Linköping University | Department of Computer and Information Science
Master's thesis, 30 ECTS | Statistics and Machine Learning
2021 | LIU-IDA/LITH-EX-A--2021/001--SE

Classification of brain tumor types in weakly annotated histopathology images with deep learning

Dávid Hrabovszki

Supervisor : Anders Eklund
Examiner : Annika Tillander

Linköpings universitet
SE–581 83 Linköping
+46 13 28 10 00 , www.liu.se

Upphovsrätt

Copyright

Abstract

The abstract resides in file `Abstract.tex`. Here you should write a short summary of your work.

Acknowledgments

Acknowledgments.tex

Contents

List of Figures

List of Tables

1 Introduction

1.1 Motivation

It is estimated that around 300,000 new brain and nervous system cancer cases occurred in 2020 worldwide, and around 250,000 deaths occurred from this type of cancer in the same year [38]. The World Health Organization classifies tumors into grades based on their malignancy, where grade I is the least malignant and grade IV is the most malignant ([26]). Grade II and III cancers are called Lower Grade Gliomas (LGG), and grade IV cancers are called Glioblastoma or Glioblastome Multiforme (GBM) [28].

It is important to diagnose cancer types correctly, because treatment options and survival expectancy depend largely on how malignant a tumor is and what characteristics it has. There are histological differences between different types, which helps the expert pathologist in the decision making. Grade I lesions have the possibility of cure after surgery alone, grade II tumors are more infiltrative, can progress to higher grades, and often recur, and grade III is reserved for cancer that has some evidence of malignancy. The treatment of grade III lesions usually include radiation and chemotherapy. Grade IV tumors are malignant, active, necrosis-prone (death of the tissue), progress quickly and often cause fatality [25].

The branch of biology concerned with biological tissues is called histology, and its aim is to discover structures and patterns of cells and intercellular substances. Histologists examine tissue samples that have been removed from the living body through surgery or biopsy. These samples are processed and stained with chemical dyes to make the structures more visible. The samples are then cut into very thin slices that can be placed under an optical microscope and examined further [39].

Then it is the pathologists' job to determine the causes of disease [40], and histopathology connects the two fields by studying the diseases of tissues under a microscope. Histopathologists make diagnoses based on tissues and help clinicians in the decision making process. Specifically, they often provide diagnostic services for cancer, by reporting its malignancy, grade and possible treatment options [6].

With the advance of technology, it is now possible to scan, save, analyze and share tissue images using virtual microscopy. This technology scans a complete microscope slide and creates a single high resolution file called whole slide image (WSI). These files take up substantial storage and require specific software to view and manipulate them, because they are stored in special file formats [3].

In this thesis, whole slide images from The Cancer Genome Atlas (TCGA) are used [41], which is a publicly available dataset that contains tissues from GBM and LGG brain tumor types from many different clinics. The images are labeled as a whole, therefore no pixel-wise annotation is available. The files can be more than 3 GB in size, and their resolution is often higher than 100,000 x 50,000 pixels, therefore smaller sized patches need to be extracted that can be more easily analyzed by computers at a large magnification.

1.2 Aim

The aim of this thesis is to classify two different types of brain tumor (Glioblastoma and Lower Grade Glioma) from whole slide histology images using deep learning, where pixel-level annotation is unavailable. Specifically, convolutional neural networks (CNNs) will be used to classify individual patches from slides as GBM and LGG, and in a second step these patch predictions will be combined to a single prediction for each slide using different approaches.

This thesis intends to find the answer to the following research questions:

1. How well can a deep learning model classify whole slide images as Glioblastoma or Lower Grade Glioma without pixel-level annotation?
2. Is a pre-trained convolutional neural network significantly better than one trained from scratch?
3. Does image augmentation significantly improve classification accuracy?
4. What is the best approach for combining the patch level predictions to a slide level prediction?

The competing approaches are compared using statistical methods. There are several challenges regarding the research question, one of which is that the slide images are large in size, therefore they need to be divided into patches. They also come from different sources, so they must be normalized. The biggest challenge however, is that there is no annotation available on a pixel level, so it is possible that patches in a slide are cancer-free, or belong to the other class. The patches need to be combined to slide level in the prediction phase, which is not a straight-forward task. In this thesis, we experiment with five different slide aggregation techniques to tackle this problem.

1.3 Related work

Numerous researchers have applied computer algorithms for histology image analysis, and the used methods can be split into two groups. One requires the extraction of hand-crafted features that expert pathologists would recognize from the slide images, the other does not. The second group can afford to not explicitly obtain these features, because they use deep convolutional networks to automatically do that within the model. The first group therefore uses more traditional machine learning algorithms that accept well defined features. This thesis belongs in the second group, and the author aims to achieve higher accuracy scores with deep learning than the researchers did in the first group.

This is not an easy task, because some researchers achieved very good results without using deep learning. Barker et al. [2] extracted coarse features of shape, color and texture from the image patches focusing on cell nuclei, reduced the dimensionality of the features, and created clusters representing similar patterns. More detailed fine features about nuclear morphology were extracted from a select few patches from each cluster, then an Elastic Net was used on these patches to make the diagnostic decision of classifying a slide into GBM or LGG. Extracting fine features was a very computationally expensive task, this is why only

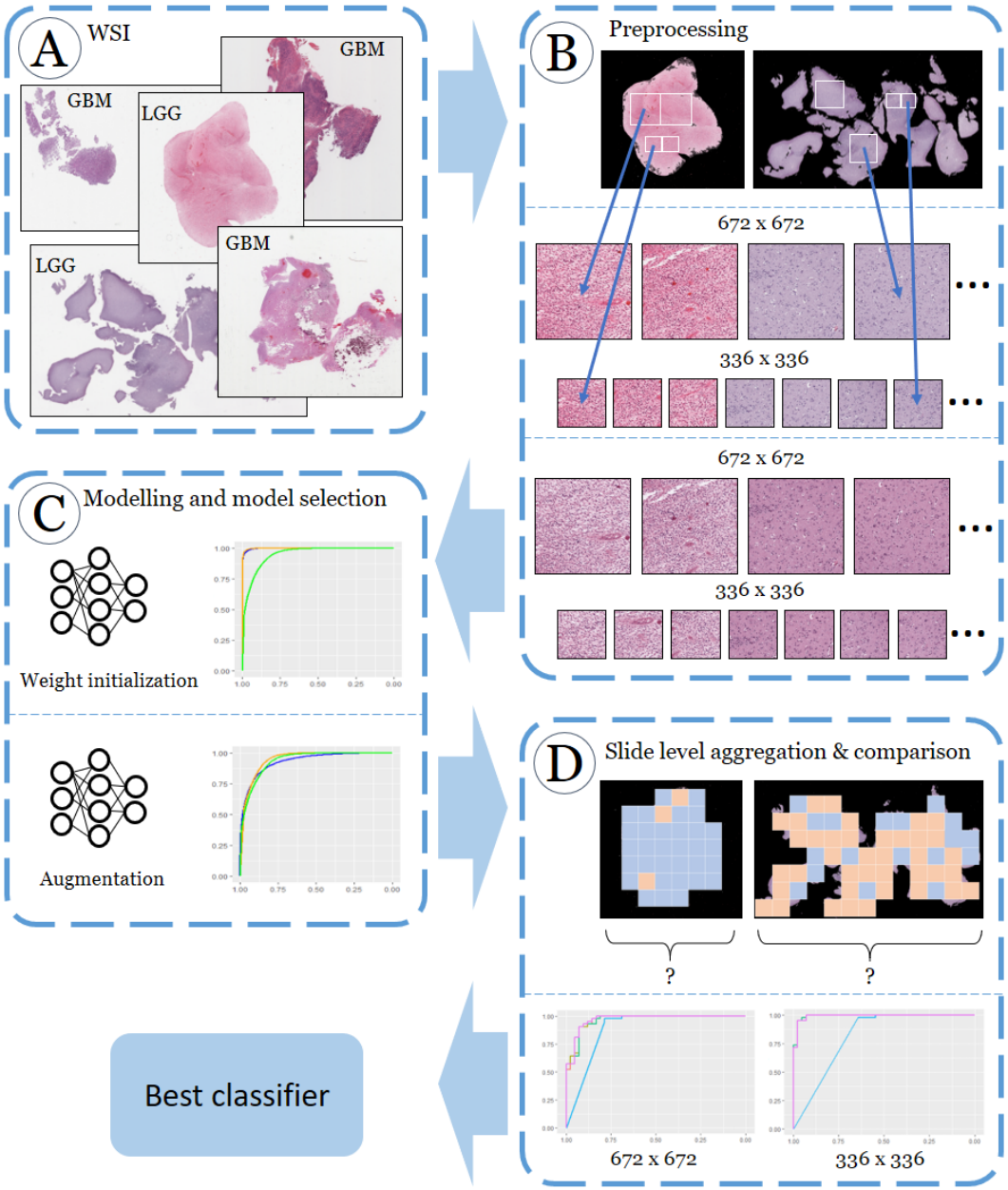


Figure 1: Workflow of this thesis

a smaller number of representative patches were a part of this step. Slide level aggregation was done by weighted voting of the predictions of the patches. All whole slide images came from the TCGA data repository, and were resized to 20x magnification level using bicubic interpolation, and the patches were 1024 x 1024 pixels. They achieved an accuracy of 93.1% on a dataset containing 302 brain cancer cases.

Rathore et al. [31] approached the problem similarly, by using a more traditional machine learning model, the support vector machine. They extracted features from the texture, intensity and morphology along with several clinical measures, and trained the SVM model on them. The creation of features required knowledge about what differentiates GBM from LGG, such as microvascular proliferation, mitotic activity and necrosis. The validation accuracy was 75.12% on images obtained from TCGA.

There is a lot more literature in this field that utilized the power of deep learning for image analysis. Kurc et al. [21] presented the three best performing methods from the 21st International Medical Image Computing and Computer Assisted Intervention (MICCAI 2018) conference for classification of oligodendroglioma and astrocytoma (which are two subclasses of LGG) patients. They all used a combination of radiographic and histologic image dataset, where the histologic images were obtained from TCGA, but they processed the two types of images separately. The three methods achieved accuracy scores of 90%, 80% and 75% respectively.

The best one, [1] applied several preprocessing steps on the images, including region of interest detection, stain normalization and patch extraction (224 x 224). They trained an autoencoder with convolutional layers to extract features from each patch, then used these features to identify patches that can potentially contain tumor regions using anomaly detection (Isolation Forest), where tumor was considered the anomaly. The DenseNet-161 network, pretrained on ImageNet, was then trained on these anomaly patches only, and the final prediction was done according to majority voting.

The second best approach, [27] argues that since only whole slide level annotation is available, but the training is done on patches, this is a weakly-supervised learning problem. To tackle this, they incorporated a Multiple Instance Learning (MIL) framework into the CNN architecture, which helps combine the patch predictions to slide level intelligently. The preprocessing steps were similar to [1], but they used 256 x 256 patches and a more simple histogram equalization for color normalization. Here a pretrained DenseNet-169 model was used with dropout regularization, which produced an average slide level score from all sampled patches for that slide. They concluded that the dropout technique did not improve the accuracy significantly.

The third best solution (only described in [21]), used a different approach. They identified tissue characteristics that differentiate the two classes, such as necrosis, cell density, cell shape and blood vessels. The images were then partitioned into 512 x 512 patches, and fed into a VGG16 CNN network with data augmentation to tackle class imbalance, and dropout and batch normalization layers to reduce overfitting.

A mixture of traditional machine learning and deep learning approaches exists, when the CNNs are only used for automatic feature extraction, but the classification is done by another machine learning algorithm. Xu et al. [47] used a pretrained AlexNet CNN for extracting 4096 features, some of which revealed biological insights, and then employed a SVM. They achieved 97.5% accuracy and concluded that these CNN features are considerably more powerful than expert-designed features.

Campanella et al. [8] conducted a very extensive research, where they used a deep learning approach to classify whether a slide image has cancer in it or not. They tested their methods on very large datasets of different types of cancer, and different slide preparation methods. The datasets were similar to TCGA in a way that they were also not labeled at pixel level, therefore the authors presented different Multiple Instance Learning approaches to tackle this weakly supervised problem in the form of slide aggregation models. These models included random forest and recurrent neural networks that were trained on the validation set to avoid overfitting. They showed by statistical comparisons that fully supervised learning models based on curated datasets do not generalize well to real world data, where detailed annotation is not available. Even though the authors did not use brain cancer data, some very useful findings and methods can be applied to the TCGA brain tumor dataset, including the statistical comparison of different models.

Other papers have experimented with deep learning for digital pathology, from which this research can benefit in questions such as optimal patch size, architecture, data augmentation methods, pre-processing steps, and slide level aggregation techniques [19], [13], [35], [20], [44], [18].

1.4 Background

There are several histologic features, such as mitotic activity and necrosis, that distinguish Lower Grade Glioma from Glioblastoma, which makes it possible for experts to make a diagnosis after inspecting the tissue sections under a microscope. Mitotic activity means the presence of dividing cells, and necrosis is the death of cells. The type of the cells also needs to be taken account, because although these two features indicate GBM in astrocytoma, they are still graded as LGG in oligodendrogliomas (astrocytoma and oligodendroglioma are two types of brain tumor emanating from different types of cells) [29].

Tissue samples removed from the body are colorless, therefore histologists soak them in hematoxylin and eosin dyes, which highlight different cellular details of the tissue, thus revealing important information. Cellular constituents that are basophilic (have an affinity for basic dye, like hematoxylin), such as the nucleus, are colored blue. Acidophilic (affinity for acid dye, like eosin) components are colored pink, like the cell membrane or the mitochondria. The process is called H&E staining, and it is the standard for histologic examination of tissues [9].

!!! How are histology slides prepared? !!!

2 Data

In this thesis we analyze microscopic histology images that are publicly available from The Cancer Genome Atlas (TCGA) [41]. There are two types of histology images, tissue slides and diagnostic slides. Tissue slides come from frozen samples, whereas diagnostic slides are results of samples being treated with formalin and paraffin to conserve their structure. In both cases, the samples are very thinly cut and put on a glass slide under the microscope. Frozen samples are more difficult to handle, however, and can more often have artifacts in them as a result of incorrect freezing. Therefore, it is more appropriate to use formalin-fixed paraffin-embedded (FFPE) tissues or diagnostic slides, as TCGA calls them, for histology analysis [36].

There are 860 examples of GBM and 844 examples of LGG available as diagnostic slides in TCGA, the whole dataset taking up 1.4 TB of storage space. The files are saved in a special format (svs) that allows for efficient storage of large images. This file format also makes it possible to obtain the slides at different magnification levels at which the microscope scanned them. The maximum magnification level is either 40x or 20x, and at this level a typical resolution would be 100,000 x 50,000 pixels, but this can vary largely. All LGG labelled images were obtained at 40x magnification (0.25 μm/pixel), while 77% of GBM images have only 20x magnification (0.5 μm/pixel) available. In order to analyze them together, all images need to be obtained at the same level, so those scanned at 40x are downscaled to 20x using bicubic interpolation.

Utilizing all of these images requires a massive cluster of computers, so this thesis focuses on a smaller subset of images that can be processed by the computer available in reasonable time. 220 whole slide images were sampled randomly and visually inspected for data quality issues. 26 images were excluded due to blur, or red, green, blue or black pen markings to make it easier to train the classifier. An example of each data quality issue can be seen on Figure 3. This adds bias to the estimated performance of the model, if it is later expected to classify all slides that contain both high and low quality images. It would have been possible to satisfactorily filter out black, blue and green pen markings, but not red, because its RGB color components can be exactly the same as the tissue on some of the more reddish slides, as well as blood cells. We decided not to make an exception with red color only, so we excluded slides with any colored pen markings.

We use 194 whole slide images, 97 from each class to create a classifier. 84 of them are used for training the model (43.3%), 26 for validation and parameter tuning (13.4%), and 84

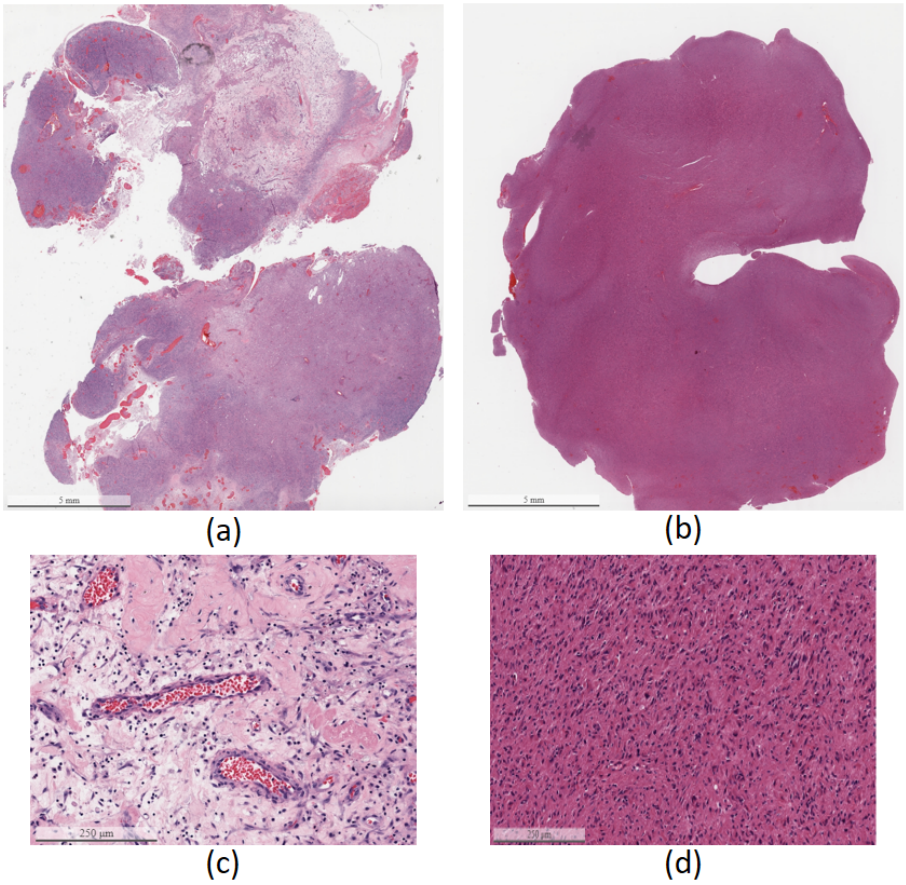


Figure 2: Glioblastoma Multiforme (GBM) and Lower Grade Glioma (LGG) whole slide images

for testing (43.3%). The relatively large ratio of the test set is necessary, because conclusions about the generalization error cannot be drawn from a small number of test examples.

Since the images are so large, it is impossible to process them as a whole without losing important morphological details, therefore patches or tiles are extracted from them, that are easier to handle for a neural network. We experiment with different patch sizes later on in this thesis, but every slide contains a few thousand patches on average, so we work with a large amount of data.

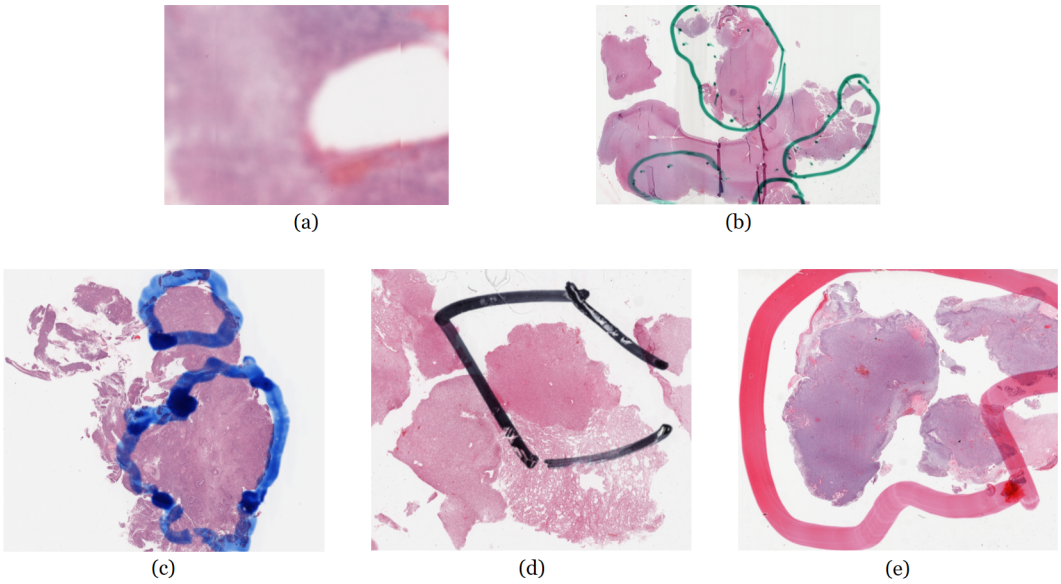


Figure 3: Examples of whole slide images excluded from the analysis

3 Theory

3.1 Deep learning

Deep learning allows artificial neural networks to learn complex representations of data by discovering structures and patterns automatically. In contrast with traditional machine learning methods, deep learning models do not require domain expertise in designing features, therefore very little engineering is necessary. Deep learning can be used for classification tasks very well, because the network is able to recognize features that are important for discrimination and will be robust to irrelevant variations. The learning is done in an iterative way by modifying the weights of the connections between the nodes in the network using the backpropagation algorithm. Each layer learns to recognize features on a higher level of abstraction then the previous one. Images are one type of data that can be analyzed very successfully with deep learning architectures, especially with convolutional neural networks (CNN), because they are much easier to train and generalize much better due to the fact that their adjacent layers are not connected fully [22].

3.1.1 Convolutional neural networks

The first paper that used convolutional networks trained by backpropagation for classifying hand-written digits was published in 1990 [23], but CNNs became increasingly popular with the arrival of fast graphics processing units (GPUs), and their ability to compute tasks in a massively parallel way making the training process much faster [30]).

Convolutional networks are superior to fully connected networks in image processing, because they are robust to geometric distortions, and the location of features does not matter too much. They also require far fewer images to train thanks to the lower number of connections inside the network [24].

3.1.2 Uncertainty estimation

It would be ideal to obtain the level of uncertainty of predictions of a neural network, because then we can decide if we can rely on its predictions or not. It could be especially useful in the case of GBM versus LGG classification, because the network is trained on patch level, but there are only slide level labels, so we expect the network to be uncertain about some patches.

There are also patches that contain neither of the classes (healthy tissue), but still inherit the label from the slide level, which makes it even more difficult for the neural network to confidently predict every instance.

The last layer of a network is usually a softmax layer that outputs values between 0 and 1 for the possible classes, but it would be a mistake to interpret these as the confidence of the model about the current prediction, because it is still possible that a model is uncertain despite a high softmax output [15].

One option is to use Bayesian Neural Networks, where we place prior probability distributions over all the weights, and as more and more evidence (data) is introduced, we obtain the posterior distributions for each and every weight in the neural network according to Bayes' theorem. The prior distribution is usually very general and wide, like the Gaussian distribution, but the posterior is narrower, since we have more information about that specific weight at the end of the training. The weights are still not point estimates, but probability distributions. This is the fundamental difference between Bayesian and traditional neural networks, because usually we perform mathematical optimization by the backpropagation algorithm that has the goal of minimizing the loss function and results in Maximum Likelihood Estimates of the weights [4].

This Bayesian approach is a mathematically sound framework, but it comes with such a large computational cost, that it renders it unfeasible for deep networks with large number of weights. An alternative to using Bayesian Networks is to use the traditional approach, but try to approximate Bayesian inference. It is possible to approximate Bayesian networks using variational inference, but it is still too computationally expensive, because they double the number of parameters in the models compared to a network with the same size. Another route is probabilistic backpropagation, but there is a simpler and better performing method that can be used with very little modification of the traditional neural network [15].

Gal and Ghahramani [15] showed that a neural network with dropout layers active in test time is a mathematically good approximation of the probabilistic deep Gaussian process marginalized over its covariance function parameters. This is because by repeatedly dropping out weights at random, we approximate integration over them. This approach is called Monte Carlo Dropout, and it can be used in any neural network that employs random dropout layers before its weight layers. In CNNs, that might only have one dropout layer before the last fully connected layer, it is still possible to obtain uncertainty estimates from this one dropout. Monte Carlo Dropout requires a relatively small number of forward passes (10-100) to create the predictive distribution of each test instance, then uncertainty can be deduced from it.

!!! Write about disadvantages of Dropout !!!

3.2 Logistic regression

Logistic regression is a statistical model for classification that uses the logistic sigmoid function

$$\sigma(a) = \frac{1}{1 + exp(-a)}$$

to output the probability of an instance belonging to a class $\mathcal{C}_1$ by applying the sigmoid function on the linear combination of its feature vector ϕ and some weights w [5]:

$$p(\mathcal{C}_1|\phi) = y(\phi) = \sigma(w^T\phi)$$

In binary classification the other class is given by:

$$p(\mathcal{C}_2|\phi) = 1 - p(\mathcal{C}_1|\phi)$$

If the feature vector ϕ has M dimensions, the model also has M adjustable parameters. We find the best fitting model by optimizing these parameters using maximum likelihood. For a dataset $\{\phi_n, t_n\}$, where $t_n \in \{0, 1\}$ and $\phi_n = \phi(x_n)$, $n = 1, ..., N$, the likelihood function is:

$$p(\mathbf{t}|w) = \prod_{n=1}^{N} y_n^{t_n} \{1 - y_n\}^{1-t_n},$$

where $\mathbf{t} = (t_1, ..., t_N)^T$ and $y_n = p(\mathcal{C}_1|\phi_n)$. To define an error function that we want to minimize, we form the negative logarithm of the likelihood function, which gives the cross-entropy error function:

$$E(w) = -\ln p(\mathbf{t}|w) = -\sum_{n=1}^{N} \{t_n \ln y_n + (1 - t_n) \ln(1 - y_n)\},$$

where $y_n = \sigma(a_n)$ and $a_n = w^T \phi_n$. When we want to minimize this function by changing its parameters, we need to calculate the gradient of the function with respect to the parameters w:

$$\nabla E(w) = \sum_{n=1}^{N} (y_n - t_n)\phi_n.$$

This does not lead to a closed-form solution, however, so optimization has to be done in an iterative way. The error function is a concave function of w, therefore it has a unique minimum, which we can reach with a suitable optimization technique [5].

3.3 DeLong's test for correlated ROC curves

When one performs multiple tests on the same examples and constructs the empirical receiver operating characteristic (ROC) curves, the correlation of the data must be taken into account when conducting statistical comparisons [10]. DeLong et al. [10] presented a nonparametric approach for comparing areas under the curve (AUC) of correlated ROC curves, which we describe below.

Let us assume a binary classification problem in which we have N examples, and class 1 denotes the presence of a disease (for example GBM in our case). Some of the N examples actually have the disease (m), and others do not ($n = N - m$). Let C_1 denote the first group and C_2 the second. Suppose that we have a binary classifier that predicts the probability of the disease being present in an example, and let these probabilities be denoted by X_i, $i = 1, 2, ..., m$ and Y_j, $j = 1, 2, ..., n$ for members of C_1 and C_2 respectively.

1. First, we need to calculate the empirical AUCs by the trapezoidal rule.

$$\hat{\theta} = \frac{1}{mn} \sum_{j=1}^{n} \sum_{i=1}^{m} \psi(X_i, Y_j),$$

where

$$\psi(X, Y) = \begin{cases} 1, & Y < X \\ \frac{1}{2}, & Y = X \\ 0, & Y > X \end{cases},$$

which means intuitively that the AUC increases by $\frac{1}{mn}$ if the predicted disease probability of a member of C_2 is less than that of a member of C_1, and it increases by $\frac{0.5}{mn}$

if they are the same. The AUC does not increase if the predicted disease probability of a member of C_1 is less than that of a member of C_2, because this surely results in misclassification.

The estimate $\hat{\theta}$ has been shown to be equal to the Mann-Whitney statistic, which is a generalized U-statistic, therefore asymptotic normality and an expression for the variance can be derived from the theory for generalized U-statistics (the theory itself is not discussed in this thesis).

If we have k different models to compare, we obtain a vector of AUC estimates $\hat{\boldsymbol{\theta}} = (\hat{\theta}^1, \hat{\theta}^2, ..., \hat{\theta}^k)$ of the true AUCs $\boldsymbol{\theta} = (\theta^1, \theta^2, ..., \theta^k)$.

2. For the AUC estimate (U-statistic) $\hat{\theta}^r$, where $1 \leqslant r \leqslant k$, we can calculate structural X and Y components in the following way:

$$V_{10}^r(X_i) = \frac{1}{n}\sum_{j=1}^{n}\psi(X_i^r, Y_j^r), (i = 1, 2, ..., m)$$

$$V_{01}^r(Y_j) = \frac{1}{m}\sum_{i=1}^{m}\psi(X_i^r, Y_j^r), (j = 1, 2, ..., n).$$

3. Now we define the $\boldsymbol{S_{10}}$ and $\boldsymbol{S_{01}}$ $k \times k$ matrices in a way that their (r, s)th element is:

$$S_{10}^{r,s} = \frac{1}{m-1}\sum_{i=1}^{m}[V_{10}^r(X_i) - \hat{\theta}^r][V_{10}^s(X_i) - \hat{\theta}^s]$$

$$S_{01}^{r,s} = \frac{1}{n-1}\sum_{j=1}^{n}[V_{01}^r(Y_j) - \hat{\theta}^r][V_{01}^s(Y_j) - \hat{\theta}^s].$$

4. We combine $\boldsymbol{S_{10}}$ and $\boldsymbol{S_{01}}$ to get the estimated covariance matrix of the parameter estimates $\hat{\boldsymbol{\theta}} = (\hat{\theta}^1, \hat{\theta}^2, ..., \hat{\theta}^k)$:

$$\boldsymbol{S} = \frac{1}{m}\boldsymbol{S_{10}} + \frac{1}{n}\boldsymbol{S_{01}}.$$

5. Using the asymptotic theory for U-statistics,

$$\frac{\boldsymbol{L}\hat{\boldsymbol{\theta}}^T - \boldsymbol{L}\boldsymbol{\theta}^T}{\sqrt{\boldsymbol{L}(\frac{1}{m}\boldsymbol{S_{10}} + \frac{1}{n}\boldsymbol{S_{01}})\boldsymbol{L}^T}}$$

has a standard normal distribution, where $\boldsymbol{L}$ is a row vector of coefficients. If we only want to compare two AUCs for difference, we can set $\boldsymbol{L} = [1 - 1]$, in which case the null hypothesis is that there is no statistically significant difference, so $\boldsymbol{L}\boldsymbol{\theta}^T = 0$.

$$H_0 : \hat{\theta}^1 = \hat{\theta}^2$$

The previous expression becomes

$$\frac{\hat{\theta}^1 - \hat{\theta}^2}{\sqrt{\boldsymbol{L}(\frac{1}{m}\boldsymbol{S_{10}} + \frac{1}{n}\boldsymbol{S_{01}})\boldsymbol{L}^T}},$$

or, according to [37],

$$\frac{\hat{\theta}^1 - \hat{\theta}^2}{\sqrt{\mathbb{V}[\hat{\theta}^1 - \hat{\theta}^2]}} = \frac{\hat{\theta}^1 - \hat{\theta}^2}{\sqrt{\mathbb{V}[\hat{\theta}^1] + \mathbb{V}[\hat{\theta}^2] - 2\mathbb{C}[\hat{\theta}^1, \hat{\theta}^2]}} = z,$$

where the variances of the AUC estimates are the diagonal values of $\boldsymbol{S}$ and the covariance is the off-diagonal value.

6. With the z score obtained above, we can perform a two-tailed test to determine whether we can or cannot reject the null hypothesis at a given significance level. The alternative hypothesis is that there is a statistically significant difference between the two AUC estimates $\hat{\theta^1}$ and $\hat{\theta^2}$.

4 Methods

4.1 Preprocessing

Whole slide images cannot be directly processed due to their large size, therefore certain preprocessing steps need to be made. Ignoring the glass background of the scans and accounting for the variability of the stain colors across the dataset also need to be addressed.

!!! Insert image of preprocessing pipeline !!!

4.1.1 Filtering

WSIs generally contain at least 50% white background, which needs to be filtered out, because it carries no useful information. We experimented with different methods, and chose a simple approach that looks at each pixel's green channel value, and filters out those that are above the intensity threshold of 200. If the image is particularly bright, the resulting binary mask can easily cover more than 90% of the image. In this case, the threshold is automatically raised, until the mask cover rate drops below 90%. For this task, the *deep-histopath* package from IBM CODAIT was used [12]. This approach can be used for H&E-stained histology datasets, because tissue is colored pink or purple, which have very low green components, in contrast with the white background.

The goal of background filtering is to remove the white pixels surrounding the tissue, but it is important to note that the methods tried often filtered out pixels inside the tissue area that had a brighter color. To prevent this, another filter was used from the *scikit-image* Python package [43] to remove these small holes resulting from the background filter.

To apply the above mentioned filters on WSIs of such large sizes is very computationally expensive, therefore they were applied on their thumbnail versions instead, and the resulting binary mask was upscaled to the full resolution. The thumbnails are the lowest resolution versions of the full sized images, and are typically around 3,000 x 1,500 pixels, so they are the equivalent of downscaling the full image by a factor of either 16x or 32x, depending on the slide.

4.1.2 Patching

As mentioned earlier, WSIs are too large to be analyzed by neural networks as a whole, therefore small patches need to be extracted from them. One logical approach is to select a size on which trained pathologists can form diagnosis about the tumor type, which is 1024 x 1024 pixels at 20x magnification ($0.262mm^2$) [2]. Convolutional neural networks that were trained on the ImageNet dataset however, usually expect images of size 224 x 224 as inputs, so our patches will need to be downscaled to this size for modelling. In the case of the recommended patch size, this would mean a 21x reduction of area, and a significant information loss, therefore the choice of a smaller patch size is more reasonable.

In [47], the highest accuracy was achieved when using 336 x 336 patches at 20x magnification. The authors of that paper experimented with 2 smaller patch sizes as well, but they performed worse. Building on these findings, we try out two patch sizes in this thesis, small (336 x 336) and large (672 x 672), where small and large patches have the area of $0.028mm^2$ and $0.113mm^2$ respectively.

The patches were created using the *patchify* Python package [45] without overlap. We extracted 745,691 patches of the small size, and 169,308 patches of the large size in total.

4.1.3 Stain normalization

The WSIs in the TCGA dataset were all stained with hematoxylin and eosin compounds that give different colors to different histological structures, but they were prepared at different clinics, where the exact practices might not match completely. Slide staining is prone to environmental conditions, and the final result is influenced by the duration of staining, pH balance, temperature, and other conditions [16]. All this introduces variations in the stain colors, that otherwise have no importance on distinguishing between GBM and LGG classes. In fact, it might make it harder for the neural network to learn the features that set the classes apart, if the stain colors are not normalized.

Even though stain normalization is considered an essential preprocessing step [16], not everyone follows this approach. Hou et al. [18] applies no stain normalization techniques, but randomly adjusts the amount of the two stains as a data augmentation step, thus making the model more robust.

In this paper, the Vahadane algorithm [42] is used for stain normalization of patches, which is widely used, because it preserves biological structures well. Roy at al. [33] conducted quantitative and qualitative comparisons of the state of the art color normalization methods for histopathology images, and concluded that Vahadane's approach provided the best results in four different cancer datasets. The algorithm requires a target image to which all source images are normalized with no color distortion. The source images are decomposed into stain density maps that record the concentration levels of both stain colors, which hold important information about the biological structures. Then these density maps are combined with the stain color basis of the target image, this way only the colors are changed, their intensity (biological structure) remains the same.

The algorithm is implemented in Python in the *StainTools* package [7]. Every patch is normalized individually, because it is recommended to first remove the white pixels of background, since they are not only composed of the two basis stain colors. When attempting to normalize WSIs prior to filtering and patching, certain artifacts were visible in the normalized version, which is why stain normalization takes its place at the end of the preprocessing pipeline. It is a very computationally heavy step that takes roughly 1 second for each patch.

4.2 Modelling

4.2.1 Training

We opted to use a ResNet50 model, similarly to [8], because it has already proven its capabilities in medical image processing. Residual networks were introduced in [17], and different versions of it exist, but here the shallowest architecture is used that is implemented in Keras.

Residual nets were a breakthrough in image classification, because they solved the degradation problem, which we experience, when networks perform worse as they get deeper after a certain point. This went against the general idea that deeper networks should produce no higher training error than a shallower version of the same architecture. The assumption is that if we add more layers, even if they do not add anything to the performance, at least they should pass on the learned information from the previous layers, acting as simple identity mapping. The fact that this did not happen indicates that the solvers have difficulty approximating identity mappings with nonlinear layers, therefore it was suggested that shortcuts are created between small blocks of nonlinear layers where the information can pass more easily, if that is what is needed. ResNets also allow deep architectures to have lower complexity than previous networks with fewer layers [17]. ResNet50 building block Figure 4.

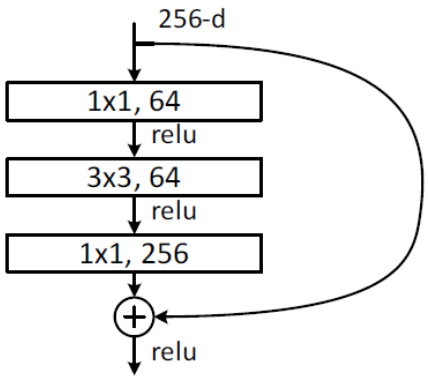


Figure 4: Building block for ResNet50 [17] © 2016 IEEE

In this thesis we investigate whether a pretrained network has an advantage over a CNN trained from scratch, so we will compare the predictive performances of both. We also examine if data augmentation has a significance during training the models, therefore the final CNN will be chosen after careful consideration while answering these questions. The competing methods are trained on the dataset consisting of the smaller size patches.

A pretrained network is a network initialized with weights pretrained on the ImageNet dataset, which means that we can expect the model to perform relatively well in the early stages of training. ImageNet contains 1000 image classes, so the last layer has 1000 units, therefore we have to create our own top layers to tailor the architecture for the GBM versus LGG binary classification problem (Figure 5). Specifically, an average pooling layer, a flatten layer, and two fully connected layers with dropout are added. The first dense layer has 100 nodes and ReLu activation, while the second one has 2 output nodes with softmax function, each corresponding to one of the classes. The dropout layers have a probability of 50% for randomly dropping connections, and they help with reducing overfitting by regularisation, while allowing for Monte Carlo Dropout, since they are also activated in test time. This way network's predictions are not deterministic, the output is different every time a forward pass is run on a test image, and they form a distribution from which uncertainty can be deduced.

The ResNet50 architecture consists of five convolutional blocks that each include 1,3,4,6, and 3 smaller blocks, visualised in Figure 5. With a colored input image of size 224 x 224 pixels, the first block outputs 64 convolutional filters of size 112 x 112 pixels. The number of

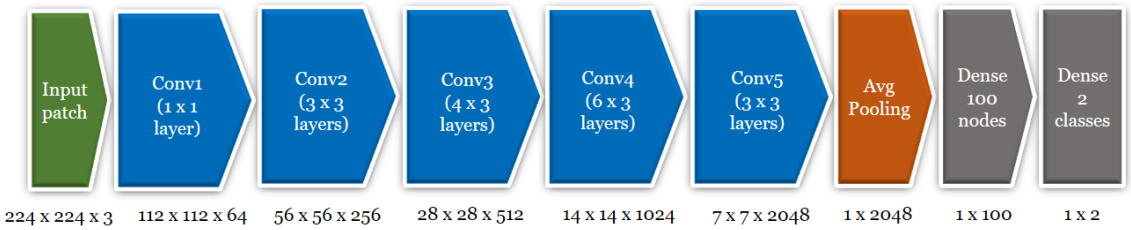


Figure 5: ResNet50-based modified architecture (with output dimensions)

filters grows as the depth increases, in the end, the network extracts 2048 features of size 7 x 7 pixels. The building block described in Figure 4 is one of the three smaller blocks in Conv2 here.

Data augmentation is a method for creating more training examples from the existing ones, so the model does not see the same images over and over again. Augmentation techniques include mirroring, rotating, shearing, cropping, and color jitter, among others. This helps reduce overfitting, which happens when the model learns the training examples too well, and cannot generalize to previously unseen data. We will experiment with different techniques to determine which ones work best in our specific case with histology images.

The idea of initializing the network's weights that were trained on a dataset, and then training the network again on a new dataset is called transfer learning (!!! cite properly https://keras.io/guides/transfer_learning/?fbclid=IwAR2sT2PIA2RUWP1UHuW7HwQII4nRpJIGEi0Z4itj2V0dh !!!), and its advantage is that we can leverage the already learned convolutional filters that recognise basic shapes and forms, thus making the fine-tuning fast and require less data. First, the added custom layers that implement the classification need to be trained until convergence, while the rest of the network is frozen, to avoid destroying the pre-trained weights. After this is achieved, the last two block of the ResNet50 model gets unfrozen and fine-tuned along with the added layers at the top. This way, the last blocks can learn filters specific to the tumor classification problem, while still building on the knowledge of the first three blocks that capture simpler patterns and shapes. Fine-tuning is done with a smaller learning rate to avoid completely overwriting the pre-trained weights.

The following three models are compared to determine whether pre-training gives an advantage:

- *Random weight initialization*: weights are initialized randomly, and the model is trained for 40 epochs with a learning rate of 10^{-6}.
- *Pre-trained with ImageNet*: weights trained on the ImageNet dataset are loaded at the beginning. The added layers are trained first for 10 epochs with 10^{-4} learning rate to allow them to converge, while the feature extraction part of the network is frozen. This is done to avoid destroying the pre-trained features by the large gradient updates the top layers must go through (!!! cite https://keras.io/guides/transfer_learning !!!). After convergence, all the layers get unfrozen, and we train the whole model for another 10 epochs with a lower learning rate (10^{-5}). The learning rate is lower to avoid changing the potentially valuable weights too much.
- *Pre-trained with ImageNet and fine-tuned*: Identical to the previous approach, except that not all the convolutional layers get trained in the second training phase. We keep the first three convolutional blocks (Conv1, Conv2, and Conv3 in Figure 5) frozen. The idea behind this approach is that since the first blocks of the network are pre-trained to recognize basic features and shapes, it might be useful to keep them as they are. The later blocks are responsible for more complex patterns, where the weights learned

from ImageNet might be less useful, since our histology dataset is quite different from ImageNet.

All three models apply the same data augmentation methods, which are horizontal and vertical mirroring.

After training these three models, we compare their predictive performances by ROC-curve analysis described in section 3.3. We run 30 forward passes on the whole of the validation set to obtain the Monte Carlo samples, and then construct the ROC-curves on patch level. Statistical significance of their difference is then examined, and the best one is chosen for further analysis.

After it has been established if it is beneficial to use pre-trained weights or not, we investigate the effects of data augmentation on the training process and the predictive performance. The following three models compete against each other.

- *No data augmentation*:
- *Mirroring*:
- *Mirroring, rotating*:

All models are optimized with the Adam stochastic gradient descent algorithm, and the binary cross-entropy loss function is used. The batch size is 64 in all cases, which is necessary to be able to fit all the information in the GPU memory. In every epoch, all the training images are used for weight update, and all the validation images are used for measuring the generalization accuracy.

4.3 Slide aggregation

The convolutional neural network described above predicts a class label for every single patch, but the aim of this thesis is to classify whole slide images, which can be interpreted as bags of patches, usually a few thousands of them. In the optimal scenario, ground-truth would be available on pixel level, that way we could be sure that the true labels of the patches are correct in all cases. It would require tremendous effort from experts to annotate on such low granularity, however, so this approach is generally not feasible in real life.

When obtaining fully ground-truth labels is impossible, machine learning methods work with weak supervision, instead of the more traditional strongly supervised learning. One branch of weakly supervised learning is inexact supervision, where only coarse-grained labels are available [48]. The general term for entities for which we know the labels is *bags*, and for smaller objects belonging to a bag is *instances*. This is also called *multiple-instance learning*, where the goal is to predict labels of new bags ([48], [11]).

Formally, the goal is to learn a function $f : \mathcal{X} \mapsto \mathcal{Y}$ given a training dataset $D = (X_1, y_1), ..., (X_m, y_m)$, where $X_i = \{x_{i,1}, ..., x_{i,m_i}\} \subseteq \mathcal{X}$ is a bag, $x_{i,j} \in \mathcal{X} (j \in \{1, ..., m_i\})$ is an instance, m_i is the number of instances in X_i and $y_i \in \mathcal{Y} = \{Y, N\}$, Y and N being the two possible classes [48].

To learn such a function, we must find a way to combine the patch-level predictions to slide-level predictions. In the next subsections we present five slide aggregation techniques, the performance of which we will compare later on in this thesis. At this stage in the analysis, we have obtained T soft predictions from the CNN model, where T is the number of Monte Carlo samples (stochastic forward passes), for every patch in our dataset.

Uncertainty is measured as the standard deviation of the T Monte Carlo samples multiplied by 2, to stretch the interval to between 0 and 1, making interpretation easier. Obtaining the *certainty estimates* or *weights* on a scale of 0 to 1 is a simple linear mapping afterwards.

It is important to note that since GBM is a more malignant type of tumor than LGG, expert pathologists classify a slide as Glioblastoma if any area in the slide can be diagnosed as such.

Keeping this in mind, it might be straightforward to simply predict GBM for every slide that has at least one GBM predicted patch in it. This would, in fact, be the optimal solution, if we could assume that each training patch had the correct ground-truth label assigned to it. Since the training slides do not have annotations on such low granularity, we cannot make this assumption. As mentioned earlier, it is entirely possible (and expected) that some patches have incorrect true labels when training, because it is possible that a patch only contains Lower Grade Glioma in a slide that is otherwise diagnosed as Glioblastoma, or it might just be a tumor-free area of the tissue. Expecting the CNN to correctly classify every patch under these circumstances is unrealistic, therefore its predictions should not be taken for granted. The layer of aggregation could be a way to smooth out the model's errors and provide a more robust classification pipeline, at the expense of abandoning the pathologists' way of diagnosis. The reasoning is that it is more important to arrive at the correct solution at the end (correct classification of test slides), than to use the same method as the experts. The other option would be to ask them to manually annotate slides at a much lower level, but this is an arduous task that takes too much time and resources.

4.3.1 Majority voting

The simplest method is using majority voting. In this case, the uncertainty of the patch predictions is ignored, and the predicted class of every patch is determined by the mean of the Monte Carlo samples. The probability of a slide belonging to the GBM class is the ratio of the patches classified as GBM. The probability is larger than 0.5 if more than half of the patches are predicted GBM.

4.3.2 Logistic regression

We would like to learn a function that maps patch predictions to slide predictions, so implementing a second-level classifier model after the CNN is a reasonable step to take.

Campanella et al. [8] conducted a large-scale experiment, in which they experimented with random forests and Recurrent Neural Networks (RNNs) as second-level classifiers (the RNN was trained on the extracted features of the first-level CNN, not its class predictions) for various types of tumor classification problems.

Hou et al. [18] compared 14 aggregation methods, among which were logistic regression and support vector machine models trained on the output of a CNN for multi-class glioma classification. Two setups of the logistic regression variant achieved the two highest accuracy scores, so it seems promising to implement it in this study, as well.

The logistic regression is learned on the validation set after the training is done to avoid over-fitting. The features are derived from the CNN predictions, taking into account their uncertainty estimates or rather, their weights, as defined previously. Statsmodels' Python implementation [34] was used, and the following features were included for fitting the model:

- Ratio of patches predicted GBM with at least 95% certainty
- Ratio of patches predicted LGG with at least 95% certainty

4.3.3 Spatial smoothing

We assume that one tumor type does not occur sparsely all over the tissue, but within well-defined borders that takes up a larger area. It is unlikely that inside an area of Glioblastoma, Lower Grade Glioma spots are present (and vice versa), therefore, as a post-processing step we apply spatial smoothing on the slides to create these larger areas of homogeneous tumor. This is achieved by covering the slide in non-overlapping windows of the same size, and assessing each window independently in terms of the more likely tumor type for that area.

In this approach, multiple criteria need to be fulfilled in order to flip the class prediction of some patches in a window. We use a window size of 9x9 patches with possible missing patches treated as unknowns, where the majority class has a chance of overwriting the labels of the minority class. Minority patches are only flipped, if at least one patch from the majority class has a higher certainty (weight) than the threshold, and all patches from the minority class have lower certainties than the threshold. This threshold value is considered a hyperparameter of the method, therefore it is optimized on the validation set with grid search. The algorithm uses majority voting at the end, and outputs the probability of GBM for every slide.

With this method, we hope to smooth out some of the patch prediction errors made by the CNN by taking into account their spatial location, and examining what label is most likely for them given this information.

!!! Insert illustration of flipping !!!

4.3.4 Standard MIL assumption

The most common method in the multiple instance learning framework was introduced in [11], and is generally referred to as the standard assumption. This states that a bag is considered positive (in our case GBM), if and only if at least one of its instances is positive.

Given that our CNN patch-level predictions are not expected to be completely accurate, this assumption might not work well in this case. Uncertainty is not taken into account here, the patch predictions are simply the means of the Monte Carlo samples. The probability of a slide belonging to the GBM class is equal the highest probability among all of the patches in that slide.

4.3.5 Weighted collective MIL assumption

The problem with the recently described standard assumption is that only one patch can decide the label of the slide, whereas the collective assumption of the multiple instance learning framework lets all instances contribute equally to the slide label prediction [46]. This assumption sees a bag as samples of a probability distribution that describe the population of that bag, and instances are assumed to be assigned labels according to some probability function $g(x) = Pr(Y|x)$. This function in our case is the CNN model itself with softmax output. The bag-level class probability function is then simply the expected class value of the population of that bag. Let $c \in Y = \{0, 1\}$ be the class label and b be a bag. Then the probability of class c given b is

$$Pr(c|b) = E_X[Pr(c|x)|b] = \int_X Pr(c|x)Pr(x|b)dx$$

The probability distribution of the bag $Pr(x|b)$ is usually not known, so it is substituted by the sample of instances present in the bag. Therefore:

$$Pr(c|b) = \frac{1}{n_b} \sum_{i=1}^{n_b} Pr(c|x_i),$$

where n_b is the number of instances in the bag [14].

But in our case, the CNN is not equally certain about the patch predictions (the output class probabilities), so we would like those patches that the model is more certain about to have a larger weight. The weighted collective assumption [14] makes this possible by incorporating a weight function into the collective assumption:

$$Pr(c|b) = \frac{1}{\sum_{i=1}^{n_b} w(x_i)} \sum_{i=1}^{n_b} w(x_i) pr(c|x_i),$$

where $w(x) : \mathcal{X} \to \mathbb{R}^+$ is the weight function determining the influence an instance has on the bag-level label. In our case, this corresponds to the weight (or certainty) derived from the

distribution of the Monte Carlo predictions. The algorithm outputs the probability of class 1 (GBM) for each slide.

4.4 Evaluation methods

Receiver operating characteristic (ROC) curves and the area under the curve (AUC) are among the most popular evaluation methods for classification models. One can compare AUCs of different models statistically to test if they are significantly different with the widely used DeLong's algorithm [10], which has been employed to compare AUCs of different slide aggregation techniques in weakly supervised histology image classification in [8].

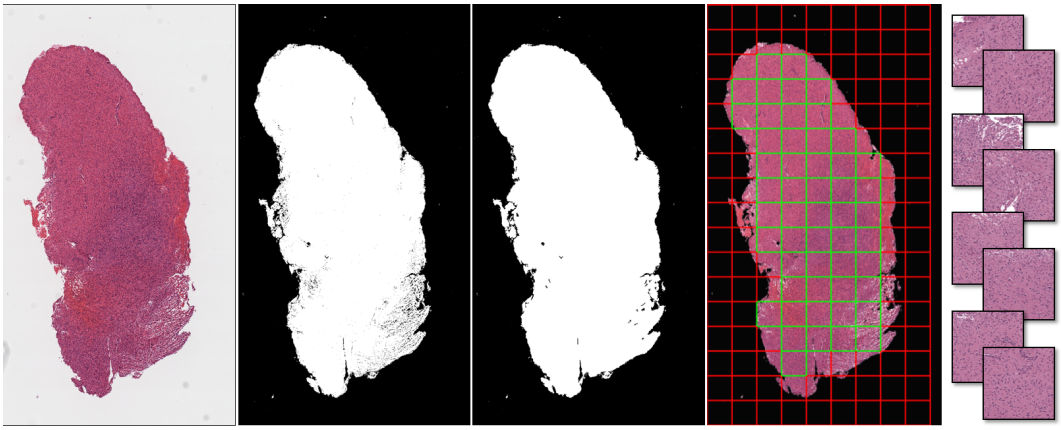
DeLong's test for comparing the AUCs of correlated ROC curves [10] is implemented in R in the pROC package [32]. Only comparison of two curves is implemented, therefore we will conduct pairwise comparisons of the methods.

!!! Delongs' test for comparing 2 correlated ROC curves !!!

!!! Accuracy, recall, specificity, F-score, ... !!!

!!! t-test for evaluating patch level pre-models (scratch, ImageNet, no augment) !!!

5 Results



(a) (b) (c) (d) (e)

Figure 6: Pre-processing pipeline

	GBM		LGG		Total	
	slides	patches	slides	patches	slides	patches
training	42	167,183	42	168,346	84	335,529
validation	13	48,804	13	51,362	26	100,166
test	42	151,336	42	158,660	84	309,996
Total	97	367,323	97	378,368	194	745,691

Table 1: Dataset of small patch size

!!! insert example of background filtering and patching!!!
!!! Insert examples of stain normalizing: target, before and after !!!
!!! Visualise grid on slide with predicted patch classes !!!

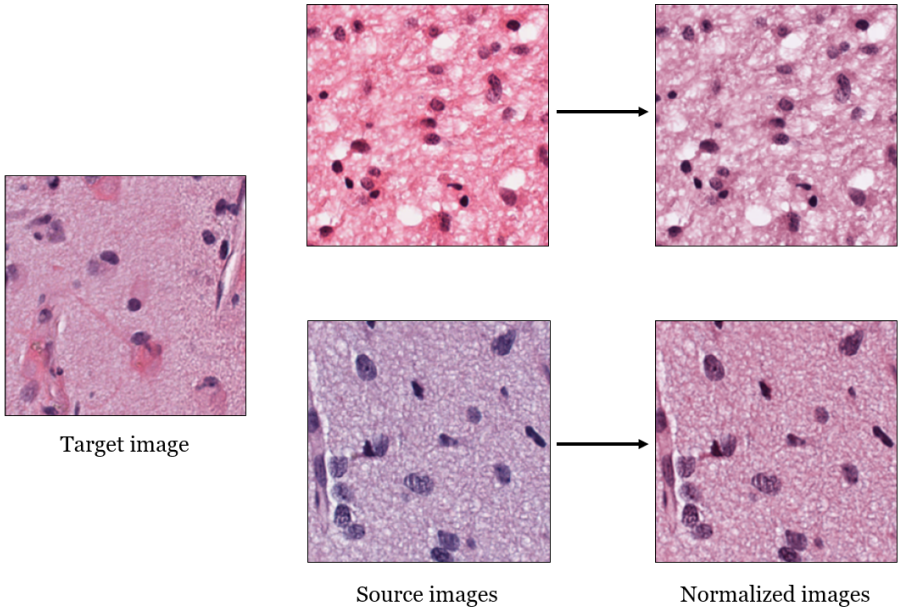


Figure 7: Stain normalization using the Vahadane method

	GBM		LGG		Total	
	slides	patches	slides	patches	slides	patches
training	42	38,215	42	38,716	84	76,931
validation	13	11,055	13	11,746	26	22,801
test	42	33,314	42	36,262	84	69,576
Total	97	82,584	97	86,724	194	169,308

Table 2: Dataset of large patch size

!!! Histogram of patch level accuracy for small and large patches !!!
!!! ROC curves, metrics comparison table !!!
!!! Visualise patch at different stages of CNN, to see what it recognises !!!
!!! Write down what features contribute most in logreg !!!
!!! Flipping only makes the majority decision easier in most cases, rare that the decision changes !!!

6 Discussion

6.1 Results

6.2 Method

6.3 Future work

!!!

When the logreg was trained without intercept, the GBM95 covariate was significant at 5 percent, but it incorrectly assumes that with all predictors = 0, the response has 50 percent probability of being 1 (source).

!!!

!!! [33] didn't use brain cancer data, so i should have done my own comparison between stain normalization methods taking into account computational complexity, because normalizing took too much time. !!!

!!! The grading methods are slightly different for astro and oligo, so MVP and necrosis are not necessarily GBM there (Perry). WHO grading is a bit blurry, because oligo can have necrosis and MVP, but they max out at grade III. !!!

!!! Future improvements: Instead of the linear weight function (certainty -> weight), create kernel. This way there are much more certain patches over 95 than anywhere else.

Would be better if we had examples of fully healthy wsis labeled as healthy. Then MIL would work better.

Could have experimented with more architectures other than resnet, and more optimizers, learning rates, etc., but this was not the focus of the thesis.

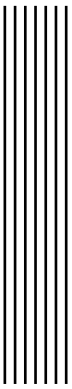
CNN model selection was done incrementally to lower training time (otherwise would have had to train many more models)

Validation set could be larger, better logistic regression could be trained (with more covariates). 10 percent of the training set could be used as validation, as in one of the sources.

Could have weighted false negatives more heavily during training to obtain high sensitivity.

No cropping and color change was an augmentation method, because we worked on eliminating these in the beginning. !!!

7 Conclusions



Bibliography

[1] Aditya Bagari, Ashish Kumar, Avinash Kori, Mahendra Khened, and Ganapathy Krishnamurthi. "A Combined Radio-Histological Approach for Classification of Low Grade Gliomas". In: *Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries*. Ed. by Alessandro Crimi, Spyridon Bakas, Hugo Kuijf, Farahani Keyvan, Mauricio Reyes, and Theo van Walsum. Cham: Springer International Publishing, 2019, pp. 416–427. ISBN: 978-3-030-11723-8.

[2] Jocelyn Barker, Assaf Hoogi, Adrien Depeursinge, and Daniel L. Rubin. "Automated classification of brain tumor type in whole-slide digital pathology images using local representative tiles". In: *Medical Image Analysis* 30 (2016), pp. 60–71. ISSN: 1361-8415. DOI: https://doi.org/10.1016/j.media.2015.12.002. URL: https://www.sciencedirect.com/science/article/pii/S1361841515001838.

[3] MBF Bioscience. *What is Whole Slide Imaging?* URL: https://www.mbfbioscience.com/whole-slide-imaging (visited on 03/15/2021).

[4] Christopher M. Bishop. *Neural Networks for Pattern Recognition*. USA: Oxford University Press, Inc., 1995. ISBN: 0198538642.

[5] Christopher M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006. ISBN: 978-0387-31073-2.

[6] Dr Rachel Brown. *Histopathology*. URL: https://www.rcpath.org/discover-pathology/news/fact-sheets/histopathology.html (visited on 03/15/2021).

[7] Peter Byfield. *Peter554/StainTools: Patch release for DOI*. Version v2.1.3. Sept. 2019. DOI: 10.5281/zenodo.3403170. URL: https://doi.org/10.5281/zenodo.3403170.

[8] Gabriele Campanella, Matthew Hanna, Luke Geneslaw, Allen Miraflor, Vitor Silva, Klaus Busam, Edi Brogi, Victor Reuter, David Klimstra, and Thomas Fuchs. "Clinical-grade computational pathology using weakly supervised deep learning on whole slide images". In: *Nature Medicine* 25 (Aug. 2019), p. 1. DOI: 10.1038/s41591-019-0508-1.

[9] John K. C. Chan. "The Wonderful Colors of the Hematoxylin–Eosin Stain in Diagnostic Surgical Pathology". In: *International Journal of Surgical Pathology* 22.1 (2014). PMID: 24406626, pp. 12–32. DOI: 10.1177/1066896913517939. eprint: https://doi.org/10.1177/1066896913517939. URL: https://doi.org/10.1177/1066896913517939.

[10] Elizabeth R. DeLong, David M. DeLong, and Daniel L. Clarke-Pearson. "Comparing the Areas under Two or More Correlated Receiver Operating Characteristic Curves: A Nonparametric Approach". In: *Biometrics* 44.3 (1988), pp. 837–845. ISSN: 0006341X, 15410420. URL: http://www.jstor.org/stable/2531595.

[11] Thomas G. Dietterich, Richard H. Lathrop, and Tomás Lozano-Pérez. "Solving the Multiple Instance Problem with Axis-Parallel Rectangles". In: *Artif. Intell.* 89.1–2 (Jan. 1997), pp. 31–71. ISSN: 0004-3702. DOI: 10.1016/S0004-3702(96)00034-3. URL: https://doi.org/10.1016/S0004-3702(96)00034-3.

[12] Mike Dusenberry and Fei Hu. *Deep Learning for Breast Cancer Mitosis Detection*. May 2018. URL: https://github.com/CODAIT/deep-histopath.

[13] Mehmet Ertosun and Daniel Rubin. "Automated Grading of Gliomas using Deep Learning in Digital Pathology Images: A modular approach with ensemble of convolutional neural networks". In: *AMIA Annu Symp Proc* 2015 (Nov. 2015), pp. 1899–1908.

[14] James Foulds and Eibe Frank. "A review of multi-instance learning assumptions". In: *The Knowledge Engineering Review* 25.1 (2010), pp. 1–25. DOI: 10.1017/S026988890999035X.

[15] Yarin Gal and Zoubin Ghahramani. "Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning". In: *Proceedings of The 33rd International Conference on Machine Learning* (June 2015).

[16] Caleb Grenko, Angela Viaene, MacLean Nasrallah, Michael Feldman, Hamed Akbari, and Spyridon Bakas. "Towards Population-Based Histologic Stain Normalization of Glioblastoma". In: vol. 11992. May 2020, pp. 44–56. ISBN: 978-3-030-46639-8. DOI: 10.1007/978-3-030-46640-4_5.

[17] K. He, X. Zhang, S. Ren, and J. Sun. "Deep Residual Learning for Image Recognition". In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016, pp. 770–778. DOI: 10.1109/CVPR.2016.90.

[18] L. Hou, D. Samaras, T. M. Kurc, Y. Gao, J. E. Davis, and J. H. Saltz. "Patch-Based Convolutional Neural Network for Whole Slide Tissue Image Classification". In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016, pp. 2424–2433. DOI: 10.1109/CVPR.2016.266.

[19] Andrew. Janowczyk and Anant. Madabhushi. "Deep learning for digital pathology image analysis: A comprehensive tutorial with selected use cases". In: *Journal of Pathology Informatics* 7.1 (2016), p. 29. DOI: 10.4103/2153-3539.186902.

[20] Justin Ker, Yeqi Bai, Hwei Yee Lee, Jai Rao, and Lipo Wang. "Automated brain histology classification using machine learning". In: *Journal of Clinical Neuroscience* 66 (2019), pp. 239–245. ISSN: 0967-5868. DOI: https://doi.org/10.1016/j.jocn.2019.05.019. URL: https://www.sciencedirect.com/science/article/pii/S0967586819306563.

[21] Tahsin Kurc, Spyridon Bakas, Xuhua Ren, Aditya Bagari, Alexandre Momeni, Yue Huang, Lichi Zhang, Ashish Kumar, Marc Thibault, Qi Qi, Qian Wang, Avinash Kori, Olivier Gevaert, Yunlong Zhang, Dinggang Shen, Mahendra Khened, Xinghao Ding, Ganapathy Krishnamurthi, Jayashree Kalpathy-Cramer, James Davis, Tianhao Zhao, Rajarsi Gupta, Joel Saltz, and Keyvan Farahani. "Segmentation and Classification in Digital Pathology for Glioma Research: Challenges and Deep Learning Approaches".

In: *Frontiers in Neuroscience* 14 (2020), p. 27. ISSN: 1662-453X. DOI: 10.3389/fnins.2020.00027. URL: https://www.frontiersin.org/article/10.3389/fnins.2020.00027.

[22] Yann LeCun, Y. Bengio, and Geoffrey Hinton. "Deep Learning". In: *Nature* 521 (May 2015), pp. 436–44. DOI: 10.1038/nature14539.

[23] Yann Lecun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L.D. Jackel. "Handwritten digit recognition with a back-propagation network". English (US). In: *Advances in Neural Information Processing Systems (NIPS 1989), Denver, CO*. Ed. by David Touretzky. Vol. 2. Morgan Kaufmann, 1990.

[24] Yann LeCun, Patrick Haffner, Léon Bottou, and Yoshua Bengio. "Object Recognition with Gradient-Based Learning". In: *Shape, Contour and Grouping in Computer Vision*. Berlin, Heidelberg: Springer-Verlag, 1999, p. 319. ISBN: 3540667229.

[25] David Louis, Hiroko Ohgaki, Otmar Wiestler, Webster Cavenee, Peter Burger, Anne Jouvet, Bernd Scheithauer, and Paul Kleihues. "The 2007 WHO Classification of Tumors of the Central Nervous System". In: *Acta neuropathologica* 114 (Sept. 2007), pp. 97–109. DOI: 10.1007/s00401-007-0243-4.

[26] David Louis, Arie Perry, Guido Reifenberger, Andreas Deimling, Dominique Figarella-Branger, Webster Cavenee, Hiroko Ohgaki, Otmar Wiestler, Paul Kleihues, and David Ellison. "The 2016 World Health Organization Classification of Tumors of the Central Nervous System: A summary". In: *Acta Neuropathologica* 131 (June 2016). DOI: 10.1007/s00401-016-1545-1.

[27] Alexandre Momeni, Marc Thibault, and Olivier Gevaert. "Dropout-Enabled Ensemble Learning for Multi-scale Biomedical Data". In: *Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries*. Ed. by Alessandro Crimi, Spyridon Bakas, Hugo Kuijf, Farahani Keyvan, Mauricio Reyes, and Theo van Walsum. Cham: Springer International Publishing, 2019, pp. 407–415. ISBN: 978-3-030-11723-8.

[28] Quinn Ostrom, Haley Gittleman, Gabrielle Truitt, Alexander Boscia, Carol Kruchko, and Jill Barnholtz-Sloan. "CBTRUS Statistical Report: Primary Brain and Other Central Nervous System Tumors Diagnosed in the United States in 2011–2015". In: *Neuro-Oncology* 20 (Sept. 2018), pp. iii1–iii86. DOI: 10.1093/neuonc/noy131.

[29] Arie Perry and Pieter Wesseling. "Chapter 5 - Histologic classification of gliomas". In: *Gliomas*. Ed. by Mitchel S. Berger and Michael Weller. Vol. 134. Handbook of Clinical Neurology. Elsevier, 2016, pp. 71–95. DOI: https://doi.org/10.1016/B978-0-12-802997-8.00005-0. URL: https://www.sciencedirect.com/science/article/pii/B9780128029978000050.

[30] Rajat Raina, Anand Madhavan, and Andrew Ng. "Large-scale deep unsupervised learning using graphics processors". In: vol. 382. Jan. 2009, p. 110. DOI: 10.1145/1553374.1553486.

[31] Saima Rathore, Tamim Niazi, Muhammad Aksam Iftikhar, and Ahmad Chaddad. "Glioma Grading via Analysis of Digital Pathology Images Using Machine Learning". In: *Cancers* 12.3 (2020). ISSN: 2072-6694. DOI: 10.3390/cancers12030578. URL: https://www.mdpi.com/2072-6694/12/3/578.

[32] Xavier Robin, Natacha Turck, Alexandre Hainard, Natalia Tiberti, Frederique Lisacek, Jean-Charles Sanchez, and Markus Müller. "pROC: An open-source package for R and S+ to analyze and compare ROC curves". In: *BMC bioinformatics* 12 (Mar. 2011), p. 77. DOI: 10.1186/1471-2105-12-77.

[33] Santanu Roy, Alok kumar Jain, Shyam Lal, and Jyoti Kini. "A study about color normalization methods for histopathology images". In: *Micron* 114 (2018), pp. 42–61. ISSN: 0968-4328. DOI: https://doi.org/10.1016/j.micron.2018.07.005. URL: https://www.sciencedirect.com/science/article/pii/S0968432818300982.

[34] Skipper Seabold and Josef Perktold. "statsmodels: Econometric and statistical modeling with python". In: *9th Python in Science Conference*. 2010.

[35] Amin Shirazi, Eric Fornaciari, Narjes Bagherian, Lisa Ebert, Barbara Koszyca, and Guillermo Gomez. "DeepSurvNet: deep survival convolutional network for brain cancer survival rate classification based on histopathological images". In: *Medical & Biological Engineering & Computing* 58 (Mar. 2020). DOI: 10.1007/s11517-020-02147-3.

[36] Caitlin Smith. *FFPE or Frozen? Working with Human Clinical Samples*. Nov. 2014. URL: https://www.biocompare.com/Editorial-Articles/168948-FFPE-or-Frozen-Working-with-Human-Clinical-Samples/ (visited on 03/15/2021).

[37] Xu Sun and Weichao Xu. "Fast Implementation of DeLong's Algorithm for Comparing the Areas Under Correlated Receiver Operating Characteristic Curves". In: *Signal Processing Letters, IEEE* 21 (Nov. 2014), pp. 1389–1393. DOI: 10.1109/LSP.2014.2337313.

[38] Hyuna Sung, Jacques Ferlay, Rebecca Siegel, Mathieu Laversanne, Isabelle Soerjomataram, Ahmedin Jemal, and Freddie Bray. "Global cancer statistics 2020: GLOBOCAN estimates of incidence and mortality worldwide for 36 cancers in 185 countries". In: *CA: a cancer journal for clinicians* (Feb. 2021). DOI: 10.3322/caac.21660.

[39] Britannica T. Editors of Encyclopaedia. "Histology". In: *Encyclopedia Britannica* (Oct. 2013). URL: https://www.britannica.com/science/histology.

[40] Britannica T. Editors of Encyclopaedia. "Pathology". In: *Encyclopedia Britannica* (Nov. 2014). URL: https://www.britannica.com/science/pathology.

[41] *The Cancer Genome Atlas Program*. https://www.cancer.gov/about-nci/organization/ccg/research/structural-genomics/tcga. Accessed: 2021-03-15.

[42] Abhishek Vahadane, Tingying Peng, Amit Sethi, Shadi Albarqouni, Lichao Wang, Maximilian Baust, Katja Steiger, Anna Schlitter, Irene Esposito, and Nassir Navab. "Structure-Preserving Color Normalization and Sparse Stain Separation for Histological Images". In: *IEEE Transactions on Medical Imaging* 35 (May 2016), pp. 1–1. DOI: 10.1109/TMI.2016.2529665.

[43] Stéfan van der Walt, Johannes L. Schönberger, Juan Nunez-Iglesias, François Boulogne, Joshua D. Warner, Neil Yager, Emmanuelle Gouillart, Tony Yu, and the scikit-image contributors. "scikit-image: image processing in Python". In: *PeerJ* 2 (June 2014), e453. ISSN: 2167-8359. DOI: 10.7717/peerj.453. URL: https://doi.org/10.7717/peerj.453.

[44] Xi Wang, Hao Chen, Caixia Gan, Huangjing Lin, Qi Dou, Efstratios Tsougenis, Qitao Huang, Muyan Cai, and Pheng-Ann Heng. "Weakly Supervised Deep Learning for Whole Slide Lung Cancer Image Analysis". In: *IEEE Transactions on Cybernetics* PP (Sept. 2019), pp. 1–13. DOI: 10.1109/TCYB.2019.2935141.

[45] Weiyuan Wu. *Patchify*. https://pypi.org/project/patchify/. 2021.

[46] Xin Xu. "Statistical Learning in Multiple Instance Problems". In: (Aug. 2009).

[47] Y. Xu, Zhipeng Jia, L. Wang, Yuqing Ai, F. Zhang, Maode Lai, and E. Chang. "Large scale tissue histopathology image classification, segmentation, and visualization via deep convolutional activation features". In: *BMC Bioinformatics* 18 (2017).

[48] Zhi-Hua Zhou. "A Brief Introduction to Weakly Supervised Learning". In: *National Science Review* 5 (Aug. 2017). DOI: 10.1093/nsr/nwx106.