground truth and prediction by the area of union. The value can range from 0 to 1, 0 being no overlap is present, 1 meaning the prediction is the same as ground truth.

$$IoU = \frac{|A \cap B|}{|A \cup B|} = \frac{TP}{TP + FP + FN} \tag{1.1}$$

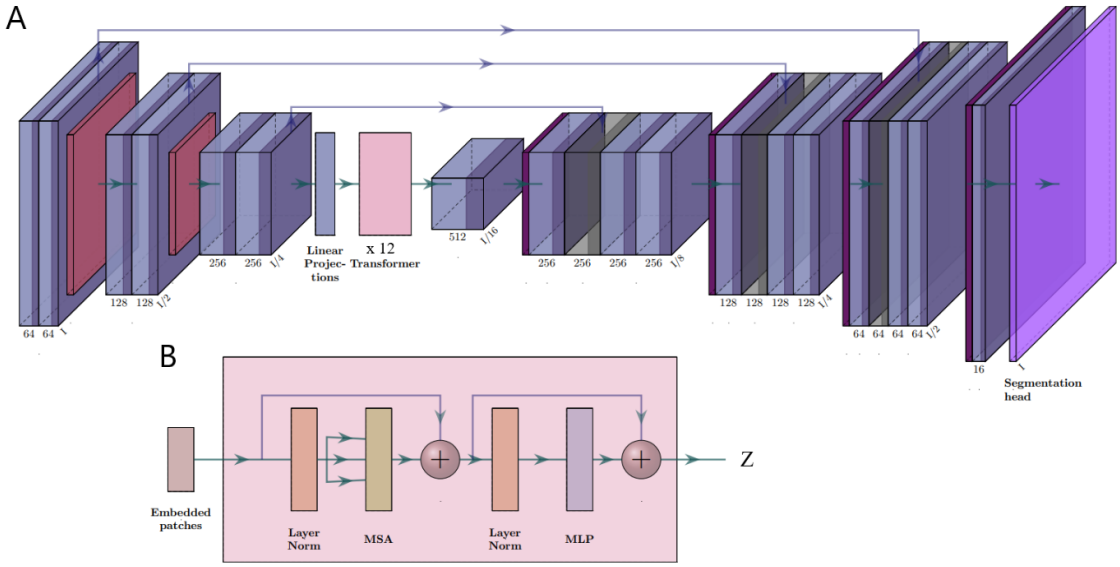

Figure 1.5: TransUNet architecture. Diagram of TransUNet architecture. **(A)** full architecture of TransUNet consisting of CNN-Transformer Hybdrid as an encoder and a cascaded upsampler as a decoder, resembling initial UNet structure. Encoder has convolutional layers, followed by 12 layers of Vision Transformer. **(B)** detailed schematic of the Transformer layer. Transformer layer receives embedded patches as input from CNN, which are fed into Multihead Self-Attention (MSA) and Multi-Layer Perceptron (MLP) blocks. Illustration made using [77].

Dice similarity coefficient

Dice coefficient (1.2) is very similar to IoU metric, however it is more frequently used in image segmentation tasks, when IoU is used in detection [84].

$$Dice = \frac{2|A \cap B|}{|A| + |B|} = \frac{2TP}{2TP + FP + FN} \tag{1.2}$$

2 The aims of the thesis

To facilitate glioblastoma diagnosis and prognosis through utilizing automated methods, this thesis proposes using deep learning approaches for structural features segmentation on tissue level in whole slide images. The primary aim is to test and analyze different segmentation models capable of working on histopathological data. It can help identify any bottlenecks interfering with the successful clinical implementation either from models or data sides. Through this work, the performance of deep learning convolution- and Transformer-based models are compared. Additionally, attention is needed to the medical relevance of this application, giving an informative overview of data and deep learning applicability with intricate histology data.

The steps to achieve the aims are:

1. **Pre-processing:** this is done by extracting patches from whole slide images. It is an important step because currently available hardware does not support processing such huge images as WSI at once, and another possible solution - severe downsampling - will cause details loss in high resolution images.

2. **Identification of relevant labels:** this has to be done to identify labels containing the most important clinical significance and their compatibility with training limitations.

3. **Model training:** this involved selecting and preparing models for training, including implementing architecture into the used framework and making model configurations.

4. **Experimentation:** Run experiments with different types of data (i.e., different input size, single- or multi-labeled training), and different training hyperparameters.

3 Experimental part

3.1 Materials and methods

3.1.1 Data

Data description

The data was taken from Ivy Glioblastoma Atlas Project (Ivy GAP), which is a publicly accessible database created by Allen Institute for Brain Science [85]. This published valuable resource investigates the expression of selected genes on different histological regions of GBM blocks, collecting genomic profiles and clinical data of patients. It contains entries of 42 tumors taken from 41 patients, with each person's information that might be needed for research purposes (Table 3.1). There are different types of data: MRI images, gene expression information, pathological samples, etc. In Figure 3.1 there is a pipeline of data creation for Ivy GAP, describing data generation. Therefore, the Ivy GAP database is a reliable source for GBM research with abundant information.

Table 3.1: Patient information in Ivy GAP database

Donor ID	Tumor name	Molecular subtype	Extent of Resection	Surgery	MGMT methylation	Survival days	EGFR amplification	Initial KPS	Age in years
12111	W10-1-1	Proneural	Complete	Primary	Yes	-	No	100	44
12112	W11-1-1	Classical, Mesenchymal	Complete	Primary	Yes	1076	No	100	57
10865	W1-1-2	Classical	Complete	Primary	No	105	Yes	100	66
12165	W12-1-1	Classical	Sub-total	Primary	No	80	Yes	70	61
12877	W13-1-1	Mesenchymal	Complete	Primary	No	250	Yes	90	59
12996	W16-1-1	Neural, Proneural	Complete	Primary	No	353	Yes	100	76

In this work, only WSIs with labeled structural features are used without any relation to patients and expressed genes. In the preparation stage in Ivy GAP, samples were taken from en bloc resected tumor tissues, subdivided into $9 \times 7.5 \times 9$mm blocks. After rapid freezing at -80°C, samples were sectioned at 20µm with object temperature of -10-11°C to eliminate any damages to the tissue that had contact with the blade, then they were put onto glass slides. Then H&E (hematoxylin and eosin) staining was performed to facilitate tissue assessment. Whole slides were scanned at 0.5µm/pixel resolution without downsampling. Image pre-processing included

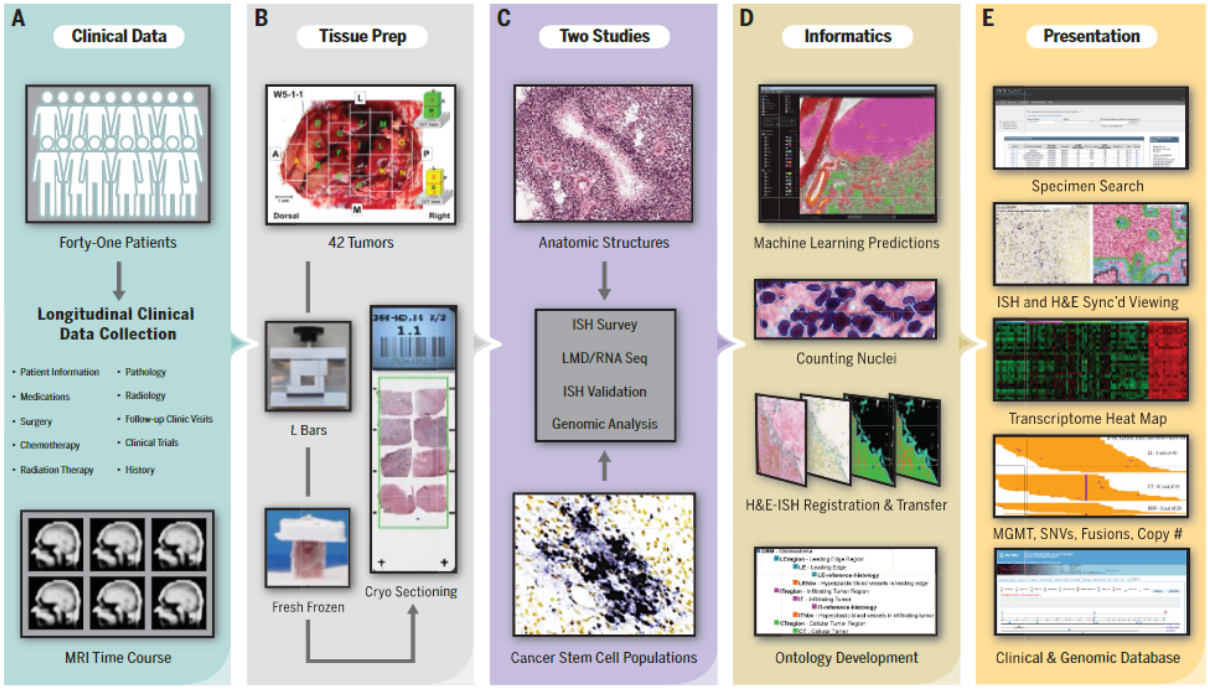

Figure 3.1: Ivy GAP data creation pipeline. **(A)** Clinical data like chemotherapy and surgeries details were collected from 41 selected patients. **(B)** Tissue preparation included tumor resection and cryo sectioning. **(C)** Anatomic feature–based profiling and cancer stem cell marker–based profiling, created a framework for the *in situ* hybridization surveys and validations, RNA-seq experiments. **(D)** Next steps included ontology development for anatomic features and their prediction using a semi-automated ML algorithm on histological data. **(E)** Open-source database was created (http://glioblastoma.alleninstitute.org/). Image is taken from [85].

white balance correction and compression as raw images weighted 5GB. Tissue blocks with excessive necrosis were eliminated (more than 75% necrosis per unit area). Images were labeled using a semi-automated annotation application called "Mill", and is based on leveraging decision trees. During training, some produced annotations were reviewed by experts in neuropathology and corrected, improving accuracy [85].

Inclusion/exclusion criteria

Annotation labels present in the Ivy GAP dataset do not comply with current World Health Organization (WHO) classification, and have 60% accuracy due to their semi-supervised annotation approach which is low [85]. The histopathological data provided for this work – 2176 WSIs - was analyzed by two board-certified neuropathologists to check consistency and compliance with the clinical evaluation WHO criteria during standard practice. The structural features that Ivy GAP's semi-automated annotation tool accurately recognized according to published criteria [85] were marked for inclusion in this study for each image, while others were omitted. To make sure the most clinically relevant labels are in abundance, checking label distribution

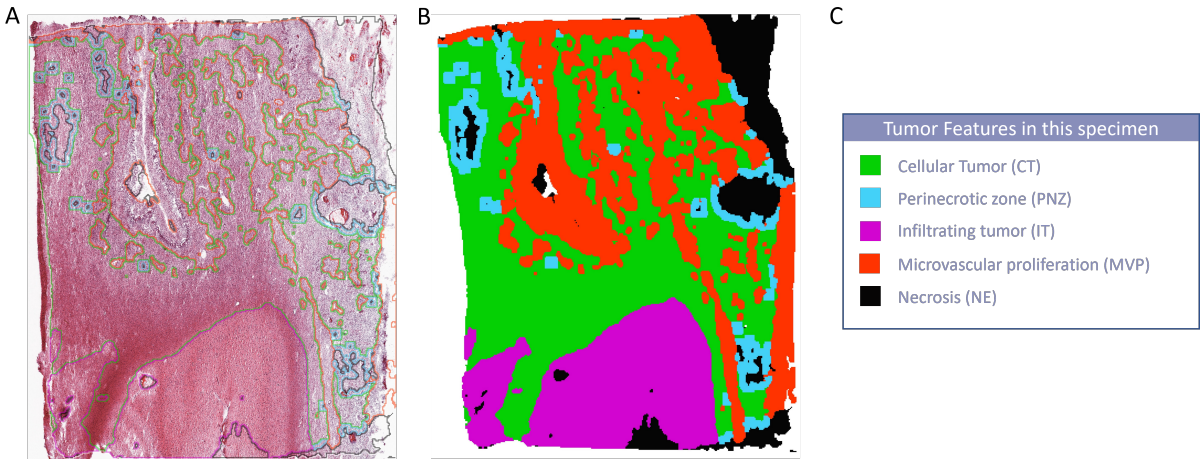

Figure 3.2: Example of annotated image. **(A)** H&E WSI with marked region boundaries. **(B)** WSI mask annotation for multiple labels. **(C)** Label colors with corresponding GBM structural feature label names present in this sample.

over all items in the dataset was done. In Figure 3.3A, it is visible that label 3 (cellular tumor) is most prevalent, and two others - label 2 (infiltrating tumor) and 4 (necrosis) - have notable value as well. Figure 3.3B provides a distribution over 100 selected images on which most of the training experiments were done; the histogram's pattern resembles the one with all samples; hence, there is the same label imbalance, and it can serve as a representative selection. Labels 7 and 9 - pseudopalisading cells with no visible necrosis and microvascular proliferation are almost non-existing in this selection. Label 0, which is marked as problematic pixels, usually is just a case of a white background with a possible small amount of cells there.

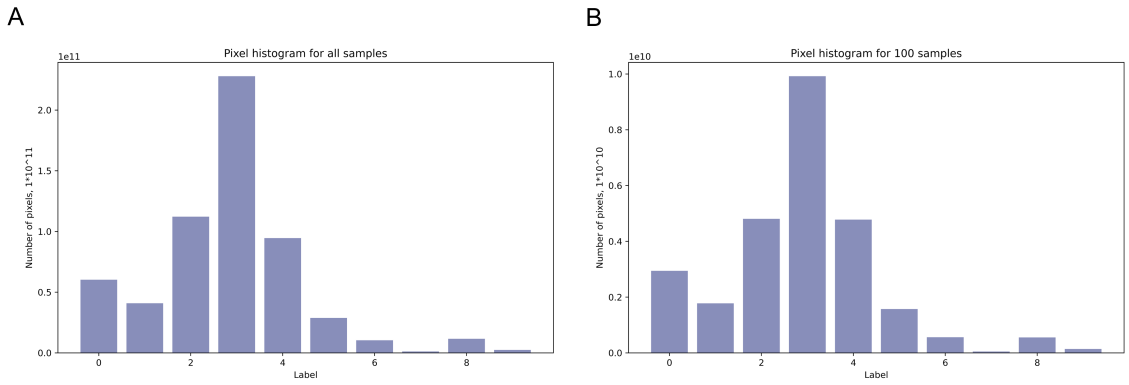

Figure 3.3: Histograms of labels distribution in selected part of Ivy GAP dataset. **(A)** Distribution of pixel labels over all 2167 images. **(B)** Distribution of pixel labels over 100 images. Both distributions have similar pattern to each other, just histogram **(A)** has 20 times more pixels. The most prevalent label is 3 - CT, and two others are 2 and 4 - IT and NE.

1. Leading Edge (LE) is the outermost boundary of the tumor, where the ratio of tumor to normal cells is about 1-3/100, and cortical layers architecture is often observed.

Table 3.2: Data segmentation labels

Gray Level	RGB Color	Feature Label
0	255 255 0	Problematic pixels
1	33 143 166	Leading Edge
2	210 5 208	Infiltrating Tumor
3	5 208 4	Cellular Tumor
4	5 5 5	Necrosis
5	67 209 247	Perinecrotic Zone
6	67 208 170	Pseudopolisading Cells around Necrosis
7	6 4 209	Pseudopolisading Cells but no visible Necrosis
8	255 102 0	Hyperplastic Blood Vessels
9	255 51 0	Microvascular Proliferation

2. Infiltrating Tumor (IT) is the intermediate zone between the Leading Edge (LE) and Cellular Tumor (CT), where the ratio of tumor to normal cells is 10-20/100 and is frequently marked by perineuronal satellitosis.

3. Cellular Tumor (CT) constitutes the major part of the core, where the ratio of tumor cells to normal cells is about 100/1 to 500/1. Density can range, exceeding typical cell mass levels or lower, possibly due to early necrosis.

4. Necrosis (NE) is dead or dying tissue, marked by the presence of karyorrhectic or cellular debris and the absence of crisp cytological architecture. It can be found in the core of a tumor.

5. Perinecrotic zone (PNZ) refers to a boundary of tumor cells in the tumor core along the edge of necrosis that lacks a clear demarcation of PAN.

6. Pseudopalisading cells around necrosis (PAN) are aggregated tumor cells, forming a narrow boundary of cells arranged like pseudopalisades along the NE zone in the core. In immature PAN, there is little NE and a zone of a high-density of pseudopalisading cells, whereas, in mature PAN, there is extensive NE and a low-density region or faint ridges of pseudopalisading cells.

7. Pseudopalisading cells but no visible necrosis (PNN) was converted from PAN feature if there was no necrosis (NE) within 900 pixels of PAN.

8. Hyperplastic Blood Vessels (HBV) are blood vessels with thickened walls found throughout a tumor and can be in different sizes and shapes. HBV is marked if there is an in-

creased density of blood vessels, specifically with thickened walls.

9. Microvascular Proliferation (MVP) refers to two or more blood vessels sharing a common vessel wall of endothelial and smooth muscle cells, typically located in the tumor core. They can appear in the shape of a glomerulus or as a garland of multiple interconnected blood vessels.

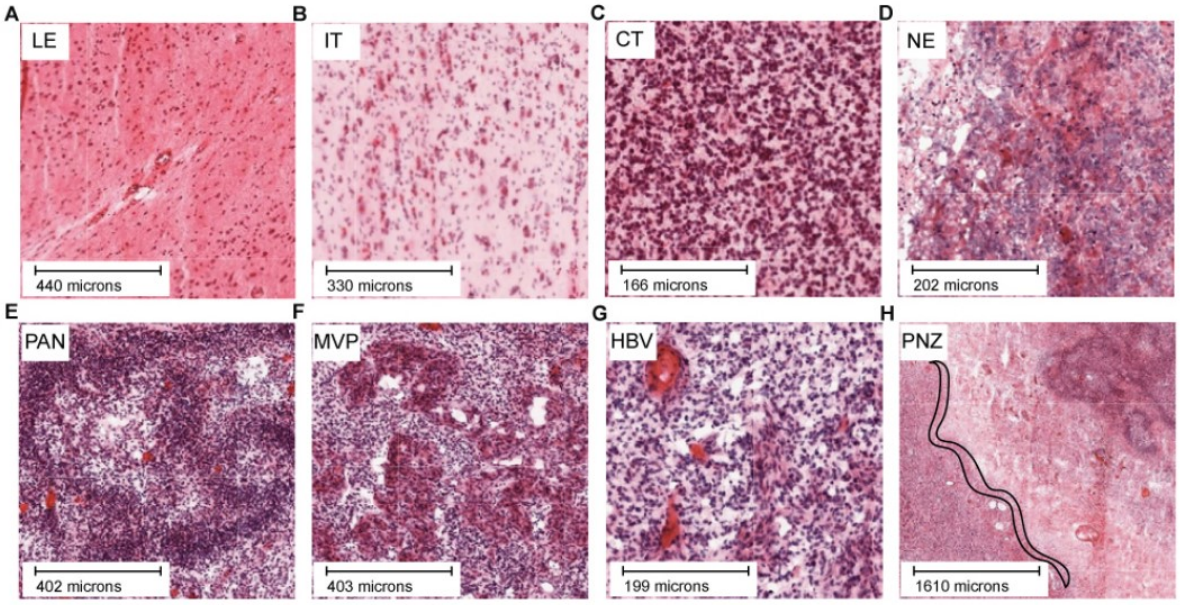

Figure 3.4: GBM histological features. Example samples from Ivy GAP dataset annotations showing GBM structural features on tissue level in H&E stained images. **(A)** Leading Edge (LE). **(B)** Infiltrating Tumor (IT). **(C)** Cellular Tumor (CT). **(D)** Necrosis (NE). **(E)** Pseudopalisading Cells around Necrosis (PAN). **(F)** Microvascular Proliferation (MVP). **(G)** Hyperplastic Blood Vessels (HBV). **(H)** Perinecrotic zone (PNZ). PNZ is depicted in a black outline.

Patch extraction

Patch extraction is common practice for working with WSI [86, 87]. This process is required because WSI cannot always be processed in its original size due to computational limits. Open-source Open Patch Miner (implemented in GaNDLF [29]) provides simple processing of patches. Upon receiving the whole slide image, the white background is first masked out using Otsu thresholding [88]. Therefore, it leaves only relevant tissue and can significantly reduce the computational search space. Then the program extracts non-overlapping patches of the desired size until exhaustion using a pseudo-grid and parallel sampling. The process stops when the slide is saturated with patches, and no more can be created. Corresponding coordinates of patches' locations are saved.

3.1.2 Training and evaluation

The models were trained with NVIDIA A40 graphical processing unit from the CUBIC cluster at the Center for Biomedical Image Computing and Analytics at the University of Pennsylvania. Experiments were done using Generally Nuanced Deep Learning Framework (GaNDLF) [29]. GaNDLF is based on PyTorch and can be used for classification, segmentation, and regression tasks using various medical imaging DL models. It is a versatile tool for training DL models, with support for a variety of imaging modalities and output classes, as well as 2D and 3D datasets. The user provides a CSV file with paths to all required input images, and masks for the training process. The subject identifiers are used to divide the complete dataset into three subsets: training, validation, and testing. A configuration file is utilized to regulate the training hyperparameters, including data split proportions, data pre-processing, data augmentations, model parameters, and training queue parameters [29]. Example configuration file can be found in Appendix A. UNet and ResUNet are already implemented there and can be directly used with specified configurations for given data and tasks. As part of this thesis work, TransUNet was implemented into GaNDLF [1].

Extracted patches are split into sets on the basis of their subject identifiers to ensure no data leakage takes place. Therefore, patches from the same subject won't be in different splits. The data is split into train, validation, and test sets with 7:1:1 ratio, and corresponding label patches go with their images. Train set with images and ground truth labels is fed into a DL algorithm. The labels indicate what the network has to learn, hence, it is a supervised learning approach. Comparing learned features and actual train labels by calculating the loss function, the model can adjust its weights. This gives feedback if the model is learning in the right direction. After each epoch, which is defined when a network has seen all training data once (each image has been propagated through it), the model goes into validation mode. The model performs forward pass across the entire validation set, resulting in predictions that can be compared with ground truth labels of this set using the specified validation metrics. Model selection can be done for the model that performs the best on the validation set at a particular epoch. A visual representation of this learning process can be seen in Figure 3.5.

Loss

In model design, the choice of loss is another important factor apart from the actual network architecture. It is an objective function that estimates model error by comparing ground truths and predictions. The loss function indicates how well the network has learned to map data features from a set of inputs to a set of outputs. Helping to optimize weight changes in the right direction, it navigates gradient so the next evaluation will have a smaller error. Choosing the loss is challenging since the function must represent the properties of the data and the problem to

[1] https://github.com/katesedykh/GaNDLF/tree/transunet

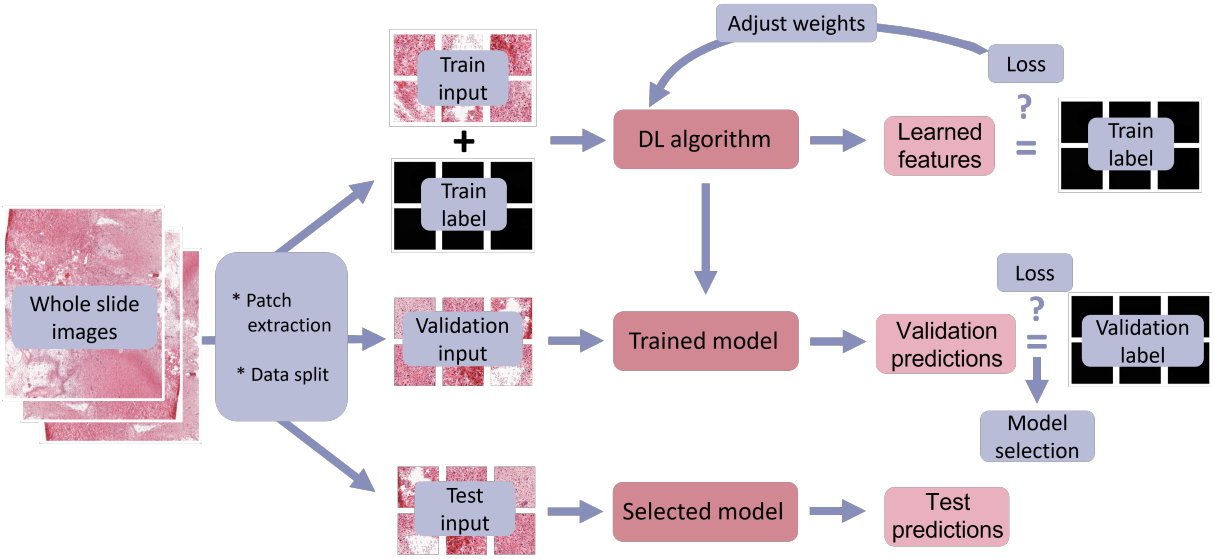

Figure 3.5: Training pipeline. Supervised learning approach is applied due to label data availability, where predictions can be compared with ground truths. Diagram shows data splits into train, validation and test with their own paths.

be solved. For semantic segmentation, the cross-categorical entropy loss function is frequently used, but for detailed segmentation of intricate data, more complex loss should be integrated. Dice loss is one of the most commonly used alternatives in medical image segmentation tasks. It improves a network based on the overlap dice coefficient between the ground truth label and the prediction of the system. The dice coefficient (shown in equation 1.2) ranges from 0 to 1, with '0' representing no overlap between predicted and ground truth labels to '1' representing perfect overlap. Since the loss itself has to be minimized, it can be defined as:

$$Dice\ loss = 1 - Dice\ coefficient \tag{3.1}$$

To improve accuracy in multi-class imbalanced segmentation, a weighted loss is used. The main idea of weighted loss is giving different weights to different classes based on the quantity for each class. Classes that can be predicted worse are assigned higher coefficients; therefore, error for this label would be higher. Including this part helps to learn different classes more uniformly, rather than predicting one class with high accuracy but skipping other classes due to insufficient data.

Hyperparameters

Hyperparameters control the learning process of a DL algorithm; they contribute to weights finding optimization. As mentioned in Section 1.4, learning rate determines the step with which the network updates its weights. If it is too big, the network will take larger steps and can easily miss a local minimum; while if it is too low, updates are tiny, and the training process progresses slowly. An optimizer changes the learning rate and weights of neurons. Stochastic Gradient Descent optimizer single records to updates weights and not the whole data as just

Gradient Descent, but there is a drawback: finding loss minimum can be quite noisy and also slow to reach stability (converge) as it needs forward and backward propagation for each step. Due to those noisy peaks, the learning rate has to be on the lower side. Momentum can solve this issue by introducing memory of gradient from previous steps and based on that determining the right direction to go. Adam optimizer utilizes momentum concept and adaptive learning rate. It is computationally efficient, and also beneficial as it requires little hyperparameter tuning [89]. A scheduler is parameter that adjusts learning rate in training process based on defined schedule, for example, exponential decay or step decay.

3.1.3 WSI inference

Inference is also based on patch-based mechanism, but here patches do not need to be extracted before the process, WSI is passed directly to the framework. The trained network's forward pass on patches is done, making predictions for each individual patch. Then the predictions on overlapping patches are stitched back together, forming a "segmentation probability map". This map indicates which regions are most likely correspond to the label, with final probability being between 0 and 1 and threshold for segmenting equals 0.5.

3.2 Results

3.2.1 Data preparation

Alongside WSIs, the segmentation annotations were provided as gray-scale images with pixel values corresponding to a number of the label. They were converted from the original RGB label using values in Table 3.2. Images in the dataset contained 4 channels: RGBA. As the alpha channel didn't contain any useful information - all the values were equal to 255 - it was removed, leaving images with RGB channels. As initial images and label maps represent whole slide and are quite large to be processed as one - $\sim 18000 \times 15000$ pixels, they were separated into patches of size 1024×1024 or 512×512 using method described in 3.1.1 subsection under patch extraction.

3.2.2 Training results

Considering the fact that TransUNet cannot work with bigger patches in set limitations, it was crucial to check difference in performance depending on patch size (Table 3.3). This was done with ResUNet architecture as it can work with both sizes of patches.

Table 3.4 provides dice scores for all networks trained with the same setting to properly compare them as not all can work on the same size of data, however bigger patches can result in better result. Hence, 512×512 patches were used for these single-label training experiments.

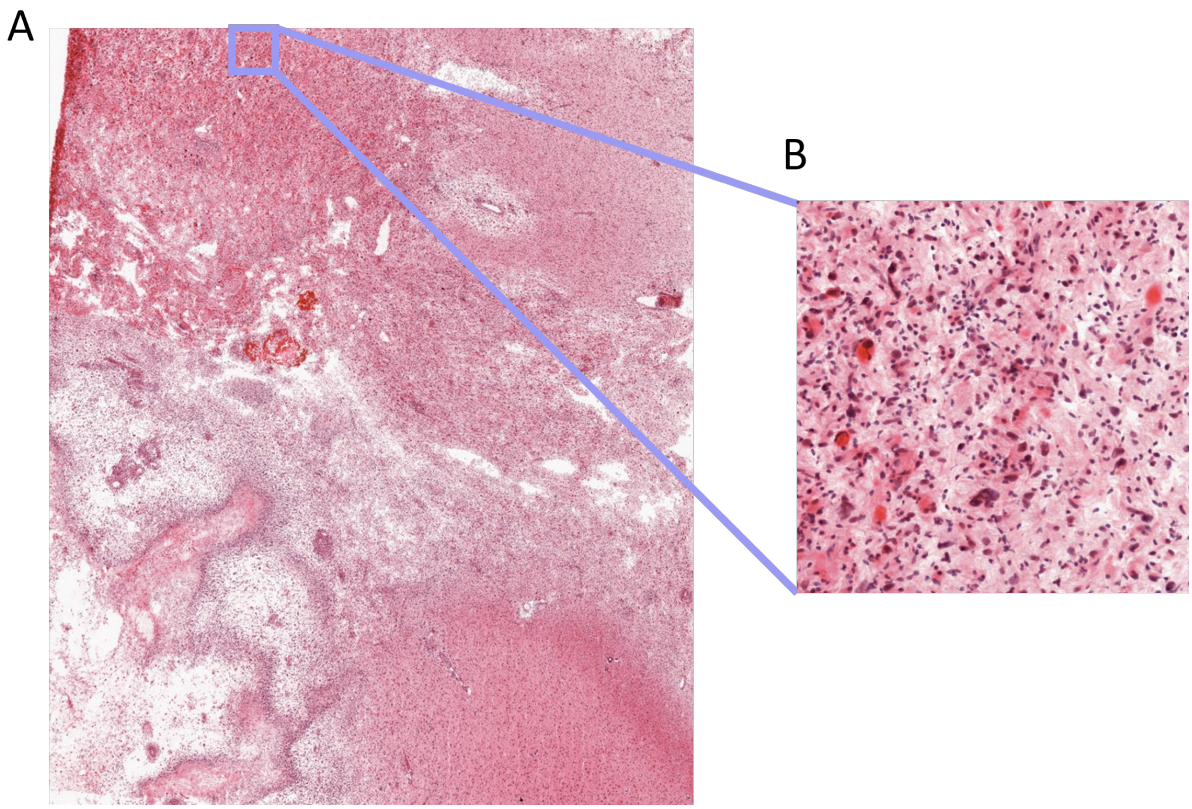

Figure 3.6: Sample whole slide image and an extracted patch. **(A)** An example of whole slide histopathology image from the Ivy Glioblastoma dataset with original dimensions $\sim 18000 \times 15000$ pixels. **(B)** A 1024×1024 patch extracted from WSI. This example shows the amount of details in one WSI and what context is captured in patches.

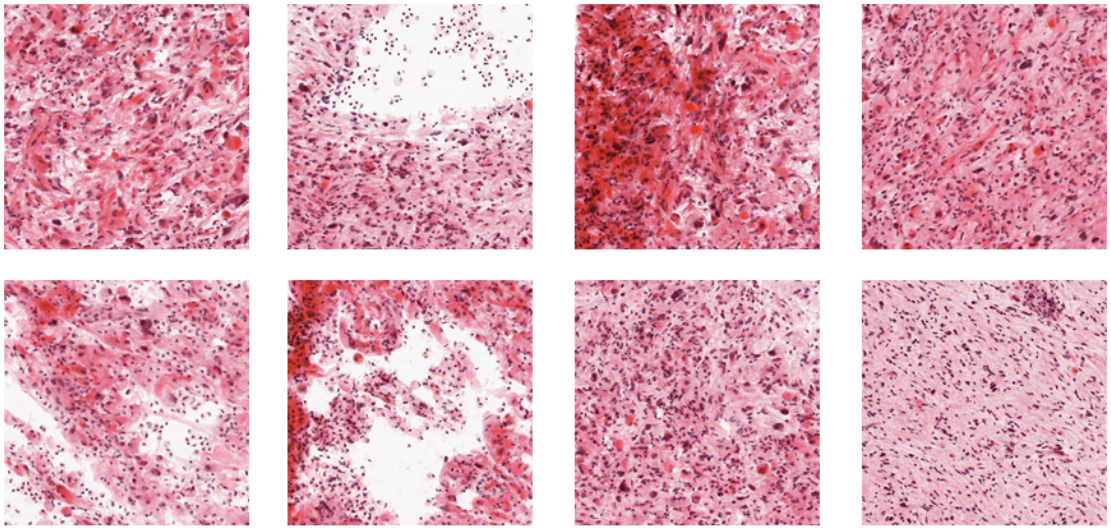

Figure 3.7: Example of extracted patches. Sample set of patches with 1024×1024 pixels dimension, extracted from whole slide image.

Table 3.3: ResUNet training with differently sized patches

Label	Hyperparameters		Architecture	Dice score
3	Adam optimizer, exponential scheduler learning rate 0.01	patch size 512x512	ResUNet	0.4047
		patch size 1024x1024		0.6379

To evaluate the role of hyperparameters in the task of semantic segmentation of the given data, it was decided to try out different learning rate and optimizers (Table 3.5). ResUNet was chosen as it provided good results and trained relatively fast.

Eventually, multiple label training was also tested with labels 2, 3 and 4. The results in Table 3.6 shows dice over all labels and then dice per each label.

3.3 Discussion

The experimentation started with training only on a subset of the entire cohort, i.e., 100 out of a total of 2167 subjects. This was done to form an understanding of the computational requirements. After the patches extraction process, this dataset part ended up consisting of $22,043$ patches of size 1024×1024 or $89,722$ of size 512×512, which was deemed minimum to provide a trained model with enough global context per patch. Those 100 subjects contained a similar label distribution compared to the whole dataset, which can be seen in Figure 3.3, hence, these experimental results can provide preliminary insight into the patterns that could

Table 3.4: Comparison results between networks trained with same hyperparameters and on same patch size

Label	Hyperparameters	Architecture	Dice score
3	Patch size 512x512, Adam optimizer, exponential scheduler, learning rate 0.001	UNet	0.4054
		ResUNet	0.4049
		TransUNet	0.4015

Table 3.5: ResUNet single label training for cellular tumor with different optimizers and learning rates

Label	Architecture	Hyperparameters			Dice score
3	ResUNet	Patch size 1024x1024, exponential scheduler	Adam optimizer	lr = 0.1	0.6551
				lr = 0.01	0.6379
				lr = 0.001	0.6714
			SGD optimizer	lr = 0.1	0.6183
				lr = 0.01	0.6787
				lr = 0.001	0.6725

arise while training on the whole dataset.

Characteristic features for GBM are pseudopalisades around the necrotic core, which are present in the dataset, and microvascular hyperplasia, which can be closely chosen as hyperplastic blood vessels (HBV) and microvascular proliferation (MVP) labels in the existing dataset. However, there are other important labels that have clinical significance, like infiltrating tumor, as it can provide insight about the probability of tumor recurrence. Since the whole dataset is extensive, it was decided to train on a smaller amount of classes. Taking into consideration clinically important features and available dataset, the focus can be on three labels: 2 - infiltrating tumor, 3 - cellular tumor, and 4 - necrosis. They can be used simultaneously for multi-label training or separately for single-label experiments. MVP, HBV and PAN labels have a low occurrence of pixels in the dataset, which is a limiting factor for this work. Hence, by training only on selected labels 2,3,4, which take over 70% of labels over the entire dataset, the most important regions were chosen, and the number of classes to predict was eliminated.

When choosing a network, it was decided to compare a common convolutional network with the one containing a Transformer part in the encoder to see how differently it can extract features from given data. The model selection included UNet and ResUNet for convolutional models, TransUNet for Transformer-based models. All those are based on encoder-decoder architectures were designed for biomedical data. This type of architecture gained popularity among the research community due to its ability to capture image context and work with a little amount of data. The initial hypothesis in comparing purely convolutional and transformer networks was

Table 3.6: ResUNet multi-label training for infiltrating tumor, cellular tumor and necrosis

Label	Architecture	Hyperparameters		Eval label	Dice score
2,3,4	ResUNet	Patch size 1024x1024, Adam optimizer, exponential scheduler	lr = 0.1	Cumulative	0.2704
				2	1.31E-11
				3	0.4747
				4	0.3364
			lr = 0.01	Cumulative	0.2798
				2	1.77E-10
				3	0.4627
				4	0.3690
			lr = 0.001	Cumulative	0.2687
				2	4.28E-13
				3	0.4067
				4	0.3994

that, for example, UNet/ResUNet can perform better while looking at a bigger picture. Transformers may perform well on small local patches, but then putting global context, there might be issues. Specifically, with conducted experiments, this can also be concluded because it was possible to train UNet/ResUNet on 1024×1024 patches, but TransUNet in available hardware settings was working only with 512×512, giving Transformers less context.

To analyze the differences between selected networks, it was decided to run experiments with the chosen architectures with the same hyperparameters and same data size, i.e., CT single label for 100 subjects as it is the most abundant label in the dataset. Considering Transformer's high computational demands, 512×512 patches were used to track models' differences. The acquired results did not show significant differences in performance between networks, slightly varying in dice score around 0.40. A hypothesis about the difference between convolutional networks was that ResUNet would perform better than UNet since residual connections in blocks have tended to produce better results [78]. This network comparison experiment can eliminate the hypothesis as UNet performed on par with ResUNet, but it is possible to assume that there might be small variations in dice score results if there would be a different data splits. Figure 3.8 depicts an example difference between those two networks. It can be seen that ResUNet has sharper boundaries which could be beneficial it there would be very detailed ground truth annotations, whilst UNet has smoother borders. In this particular example UNet performed worse by segmenting more area. TransUNet did not provide any advantage over purely convolutional architectures, resulting in 0.4015 dice score on given validation data split and segmenting almost the whole WSI in single-label inference. Here it is also worth mentioning a comparison of ResUNet training with the same setting, but only differing in patches size. The initial hypothesis was that dice score would be higher in larger patches, and indeed it was confirmed. While ResUNet trained on smaller patches - 512×512 - performed with dice of 0.4047, the model

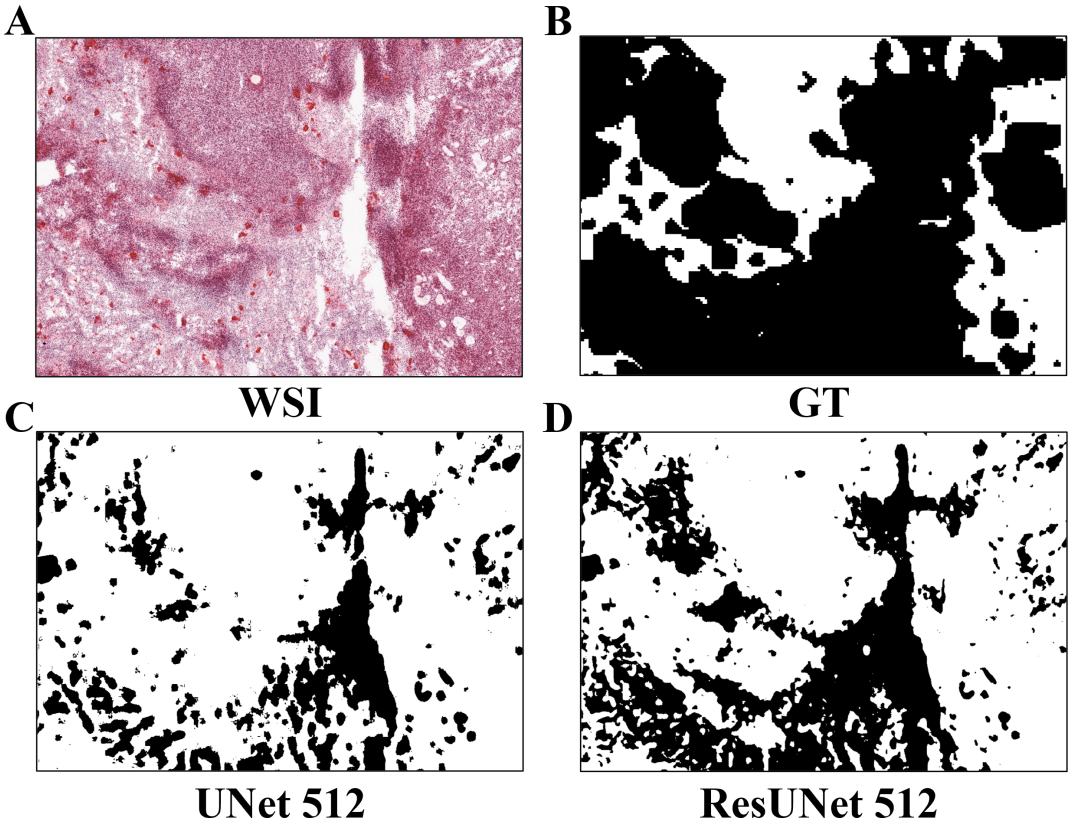

Figure 3.8: Sample result of CT label inference with UNet and ResUNet. **(A)** Part of WSI. **(B)** Ground truth label for CT. **(C)** UNet trained on 512×512 patches. **(D)** ResUNet trained on 512×512 patches.

trained with the same parameters but on larger patches - 1024×1024 - resulted in 0.6379 dice score (Table 3.3). Larger patches have 4 times more tissue sample than smaller ones in this case, hence, there is more context of surrounding areas.

For evaluation of role of hyperparameters on results in experiments for this work, it was decided to run several experiments with varying learning rates and two different optimizers - SGD and Adam. ResUNet was used in these experiments just to provide an example as there weren't significant difference between network performance. Table 3.5 provides results for three different learning rates: 0.1, 0.01, 0.001 for each optimizer. As these experiments were done on the same data split, there is no consideration of variation depending on data. One of the expected highlight differences between optimizers was that Adam provided a smoother performance and loss with SGD was fluctuating more. It can be concluded that overall smaller learning rate would give a higher dice score. With Adam optimizer the best result of 0.6714 was with smallest learning rate of 0.001, whilst others gave a bit smaller dice. For experiments with SGD, there is not much difference between 0.01 and 0.001 learning rate, with former being slightly better - 0.6787 and 0.6725. Training with 0.1 initial learning rate gave lower result of 0.6183. It can be assumed that lower learning rate can be more beneficial as it allows to learn more optimal

parameters by taking small steps.

Previously discussed results referred to training only on a single label, however, it is also important to assess multi-label training results. The training was done on selected labels that are abundant: IT, CT, NE. While results of CT and NE are consistent with the results seen from training on only label CT, but still being lower due to the variety of labels to predict. Label IT has a drastically decreased dice score compared to others (Table 3.6) with all learning rates. To find the reasoning behind this low dice score for label IT in multi-label training, an experiment with only this label was performed. Interestingly, additional single-label model training revealed that indeed infiltrating tumor is a complex label to predict, resulting in a similar dice score to the one in multi-label training. The complexity of IT can be explained using medical knowledge about what it consists of. IT is healthy tissue that was infiltrated by cancerous cells; this infiltration can be sparse, hence, mostly providing normal brain tissue or higher, making it more close to cellular tumor.

One of the limiting factors of this work is the data and its credibility and complexity. The initial annotations were based off of a semi-automated approach with low accuracy as described in 3.1.1, and even then, not all labels were checked by professionals in the initial dataset. The boundary regions of the borders are difficult to annotate in any dataset, and even more so here, where the number and complexity of boundary definition is high, which causes issues even for clinical experts. Especially since some features are closely connected, for example, leading edge and infiltrating tumor, one defined by tumor to normal cells ratio $\sim 1-3/100$ and the latter $10-20/100$. Another example can be presence of blood vessels in another label, there might be some amount but not enough to assign it to corresponding label, hence the feature occupying most of selected region will be assigned. Ivy GAP annotations try to cover larger areas, while this work, and specifically usage of patches is mostly focused on getting detailed segmentation. Figure 3.9 shows several inference examples for different configurations alongside initial WSI and ground truth label. Annotation (Fig.3.9.B) does not depict much details, considering that it is made for an image with size of $\sim 18000 \times 15000$ pixels, but output masks look more detailed. It is worth noting that in all outputs had more CT label segmented that in real annotation. Most probable explanation lies in patch-based method of pipeline. The annotations might work better with downsampled images so the amount of context put into annotations can match data loss in WSI. Capturing more of WSI into a patch could assist the model in learning the complex data structures and predict features more precisely.

For further research, it would be useful to check predictions on each label in single and multi-label training experiments to investigate correlation patterns of the amount of data and dice score. As it was noticed, the infiltrating tumor had an unexpectedly low testing score despite being in the top three abundant labels for which there might be an explanation from understanding the medical basis on those features. Despite low metrics in some experiments which can

be explained by different possible factors, it is still worth to continue exploring segmentation techniques on the Ivy GAP histopathology dataset. These methods can provide new opportunities for tissue level data characterization in research, prognosis, and treatments by speeding up the process. In this work, it was clear that intricate histopathology data is hard to analyze, even with state-of-the-art models, and further experiments are required. Overall, results look promising and warrant further research to be applicable in clinical usage.

3.3.1 Future work

Although progress for the sake of progress should be avoided, this study has shown the application of semantic segmentation in the field of WSI provides fertile research opportunities. There are multiple prospects for this study to make it more extensive, but that falls out of the scope of the thesis. It would be worth investigating the effect of resizing WSI and data loss. Whole slide images are hard to process due to their large size, hence there is a need to extract patches which might not be beneficial in some cases as the network doesn't see the context of surrounding areas. Obviously, with a size decrease, there will be some data loss, but there is a possibility that resizing the initial WSI and then extracting the smaller amount of patches, each covering overall more data than in experiments without resizing, may enhance a model's performance. Additionally, further experiments on all labels, separately and simultaneously, are needed to garner deeper understanding of the problem. Depending on needed application, multi-label training can be done in a single-label style, meaning it is possible to consider several labels as one. This might be usable in cases when the area of specific labels needs to be known and now per label composition.

CAD is not routinely used by clinicians, but it has the potential to contribute towards potentially simplifying or altogether removing monotonous work, and improving diagnosis accuracy and repeatability. The clinical usefulness of this type of work can only be assessed after on-site deployment, which is an extremely challenging task. Even though sufficient performance might be the bottleneck holding back CAD, the speed and computation requirements also play a role. DL methods are well-known to require a high amount of computational footprint which "standard" machines cannot supply, thereby requiring machines powerful "DL-accelarator" cards (such as graphical or tensor processing units) to run the neural network inference. A lot of these requirements can be circumvented by post-training model optimization. For example, GaNDLF has an option for post-training optimization using OpenVINO Toolkit [90], which can allow a trained model to be inferred on laptop-grade central processing units, thereby allowing for integration in clinical workflows.

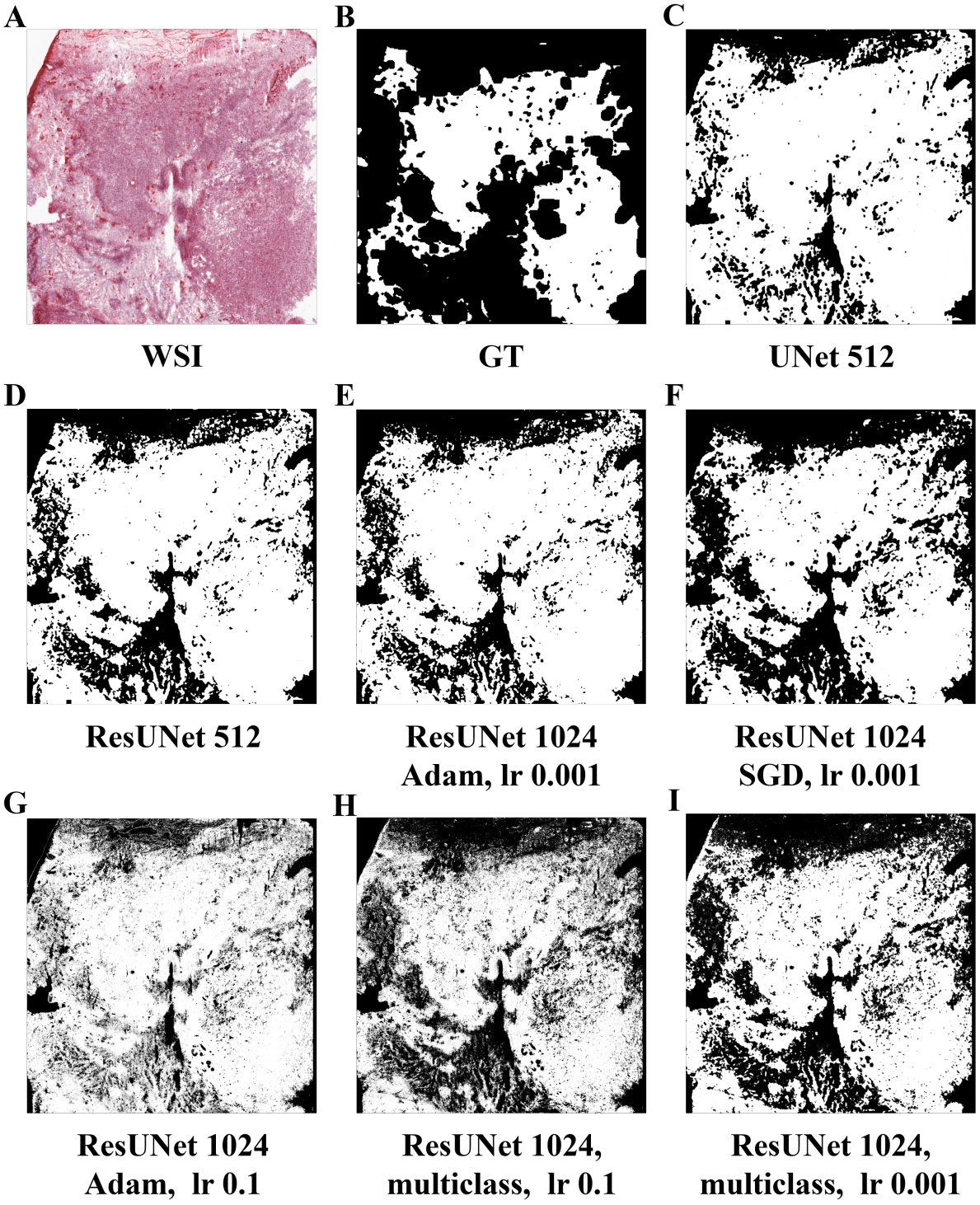

Figure 3.9: Inference result of different networks and configurations on WSI for CT label. **(A)** WSI. **(B)** Ground truth for CT label. **(C)** UNet trained on 512×512 patches. **(D)** ResUNet trained on 512×512 patches. **(E)** ResUNet trained on 1024×1024 patches with Adam optimizer and initial learning rate 0.001. **(F)** ResUNet trained on 1024×1024 patches with SGD optimizer and initial learning rate 0.001. **(G)** ResUNet trained on 1024×1024 patches with Adam optimizer and initial learning rate 0.1. **(H)** ResUNet trained on 1024×1024 patches with Adam optimizer and initial learning rate 0.1 for three labels, prediction is only for CT label. **(I)** ResUNet trained on 1024×1024 patches with Adam optimizer and initial learning rate 0.001 for three labels, prediction is only for CT label.

Summary

Glioblastoma is a malignant brain tumor with a low survival rate. It is highly heterogeneous and can infiltrate surrounding tissues. Analysis of its composition can help understand disease progression. Whole slide imaging has become an important part of a diagnostic routine, giving an understanding on tissue level. There are various features that either can support the characterization of GBM or get insights into a tumor's state and possible survival rate. Deep learning approaches can be used to get automated segmentation of these structural features in histopathological images.

In this work, convolution and transformer-based neural networks with encoder-decoder architecture were tested using the Ivy Gap histology images dataset, containing whole slide images of glioblastoma specimens. In a pre-processing step, images were split into smaller patches, so it is feasible to process them. Trying different networks and hyperparameters, it is possible to get an understanding that this GBM histopathology data is intricate and there should be more research done, possibly paying attention to the medical side as well. Tuning only the learning rate within other same conditions improves network performance. Patch size also has an effect on performance; bigger patches provide more image context, which is better for WSI analysis. However, manually tuning hyperparameters is time and computation power consuming, and for further research, the usage of automatic hyperparameter search can be done. Unfortunately, the dice score of the best-trained model is still low for it to be used in clinical applications. There are future modifications that can be done to improve this score, focusing on both data processing and network training.

Acknowledgments

I am grateful to my thesis supervisors Sarthak Pati for the guidance on how to make this work better, encouragement and even possibility to work on this project and Prof. Gholamreza Anbarjafari (Shahab) for valuable advice and feedback he has provided to me. Additionally, I would like to thank Ujjwal Baid who helped with his knowledge in Ivy GAP dataset and histopathological images despite not being an official supervisor. Their help and support made possible for me to work in the right direction. I am particularly grateful to Shahab for an opportunity of joining the iCV Research Lab during my studies, where was introduced to the field of Artificial Intelligence, learned about many interesting topics and gained various research skills. My time in the university would not be the same without this experience. I would also like to thank Assoc. Prof. Ilona Faustova and Artemi Maljavin for their input in my studies and moral support. Finally, I'm grateful for the support of my family and friends.

References

[1] Ziba Gandomkar, Patrick C Brennan, and Claudia Mello-Thoms. "Computer-based image analysis in breast pathology". In: *Journal of pathology informatics* 7 (2016).

[2] Mitko Veta et al. "Breast cancer histopathology image analysis: A review". In: *IEEE transactions on biomedical engineering* 61.5 (2014), pp. 1400–1411.

[3] Tahsin Kurc et al. "Segmentation and classification in digital pathology for glioma research: challenges and deep learning approaches". In: *Frontiers in neuroscience* 14 (2020), p. 27.

[4] Luke Oakden-Rayner. *The rebirth of CAD: how is modern AI different from the CAD we know?* 2019.

[5] Tao Zeng, Daming Cui, and Liang Gao. "Glioma: an overview of current classifications, characteristics, molecular biology and target therapies". In: *Front Biosci (Landmark Ed)* 20 (2015), pp. 1104–1115.

[6] Wojciech Szopa et al. "Diagnostic and therapeutic biomarkers in glioblastoma: current status and future perspectives". In: *BioMed research international* 2017 (2017).

[7] Lawrence F Marshall, Bryan Jennett, and Thomas W Langfitt. "Needle biopsy for the diagnosis of malignant glioma". In: *JAMA* 228.11 (1974), pp. 1417–1418.

[8] KH Fulling and JS Nelson. "Cerebral astrocytic neoplasms in the adult: contribution of histologic examination to the assessment of prognosis." In: *Seminars in Diagnostic Pathology*. Vol. 1. 2. 1984, pp. 152–163.

[9] Hans-Georg Wirsching and Michael Weller. "Glioblastoma". In: *Malignant Brain Tumors* (2017), pp. 265–288.

[10] Quinn T Ostrom et al. "CBTRUS statistical report: primary brain and central nervous system tumors diagnosed in the United States in 2008-2012". In: *Neuro-oncology* 17.suppl_4 (2015), pp. iv1–iv62.

[11] Spyridon Bakas et al. "Identifying the best machine learning algorithms for brain tumor segmentation, progression assessment, and overall survival prediction in the BRATS challenge". In: *arXiv preprint arXiv:1811.02629* (2018).

[12] Spyridon Bakas et al. “Overall survival prediction in glioblastoma patients using structural magnetic resonance imaging (MRI): advanced radiomic features may compensate for lack of advanced MRI modalities”. In: *Journal of Medical Imaging* 7.3 (2020), p. 031505.

[13] Anahita Fathi Kazerooni et al. “Cancer imaging phenomics via CaPTk: multi-institutional prediction of progression-free survival and pattern of recurrence in glioblastoma”. In: *JCO clinical cancer informatics* 4 (2020), pp. 234–244.

[14] Xiang Zhang et al. “Glioblastoma multiforme: Molecular characterization and current treatment strategy”. In: *Experimental and therapeutic medicine* 3.1 (2012), pp. 9–14.

[15] William A Freije et al. “Gene expression profiling of gliomas strongly predicts survival”. In: *Cancer research* 64.18 (2004), pp. 6503–6510.

[16] Roel GW Verhaak et al. “Integrated genomic analysis identifies clinically relevant subtypes of glioblastoma characterized by abnormalities in PDGFRA, IDH1, EGFR, and NF1”. In: *Cancer cell* 17.1 (2010), pp. 98–110.

[17] Peter C Burger and Robin T Vollmer. “Histologic factors of prognostic significance in the glioblastoma multiforme”. In: *Cancer* 46.5 (1980), pp. 1179–1186.

[18] Peter A Steck et al. “Functional and molecular analyses of 10q deletions in human gliomas”. In: *Genes, Chromosomes and Cancer* 24.2 (1999), pp. 135–143.

[19] Daniel J Brat and Erwin G Van Meir. “Vaso-occlusive and prothrombotic mechanisms associated with tumor hypoxia, necrosis, and accelerated growth in glioblastoma”. In: *Laboratory investigation* 84.4 (2004), pp. 397–405.

[20] Daniel J Brat. “Glioblastoma: biology, genetics, and behavior”. In: *American Society of Clinical Oncology Educational Book* 32.1 (2012), pp. 102–107.

[21] Fidan Seker-Polat et al. “Tumor Cell Infiltration into the Brain in Glioblastoma: From Mechanisms to Clinical Perspectives”. In: *Cancers* 14.2 (2022), p. 443.

[22] Navid Farahani, Anil V Parwani, and Liron Pantanowitz. “Whole slide imaging in pathology: advantages, limitations, and emerging perspectives”. In: *Pathol Lab Med Int* 7.23-33 (2015), p. 4321.

[23] Matthew G Hanna, Anil Parwani, and Sahussapont Joseph Sirintrapun. “Whole slide imaging: technology and applications”. In: *Advances in Anatomic Pathology* 27.4 (2020), pp. 251–259.

[24] Arthur L Samuel. “Some studies in machine learning using the game of checkers”. In: *IBM Journal of research and development* 3.3 (1959), pp. 210–229.

[25] Tom Mitchell. “Machine learning”. In: (1997).

[26] Eric J Topol. “High-performance medicine: the convergence of human and artificial intelligence”. In: *Nature medicine* 25.1 (2019), pp. 44–56.

[27] Eric Topol. *Deep medicine: how artificial intelligence can make healthcare human again*. Hachette UK, 2019.

[28] Konstantina Kourou et al. "Machine learning applications in cancer prognosis and prediction". In: *Computational and structural biotechnology journal* 13 (2015), pp. 8–17.

[29] Sarthak Pati et al. "Gandlf: A generally nuanced deep learning framework for scalable end-to-end clinical workflows in medical imaging". In: *arXiv preprint arXiv:2103.01006* (2021).

[30] Michael I Jordan and Tom M Mitchell. "Machine learning: Trends, perspectives, and prospects". In: *Science* 349.6245 (2015), pp. 255–260.

[31] Brian D. Ripley. *Pattern Recognition and Neural Networks*. Cambridge University Press, 1996.

[32] Kevin P Murphy. *Probabilistic Machine Learning: An introduction*. MIT Press, 2022.

[33] Christopher JC Burges. *Dimension reduction: A guided tour*. Now Publishers Inc, 2010.

[34] Glenn Fung. "A comprehensive overview of basic clustering algorithms". In: (2001).

[35] Marco A Wiering and Martijn Van Otterlo. "Reinforcement learning". In: *Adaptation, learning, and optimization* 12.3 (2012), p. 729.

[36] Lucian Buşoniu et al. "Reinforcement learning for control: Performance, stability, and deep approximators". In: *Annual Reviews in Control* 46 (2018), pp. 8–28.

[37] István Szita. "Reinforcement learning in games". In: *Reinforcement Learning*. Springer, 2012, pp. 539–577.

[38] *Machine learning types with examples*. `https://wordstream-files-prod.s3.amazonaws.com/s3fs-public/machine-learning.png`. Accessed: 2022-06-03.

[39] Keri Stephens. "Deep Learning Outperforms Standard Machine Learning in Biomedical Research Applications". In: *AXIS Imaging News* (2021).

[40] Chao Ning and Fengqi You. "Optimization under uncertainty in the era of big data and deep learning: When machine learning meets mathematical programming". In: *Computers & Chemical Engineering* 125 (2019), pp. 434–448.

[41] Najib J Majaj and Denis G Pelli. "Deep learning—Using machine learning to study biological vision". In: *Journal of vision* 18.13 (2018), pp. 2–2.

[42] Bjoern H Menze et al. "The multimodal brain tumor image segmentation benchmark (BRATS)". In: *IEEE transactions on medical imaging* 34.10 (2014), pp. 1993–2024.

[43] Oskar Maier et al. "ISLES 2015-A public evaluation benchmark for ischemic stroke lesion segmentation from multispectral MRI". In: *Medical image analysis* 35 (2017), pp. 250–269.

[44] Philipp Tschandl, Cliff Rosendahl, and Harald Kittler. “The HAM10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions”. In: *Scientific data* 5.1 (2018), pp. 1–9.

[45] Ping Li et al. “A Dataset of Pulmonary Lesions With Multiple-Level Attributes and Fine Contours”. In: *Frontiers in Digital Health* 2 (2021), p. 55.

[46] Adam Paszke et al. “Pytorch: An imperative style, high-performance deep learning library”. In: *Advances in neural information processing systems* 32 (2019).

[47] Martín Abadi et al. “Tensorflow: A system for large-scale machine learning”. In: *12th {USENIX} symposium on operating systems design and implementation ({OSDI} 16)*. 2016, pp. 265–283.

[48] Yangqing Jia et al. “Caffe: Convolutional architecture for fast feature embedding”. In: *Proceedings of the 22nd ACM international conference on Multimedia*. 2014, pp. 675–678.

[49] *Artificial neural network structure with input, hidden, and output layers*. `https://www.knime.com/sites/default/files/3-intro-deep-neural-networks.png`. Accessed: 2022-20-03.

[50] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep learning*. MIT press, 2016.

[51] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. “Learning representations by back-propagating errors”. In: *nature* 323.6088 (1986), pp. 533–536.

[52] Yann LeCun et al. “A theoretical framework for back-propagation”. In: *Proceedings of the 1988 connectionist models summer school*. Vol. 1. 1988, pp. 21–28.

[53] Alexander Kirillov et al. “Panoptic segmentation”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2019, pp. 9404–9413.

[54] Juan Eugenio Iglesias and Mert R Sabuncu. “Multi-atlas segmentation of biomedical images: a survey”. In: *Medical image analysis* 24.1 (2015), pp. 205–219.

[55] Leena Chandrashekar and A Sreedevi. “Advances in biomedical imaging and image fusion”. In: *International Journal of Computer Applications* 179.24 (2018), pp. 1–9.

[56] Dimitrios Bounias et al. “Interactive Machine Learning-Based Multi-Label Segmentation of Solid Tumors and Organs”. In: *Applied Sciences* 11.16 (2021), p. 7488.

[57] Paul A. Yushkevich, Yang Gao, and Guido Gerig. “ITK-SNAP: An interactive tool for semi-automatic segmentation of multi-modality biomedical images”. In: *2016 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)* (2016), pp. 3342–3345.

[58] Christos Davatzikos et al. “Cancer imaging phenomics toolkit: quantitative imaging analytics for precision diagnostics and predictive modeling of clinical outcome”. In: *Journal of medical imaging* 5.1 (2018), p. 011018.

[59] Michela Antonelli et al. “The medical segmentation decathlon”. In: *arXiv preprint arXiv:2106.05735* (2021).

[60] Sarthak Pati et al. “The cancer imaging phenomics toolkit (CaPTk): technical overview”. In: *International MICCAI Brainlesion Workshop*. Springer. 2019, pp. 380–394.

[61] Paul A. Yushkevich et al. “User-Guided 3D Active Contour Segmentation of Anatomical Structures: Significantly Improved Efficiency and Reliability”. In: *Neuroimage* 31.3 (2006), pp. 1116–1128.

[62] Spyridon Bakas et al. “Identifying the best machine learning algorithms for brain tumor segmentation, progression assessment, and overall survival prediction in the BRATS challenge”. In: *arXiv preprint arXiv:1811.02629* (2018).

[63] OS Pianykh. *Digital Imaging and Communications in Medicine (DICOM): A Practical Introduction and Survival Guide*. 2009.

[64] Yuxin Cui et al. “A deep learning algorithm for one-step contour aware nuclei segmentation of histopathology images”. In: *Medical & biological engineering & computing* 57.9 (2019), pp. 2027–2043.

[65] Peter Naylor et al. “Nuclei segmentation in histopathology images using deep neural networks”. In: *2017 IEEE 14th international symposium on biomedical imaging (ISBI 2017)*. IEEE. 2017, pp. 933–936.

[66] Ujjwal Baid et al. “Federated Learning for the Classification of Tumor Infiltrating Lymphocytes”. In: *arXiv preprint arXiv:2203.16622* (2022).

[67] Joel Saltz et al. “A containerized software system for generation, management, and exploration of features from whole slide tissue images”. In: *Cancer research* 77.21 (2017), e79–e82.

[68] Shidan Wang et al. “Pathology image analysis using segmentation deep learning algorithms”. In: *The American journal of pathology* 189.9 (2019), pp. 1686–1698.

[69] William K Funkhouser. “Pathology: the clinical description of human disease”. In: *Essential Concepts in Molecular Pathology*. Elsevier, 2020, pp. 177–190.

[70] Alessio D’Alessio et al. “Pathological and molecular features of glioblastoma and its peritumoral tissue”. In: *Cancers* 11.4 (2019), p. 469.

[71] Milan Decuyper et al. “Artificial intelligence with deep learning in nuclear medicine and radiology”. In: *EJNMMI physics* 8.1 (2021), pp. 1–46.

[72] Phillip M Cheng et al. “Deep Learning: An Update for Radiologists”. In: *RadioGraphics* 41.5 (2021), pp. 1427–1445.

[73] Guangming Zhu et al. “Applications of deep learning to neuro-imaging techniques”. In: *Frontiers in neurology* (2019), p. 869.

[74] Biswas Mainak et al. “State-of-the-art review on deep learning in medical imaging”. In: *Frontiers in Bioscience-Landmark* 24.3 (2019), pp. 380–406.

[75] Xiaoxuan Liu et al. “A comparison of deep learning performance against health-care professionals in detecting diseases from medical imaging: a systematic review and meta-analysis”. In: *The lancet digital health* 1.6 (2019), e271–e297.

[76] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. “U-net: Convolutional networks for biomedical image segmentation”. In: *International Conference on Medical image computing and computer-assisted intervention*. Springer International Publishing, 2015, pp. 234–241.

[77] Haris Iqbal. “Harisiqbal88/plotneuralnet v1. 0.0”. In: *URL: https://doi.org/10.5281/Zenodo* (2018).

[78] Anita Khanna et al. “A deep Residual U-Net convolutional neural network for automated lung segmentation in computed tomography images”. In: *Biocybernetics and Biomedical Engineering* 40.3 (2020), pp. 1314–1327.

[79] Zhengxin Zhang, Qingjie Liu, and Yunhong Wang. “Road extraction by deep residual u-net”. In: *IEEE Geoscience and Remote Sensing Letters* 15.5 (2018), pp. 749–753.

[80] Ashish Vaswani et al. “Attention is all you need”. In: *Advances in neural information processing systems*. 2017, pp. 5998–6008.

[81] Adrian MP Braşoveanu and Răzvan Andonie. “Visualizing transformers for nlp: a brief survey”. In: *2020 24th International Conference Information Visualisation (IV)*. IEEE. 2020, pp. 270–279.

[82] Jieneng Chen et al. “Transunet: Transformers make strong encoders for medical image segmentation”. In: *arXiv preprint arXiv:2102.04306* (2021).

[83] Alexey Dosovitskiy et al. “An image is worth 16x16 words: Transformers for image recognition at scale”. In: *arXiv preprint arXiv:2010.11929* (2020).

[84] Alex P Zijdenbos et al. “Morphometric analysis of white matter lesions in MR images: method and validation”. In: *IEEE transactions on medical imaging* 13.4 (1994), pp. 716–724.

[85] Ralph B. Puchalski et al. “An anatomic transcriptional atlas of human glioblastoma”. In: *Science* 360.6389 (2018), pp. 660–663.

[86] Melih Kandemir and Fred A Hamprecht. “Computer-aided diagnosis from weak supervision: A benchmarking study”. In: *Computerized medical imaging and graphics* 42 (2015), pp. 44–50.

[87] Teresa Araújo et al. “Classification of breast cancer histology images using convolutional neural networks”. In: *PloS one* 12.6 (2017), e0177544.

[88] Nobuyuki Otsu. “A threshold selection method from gray-level histograms”. In: *IEEE transactions on systems, man, and cybernetics* 9.1 (1979), pp. 62–66.

[89] Diederik P Kingma and Jimmy Ba. “Adam: A method for stochastic optimization”. In: *arXiv preprint arXiv:1412.6980* (2014).

[90] Yury Gorbachev et al. *Openvino deep learning workbench: Comprehensive analysis and tuning of neural networks inference*. 2019. DOI: 10.1109/ICCVW.2019.00104.

Appendix A

A Sample GaNDLF configuration

```
version:
  {
    minimum: 0.0.14,
    maximum: 0.0.14
  }
# Model parameters
model:
  {
    dimension: 2, # dataset dimension
    base_filters: 32, # number of filters in the initial model convolution;
    architecture: resunet,
    final_layer: sigmoid,
    class_list: [3], # list of labels the model should train on and predict
    n_channels: 3, # number input channels (3 - RGB)
  }
# metrics to evaluate the validation performance
metrics:
  - dice

modality: path

# Patch's settings
enable_padding: True
# Patch size during training
```

```yaml
patch_size: [1024, 1024]
# Set the stride size : needed only during inference.
stride_size: [256, 256]
# uniform: UniformSampler or label: LabelSampler
patch_sampler: uniform

# Number of epochs
num_epochs: 50
patience: 25
# Batch size
batch_size: 6
# Initial learning rate
learning_rate: 0.001
scheduler:
  { type: exp,
    gamma: 0.1 }
optimizer: adam #adam or sgd

# Loss function to use: 'dc' - dice loss
loss_function: dc
weighted_loss: False

nested_training:
  { testing: -8,
    validation: -8 }

# Histopathology slide level
slide_level: 0

# maximum number of patches that can be stored in the queue
q_max_length: 20
```

Non-exclusive licence to reproduce thesis and make thesis public

I, Ekaterina Sedykh,

1. grant the University of Tartu a free permit (non-exclusive licence) to: reproduce, for the purpose of preservation, including for adding to the DSpace digital archives until the expiry of the term of copyright, my thesis

   "Automated segmentation of various features of glioblastoma in histopathological images",

   supervised by Sarthak Pati and Prof. Gholamreza Anbarjafari.

2. I grant the University of Tartu the permit to make the thesis specified in point 1 available to the public via the web environment of the University of Tartu, including via the DSpace digital archives, under the Creative Commons licence CC BY NC ND 4.0, which allows, by giving appropriate credit to the author, to reproduce, distribute the work and communicate it to the public, and prohibits the creation of derivative works and any commercial use of the work from **27/05/2025** until the expiry of the term of copyright,

3. I am aware that the author retains the rights specified in points 1 and 2.

4. I confirm that granting the non-exclusive licence does not infringe other persons' intellectual property rights or rights arising from the personal data protection legislation.

Ekaterina Sedykh
27/05/2022