Visual Analytics Framework for Training and Evaluation of Online Random Forests for Tissue Classification in Large Histopathological Slides

B.Sc. Hannes Seibt

Mat.-Nr: 209851

Hannes.Seibt@st.ovgu.de

Supervisor

Dr.-Ing. Dirk Joachim Lehmann

dirk@isg.cs.uni-magdeburg.de

Otto von Guericke University Magdeburg

Department of Simulation and Graphics

11. December 2017

FAKULTÄT FÜR
INFORMATIK

Statement of Authorship of the Student

Thesis:	Visual Analytics Framework for Training and Evaluation of Online Random Forests for Tissue Classification in Large Histopathological Slides		
Name:	Hannes	Surname:	Seibt
Date of birth:	10.03.1992	Matriculation no.:	209851

I herewith assure that I wrote the present thesis independently, that the thesis has not been partially or fully submitted as graded academic work and that I have used no other means than the ones indicated. I have indicated all parts of the work in which sources are used according to their wording or to their meaning. I am aware of the fact that violations of copyright can lead to injunctive relief and claims for damages of the author as well as a penalty by the law enforcement agency.

Magdeburg, 11.12.2017

Hannes Seibt

Abstract

Histopathology is the gold standard for the diagnosis, grading, and staging of a considerable number of diseases, among which are almost all cancer types and also breast cancer which serves as the application example in this work. The diagnosis of breast cancer has a high clinical relevance. Not only is breast cancer among the top three of the most commonly diagnosed cancer types (1.67 million cases per year [1]) and the most commonly diagnosed cancer type in women but also is the leading cause of cancer-related deaths in women worldwide [2].

In traditional histopathology, tissue that got extracted from the patients body is prepared and mounted on a glass slide in order to be examined under a standard optical microscope. This technique nowadays gets more and more replaced by digital pathology. In digital pathology whole-slide scanners are used to digitize glass slides containing tissue specimens at high resolution (up to 160nm per pixel). The resulting image is called *Whole Slide Image* (WSI). It was shown that digital examination reaches similar diagnosis performance compared to analogous examination using a standard, optical microscope [3, 4].

Since the size of WSIs commonly reaches the order of a billion of pixels, the examination is tedious and error prone, especially since the pathologists spends a lot of time navigating through uninformative areas. Furthermore pathologists never complete examine the whole WSI and therefore base their diagnosis on different data bases. This might be the reason why the variability between experts remains significant for several applications [5, 6, 7, 8].

With digital pathology it became possible to support the pathologist in examining the WSI. Depending on the task, solutions might be fully automatic deep learning approaches or semi-automatic CAD (computer-aided diagnosis) applications. While deep learning approaches have a high potential to exceed the accuracies reached by common machine learning algorithms, they also require large sets of data samples and are less generic. Semi-automatic approaches on the other hand have the advantage to make use of the user's knowledge and capabilities. Therefore they are better suited to adapt to new scenarios.

The method proposed in this work is designed as a CAD application which is able to adapt to many different tissue classification tasks. The method aims at reducing inter-rater variability by presenting the pathologist with similar WSI sections and by supporting them during the training procedure with fast and adequate feedback. To do so it makes use of superpixels which are used to accelerate computation and furthermore to display context information. An online random forest implementation ensures the real-time capability and furthermore that the proposed method is as generic as possible.

The method was evaluated in a small user study with three pathologists. Their task was to classify three WSIs using the proposed method. The dataset alongside with the ground truth data is provided by the Camelyon challenge, which is addressed at comparing novel full-automatic approaches for tissue classification in hematoxylin and eosin (H&E) stained WSIs of lymph node sections. The results of the classification were quantitatively evaluated using the statistical measure Cohen's kappa, as well as qualitatively by overt observation during the evaluation and an unstructured interview after the evaluation.

For the quantitative evaluation the average inter-rater concordance rate is compared with the outcome of a large study by Elmore et al. [9] which investigates the inter-rater concordance in a real-world scenario. The evaluation showed that using the proposed method a higher average inter-rater concordance rate was achieved.

Contents

List of Abbreviations

CAD Computer Aided Diagnosis 11–13, 16, 17, 24, 27, 84
CORRLDA Correlation LDA 25

D Differentiated 25, 26
DCIS Ductal Carcinome in Situ 15

GLCM Gray-Level Co-Occurence Matrix 56, 57
GPU Graphics Processing Unit 73, 74

H&E Hematoxylin and Eosin 11, 23, 26, 73
HDD Hard Disk Drive 73, 74
HIC High-Income Countries 8

IARC International Agency for Research on Cancer 8
IHC Immunohistochemistry 11, 61

KNN K-Nearest Neighbor 25

LBP Local Binary Pattern 22, 44, 55–57, 98
LDA Linear Discriminant Analysis 25
LMIC Low- and Middle-Income Countries 8

MR Magnet Resonance 10, 87

NB Neuroblastoma 24
NM Nearest Mean 25

PCBR Population-Based Cancer Registries 8
PD Poorly Differentiated 25, 26

RAM Random-Access Memory 40, 53, 57, 61, 74
RMSD Root-Mean-Square Deviation 21, 97
ROI Region of Interest 19–21, 23, 24, 27, 28, 42, 44–46, 52, 53, 71, 81–83, 87, 88, 98

SFFS Sequential Floating Forward Selection 25
SLIC Simple Linear Iterative Clustering 29, 33, 34, 62, 97
SLNB Sentinel Lymph Node Biopsy 8, 9, 54
SSD Solid-State Drive 74
SVM Support Vector Machine 23, 25

UD Undifferentiated 24, 26
ULS Update Lead Statistics 52
US Ultrasonography 10

WSI Whole Slide Image 2, 7, 11–14, 16, 17, 19–28, 39–44, 47, 49, 53, 54, 57–61, 66, 71–77, 79, 81–84, 97, 99, 101

1 Introduction

Examining Whole Slide Images (WSI) is part of a variety of very different fields of daily clinical routine. It can be found in the diagnosis of diseases affecting the colon, the liver, the breast and many other types of tissue. Because it is used for diagnosing such a wide variety of diseases this work mainly focuses on the diagnosis of breast cancer. This does not mean the proposed method is not applicable to other kinds of diseases, it just means the proposed method was designed to fit the needs of breast cancer diagnosis and therefore was evaluated using labeled breast cancer data.

To fully understand the motivation of this work it is important to present all relevant aspects in this context. Some may seem superfluous to discuss but it is important to substantiate the whole argumentation with facts and numbers. For instance, breast cancer incidence plays an important role in justifying this thesis' topic. Section 1.1 gives an overview about incidence, mortality, prevalence and other important characteristics of breast cancer.

Secondly, the role of histopathology needs to be clarified and why it is the gold standard in diagnosing various diseases, among which are almost all cancer types and therefore breast cancer as well [10]. Therefore a short introduction to histopathology is given in 1.2. Historically this procedure was done using a standard optical microscope. Nowadays it gets more and more replaced by digital pathology in general. Digital pathology is a new, rapidly expanding field in medical imaging. It was shown that digital examinations reach similar diagnosis performance compared to analogous examination using an optical microscope [3, 4]. In section 1.2 and 1.3 a short introduction is given to these two topics.

After the theoretical introduction a problem statement is given as well as the precise aim of this work. This will include a description of the method as well as success criteria. In this work a new method is proposed that uses super pixels to realise an easy user interaction which is intended to produce high quality input for a random forest implementation. The output of the random forest is a segmentation of a WSI according to the user's input.

1.1 Motivation

Global cancer incidence data is derived from population-based cancer registries (PCBR). But especially in developing countries these statistics mostly cover only major cities but only rarely smaller, subnational areas. For instance in 2006 only 21% of the world's population was covered by PCBR. Regions with the sparsest registration rates were Asia (8% of the population) and Africa (11%) [1]. When speaking of records that meet the standards of the International Agency for Research on Cancer (IARC) these numbers become even lower (5% coverage in Asia and only 2% coverage in Africa)[11].

Even though this shows how difficult it is to maintain worldwide cancer statistics of good quality, some facts about certain types of cancer can be stated with certainty. In their annual report of 2016 the American Cancer Society stated a decline of the incidence in cancer in men of 3.1% but also stated a stable cancer incidence rate in women [12]. Furthermore breast cancer is among the top three of the most commonly diagnosed cancer types with 1.67 million cases per year [1]. In Figure 1 the most commonly diagnosed cancer types worldwide are visualized. Not only is breast cancer the most commonly diagnosed type of cancer in women but it is also the leading cause of cancer-related deaths among females worldwide [2]. Nonetheless breast cancer mortality rates are decreasing in many high-income countries (HIC) since 1990. Accountable for those changes are early detection and improved treatment [13]. Although in HIC the mortality rates are decreasing, the rates tend to increase in countries with historically lower rates, which usually are low- and middle-income countries (LMIC), such as Latin America and the Caribbean parts of America. These increases are not explainable by only one single cause; the increases in breast cancer incidence rates could be caused by a better registration procedure as well as by changes in risk factors and limited access to early detection and treatment [14, 15, 16]. The dynamics of different reasons that might cause these rates to increases are not completely understood [2].

Most important for the treatment of breast cancer is the stage in which it gets detected; the earlier it gets detected the more effectively it can be treated [2]. When breast cancer was diagnosed during an early stage, a sentinel lymph node biopsy (SLNB) is the standard of care [17]. It helps the physician to determine whether the tumor developed the ability to spread through the body's blood or lymph fluid. It also helps in staging the tumor and planning treatment [18].

The sentinel lymph node is the lymph node which is located closest to the tumor [18]. Therefore it is possible that there are more than one sentinel lymph node. When a tumor starts spreading these sentinel lymph nodes are the first lymph nodes to which

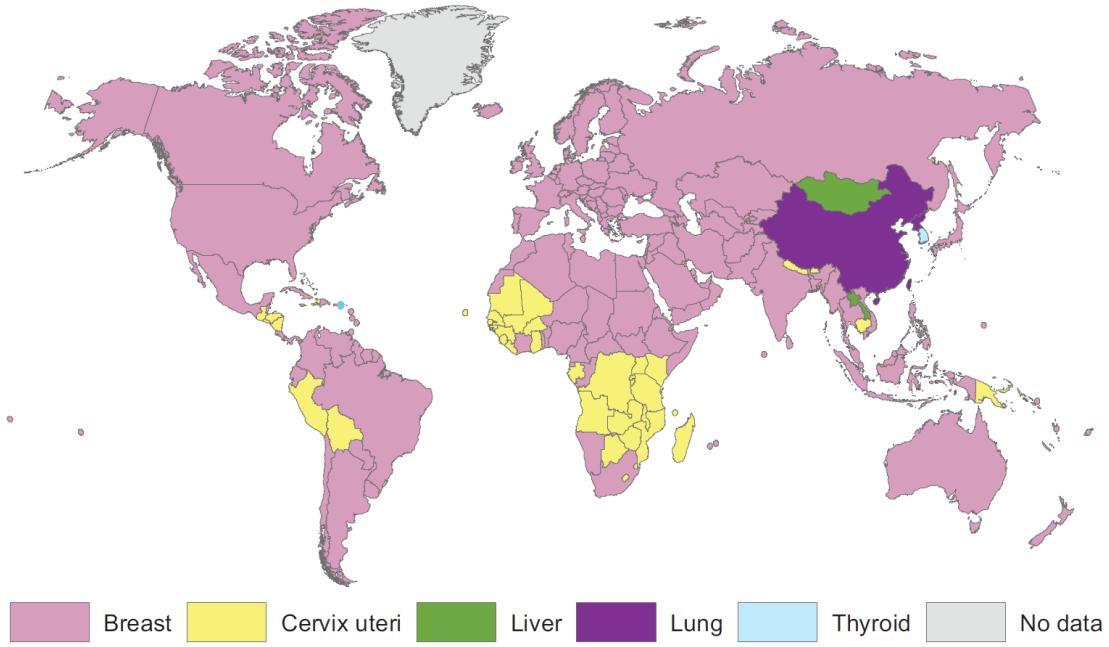


Figure 1: Most commonly diagnosed cancer types in women, 2012 [2].

cancerous cells of a primary cancerous tumor would spread. During an SLNB these lymph nodes are getting removed using a radioactive substance or blue dye to localize the lymph nodes (see figure 2).

After the tissue specimen got extracted it gets processed further in order to examine it and draw diagnostic conclusions from it. This step is covered in section 1.2 .

1.2 Histopathology

The word 'Histopathology' is compound of three Greek words: *histos* which means 'tissue', *pathos* which means 'suffering' and *logia* which translates to 'study of'. In general it is a sub domain of pathology that deals with diseased tissue. It is the study of signs of the disease by analysing tissue that had been processed and got fixed on glass slides [10].

Traditionally this examination was done using visual analysis through optical microscopes which gets more and more replaced by digital pathology. In his section a brief overview of histopathology is given, before digital pathology is covered in subsection 1.3. Its advantages over other diagnosis techniques is discussed and a rationale why it is still the gold standard in diagnosing a considerable number of diseases including breast cancer is given [19].

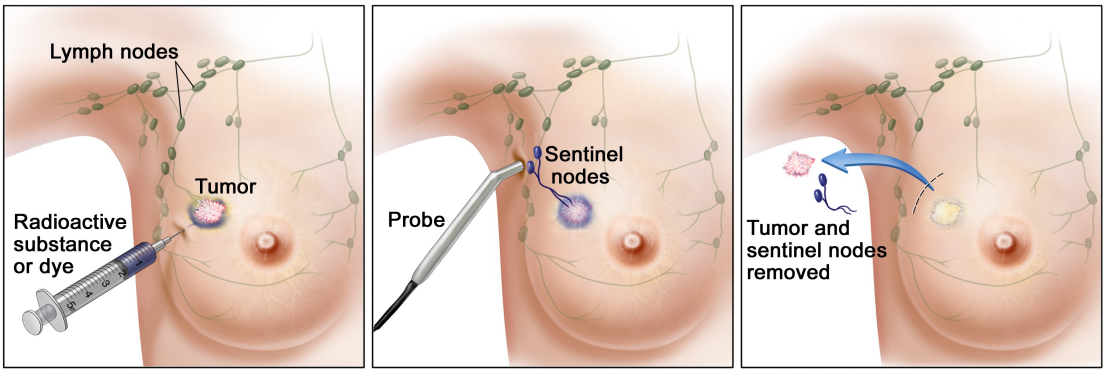


Figure 2: Sentinel lymph node biopsy of the breast. A marker gets injected close to the tumor which helps the surgeon to locate the sentinel lymph nodes. When found the sentinel lymph nodes are removed and the adjacent tissue gets checked for further cancer cells [18].

In a comparative study of different image modalities, Berg et al. [20] compared mammography, ultrasonography (US), and magnetic resonance (MR) regarding their diagnostic accuracy. Even though different modalities reached up to 96% accuracy for certain cancer types, they stated that histopathological proof of the diseases extent is imperative whenever possible to assist in therapy planning [20]. Further diagnosis techniques that could be considered are electron microscopy and molecular tests, especially gene expression profiling. Rubin et al. [19] state that no specific determinants of malignancy exist that can be detected by electron microscopy. However it may aid in the diagnosis of poorly differentiated cancer types by searching for certain signifiers that may distinguish cancer types that are problematic to classify using optical microscopes.

Molecular tests on the other hand have great potential to refine breast cancer classification and enhance our understanding of the disease biology [21]. However, Schnitt et al. [21] also state that it may take years before these tests are used in patient management and therapy decisions. This is where histopathology is at an advantage as it can rely on decades of data. In general histopathological research is symptomatical in its nature. The criteria that distinguish malign from benign tissue are not based on scientific criteria but rather on a historical correlation of histhological patterns with clinical outcome [19].

Hereinafter all preprocessing steps which both digital and analogue pathology have in common are introduced.

The first preprocessing step is the fixation of the tissue which is necessary in order to be able to slice it into 3 µm to 5 µm sections. Therefore the tissue gets fixated

with formalin and embedded in paraffin [22]. A microtome, a high-precision cutting instrument, is used to cut the block after it hardened. Afterwards the thin slices get mounted on glass slides. The last step which is the same in digital and analogue pathology is the dyeing. Because important structures like nuclei and cytoplasm are not immediately visible, these regions of interest need to get highlighted through dye.

One dye in particular, hematoxylin and eosin (H&E), has already been in use for the past hundred years and is still part of today's standard staining protocol [22]. Hematoxylin stains cell nuclei blue/purple by binding to DNA, whereas eosin stains cytoplasm and connective tissue pink by binding to proteins [10]. More advanced staining techniques used in immunohistochemistry (IHC) make use of antibody reactions. Estimating the number of cells that are positive for a particular antigen and the degree of staining intensity is part of the analysis to histologically grade the tumor. Nonetheless hematoxylin is still used beforehand as a counterstain to increase the contrast [22].

The typical pathology lab workflow ends with the staining and cover slipping of the glass slides. With the rise of digital pathology, slide digitization gets added to the workflow as a final step more often [23]. Early systems for slide digitization consisted of cameras mounted on standard optical microscopes, but these days they have been replaced by WSI scanners. The current state of the art in digital pathology is outlined in section 1.3.

1.3 Digital Pathology

In the past decade, Computer-aided diagnosis (CAD) became an essential part of the clinical workflow and was more and more accepted. Early attempts to use CAD for the analysis of medical images were already made in the 1960s. Most studies investigated automated computer diagnosis since the assumption was that computers might replace pathologists one day. This changed drastically in the 1980s, when the question came up "how can radiologists' diagnosis be helped by the benefits of digital images?" which led immediately to the concept of computer-aided diagnosis. Over the years, CAD became more popular and not only the number of CAD-related scientific publications per year increased but also the performance of CAD applications. For example, the sensitivity in detecting clustered micro calcifications using CAD applications increased from 87% at 1 false positive per image in 1993 to 98% at 0.25 false positive per image in 2007 [24].

This section discusses the need for digitization and quantitative image analysis for disease grading, compares traditional examination with its digital counterpart

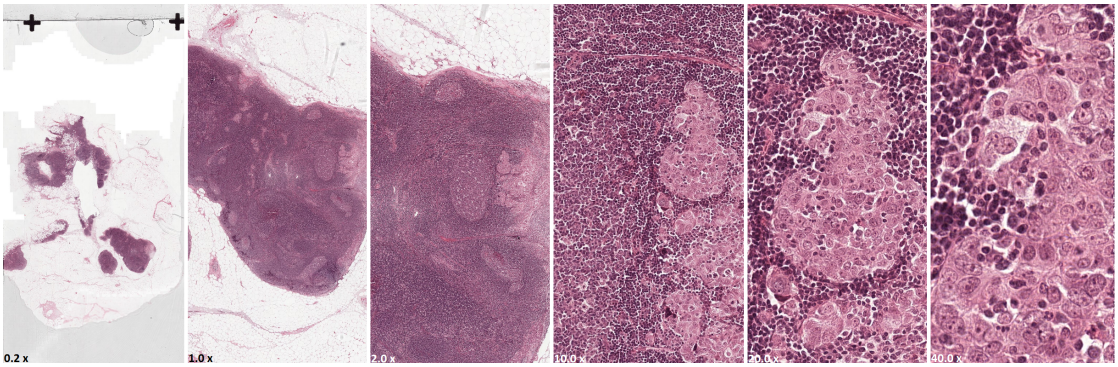


Figure 3: Whole slide image at different magnifications; from left to right: $0.2\times$, $1.0\times$, $2.0\times$, $10\times$, $20\times$, $40\times$.

regarding diagnosis accuracy and also gives an introduction to the whole image acquisition process using WSI devices.

To get an idea of the dimensions and characteristic parameters of WSIs, the following subsection gives an overview about the whole image acquisition process. In essence, a whole slide scanner is a highly specialist microscope under robotic and computer control [25]. Most WSI scanners support standard bright field images as their imaging mode but some also support fluorescent images. The image capture magnification ranges from a factor of 2 to 63 which is the main cause for differences in scan speed which ranges from < 30 seconds to around 5 minutes. Besides the magnification, the resolution plays an important role as well. The resolution for WSI scanners is defined in $\frac{\mu m}{px}$. Therefore a lower resolution value results in a higher level of detail. At a magnification of 40 $\times$, resolutions range from $0.1375\,\frac{\mu m}{px}$ to $0.25\,\frac{\mu m}{px}$ [25]. Another interesting parameter is the slide capacity. In order to enable batch processing, many scanners have a storage unit where up to 400 slides fit in for the purpose of avoiding manual insertion of each slide [25]. In figure 4, such a WSI scanner is depicted. as well as a WSI at different magnifications in figure 3 to give an idea about this dense image modality.

Just like radiology, histopathology is undergoing a process of digitalization. Even though CAD is already used for diagnoses for diseases of a wide range of body parts, it is mostly impossible to adapt well established CAD methodologies from radiology to histopathology because image characteristics are vastly different. For instance, a large radiological dataset obtained on a routine basis would be a Chest CT scan comprising of $512 \times 512 \times 512$ elements or ~ 134 million voxels [10]. For comparison a WSI scanned at 40 $\times$ on a routine basis consists of approximately $15\,000 \times 15\,000$ elements or ~ 225

million pixels [10]. Furthermore, one CT scan usually produces enough data per patient, while a single prostate biopsy can comprise of 12-30 biopsy samples. This results in $\sim 2.5 - 4$ billion pixels of data per patient study [10]. Also radiology mostly deals with gray-scale images whereas pathologists usually deal with color images. With the rise of multi-spectral and hyper-spectral imaging a pixel could be associated with even more sub-bands and wavelengths [10]. Since handling the comparatively high density of data is still an unsolved problem in image analysis research, the questions that researchers ask of pathological data are typically less well articulated than the problems investigated in radiology [10].

This is why CAD is not already a standard in clinical routine of pathologists. Currently, subjective (but well educated) tissue analysis by a pathologist is the only definitive method for (a) confirmation of the presence or absence of cancerous tissue and (b) staging, or measuring disease progression [10]. But with the rise of more specific treatment methodologies and further discoveries of more specific types of cancer it becomes more important to distinguish different types of cancer more precisely. The only way to do this is by quantifying different pathological measures like cell count, fractions of certain tissue types or staining related measures which is only possible by analysing digital representations of the tissue sections. King et al. [26] also state that by further quantification it would be possible to reduce the pathological interpretation bias discussed in section 1.1. Objective malignancy through quantification regarding breast cancer is discussed in further detail in [27].

Figure 4: Omnyx whole slide imaging scanner by GE [25].

Furthermore, WSI has a wide range of clinical application areas. It can be used in telepathology for primary diagnosis and consultation (second opinions). Compared to their analogue counterpart, WSIs are not limited to one unique instance but are rather easily reproducible. Therefore they are already in use for remotely viewing immunostains, showcasing slide sections at tumor boards and especially for archiving which is way easier than for analogue sections. Furthermore, many authors already use WSIs to report the outcome of both their primary and secondary diagnosis which is related to the aforementioned quantification [28]. Another important application

where WSI are more practical than analogue slides is teaching pathology trainees and staff pathologists [28]. Merk et al. [29] did a comparative study on the acceptance of teaching modules based on WSI systems and those based on light microscopy and glass slides. Their data suggests that today's medical trainees prefer WSI-based teaching modules.

Despite all these advantages it is still important to verify if diagnoses are as accurate using WSI compared to light microscopy and glass slides. Many validation studies have been performed but each and every one of them is focused on one particular use case and their results can not be generalized to all areas of surgical pathology [28]. Still Ghaznavi et al. looked at a large amount of validation studies which all indicated that WSI platforms are suitable for diagnoses that are as accurate as those made by light microscopy [28]. To some extent, discrepancies between WSI and glass-slide reviews can be explained by the lack of the pathologists WSI experience [25].

1.4 Problem Statement

Proceeding from the motivation presented in 1.1 and the additional information in sections 1.2 and 1.3, further problems arise. Some of these problems exist for both digital and analogue pathology, while some are just present in one of these two fields. A key problem in both fields is high inter- and intra-observer variability [9]. Intra-observer variability refers to the disagreement between two observations by one single pathologist (i.e. observations made on multiple independent occasions) whereas inter-observer variability refers to the disagreement between two or more pathologists.

The large variability rates occur due to human error, differences in opinion when interpreting tissue and also to a large extent due to the structural complexity of WSIs. As mentioned in section 1.3, WSI scans are way denser and larger than other medical imaging formats, they usually have a larger spectrum and also they are more complex in their handling. A recent study with 115 pathologists and 6900 individual cases by Elmore et al. [9] showed that diagnostic concordance rates between pathologists were significantly higher in in-person meetings compared to their reviews which they made independently on one another. The study was designed such that three experienced, internationally recognized pathologists, known for their research and continuing medical education on diagnostic breast pathology, would define the "groundtrouth" for a certain dataset which then would be compared to the observations of the other said 115 pathologists. After the initial independent evaluation, the three reference pathologists only agreed on 75% of the cases (180 of 240). After an in-person

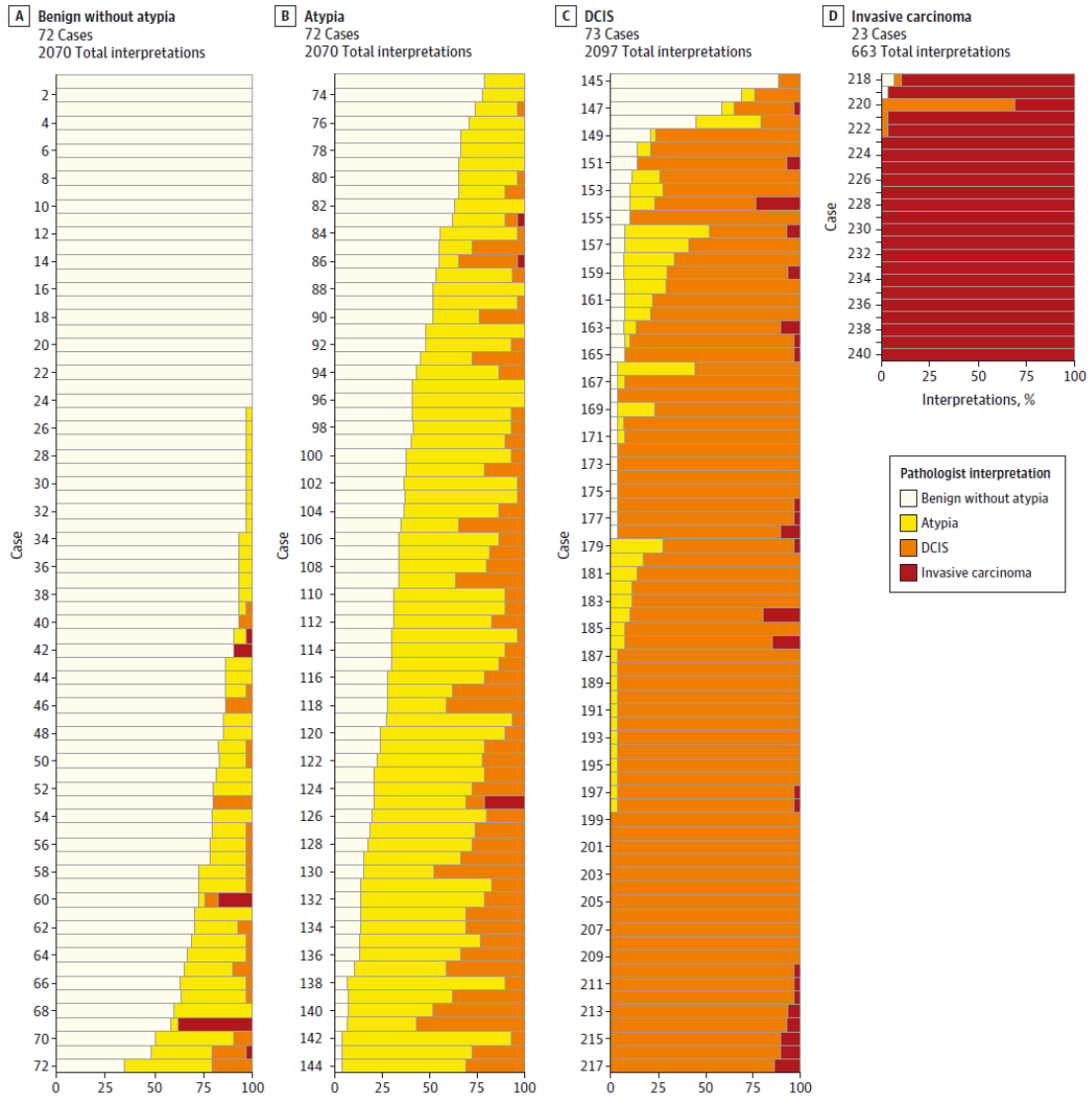


Figure 5: Participating pathologists' interpretations of each of the 240 breast biopsy test cases [9].

meeting these three pathologists managed to come to an agreement regarding the diagnosis of the 240 cases. In a second step, all 115 pathologists participating in this trail independently also had to make a diagnosis on 60 of the overall 240 cases each. Therefore they provided 6900 individual interpretations for comparison. Their comparison showed an overall concordance rate of 75,3%. Depending on the type of cancer and stage these values varied: The diagnoses of invasive breast cancer showed a high level of agreement whereas ductal carcino in situ (DCIS) and atypia showed a substantially lower level of agreement [9].

In figure 5, the statistics for each of the overall 240 cases are depicted. Each beam represents one case. It is proportionally subdivided by all diagnoses made by the participating pathologists.

It is easy to see that in a lot of cases, especially regarding atypia, the variability of the pathologists' diagnoses is noticeable. The consequences of such misinterpreted cases are different depending on the stage. When a biopsy is overinterpreted, the patient might undergo unnecessary surgeries, radiation or hormone therapy which might result in heightened surveillance, costs and anxiety [9].

Therefore it is important to provide pathologists with tools that help them examining WSIs in a way so that they do not miss important parts of the whole section. In addition, they should be able to quantify and visualize their findings in order to be able to discuss the case, to compare it to other pathologists findings and to properly archive it. Further requirements are listed in section 4.1.

Since pathologists are well paid specialists it is of interest to keep the examination time per WSI as low as possible without affecting the quality of the diagnosis. As a consequence, *eye-balling* became a widely spread method for some cancer classification techniques [30]. One major field of application is the Ki-67 index which is getting more and more important in the classification of breast cancer and also other types of cancer. It was shown that the Ki-67 nuclear antigen is expressed in only a few phases of the cell cycle and thereby it is possible to assess the growth fraction of neoplastic cell populations. The main activity for calculating the Ki-67 index hence is counting overall fractions of cells. eye-balling in essence is one of the prevalent counting methodologies where the percentage of Ki-67-positive tumor cells is roughly estimated. In order to do so, the entire WSI is scanned at intermediate power (x 10 objective) without actual counting cells. It is the most widely used method and even is advocated by some of the original authors of European Neuroendocrine Tumor Society [31] and the North American Neuroendocrine Tumor Society [32] guideline papers. But Reid et al. [30] showed in a study that while eye-balling indeed is the fastest and cheapest way to count cells, it also suffers from poor reliability and reproducibility.

Techniques like eye-balling show the main principle prevalent in the field of histopathological analysis. Although such techniques are not as well documented for other diagnosis methodologies, they might exist in other fields of diagnosis and suffer the same consequences. CAD applications need to address problems like this in a semi-automatic manner to decrease the pathologists' workload and increase accuracy and reproducibility. This item is revisited in the requirement definition in section 4.

The last issue is addressed at machine learning based solutions and especially applications based on neural networks. As Carpenter et al. [33] put it, there is a general lack of flexibility to adapt to new problems. Custom programs show the potential and power of automated image analysis but are not as adjustable as they need to be in order to be adapted to new tasks without interacting directly with the code. Hence this is simply not practical for routinely processing thousands of images [33]. Commercial software on the other hand completely lacks the ability of adaptation by its proprietary nature. Such solutions are usually closed source which prevents the researcher from knowing the strategy of a given algorithm nor can they modify it if desired.

Pathologists in their clinical work flow on the other hand usually do not need their software to adapt to new cases; they especially do not need to interact with the code itself. Software that requires the pathologist to understand the underlying algorithm poses a problem. Therefore it is necessary to let the pathologists stay in their own domain when using CAD software in medical imaging. All in all it is important to keep in mind which mental models the end user works with when developing a medical CAD application. This topic is also discussed further in section 4.1.

The major problem that will be addressed in this work is variability of the pathologists' diagnoses. This will be achieved by a method that makes eye-balling obsolete and also by bridging the gap between histopathological analysis and data science/image processing. The concrete aim of this work is further refined in section 1.5.

1.5 Aim of this Work

This work aims at developing a method for tissue classification with lower inter-observer variability compared to the study presented in section 1.1. The goal is to reach an inter-observer agreement of 80%. The proposed method consists of two parts: the interaction concept and the image analysis approach for the classification of WSIs. They are evaluated both independently and in their mutual interplay.

The main assumption is that users are able to train a classifier of a machine learning approach more precisely when they are presented with visual feedback about the current state the classifier is in. This is important since the classifier usually gets better during training but also can get worse if provided with bad or wrong training data. This way users tend to produce similar classifiers that lead to similar classification results and therefore to higher inter-observer agreement.

In order to verify its success the image processing part first undergoes an evaluation where its results are compared to existing results of a medical image challenge where

the same data was both provided and used. This part represents the hypothetical case that the classifier is provided with 100% correct input data. In a second step, pathologists use the proposed method to classify cancerous lymph node tissue. The results are compared and a median concordance rate is calculated.

1.6 Structure of this Work

This work starts with the presentation of related work. In the following section 2.2 the advantages and disadvantages are discussed as well as applicability and suitability for real world applications and also in which way these concepts contribute to this work. In section 3 used concepts are introduced. This involves image analysis methods as well as design methods. The main part of this work is section 4, which consists of the segmentation algorithm and the interaction concept for the user interaction. In section 5 a detailed description of the evaluation setup and its realization is given. Afterwards the results are presented and discussed. This work closes with a summary of the proposed method and an outlook at future work in section 7.

2 State of the Art

With the recent advent of digital pathology, many approaches for (semi-) automatic analysis of WSIs were introduced. In the following section 2.1, existing approaches are presented and in the subsequent section 2.2 they are analysed regarding their suitability for the approach presented in this work. Furthermore the question concerning their completeness is discussed and in which way this works approach contributes to the field of research in general.

2.1 Literature Studies

Many different approaches for tissue classification already exist but most of them are either very specifically designed for one use case, tied to a certain source of data or have other premises that are hard to fulfill. While their restrictions are discussed in the next section 2.2, this section introduces them in general.

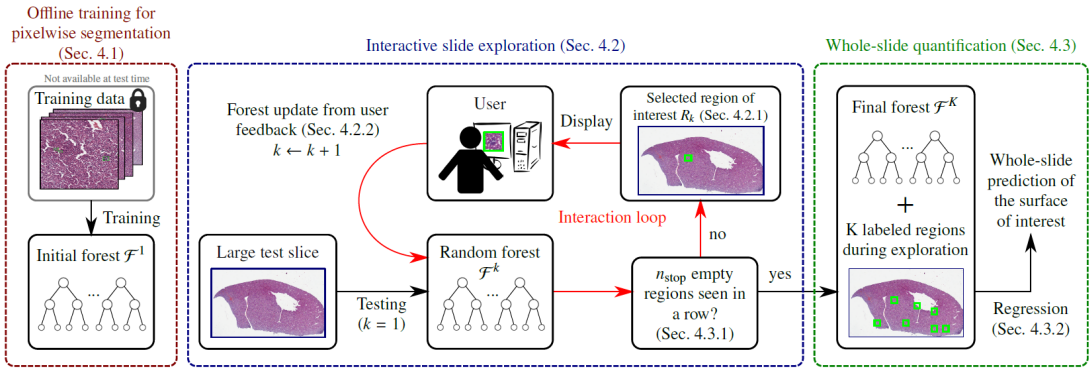


Figure 6: Workflow for assisting whole slide examination by Peter et al. which is representative for many approaches in this field of work [34].

One very popular approach is to present the pathologist with different regions of interest (ROI) one after another while simultaneously training a classifier. Peter et al. present such an approach using adaptive forests to assist the pathologist with the examination of histopathological slides [34]. Their objective was to assess the surface

covered by hematopoietic cells within mouse liver slides. To do so, they developed a region scoring function which converts pixelwise predictions into scores for each region of the whole slide. Depending on the score of a certain region it is presented to the user whose input gets used to update the random forest. In their work they use Haar-like features which describe a pixel by its visual content at offset locations in their direct vicinity. By using integral images, this approach leads to low computation times and a low memory footprint. The whole work flow is depicted in figure 6.

The scenario starts with an offline training step in which a classification forest $\mathcal{F}^1$ is trained at pixel level on some labeled examples and thereby encodes the available prior knowledge before observing the test data. This initial forest outputs for every pixel $\mathbf{x}$ the probabilistic estimate $\mathbb{P}(\mathbf{x} \in \mathcal{P}|\mathcal{F}^1) \in [0,1]$ that the label $y(\mathbf{x}) = 1$, i. e. belongs to the sought structure. In this instance $\mathcal{P} = \{x \in \mathcal{X}|y(x) = 1\}$ denotes the set of positive instances, with $\mathcal{X}$ being the set of observable samples. Since the goal of their approach is to display ROIs to the pathologist, a region scoring function ϕ is needed that uses the pixelwise forest predictions. It is defined as the sum of probabilities for having a positive label, i. e. $y(x) = 1$, of each pixel belonging to the region $\mathcal{R}$:

$$\phi(\mathcal{R}|\mathcal{F}) = \sum_{x \in \mathcal{R}} \mathbb{P}(x \in \mathcal{P}|\mathcal{F}) \tag{2.1}$$

Region $\mathcal{R}_i$ is part of a set of non-overlapping regions $\mathcal{R}^*$ of fixed size $\rho \times \rho$. The first step of the methodology of Peter et al. [34] is to determine region $\mathcal{R}_1 \in \mathcal{R}^*$ of highest interest according to the scoring function $\phi(\mathcal{R}|\mathcal{F})$ and display it to the pathologist. In the next step the pathologist has to report the relevance of region $\mathcal{R}_1$. Peter et al. considered two possibilities to do so by either fully delineating the ROI (time-consuming, accurate) or by a one-click input (fast, ambiguous). Either way the pathologists' input gets used to modify the initial forest $\mathcal{F}^1$ which thereby becomes $\mathcal{F}^2$. This procedure is repeated until a stopping criterion is reached. This stopping criterion is based on the density of positive suggestions, i. e. when most positive regions presumably were seen the exploration stops. The variable n_{stop} defines after how many negative regions that were suggested in a row the stopping criterion is reached and the whole-slide prediction can be carried out. Experimental results suggested $n_{stop} = 6$ as a suitable trade-off between accuracy and human effort.

After k iterations this generates, starting from an initial forest $\mathcal{F}^1$, a series of forests $\mathcal{F}^2$, $\mathcal{F}^3$,...$\mathcal{F}^{(k+1)}$. After the stopping criterion was met all the gathered knowledge is used to quantify the surface $\bigcap q$ covered by hematopoietic cells within the WSI via linear regression. Said knowledge constitutes of k labeled regions as well as k forests.

In their evaluation Peter et al. were able to show that their method is capable of extracting relevant ROIs. After presenting $\sim$ 20% of the WSI to the pathologist $\sim$ 98% of the surface of interest was seen. In figure 7 the correlation between estimated and true surface covered by hematopoietic cells is shown. Their dataset consisted of 70 WSI of mouse liver specimen that was scanned at a resolution of 0.5 $\frac{\mu m}{px}$ which results in image dimensions of $\sim$ 25 000 px $\times$ 30 000 px or 15 mm $\times$ 12.5 mm (187.5 mm^2). The figure also shows that the surface covered by hematopoietic cells is 0.2 mm^2 on average which puts the RMSD of $1.9 \times 10^{-2} mm^2$ into perspective. During the evaluation the stop criterion worked such that 5.5% of the WSI was shown to the pathologist in total.

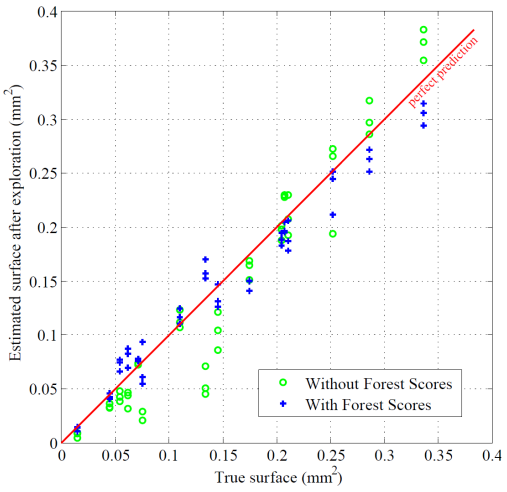


Figure 7: Correlation between true surface covered by hematopoietic cells and estimated surface by their porposed method. Root-mean-square deviation (RMSD) is $1.9 \times 10^{-2} mm^2$ [34]

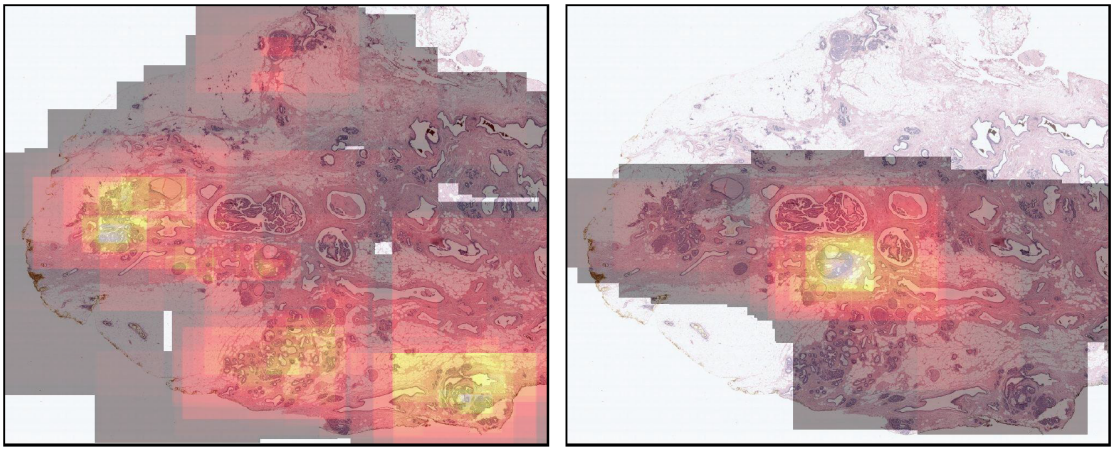


Figure 8: Visualization of the viewing behaviour of two pathologists of the same WSI illustrated as heat maps. Brighter areas indicate that said pathologists spent more time examining the tissue in these regions. Note that although the same WSI was examined the viewing behaviour of these pathologists vastly differ [35].

A different approach by Mercan et al. [35] makes use of the viewport tracking data of three pathologists with the goal to localize diagnostically relevant ROIs for the examination of breast histopathology images; an example of such a dataset is shown

in figure 8. In contrast to the approach by Peter et al. this method does not aim at discriminating between malignant and benign tissue but rather at extracting a set of regions that may have an impact on the overall diagnosis of a case.

The viewport is defined as the visible part of a WSI on the pathologist's screen. It gets stored in a viewport log, so each entry of the viewport log corresponds to a rectangular part of the actual image. In order to analyse this dataset Mercan et al. [35] define three elementary actions over the viewport tracking data: Zoom peak, slow panning and fixation. Zoom peak refers to a point where the zoom level is higher than the previous and the next viewport log, slow pan is defined as an interaction without zooming but small displacement (to avoid logging interactions with the only purpose of navigation) and fixations are the points where the viewport stays the same for more than 2 seconds.

Analysing the viewport tracking logs results in a set of all the areas in which zoom peak, slow panning or fixation took place. This set is marked as a collection of diagnostically relevant regions that Mercan et al. [35] aim to predict with their approach in a WSI.

After extracting the set of relevant regions, in the next step a bag-of-words model is used to identify salient *words*. To build a visual vocabulary all regions marked as diagnostically relevant get subdivided into 120×120 pixel image patches, which serve as *words*. A bag is a 3600×3600 pixel region also cut from the WSI and therefore a *bag of words*. In order to classify the patches with k-Means, features need to extracted. Two widely used features are used, arranged in two sets: For the first set of features a well known color deconvolution algorithm [36] is used to generate two gray-scale images, one containing all structures dyed with eosin and one containing all structures dyed with hematoxylin. For each image a texture feature called Local binary pattern (LBP) gets computed from which a histogram gets calculated. Both histograms get concatenated and thereby form the first feature. The second feature gets computed in a similar manner. It contains three histograms of each channel of a CIE-L*a*b* color space representation that also get concatenated.

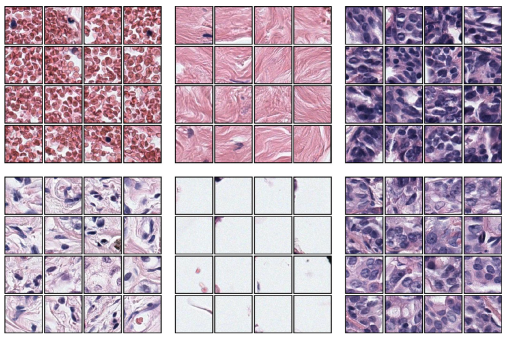


Figure 9: Example results from the K-Means Clustering. It shows how different textures and color characteristics got assigned to different centres [35].

Using the k-Means algorithm the candidate words were clustered and K cluster centres were obtained. In a sliding window approach any input image now can get classified. The sliding window is 3600 ×3600 pixel large and consists of 30 × 30 pixel patches that are visual words. Each visual word now gets assigned the cluster number of the closest cluster centre. Figure 9 shows examples of different patches assigned to different cluster centres. In a next step a histogram of cluster numbers representing the frequency of each cluster in that window gets calculated for each sliding window. In a final step a binary classifier is trained which uses the patches extracted from the expert's interactions with the slide as positive samples and samples from the rest of the image as negative samples.

Using a support vector machine (SVM) as the classifier, Mercan et al. [35] were able to reach accuracies up to 79,60% whereas logistic regression only reached 75,52% accuracy.

In an updated version of their publication Mercan et al. [37] replaced their 120 × 120 pixel patches with superpixels of similar size and compared the resulting ROI classification accuracy. In figure 10 this comparison is visualized as a function of dictionary size. The difference between the two compared modes is not significant but steady. Further publications that successfully use superpixels as building blocks in tissue analysis can be found in [39] and [40]. The general concept of superpixels is introduced in section 3.1.

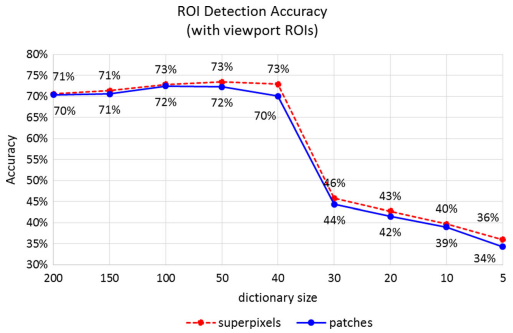


Figure 10: Comparison between the classification accuracies of two input formats: Normal image patches and similar sized super pixels [37].

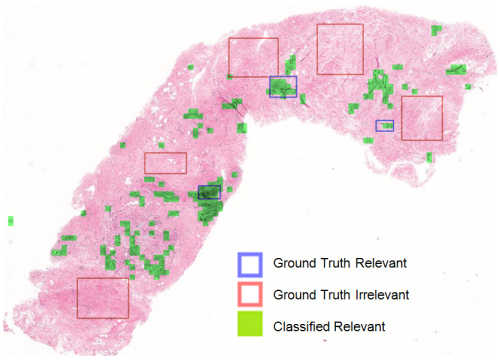


Figure 11: Classification result of an evaluation of a WSI [38].

Another approach for finding diagnostically significant ROIs by Bahlmann et al. [38] makes use of the distribution of epithelial nuclei and tubule formation and their specific characteristics they show when stained with H&E. Just like Mercan et al. [35] they do not aim for an algorithm that distinguishes between malignant and benign tissue but rather one that identifies di-

agnostically relevant ROIs. Their strategy for finding ROIs is similar to the one by a human pathologist: The decision whether a region is of interest or not is based on the distribution of nuclei and cytoplasm within a region [38]. While nuclei appear purple, cytoplasm appears pink. In order to achieve low computation times, Bahlmann et al [38] ignore higher abstraction levels like shape information, but rather apply statistical analysis on pixel level on the highest resolution. Therefore they subdivide the WSI into patches of the size of 256×256 pixel or 120×120 µmeter. Like Mercan et al. they separate the input image patches into two H and E channels. Each patch gets represented by a 22-dimensional feature vector of eleven uniformly distributed percentile ranks (at 0%, 10%, 20%,...,100%) for each channel. In the last step a linear SVM gets trained for this binary classification task.

Their goal is to achieve a 100% detection rate at the expense of a higher false positive rate. In their evaluation they were able to show that not only relevant regions were found with an accuracy of $\sim$ 100%, but also irrelevant regions were not falsely classified as relevant regions. Figure 11 visualizes this fact exemplary on a fully marked WSI. The blue and red boxes are annotations made by a pathologist, which indicates non-labeled regions can belong to both, relevant or irrelevant regions. The green regions represent areas that were classified as relevant by their approach. The visualization shows in general that Bahlmann [38] et al. reached their goals and showed that their method could serve as a proper preprocessing step for different CAD applications.

As a last method, a fully automatic tissue classification approach by Kong et al. [41] will be briefly introduced. Also discussing fully automatic approaches when designing CAD solutions is important because of two aspects: (a) some parts of the methodology can be incorporated in CAD solutions and (b) since fully automatic approaches usually are less generic they achieve higher accuracy. Therefore they give a better idea about the potential of computerized tissue analysis.

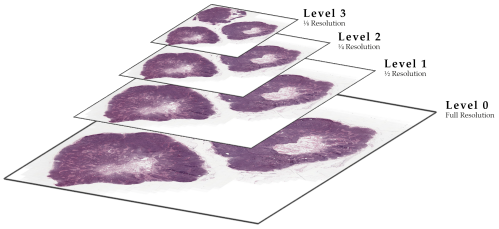


Figure 12: Typical WSI representation as a multi-resolution pyramide with image sizes scaled by half from bottom to top.

Kong et al. [41] introduced a method for tissue classification that helps in grading neuroblastoma (NB) which is a cancer type most frequently occuring in children. Three different categories of NB are distinguished: undifferentiated (UD), poorly undifferentiated (PD), and differentiated (D). Those grades are reflected in different

pathological characteristics and micro-textural features. The more differentiated the tumor is, the less aggressive they are; therefore it is important to be able to grade them.

Like in the previous approach by Bahlmann et al., this approach also operates on pixel level, however it not only uses the highest resolution but all of them; from lowest to highest. Figure 12 displays a typical pyramid image that is commonly used in multi-resolution systems. Their approach is based on the typical workflow of a pathologist. It starts on the level with the lowest resolution and goes to the next level with higher resolution if the classification accuracy is below a certain threshold. Rather than requiring the algorithm to work on the whole WSI, the WSI is partitioned into 512×512 pixel tiles on which the classification algorithm is applied.

In a preprocessing step each tile on each resolution level gets segmented in multiple cytological components (nuclei, cytoplasm, neuropil, red blood cells (RBCs), and background) using a color deconvolution algorithm called EMLDA [42]. In the actual training only the channels containing neuropils and cytoplasm are used. The textural features that are used are four of Haralick's statistical measures: entropy, mean, variance, and homogeneity (f_9, f_6, f_4 and f_1, see A.1 for reference). These features are computed on co-occurrence matrices of values that lie in a local neighborhood of the pixel. Said co-occurrence matrices are computed on the L*, a*, and b* channels of each of the segmented image components: the neuropils and the cytoplasm. This results in $2 \times 3 \times 4 = 24$ features. Therefore each image patch is represented by a 24-dimensional image vector. But not all features may increase the classification accuracy, in fact too many features may result in a phenomenon called *peaking phenomenon* as a result of Bellman's „curse of dimensionality". Further discussion on this topic can be found here [43]. In order to reduce dimensionality Kong et al. [41] insert a step into their processing pipeline, in which they select a subset of features using a popular feature selecting technique, called the sequential floating forward selection (SFFS) [44]. Depending on the input SFFS produces a minimal set of varying size of features.

This set is used to train seven different classifiers: K-nearest neighbor (KNN), linear discriminant analysis (LDA) & KNN, LDA & nearest mean (NM), correlation LDA (CORRLDA) & KNN, CORRLDA & NM, LDA & Bayesian and SVM. Using many classifiers whose classification results are getting combined is special about this approach. After training each classifier on each resolution, the results are combined in a final evaluation step by the so called classifier combiner.

The classifier combiner produces a final decision θ^* that refers to the decision supported by the majority of the K classifiers:

$$\theta^* = \underset{i \in \{1,2,\ldots,C\}}{\arg\max}\, \Psi(i) \tag{2.2}$$

where $\Psi(i)$ is the number of votes for the i-th class collected from the K classifiers and C is the number of classes. After a label is assigned to each patch, the confidence S_l of this decision gets evaluated. S_l is defined as the sum of weighted weights assigned to classifiers that concur with the combiner. The weights are computed using the leave-one-out validation process.

If the sum of weights S_l is below a certain threshold $\gamma_{l,l+1}$ the algorithm will continue its analysis on the next higher resolution, otherwise the classification result is good enough and the process is quitted. The threshold $\gamma_{l,l+1}$ defines for level l below which confidence analysis is continued on level $l+1$.

For their evaluation Kong et al. [41] et al. use 387 image tiles extracted from three representative WSIs (10 UD, 10 PD, 13 D) to train their seven classifiers. To evaluate the generalization ability of their approach Kong et al. [41] apply their method on 33 unseen WSIs with a threshold of $\gamma_{l,l+1} = 1$ for all levels. Their approach reaches an overall classification accuracy of 87.88%.

In this section four different tissue classification approaches were introduced representative for the whole field of research regarding their general strategies as well as regarding their evaluation results. In the next chapter they are discussed in terms of the design of a semi-automatic tissue classification approach.

2.2 Analysis

In this section it will be analysed to which extent the methods presented in section 2.1 are of use in the design of a CAD tissue classification approach.

One restriction that all of them have in common is their reliance on prior knowledge. None of the methods provide a way that would allow users to specify a certain class of tissue that they would like to segment in a WSI. Especially the approach by Bahlmann et al. [38] is only applicable on H&E stained tissue and furthermore is only suitable for the diagnosis of cancerous tissue, since their features are closely bound to the characteristics of such tissue. The other approaches seem to be applicable for different classification tasks depending on the training dataset.

The input dataset is another crucial part of a tissue classification method. While three of them rely on datasets, labeled by experts the approach by Mercan et al. relies completely on viewport tracking data. In order to be able to use a classification algorithm on a new type of dataset, it is worthwhile to keep the effort for the training as low as possible. While outlining ROIs in WSIs is a tedious procedure, generating a sufficiently large viewport tracking dataset is even more inconvenient. The other approaches relying on labeled datasets at least can draw on either online available datasets of common tasks provided by organizers of medical imaging challenges, or datasets published by other researchers.

Another difference between the presented approaches is their purpose. While three of them aim for extracting diagnostic relevant ROIs, the remaining one aims for a full tissue classification. Therefore an incorporation into a CADapplication would be at different points of a hypothetical interaction pipeline. The approaches that aim to extract ROIs could be used like Peter et al. [34] suggest in their method to generate sufficient input for the training of a classifier. Full automatic approaches, like the one by Kong et al. [42], on the other hand could be used in the subsequent step where the experts input is used to train a classifier that can evaluate the WSI.

To enable integration of approaches like the one of Kong et al. [42] into CAD applications, it is necessary to reduce their complexity to the point where they become real-time capable. For instance Kong et al. used a 64-node cluster for their evaluation which lead to a median computation time of 32.77 ms per image tile at the highest resolution (896×896 pixel). This results in a processing rate of $1.47 \cdot 10^8 \frac{\text{px}}{\text{min}}$. In contrast Peter et al. [34] achieve a processing rate of $2.0 \times 10^7 \frac{\text{px}}{\text{min}}$ using only consumer-level hardware. This small difference can be explained by their approaches complexity. While Peter et al. [34] only compute highly efficient Haar-like features on only one resolution and only train one classifier, Kong et al. [41] work with four more complex Haralick features on four different levels of resolution and train seven classifiers. Bahlmann et al. [38] even accomplish a processing rate of $3.93cdot10^{10} \frac{\text{px}}{\text{min}}$ which leads to a processing time of ~ 2 s for a full WSI using a standard laptop.

This section presented different main variables that are essential in a CAD applications.

2.3 Summary of Existing Approaches

In the last section, important variables of CAD applications in the field of tissue classification were presented. This section defines how this works approach fits into

the pool of related work and furthermore states which aspects are taken care of which were neglected by the methods introduced in section 2.1.

Most importantly this works approach is a real-time capable CAD application for tissue classification that works on consumer-level hardware. Its interaction pipeline will be similar to the one by Peter et al. [34] but most importantly it does not rely on a dataset of labeled WSIs which makes it applicable for a broader set of tasks. The pathologist starts with only the WSI and successively refines the classifier which suggests new diagnostically relevant ROIs.

Furthermore it features a set of interactions that allows the pathologist to outline cancerous regions with little effort. This way the classifier gets more precise input compared to the approach by Peter et al. [34].

In summary, this works approach is applicable to a larger set of tissue classification tasks, it features user interactions that result in training data of high quality without challenging the pathologist to familiarize himself with a completely unknown topic, the topic of machine learning. And finally it does not require a cluster to analyse WSIs as a whole. Neither does it need more than consumer level hardware to make real-time user interaction feasible.

3 Methodology

Besides discussing related work it is important to introduce used concepts. This section will introduce the most important used concepts from both the field of tissue classification using machine learning and the field of user interaction design.

3.1 Used Concepts

The two most important concepts used in this work are random forests by L. Breiman [45] and SLIC (Simple Linear Iterative Clustering) superpixel segmentation by Achanta et al. [46]. These are the only two concepts that will be discussed in detail since it would go beyond the scope of this work to introduce all further more general used concepts like k-Means clustering or Gaussian filtering.

3.1.1 Random Forests

The random forest algorithm is a classification and regression method proposed by L. Breiman [45] in 2001. It since got used in many medical imaging and computer vision applications, of which a few are listed in [47]. Its rising popularity is driven by many reasons: First it has proven its high efficiency during both training and evaluation (parallelization is easily achievable) while still achieving state-of-the-art results [48]. Second and most importantly it can deal with small sample sizes and high dimensional feature space without falling a victim to the "curse of dimensionality" [49]. Said "curse" refers to the problem which data analysis techniques compete with that occurs when dimensionality increases and data points become increasingly "sparse". Furthermore random forests are inherently multi-class, which means that it does not require to train multiple classifiers to solve a multi-class classification problem. Although they were introduced to be trained in offline mode many adaptations arose that enabled an online use of random forests [48].

This section will present the basic idea of random forests and will define the mathematical notation which is taken from [50]. When introducing all random forest notations

as following, vectors are denoted as boldface lowercase symbols (e.g. $\mathbf{v}$) and sets as calligraphic symbols (e.g. $\mathcal{S}$).

A random forest is an ensemble of random trees which themselves are decision trees in general. Figure 13 shows a figurative depiction of a decision tree and its internal functioning for a hypothetical classification task.

Each branch node is associated with a binary split function $h(\mathbf{v}, \boldsymbol{\theta}_j) \in \{0, 1\}$ (also called "weak learner" in literature) similar in its functioning to the questions in figure 13 ("Is bottom part blue?"). All leaf nodes on the other hand are associated with a predictor model $p(c|\mathbf{v})$ with $c \in \{c_k\}$ similar to the labels attached to the leaf nodes in figure 13 ("Outdoor", "Indoor"). It assigns either a class c or in a regression scenario a probability of $\mathbf{v}$ belonging to class c to an incoming data point $\mathbf{v}$. Both the predictor model and the split function are explained in detail in the next section.

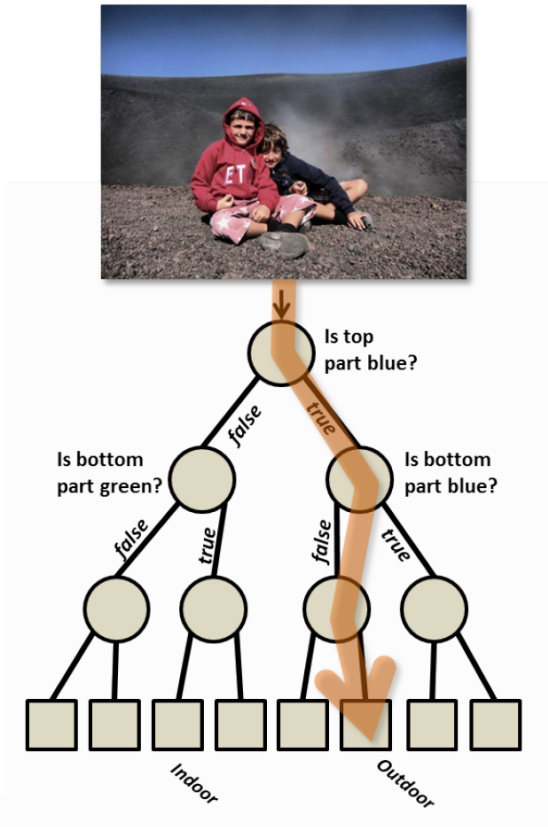


Figure 13: Each branch node of a decision tree stores a binary split function to which each incoming data point is applied. This figure shows in a figuratively manner how a decision forest is used to discriminate images containing outdoor scenes from images containing indoor scenes [50].

Training

A random forest gets trained using a set of data points $\mathcal{S} = \{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n\}$. These data points are denoted as vectors $\mathbf{v} = (x_1, x_2, \ldots, x_d) \in \mathbb{R}^d$ with x_i representing a scalar feature response. Furthermore each data point $\mathbf{v}$ is assigned to one class c_k or in a regression scenario to a vector $\mathbf{p} \in \mathbb{R}^k$ that contains the probabilities of it belonging to each of the k classes.

Each tree gets trained independently with the same dataset $\mathcal{S}$ preferably parallel since these processes are independent from one another.

Before discussing the training process in detail it is important to familiarize with the concept of information gain and its role in the training process. In general information

gain I is described in the change of entropy of two states which can be described by the following equation:

$$I = H(\mathcal{S}) - \left(\frac{|\mathcal{S}^L|}{|\mathcal{S}|} H(\mathcal{S}^L) + \frac{|\mathcal{S}^R|}{|\mathcal{S}|} H(\mathcal{S}^R) \right) \tag{3.1}$$

with the Shannon entropy defined as: $H(\mathcal{S}) = -\sum_{c \in C} p(c) \log(p(c))$ [50]. In equation 3.1 $\mathcal{S}L$ and $\mathcal{S}^R$ refer to the subsets of data points that arrive at the left and right child node with $\mathcal{S}^L \subset \mathcal{S}$, $\mathcal{S}^R \subset \mathcal{S}$, $\mathcal{S}^L \cap \mathcal{S}^R = \emptyset$ and $\mathcal{S}^L \cup \mathcal{S}^R = \mathcal{S}$. Therefore a perfect split resulting in two subsets each containing all data points of only one class c_i would have a maximal information gain of $I = 1$ depending on $\mathcal{S}$. The information gain can be defined flexibly or can be exchanged with other metrics (e.g. Gini coefficient) that fits the data best.

The training process starts at the root node and continues after each successful split at its child nodes until a stopping criterion is met. The weak learner $h(\mathbf{v}, \boldsymbol{\theta}_j)$ is characterized by its parameters $\boldsymbol{\theta} = (\boldsymbol{\phi}, \boldsymbol{\psi}, \boldsymbol{\tau})$. The parameter $\boldsymbol{\tau}$ defines the threshold for the inequalities used in the binary test, $\boldsymbol{\phi}$ contains a selection of features taken from the whole feature vector $\mathbf{v}$ and $\boldsymbol{\psi}$ defines how the data gets separated. Especially $\boldsymbol{\psi}$ is adaptable to a large extent. It comprises of the sub-attributes $\boldsymbol{\psi}_O$ and $\boldsymbol{\psi}_T$. While $\boldsymbol{\psi}_T$ defines the split function type that will be used, $\boldsymbol{\psi}_O$ generates a vector of four offset positions that are located within the patch's region. These offset positions get used by every split function type in a different way.

Usually it defines a geometric primitive that is used to separate the data (e.g. an axis-aligned hyperplane, an oblique hyperplane, a general surface *etc.* [50]). In this work the split functions are specialized for the classification of pixels respectively image patches. The implementation used in this work is based on [51] by Kainz et al. It defines four different split function types $\boldsymbol{\psi}_T$: (a) single pixel value, (b) pixel value difference, (c) Haar-like features and (d) constrained pixel value difference.

Deviating from the aforementioned definition of the input vector, $\mathbf{v}$ does not comprise of scalar values but of image patches centred around the pixel to be classified. This way a pixel is classified not only by its own feature values but also by the feature values of its surrounding.

These are the four split function types $\boldsymbol{\psi}_T$ used:

(a) The single pixel value function is the most straight forward one: It gets the feature vector $\mathbf{v}$ at the offset position $\boldsymbol{\psi}_{O_1}$, takes a random value selected by $\boldsymbol{\phi}$ from it and compares it to $\boldsymbol{\tau}$. If it is larger it gets assigned to $\mathcal{S}^L$ otherwise it gets assigned to $\mathcal{S}^R$.

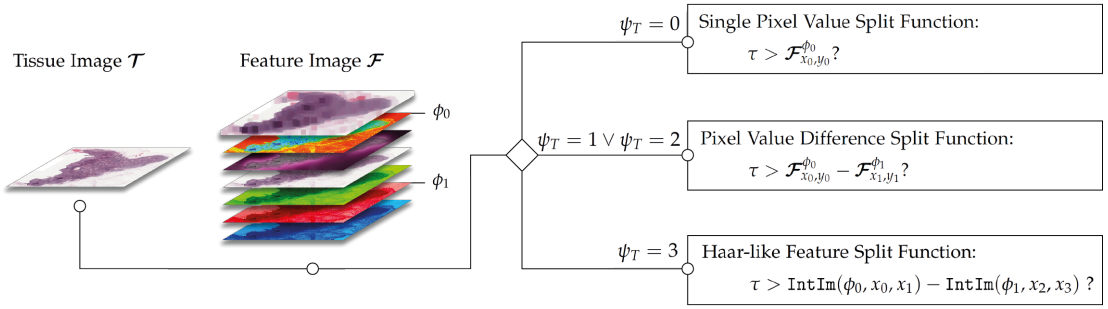


Figure 14: Functioning of the random forest: Each pixel is represented by an image patch which is centered on the current pixel. A feature image $\mathcal{F}$ is computed from said patch in the next step. Afterwards ϕ determines the feature image dimensions that are used and ψ_O defines a vector of four offset positions within the patch that are used by each split function type, defined by ψ_T, diferently. The function `IntIm` returns the integral image of the specified dimension ϕ_i of the feature image $\mathcal{F}$ and the rectangle defined by two offset positions respectively.

(b) The pixel value difference function compares the difference of two feature values. It takes two feature vectors at the offset positions $\boldsymbol{\psi}_{o_1}$ and $\boldsymbol{\psi}_{o_2}$, extracts feature values at $\boldsymbol{\phi}_1$ and $\boldsymbol{\phi}_2$ and compares their difference with $\boldsymbol{\tau}$. This way $\boldsymbol{\phi}_1$ and $\boldsymbol{\phi}_2$ might be of the same channel but most certainly compare values of two different channels.

(c) In a similar way the Haar-like feature function compares the difference between two values with a randomly chosen $\boldsymbol{\tau}$. The first value is defined as the sum of pixel values of channel $\boldsymbol{\phi}_1$ that are located inside a rectangle with $\boldsymbol{\psi}_{O_1}$ as the upper left corner and $\boldsymbol{\psi}_{O_2}$ as the lower right corner. The second value is computed analogously using the variables $\boldsymbol{\psi}_{O_3}$, $\boldsymbol{\psi}_{O_4}$ and $\boldsymbol{\phi}_2$. For the computation of this feature, integral images, first introduced by F. Crow [52], were used.

(d) The fourth split function type is the constrained pixel value difference which works the same way as the pixel value difference function that was introduced before except for one thing: The second offset position $\boldsymbol{\psi}_{O_2}$ gets chosen in dependence of $\boldsymbol{\psi}_{O_1}$. It must be located within a 10 pixel radius. Apart from this it works the same way.

These are all the alterable variables that define a split function $h(\mathbf{v}, \boldsymbol{\theta}_j)$. Figure 14 illustrates the functioning of the random forest and the influence of all relevant parameters. If $\mathcal{T}$ is the infinite set that emerges from the cartesian product of $\boldsymbol{\phi} \times \boldsymbol{\psi} \times \boldsymbol{\tau}$

a subset $\mathcal{T}_j$ with $\mathcal{T}_j \subset \mathcal{T}$ of user defined size exists. Such a subset gets generated every time the training process of a node j starts. The training starts at the root node $j = 0$ with a Set of data points $\mathcal{S}_0$ and a set of parameter combinations $\mathcal{T}_0$. The goal is to define a split function such that the information gain is as high as possible by optimizing

$$\boldsymbol{\theta}_j^* = \arg\max_{\boldsymbol{\theta}_j \subset \mathcal{T}_j} I_j \tag{3.2}$$

with

$$I_j = I(\mathcal{S}_j, \mathcal{S}_j^L, \mathcal{S}_j^R, \boldsymbol{\theta}_j). \tag{3.3}$$

After $\boldsymbol{\theta}_j$ was found the set $\mathcal{S}_j$ gets split into $\mathcal{S}_j^L$ and $\mathcal{S}_j^R$ according to the split function defined by $\boldsymbol{\theta}_j$ and the training process continues at the child nodes of $\mathcal{S}_j$ and their related sets of data points $\mathcal{S}_j^L$ and $\mathcal{S}_j^R$.

The training stops when a stopping criterion is reached. Commonly used stopping criteria are: (a) maximum tree depth is reached, (b) minimum number of samples necessary for a split is not reached or (c) information gain of a split is below a certain threshold.

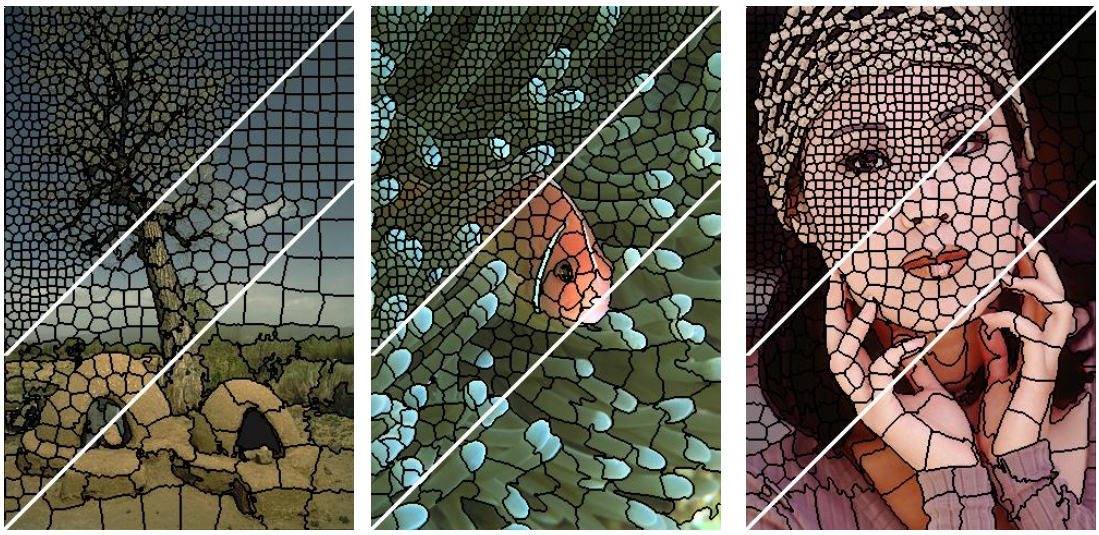
After the training has stopped the predictor model $p(c|\mathbf{v})$ of each leaf node gets updated. In a regression scenario this gets done by simply computing the mean of all the probability vectors $\mathbf{p}$ that are assigned to the data points that alighted in this leaf node. In a classification scenario $p(c|\mathbf{v})$ does not return a continuous value but just the class $c \in c_k$ that most data points in the leaf in question are assigned to. Therefore it is only necessary to find class c that occurs most frequently.

Evaluation

Compared to the training process the evaluation is more straight forward. To evaluate an unlabelled data point $\mathbf{v}$, it gets pushed through all trees t_i of the whole random forest. In a regression scenario each tree returns a probability vector $\mathbf{p_t}$ that got computed by the predictor model $p_t(c|\mathbf{v})$ of the leaf the data point alighted in. To get the final probability vector $\mathbf{p}$ again the mean of the response vectors of all the trees is computed:

$$p(c|\mathbf{v}) = \frac{1}{T} \sum_{t=1}^{T} p_t(c|\mathbf{v}) \tag{3.4}$$

In a classification scenario no averaging takes place but only the class most trees predicted gets selected.



(a) (b) (c)

Figure 15: Three different images segmented into superpixels of size $\sim 64 \times 64$ pixel, 256×256 pixel and 1024×1024 pixel using SLIC [46]

3.1.2 Superpixels

Superpixels are sets of regular pixels that share similar properties and therefore form "perceptually meaningful atomic regions" that can be used in image processing instead of the regular, rigid pixel grid [46]. The reasons for using superpixels are diverse but most important they can reduce the complexity of image processing tasks when used as primitives from which to compute image features [46]. Figure 15 shows three different superpixel images that were generated using SLIC where each superpixel is outlined in black. In order to reduce the chane that a superpixels gets centred on an edge or a noisy pixel, all centres are moved to seed locations with the lowest gradient in a 3×3 meighborhood.

The algorithm for superpixel generation used in this work is called SLIC. It is an adaptation of k-Means that is used for superpixel segmentation. The feature vector is a combination of the three color channels of the CIELAB color space $[l\ a\ b]^T$ and the pixel's position $[x\ y]^T$. The only input that the SLIC algorithm needs is k, the desired number of approximately equally sized superpixels. This value is used in an initialization step where k cluster centres $C_i = [l_i\ a_i\ b_i\ x_i\ y_i]^T$ are sampled on a regular grid with a spacing of $S = \sqrt{\frac{N}{k}}$ where N is the overall number of pixels in the image to be segmented. In a next step each pixel i gets assigned to the centre C_k, that is closest to it regarding to the distance function D. To speed up the segmentation process the

amount of distance calculations per pixel is limited to those centre points lying within a search radius of $2S$.

This distance measure D needs to combine two distances, spatial distance and the distance of two colors in the CIELAB color space. The distance d_s between two pixels is defined as the euclidean distance between two points: $d_s = \sqrt{(x_j - x_i)^2 + (y_j - y_i)^2}$. In a similar way d_c is defined: $d_c = \sqrt{(l_j - l_i)^2 + (a_j - a_i)^2 + (b_j - b_i)^2}$. The spatial distance is normalized by S because it is the maximum distance a pixel i can have to the centre pixel it might get assigned to. Since the determination of a maximum color distance is not as straightforward d_c gets normalized by a constant m. Therefore the resulting distance function D is

$$D = \sqrt{\left(\frac{d_c}{m}\right)^2 + \left(\frac{d_s}{S}\right)^2} \tag{3.5}$$

This way m can be used as an input variable that controls the superpixels compactness. The larger m becomes the more important spatial proximity becomes and vice versa; when it becomes smaller superpixels are less regular shaped and converge towards image boundaries. After all pixels were assigned to centres each centre's feature vector gets updated to the mean $[l\ a\ b\ x\ y]^T$ vector of all pixels belonging to the cluster. These steps are repeated iteratively for a fixed number of rounds; Achanta et al. [46] found that 10 iterations for most images suffice.

3.2 Design Methods

Since this work is addressing a very specific group of expert users it is of great importance to meet their needs when designing user interactions. The following section gives a brief introduction in the work of two influential experts in the field of design, usability engineering, human-computer interaction, and cognitive science: D. Norman and B. Shneiderman.

One of the most important concepts made popular by D. Norman is *user-centered design* (he refers to it as human-centered design), which initially was introduced by Plea et al. in 1987 [53, 54]. It was addressed at the problems of the prevalent technology-centered design.

Traditionally applications were designed in a technology-centered way: Engineers built systems and provided the user with displays showing information about the performance or current state of the system. With the ongoing progress of technology the amount of data grew exponentially and therefore did the information overload for

the user. In the scenario of a pilot operating an aircraft cockpit he had to find, sort, integrate, and process the information that is needed from all that which was available [55]. This means a user had to adapt to technology in order to perform a task and not vice versa.

On the contrary user-centered design reinforces the development of interfaces that consider the capabilities and needs of the user. Rather than displaying all the available information a user-centered design integrates them in a way that supports the users goals, tasks, and needs [55]. Endsley et al. [55] emphasize situation awareness of a user as a main criteria for the success of a decision process in most real-world scenarios. They formulated three principles that they think are the key to user-centered desgin: Technology should be organized...

...around user's goals, tasks, and abilities.

...around the way users process information and make decisions

...such that the user is kept in control and aware of the state of the system.

The first item of their list is pretty straight forward. But it is not addressed at simple, linear, and repetitive tasks but rather at more complex tasks with varying tasks time and no prescriptive order of tasks. In those kind of systems interfaces and capabilities need to be designed so that they support the changing goals of the user. Furthermore it is important when designing such systems to consider perceptual and physical capabilities and also mental abilities of the user.

The second item considers the fact that a difference exists between idealized model of decision making and the results from research that has been conducted on how people make decisions in real-world scenarios. Rather than optimizing across possible alternatives the human mind first classifies and tries to understand a situation which triggers the appropriate response from long-term memory, proceeding to action selection [55, 56, 57, 58]. This means situation awareness is a key feature highly influencing the success of a decision process. Therefore it is important in complex situations to not only perceive the state of the system as a user but to understand the integrated meaning of what is perceived regarding the current goal. To ensure the user is making "correct" decisions resulting in "correct" actions it is important to design interactive systems to support the user's ability to gain and maintain situation awareness in complex and dynamic environments [55].

The last item is addressing the phenomenon that users can get out-of-the-loop when using systems with a high level of automation. Endsley et al. [55] therefore state that even when a user does not need to perform a task because of automation he still needs

be in control of what the system is doing in order to maintain the situation awareness. User-centered design therefore must address in which way automation blends with user interactions.

Over the years the concept of user-centered design evolved and even though many modified versions came up through the years all versions are variants of the common theme: iterate through the stages of observation, generation, prototyping and testing. But even though this a commonly used and useful technique it is complicated to put into praxis. D.Norman puts this problem in his book "The Design of Everyday Things" [54] like this:

> "The HCD (human-centered design) process describes the ideal. But the reality of life within a business often forces people to behave quite differently from that ideal." (Don Norman)

For this work this means that the involvement of a well payed pathologist in an iterative design process was not feasible. But essential core assumptions of user-centered design, like the one presented by Endsley et al. [55], were applied during the design process of the interface and user interactions.

In contrast to Normans work Shneidermans *Visual Information Seeking Mantra* [59] is not addressed at getting to know the user but rather at designing interactive user applications in a more general sense. The most important and also most popular part of his design guideline is summarized in said Visual Information Seeking Mantra [59]:

> "Overview first, zoom and filter, then details-on-demand." (B.Shneiderman)

It serves as starting point for designing advanced graphical user-interfaces especially with regards to direct-manipulation interfaces. D.Norman summarizes his guideline at a high level of abstraction by seven tasks:

Overview Give an overview over the entire dataset. This will be the starting point of the exploration and will help the user to build a mental model

Zoom In order to explore the dataset the user needs to zoom in on items of interest. Craft et al. [60] also stated that zooming can be regarded as filtering by navigation.

Filter To avoid obstruction by irrelevant items it is important to be able to filter out these items.

Details-on-demand When the user found an item or a group of items of interest he should be able to explore it in further detail

Relate Display relationships between items and support the user to find items that relate by certain similarities.

History When exploring unseen datasets the user tends to use a trial and error method to find items of interest. Therefore it is important to enable the user to return to a previous state.

Extract Allow the extraction of single items or item groups. In a lengthy discovery process one finding might be of interest for later use in ongoing tasks or work projects [60].

Shneiderman chose to issue his guideline in a compact form without proposing large tool sets or framework. Therefore his mantra got used in many different ways usually even without actually stating in which way it was used [60]. As Craft et al. stated in their literature studies regarding the Visual Information Seeking Mantra it mostly got used as a starting point for designing applications in general. This also applies for this work: Parts of it were used during the design process others were omitted on purpose. When explaining the intentions behind different design choices the Visual Information Seeking Mantra is referenced in section 4.

3.3 Formalisms

The symbols denoted in this work are written as follows. Scalar variables: lower case (e.g. n); vector: bold lower case (e.g. $\mathbf{v}$); sets: upper case, caligraphic (e.g. $\mathcal{S}$); matrices: bold, upper case, caligraphic (e.g. $\boldsymbol{\mathcal{M}}$). The value of a pixel of channel d at position (x, y) in image $\boldsymbol{\mathcal{M}}$ is noted as $\boldsymbol{\mathcal{M}}^d_{x,y}$, whereas $\boldsymbol{\mathcal{M}}_{x,y}$ accesses a vector $\mathbf{v} \in \mathbb{R}^d$.

$\boldsymbol{\mathcal{T}}$	tissue image of dimension $d = 1$
$\boldsymbol{\mathcal{F}}$	feature image of arbitrary dimension
$\boldsymbol{\mathcal{M}}$	proximity score map of dimension $d \in [1 - 6] \subset \mathbb{N}$
$\boldsymbol{\mathcal{C}}$	temporary classification result of dimension $d = 1$
$h(\mathbf{v}\|\theta_j)$	splitfunction taking a sample point $\mathbf{v}$ and parameters θ as input
$\boldsymbol{\theta} = (\boldsymbol{\phi}, \boldsymbol{\psi}, \boldsymbol{\tau})$	Split function parameters comprising of the following values:
	$\boldsymbol{\phi}$ Features used for comparison in the split function
	$\boldsymbol{\psi}_O$ Offset positions used for valuable samples
	$\boldsymbol{\psi}_T$ split function type
	$\boldsymbol{\tau}$ threshold used to define split
$p(c\|\mathbf{v})$	predictor model, it returns the probability of $\mathbf{v}$ belonging to class c
m	number of classes the user wants to classify
c	confidence slider value

4 Proposed Method

The initial goal of this work was to help the pathologist with finding regions of interest and to support him during the training and evaluation process of the random forest algorithm.

The following section will describe how the searching process was intended to work and why it was not successful. Afterwards the proposed method for training and evaluating the random forest will be presented. After describing the main interaction pipeline the individual parts are presented in detail. Even more important to this part than the tissue classification algorithm is the user interaction which hides all implementation detail from the user so the user can focus on the actual classification process.

4.1 Requirements

The Requirements for this work's method are aligned with mostly two things: (a) the characteristics of WSIs and (b) the capabilities of pathologists working with them during their daily routines. The following list contains requirements but not features for this works' method:

Do Not Make the User Adapt Make the system adapt to the users needs so that the user can keep working in his standard domain, like Endsley et al. [55] suggested it in their design guideline (see section 3.2. This is important in the interest of keeping the mental effort as low as possible. If a pathologist had to deal with a tissue diagnosis task during his daily routine and additionally had to make an effort to cope with non domain related visualizations this would affect his performance in a negative way .

Do Not Require Prior Knowledge The point which makes this approach unique towards other methods is that it does not rely on prior knowledge. Neither labeled datasets are needed to train an initial classifier nor is the approach based on characteristics of one specific tissue type, like the approach by Bahlmann et al. [38] (see section 2.1). It is as generic as possible for a tissue classification approach; further fields of application and how this method needs to be adapted in other fields are discussed in section 7.

Real-time Capability Of great importance for the satisfaction and the overall success of the classification task is a seamless workflow. Therefore the approach shall provide the user with real-time feedback. Operations in which the user does not need to control the system, for instance when evaluating the WSI, longer computation times are acceptable as long as the user is aware of these conditions. This premise is based on the guideline about user-centered design by Endsley et al. [55] (see section 3.2).

Support Consumer-level Hardware In real world scenarios pathologists rarely are equipped with a computer cluster, like the one that was used by Kong et al. [41] ,but rather with normal consumer-level hardware. Therefore this works method should be executable on normal consumer-level hardware. On the other hand it is imaginable that heavy computation processes can be outsourced via cloud computing, which gets discussed in section 7 . But everything that involves user interaction should be computable on the user's computer.

Support Batch Processing Since WSIs at resolutions on which pathologists actual examine them can be larger than 20 000 pixels per side their corresponding feature images tend to get to large to fit into RAM (Random-access memory) at once it is of importance that is possible to evaluate separate patches without to much computational overhead. This item refers to problems with large filter kernels and discontinuous patch boundaries.

4.2 Interaction Design and -Pipeline

The main idea behind the developed interaction pipeline is to abstract the machine learning algorithm in a way that the user does not need to know how it works in order to use it. To emphasize this goal the user interaction relies on direct manipulation in the image space which is common territory for pathologists.

Hence, in the following the workflow is laid out without technical implementation details, but only from a users perspective.

After loading the WSI the user can use several options to adjust the datasets appearance. Amongst them are easy color adjustment tools for histogram manipulation as well as a widget for color channel selection, which is especially interesting for e.g. fluorescence microscopy where each channel not necessarily maps to channels of common color spaces like RGB, CIELAB or HSV.

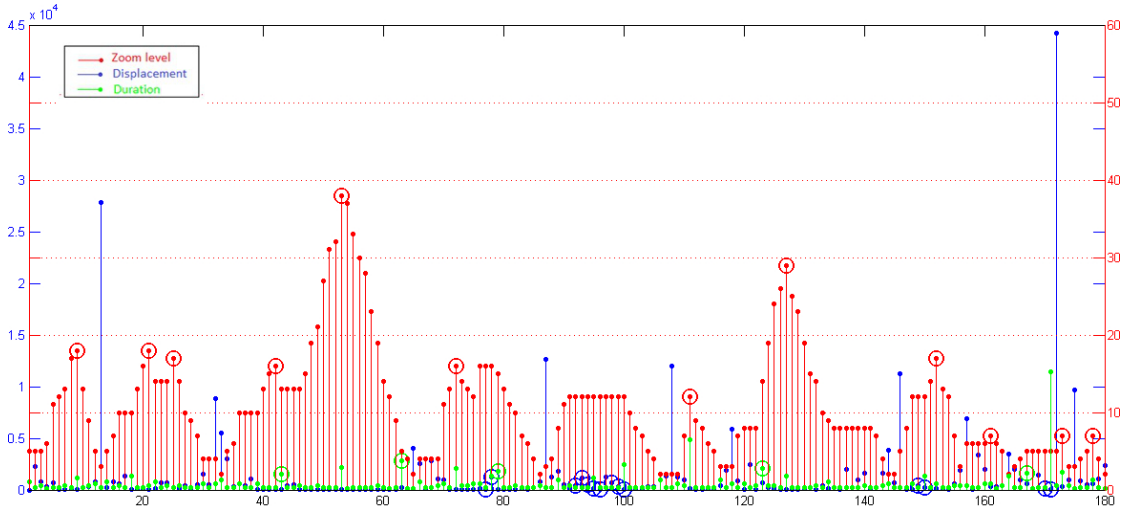
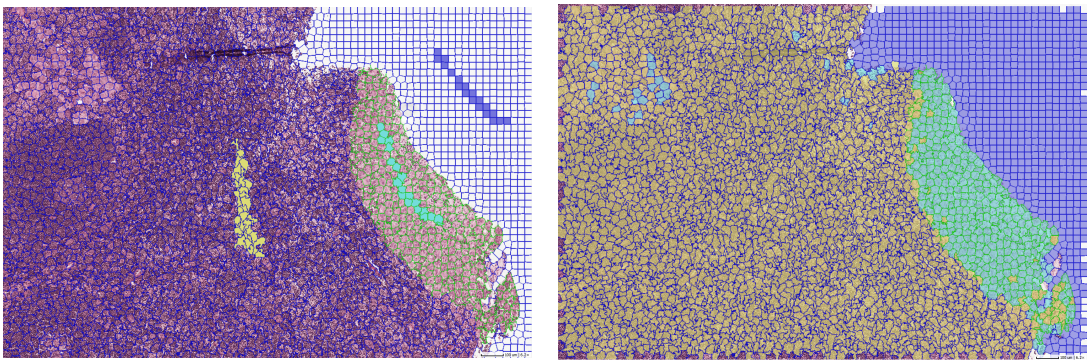


Figure 16: Characteristic sequence of zooming and displacement interactions a pathologist executes when examining a WSI [37].

When the user starts the actual tissue classification process the interface does not change a lot aside from the associated widget, which gets enabled. As mentioned earlier WSIs do not exhibit a high visual contrast or show any specific characteristics for location or connectivity of different tissue types. In contrast to this e.g. whole body scans and many other medical imaging modalities enclose such visual clues. Therefore the user can activate an overlay "heat map" that emphasizes different regions that might be of interest. When the user changes the viewport the overlay does not update, but



(a) The strokes needed for brushing a class.

(b) Initial classification result.

Figure 17: Figure (a) shows the brush strokes need for defining a tissue type of interest. The initial classification result that is computed using the input from figure (a) is depicted in figure (b). The tissue area in which the border lines of the superpixels are colored green is defined as cancerous tissue by the groundtruth provided by the dataset (see section 5.1.2 for further information about the dataset).

stays at the same place and the same resolution. If the user wants to update the overlay at a certain section of the WSI he simply has to toggle the update button. A continuous update is not feasible since the computation is to demanding to do it in real-time. The same applies to frequent updates that are toogled when the viewport did not change for longer than a threshold t_s. The user can choose different display styles for the heat map so it would not obstruct the underlying structure of the WSI to much; they are introduced in section 4.4.4.

This way the user can explore the WSI in a similar way like he would do with an analogue light microscope. He starts at a low resolution and searches for visual clues that indicate the presence of the tissue of interest. When he found such clues he proceeds at a higher resolution. This characteristic process is visualized in figure 16 by Mercan et al. [37] who analysed the exploration process pathologists undergo when examining a WSI.

When the user found a ROI he is asked to brush all the tissue types he is interested in. Short strokes that are not any longer than 100px on the screen are sufficient; an example is depicted in figure 17a. A maximum of six different tissue types can be selected. After the user is done selecting tissue types the actual training process starts. After he clicked on 'done' a rough, initial classification result like it is depicted in figure 17b is displayed to the user. With a brush and an eraser the user now starts adjusting the classification result while training the random forest in the back-end. The special

thing about this interaction is that while the user brushes a certain area the whole viewport gets updated not only the area he brushed in. This way similar types of tissue get also classified as the tissue of interest or on the contrary when using the eraser are no longer assigned to this class. The characteristics of this interaction style are depicted in figure 18. While brushing and erasing the user can switch between several different visual styles for the representation of results and context visualizations; they too are introduced in section 4.4.4.

After the user is pleased with the segmentation result he can continue his exploration in the traditional way by zooming out and searching for further clues or he could click 'next' and use the knowledge the random forest gained so far. By clicking next the random forest implementation selects a new section of the WSI where tissue of interest might be located. The segmentation continues in the new viewport with a short delay.

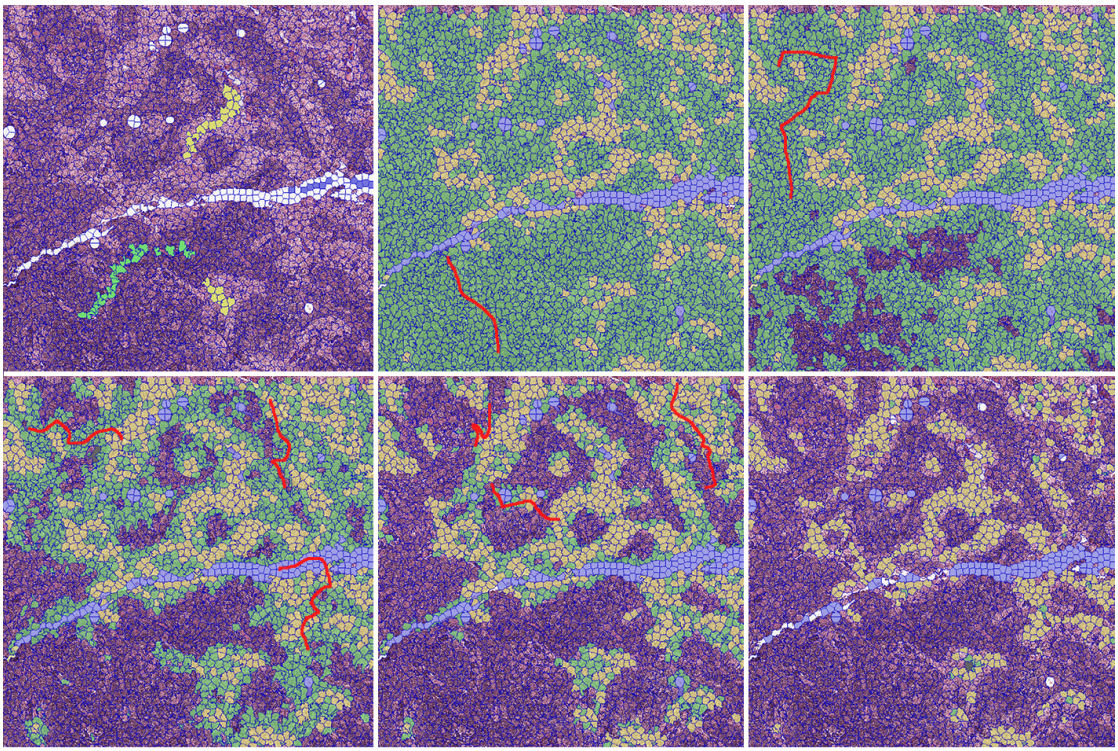


Figure 18: The interaction technique entails that not only regions which brushed are update but also similar regions. The first image shows the initialization step in which exemplarily three tissue types are selected. The red strokes indicate the use of the eraser. It is shown how the green class gets removed. The updating procedure works the same way the other way around when using the brush tool.

This procedure repeats itself a few times but at a maximum of nine times. The user can stop the process at any time by clicking on "Apply to Whole Slide". This starts the

evaluation process in which the entire WSI is classified. At this point the pathologist is not required to interact with the framework any more. Since the evaluation of a WSI takes several ours (at least on normal, consumer-level hardware) it is important to plan the point at which the evaluation should start with caution. Different ways on how the evaluation could be implemented in praxis are discussed further in section 4.4.3. Anyhow the user finished the training process for one WSI and now can either evaluate it in its entirety or proceed with a next WSI.

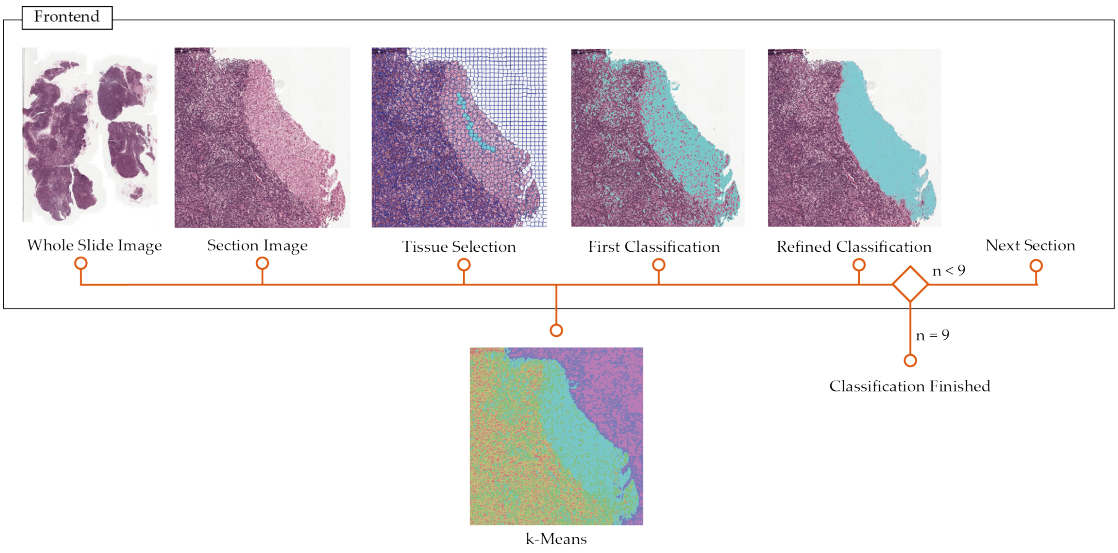


Figure 19: The steps required for the classification of one section image: The classification starts by presenting the user with the entire WSI. By using the zoom and displacement he specifies the first ROI that gets classified. The tissue of interest gets selected by simple strokes as depicted. Regardless of how many different tissue types were brushed by the user the section image is classified into six classes by a k-Means algorithm. From the classification result all classes the user is interested in, are extracted and are used to train a first classifier. Unlike in the case of the k-Means classification result the classification result generated by the random forest now gets displayed. During the next step the user is required to refine the segmentation by using the brush and eraser. After the user is satisfied with the result, he can either request a new section or continue the exploration in a classical way. If he already classified nine section images the classification is finished. Otherwise he is presented with a new section image. The procedure of finding new sections of interest is covered in 4.4.1.

The whole pipeline can be seen in figure 19 and is explained stepwise in the following sections with technical details regarding the random forest implementation included.

4.3 Finding Regions of Interest

When Examining WSIs pathologists are navigating through large areas which hold no information for them. This is due to the fact that those tissue sections are less structured than other medical data like bodyscans. Different tissue types rarely feature characteristics regarding connectivity or location. But WSIs hold a lot of texture information that can be used to guide the pathologist during the examination. This section covers the first step of the interaction pipeline in which the user is presented with the WSI at the lowest resolution and is starting to look for visual clues that indicate the presence of the tissue he is looking for.

The visualization that is offered in this step is a heat map-like overlay that increases the visual contrast and thereby helps the pathologist to detect visual clues. Implementation details are important regarding two things: (a) how the heat map gets created in the first place and (b) how does it get updated in an interactive environment.

The main premise for such an overlay is that it is fast, easy to be updated on higher resolution or other image sections and most important that it increases the users insight. Therefore this approach uses a k-Means implementation to cluster the section of the WSI that is in the viewport.

The k-Means algorithm takes 10% of the sections pixels as samples and clusters them into six classes. Each pixel is represented by a feature vector of size six. The features are taken from the following six feature images computed from the RGB section image: Feauture images 1-3 are the channels l,u and v of the LUV colorspace, feature image 4 is a histogram equalized grayscale image, and feature images 5 und 6 are local binary pattern (LBP) images with radius one and three. They were chosen because they were shown to be valuable discriminators while being relatively easy to be computed. In section 4.4.2 all features get introduced in detail and their performance get analysed (see figure 22). A total of six features were chosen because k-Means is easily susceptible to the curse of dimensionality (see section 2.1).

The amount of six k-Means cluster was chosen based on two things: (a) the human color perception and (b) the performance of the k-Means algorithm. C.G.Healey [61] recommends not to use more than five to seven colors at a time because the human cognition has troubles to remember the color coding of this many items. Furthermore a higher k would result in more operations the k-Means had to perform which would result in an increase of computation time (with features $f \in \mathbb{R}^6$, $k = 6$ and $\sim 50\,000$ samples the clustering takes ~ 70 ms).

The output of the k-Means Clustering is a set of six cluster centers $c_i \in \mathbb{R}^6$. In a next

step each pixel in the section gets assigned to the cluster i its feature vector is closest to. Each cluster is assigned to a color h_i and all pixels p_i are colored in the color of the cluster they were assigned to. This way an image segmentation is generated, which the user can use in synergy with the actual WSI to search for ROIs. Again the user can chose between different visual representations discussed in section 4.4.4 .

The next important thing is that the color coding stays the same when the user updates the overlay. This becomes an issue because the k-Means algorithm chooses its cluster centers randomly. Therefore it is unlikely that a pixel p_i is colored in the same color in two successive passes of clustering. The upper row in figure 20 shows such a random assignment of colors.

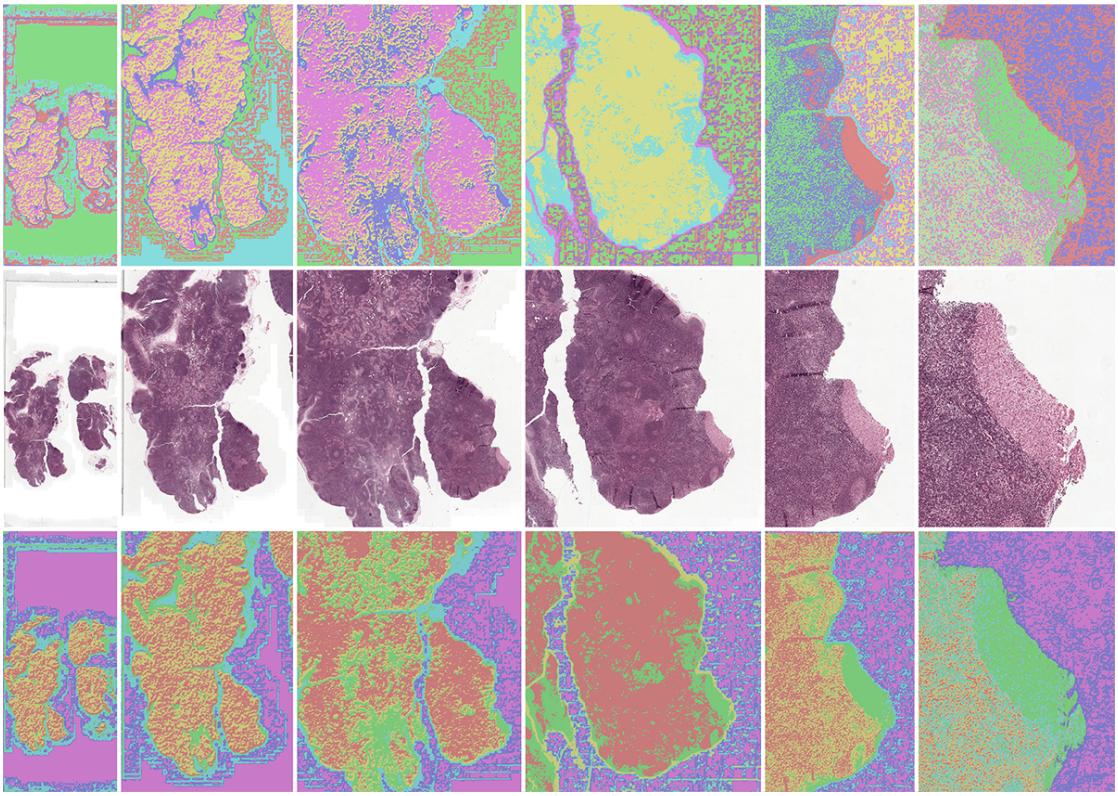


Figure 20: Zooming in on a ROI with heat map-like overlay visualizations. On top a normal k-Means with no special assignment technique of cluster centers to colors is depicted, while on the bottom an assignment technique is used which ensures a more steady scheme during zooming and displacement operations.

To deal with this problem the cluster centers c_i are brought into an order that remotely stays the same during zooming or displacement operations. Therefore the dimension of the feature space with the best separation gets determined. This is done by using the definition for the diameter D of a cluster by Zhang et al. [62]. It is defined as the

average pairwise distance between two data points within a cluster. The smaller the diameter the more compact is the cluster in question.

$$D = \sqrt{\frac{\sum_{i=1}^{N}\sum_{j=1}^{N}(\mathbf{x}_i - \mathbf{x}_j)^2}{N(N-1)}} \tag{4.1}$$

This formula now gets adapted so that it finds the dimension d^* in feature space $\mathbb{R}^6$ which has the largest diameter:

$$d^* = \arg\max_{d\in[1,6]} \sqrt{\frac{\sum_{i=1}^{k}\sum_{j=1}^{k}(\mathbf{c}_{d,i} - \mathbf{c}_{d,j})^2}{k(k-1)}}, \tag{4.2}$$

with c_d, i being the scalar in feature dimension d of the i-th cluster center c_i. Now that the dimension with the best separation is found, all k clusters are sorted in ascending order according to their scalar feature value of dimension d. Now each cluster center c_i gets assigned to color h_i. This way the resulting clusters are not assigned to random colors. Only when the difference between two successive section images is to large color assignments can become unsteady between both clustering results. But according to the work by Mercan et al. [37] this is not characteristic for pathologists examination procedure (see figure 16). Figure 20 shows the effect of this procedure resulting in more steady color schemes.

4.4 Training and Evaluation Process

The most essential part of this work lies in the user interaction framework that covers the training and evaluation process of the random forest. Given the fact the user found a ROI using the overlay visualization presented in the previous section, section 4.3 he now starts the segmentation process while training the random forest in the back-end, which is covered in the next section, section 4.4.1. The next step in the interaction pipeline is the evaluation which is covered in section 4.4.3. Different visual representations that can be chosen by the user are presented and discussed regarding their relevance and additional value for the user in section 4.4.4 and the evaluation of the used set of parameters is to be found in section 4.4.5.

4.4.1 Using Regression to Classify Tissue

This subsection covers the training-component of the method which is the basis of the whole tissue classification approach. It is loosely based on the nucleus detection

approach by Kainz et al. [51]. In section 3.1.1 some parts of it regarding the training procedure were already introduced which now get applied in praxis.

The segmentation process starts when the user found a ROI and, even as important, a resolution at which he is able to distinguish between the tissue of interest and the surrounding tissue. This is important because a multi-scale approach like the one by Kong et al. [41] presented in section 2.1 is complex and most certainly not appropriable on consumer-level hard ware. This way the task of choosing a sufficient resolution is given to the user and all further analysis will be done on this level.

At this point six cluster centers $\mathbf{c}_i$ from the k-Means clustering are already available, which now get used to compute a confidence map $\boldsymbol{\mathcal{M}}$. The confidence map is an image of the size of the viewport and therefore is as large as the tissue section image $\boldsymbol{\mathcal{T}}$ taken from the WSI which is about to get classified. In its initial form it comprises of six channels with values $x \in [0,1] \subset \mathbb{R}$. The value of a pixel in image $\boldsymbol{\mathcal{T}}$ of channel d at position (x,y) is noted as $\boldsymbol{\mathcal{T}}^d_{x,y}$. Likewise the feature image $\boldsymbol{\mathcal{F}}$ computed from the tissue section image has the same dimensions and comprises of six channels: l,u,v,histogram equalized grayscale image, lbp1, lbp3 (see section 4.3).

Hence $\boldsymbol{\mathcal{M}}_{x,y}$ of the confidence map is a vector storing the probabilities of $\boldsymbol{\mathcal{T}}_{x,y}$ belonging to each of the six cluster centers $\mathbf{c}_i$ computed by k-Means. Therefore the following applies:

$$\sum_{d=1}^{6} \boldsymbol{\mathcal{M}}^d_{x,y} = 1, \tag{4.3}$$

with (x,y) being randomly chosen indices that lie within the image dimensions. To achieve that for each pixel (x,y) the distances to each cluster center $\mathbf{c}_i$ gets computed resulting in a distance vector $\mathbf{v}_{x,y} = (v^1_{x,y}, \ldots, v^k_{x,y})^T \in \mathbb{R}^k$, with

$$v^i_{x,y} = \sqrt{(c^1_i - \boldsymbol{\mathcal{F}}^1_{x,y})^2 + \ldots + c^6_i - \boldsymbol{\mathcal{F}}^6_{x,y})^2}\ ,\ i \in [1,k] \subset \mathbb{N}. \tag{4.4}$$

Since a smaller distance to a cluster center indicates a higher probability that a pixel gets assigned to this particular cluster center, an auxiliary variable $\hat{\mathbf{v}}_{x,y}$ gets introduced. Furthermore the resulting probability needs to be within the range $[0,1] \subset \mathbb{R}$. Hence $\hat{\mathbf{v}}_{x,y}$ is defined as follows: $\hat{\mathbf{v}}_{x,y} = \mathbf{1} - \frac{\mathbf{v}_{x,y}}{|\mathbf{v}_{x,y}|}$, with $\mathbf{1} = (1, \ldots, 1)^T \in \mathbb{R}^k$. Now the values of $\hat{\mathbf{v}}_{x,y}$ are in a range of $[0,1] \subset \mathbb{R}$ and sorted by highest probability. The only thing left to do is to normalize it again, so that equation 4.3 applies; hence $\boldsymbol{\mathcal{M}}_{x,y} = \frac{\hat{\mathbf{v}}_{x,y}}{|\hat{\mathbf{v}}_{x,y}|}$.

In the following step the concept of superpixels gets used for the first time. The tissue image got segmented into a set $\mathcal{S}$ of superpixels. Each superpixel $\mathcal{P}$ again is

another set of an arbitrary amount of pixels themselves. A pixel $p_{x,y}$ that belongs to a superpixel $\mathcal{P}_i$ is noted as $p^i_{x,y}$.

When the user is told to brush tissue of interest not a regular brush tool is used but all pixels are selected that belong to a superpixel that was brushed. Hence m sets of superpixels with $m \in [1, \cdots, 6]$, comprising of sets of pixels are brushed, each one belonging to a tissue type of interest. Therefore they are noted as $\mathcal{S}_1, \ldots, \mathcal{S}_k$. Using the superpixels for refining the user input is beneficial for the user since he can use the brush less accurate and the resulting selection still will be accurate enough because the superpixel comprise of homogeneous areas.

To determine which class was meant to be brushed by each stroke each set of superpixels gets analysed. A temporary classification result $\boldsymbol{\mathcal{C}}$ with one channel and values $x \in [1, k] \subset \mathbb{N}$ get computed using the confidence map $\boldsymbol{\mathcal{M}}$. Each pixel in $\boldsymbol{\mathcal{C}}_{x,y}$ gets assigned to the index of the highest value of the vector $\boldsymbol{\mathcal{M}}_{x,y}$:

$$\boldsymbol{\mathcal{C}}_{x,y} = \underset{i \in [1,\ldots,k]}{\arg\max}\, \boldsymbol{\mathcal{M}}^i_{x,y}. \tag{4.5}$$

In a next step for each set $\mathcal{S}_i$ a histogram comprising of k bins is built from $\boldsymbol{\mathcal{C}}$. The histogram for $\mathcal{S}_i$ is defined as follows:

$$H_j = \sum_{p_{x,y} \in \mathcal{S}_i} \kappa(x, y, j) \tag{4.6}$$

with:

$$\kappa(x, y, j) = \begin{cases} 1 & \text{if } \boldsymbol{\mathcal{C}}_{x,y} = j \\ 0 & \text{otherwise,} \end{cases} \tag{4.7}$$

and $j \in [0, \ldots, k]$. Now the set $\mathcal{S}_i$ gets assigned to class k^* with the most cases in its bin:

$$k^* = \underset{j \in [1,\ldots,k]}{\arg\max}\, H_j. \tag{4.8}$$

The last check that is done regarding the user input for the tissue selection is to make sure, that the selection was not done by chance but has statistical significance. This is done using the hoeffding inequality [63] in a way Domingos et al.[64] use it to optimize the split functions of their hoeffding trees.

The hoeffding inequality states that for a random variable Z with range R its true average $\overline{Z}$ would not deviate from an observed average $\widehat{Z}$ more than ϵ with an error-likelihood of δ:

$$\left|\widehat{Z} - \overline{Z}\right| < \epsilon \text{ ,where } \epsilon = \sqrt{\frac{R^2 \ln\left(\frac{1}{\delta}\right)}{2n}}, \tag{4.9}$$

and n being the number of instances [65]. Domingos et al. [64] used this for proving that an attribute X_a with the highest score regarding a scoring function G that needs to be maximized, is the correct choice. Hence after seeing n examples and a probability of $1 - \delta$ the hoeffding bound guarantees that $\Delta G \geq \epsilon$ with $\Delta G = G(X_a) - G(X_b)$ and X_b being the attribute with the second highest score regarding G.

In the given use case this inequality is used to ensure, that the found class k^* is actually the class the user brushed intentionally. In this adapted version k^{**} refers to the class having second most items in its bin. Range R in this case simply is the amount of classes $k = 6$ and the sample count n is defined by the size of the Set $|\mathcal{S}_i|$. For this use-case an error likelihood of $\delta = 0.05$ was found useful. A realistic application example would be as follows: The user brushed 10 superpixel, each comprising of 50 pixels each. Therefore the sample count is $n = |S_i| = 10 \cdot 50 = 500$. This results in $\epsilon = \sqrt{6^2 \ln\left(\frac{1}{0.05}\right) \frac{1}{2 \cdot 500}} \approx 0.328$. In a scenario where for instance 300 of all brushed pixels belong to k^* and 100 belong to k^{**}, the proof that k^* is the class the user brushed intentionally, with $\Delta G = G(k^*) - G(k^{**}) = \frac{300}{500} - \frac{100}{500}$ would be positive, because $\Delta G > \epsilon$ with $\Delta G = 0.4$.

If this check does not turn out to be positive, the user is asked to brush more superpixels belonging to the tissue type he is looking for and therefore increasing the confidence.

Now that for each of the m sets of superpixels $\mathcal{S}_i$ a corresponding class k_i was found, the proximity score map needs to be updated. All classes the user is not interested need to be deleted from the proximity score map $\boldsymbol{\mathcal{M}}$. Afterwards the proximity score map needs to be normalized again so that equation 4.3 becomes true again. Now the proximity score map comprises of m channels each containing the probabilities for a pixel to belong to a certain class of tissue.

This proximity score map is used to train a first version of the random forest. This way the random forest produces the same result as the k-Means but more importantly learns about the textural features of the tissue of the WSI and how to distinguish them. The forest gets trained using 1% of the tissue images $\boldsymbol{\mathcal{T}}$ pixels according to the procedure described in section 3.1.1 and the set of features introduced in section 4.4.2.

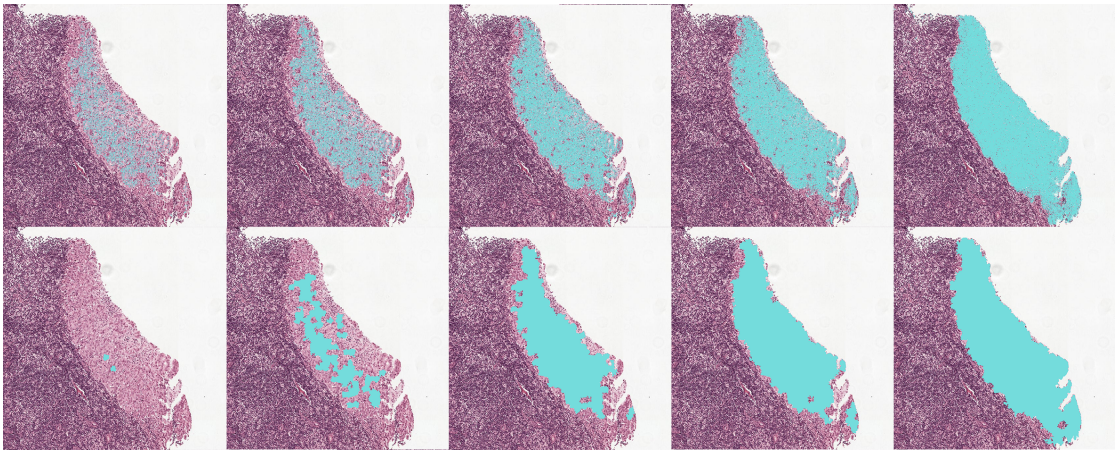


Figure 21: Distinctive features of both visualisations based on both pixel- and superpixel-wise evaluation (pixel-wise in top row, superpixel-wise in bottom row). While the pixel-wise visualisation appear more steady over time, the superpixel-wise evaluation better emphasises change and the current state of the classifier.

In a next step the section tissue image $\mathcal{T}$ gets evaluated using the built forest. The evaluation procedure also is described in section 3.1.1. The evaluation process results in a map of probability vectors $p_{x,y}$ *in* $\mathbb{R}^m$ comprising of the probabilities that pixel (x, y) gets assigned to each of the m classes of interest. This map now gets used as the new proximity score map.

Now that the first version of the forest is trained and a proximity score map $\mathcal{M}$ got computed the result needs to be visualized for the user to see. Two options of were explored: The first option already got explained earlier when describing how to compute the temporary classification result $\mathcal{C}$. This option would result in a pixel-wise evaluation displayed in comparison with other visualization styles in figure 21. The second option provides that a classification result gets computed per superpixel. This option has several advantages towards option one, but is less precise and detailed. Its advantages are that it cancels out the noise and during the interactive training part the change and the current state of the classifier is more obvious to perceive, which is depicted in figure 21. The visualization based on pixel-wise evaluation on the other hand is more susceptible to over-segmentation since changes in areas which the user does not focus go unnoticed more easily (change blindness).

The superpixel wise classification result gets computed in a similar way as equations 4.6 and 4.8 define it. In this case $\mathcal{S}_i$ is the superpixel that gets classified. After the class i was found said superpixel S_i got assigned to, all pixels $p_{x,y} \subset \mathcal{S}_i$ are classified as belonging to class i.

When assigning classes to colors an HSL-color model gets used where each color has the same saturation value S and the same lightness value L. The Hue value H gets uniformly sampled from the color space with an angle of $k/360°$ ($k = 6$) between two colors starting at 0° which corresponds to red.

The advantages and disadvantages of both visualization styles alongside with all options regarding the visual representations the user can chose from, are discussed in section 4.4.4.

At this point the user is faced with a first classification result, which most likely will not be perfect since until this point the user did not specify precisely which type of tissue he is looking for. His first and most simple option for refining the segmentation result is by adjusting the confidence slider's threshold value c. This works if the segmentation is a simple over- or under segmentation. The confidence value gets adjusted by a simple slider; to use it no special knowledge about the underlying implementation is needed. Immediate feedback gets displayed: (Super)-pixels with an average confidence value of class k^* greater than the slider value c, appear, while all others are suppressed. The mean probability $\overline{p}_{k^*}$ for superpixel $\mathcal{S}_i$ belonging to k^* can be formalized by the following equation:

$$\overline{p}_{k^*} = \frac{1}{H_{k^*}} \sum_{p_{x,y} \in \mathcal{S}_i} \gamma(x, y, k^*), \tag{4.10}$$

with

$$\gamma(x, y, j) = \begin{cases} \boldsymbol{\mathcal{M}}^j_{x,y} & \text{if } \boldsymbol{\mathcal{C}}_{x,y} = j \\ 0 & \text{otherwise.} \end{cases} \tag{4.11}$$

Using this threshold a superpixel $\mathcal{S}_i$ only gets colored in the color assigned to k^* if $\overline{p}_{k^*} > \xi$.

The second way of manipulation is to adjust the forest by using the brush and the eraser tool that were introduced in section 4.2. The main idea is that they either increase the probability of a class for a selected set of superpixels when using the brush or decrease said probabilities when using the eraser. When the user brushes the tissue (using either the brush or the eraser) after n superpixels that have been hit an update is triggered ($n = 3$ was found suitable for this use case). Each update is used for tuning the random forest depending on the implementation of the online random forest.

When an update gets triggered the following input is available to the algorithm: (a) the proximity score map, (b) three Superpixels $\mathcal{S}_1, \ldots, \mathcal{S}_3$ (with $n = 3$), (c) the index of class c_i that is of interest, and (d) ofcourse the type of brush b_t the user selected.

The brush type b_t either is assigned to $b_t = e$ when the eraser is selected, otherwise to $b_t = b$ if the brush is selected.

Many different approaches for the online adaption of a random forest exist, but not all of them are appropriate for this use case. Two common online approaches are presented in the following section plus the one that gets used in this works method. It is an adapted version of the approach that gets presented second.

The easiest and most straight forward way to implement an online random forest is to replace the worst performing trees in the forest with newly trained trees in regular intervals. Since random trees are highly efficient during evaluation it is possible to find the worst performing tree using a k-fold cross validation. After eliminating them they easily get replaced by newly trained trees at real-time. After incorporating those newly trained trees in the forest a fraction of the images pixels get evaluated by the updated forest and the proximity score map gets updated accordingly. Using the superpixel-wise representation for the classification result this results in a clearer visualization whereas the change is visualised more steadily over the curse of time using the pixelwise representation. In figure the difference between both visualizations is depicted. Another problem with this implementation is that this approach is addressed at classification tasks with changing input. When new trees are trained always a fraction of the training samples are drawn from incorrectly labeled data. Hence always a fraction of the evaluated samples also gets labeled incorrectly. Drawing training samples only from the n superpixels $\mathcal{S}_1, \ldots, \mathcal{S}_n$ the user just labeled is not an option either since such a set of training samples would not represent the whole image that gets classified. Otherwise the user needs to be asked to also brush negative samples which was shown to confuse pathologists.

This approach was shown to induce high user frustration since it only slowly converges towards a correct classification result and the user also has to deal with some pixels more than once. In general this approach is rather useful for training a model on a data stream, like an audio or video stream than on static data with a changing proximity score map.

A similar effect is observable when instead of retraining a forest only split functions are optimized so that they better adapt to new samples. The structure of the trees stays the same while the split functions threshold values are changed to increase the information gain. This technique is marginally faster but still uses incorrectly labeled data points for training and therefore produces similar results than the aforementioned method. If only data samples were used which the user approved as correctly classified

the trees would create a bias and classification accuracy would decrease for areas the user did not approve. Again this method would result in an increased effort by the user and would not hold any advantages towards simple delineation.

The third approach called ULS (Update Leaf Statics) by Peter et al. [34] does not change the structure of the forest by training new trees, but takes the trees as they are and modifies their leaf statistics. For the updating procedure many different problem specific adaptions are available, but the most common one is to add new specific samples to the set of samples and thereby adjusting the probabilities of a definite set of leaf nodes. Therefore a new labeled sample $v_i \in \mathcal{S} = \{v_1, \ldots, v_n\}$ with a probability vector $\mathbf{p}_i \in \mathbb{R}^k$ assigned to it with k being the number of classes in the regression scenario gets pushed through all the trees of the random forest. All the predictor models $p(c|\mathbf{v})$ of the leaf nodes it respectively alights in get updated. Often a weight λ is used to adjust the ratio of importance between prior knowledge and newly observed samples. When the sample point alights in a leaf node with a set of samples $\mathcal{S}_i$ already assigned to it, the leaf nodes predictor model gets updated as follows:

$$p(c|\mathbf{v})_{\text{new}} = \frac{\lambda \mathbf{p}_i(c) + |\mathcal{S}_i|\, p(c|\mathbf{v})}{|\mathcal{S}_i| + 1}. \tag{4.12}$$

This way the forest gets tuned without being retrained. Furthermore no perfect classification result is needed as input since the samples added to the forest can be drawn from selected regions. In contrast to the last two approaches this only affects the ROI while keeping all other regions unaffected.

Implementation-wise no threshold like λ is needed since the training procedure is designed such that the user can decide how important the new samples are compared to the already seen ones. When an update gets triggered first the probabilities of all pixels $p_{x,y} \in \mathcal{S}_i$ where $\mathcal{S}_I$ are the superpixels that were brushed recently get updated. If the brush type is $b_t = b$ the probability for the class of interest c_i increases while the probabilities for all other classes decrease and vice versa when $b_t = e$. This is done by altering the vector of probabilities $p_c \in \mathbb{R}^k$: When increasing the probability of class i, the new scalar is computed as follows: $\mathbf{p}_i^{\text{new}} = \frac{\mathbf{p}_i + 1}{2}$. All other scalars of the vector are simply divided by 2 resulting in a vector of length 1.

When the eraser is selected and therefore $b_t = e$ the class probability for c_i gets divided by two while all other probabilities are modified like this: $\mathbf{p}_i^{\text{new}} = \frac{\mathbf{p}_i + \frac{1}{k-1}}{2}$, which also results in a vector of length 1.

Again this is only done for the pixels belonging to the set of brushed superpixels. Using this updated proximity score map the forest gets modified in the next step similar

to equation 4.12. By allowing to add a datapoint several times, no λ is needed since the user can choose how important a certain sample is by brushing a certain superpixel more than once.

After each user input, the leaf statistics are updated immediately and instant feedback is provided in real-time by evaluating 10% of the image's pixels. These are now colored according to their new classification result. Depending on the chosen visualisation style the evaluated samples either are displayed directly or contribute to the score of single superpixels which are colored according their score (see figure 21).

When the user is satisfied with the classification result of one class he either continues with a next class or he continues to classify further sections of the WSI. This can be done by exploring the WSI in a traditional way or by jumping to the next section by clicking on "next".

When the user clicked on "next" the random forest chooses random positions in the whole slide image and evaluates a patch around this position of the approximate size of a superpixel. This is done until a patch was evaluated positive for one the classes of interest. When a patch was found the viewport gets centered on this patch at the same level of detail that the user evaluated the previous section image on. Again the pre-classification result gets displayed which the pathologist is meant to adjust.

This procedure is done nine times or until the user clicks on "Evaluate whole slide" which starts the evaluation discussed in the next section, section 4.4.3. The reasons for choosing nine are diverse: The first on is based on the research of Mercan et al. [37], who found out that on average a pathologist zooms on nine ROIs when examining a WSI. This behaviour is depicted in figure 16. The second reason is addressed at requirement number four of the list in section 4: Support user-level hardware. When evaluating the WSI one option is to retrain a forest on all the labeled section images the user did work with. Since this requires a lot of RAM the restriction to nine section images comes in handy. The reason why the user is kept in control is because from a user-centered design perspective the user knows when he saw enough of the image. What he does not know is when the random forest saw enough samples. But since these two conditions correlate it is safe to ask this of the user.

This subsection covered the whole training procedure whose output now gets used to evaluate the complete WSI. This step is covered in section 4.4.3 after introducing the used set of features in the next section.

4.4.2 Features

Even though the sample application in this work is the detection of cancerous tissue during the procedure of a SLNB this approach is designed to be as generic and adaptable to similar task as possible. This i.a. reflects in the set of features and in the way they are used.

First of all it is important to determine whether the approach is meant to classify objects - "things" - or materials - "stuff" [66]. Materials are defined by repetitive patterns of fine scale properties without a specific or distinctive outline, shape or extent. On the contrary objects (e.g. nuclei) appear as regions of distinctive shape and size. On the other hand many materials are distinguishable because they are part of an object that was recognized beforehand [66]. In this context it is obvious that this works method aims at classifying materials. Therefore no features like shape descriptors or graph-based descriptors are used but rather texture features.

When deciding upon a set of features to use in a machine learning based classification algorithm two things need to be taken into account:

(a) The size of the feature space should fit the capabilities of the machine learning algorithm. Not only does the dimensionality of the feature space affect the algorithm's performance but also its accuracy. Not all features contribute to the classification result equally or even in a positive way at all. Depending on the used algorithm the classification performance can also deteriorate if more than the necessary number of features are used; this behaviour is called the "peaking phenomenon" [67]. By using a random forest algorithm both specifying a too small, non generic set of features and specifying a too large set of features, which affects the classification performance in a negative way, is avoided. Random forests have shown to perform well in large feature spaces, since they intelligently choose relevant feature dimensions themselves while disregarding irrelevant ones.

(b) Obviously the set of features shall be designed such that it can discriminate the same structures a pathologist is able to discriminate. Therefore it is important to investigate the coherence between e.g. local statistical measures that can serve as features and pathological and morphological characteristics a pathologist tends to use. Ambros et al. [68] found that even though pathologists are heavily relying on morphological features, like nuclear size or cellularity, they still implicitly integrate a large amount of information derived from textures of different histological structures in their decision making. Kong et al. [41] extracted a set of features

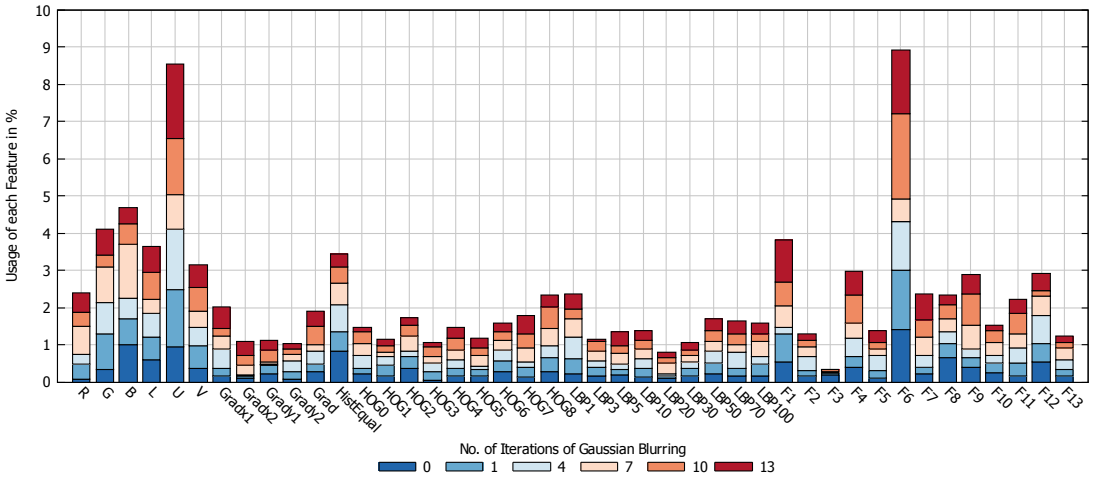


Figure 22: Partial contribution of each feature to the whole random forest.Furthermore it shows after how many iterations of Gaussian blurring the features are most valuable.

commonly used by pathologist for discriminating between different components. This set of features is listed in Table .

To be able to represent such complex characteristics the feature vector used in this approach comprises of a large variety of texture features. Again it is left to the random forest to decide which on separates the given data best. In figure 22 one can see that for the used data set some features were more distinctive than others and therefore were chosen more often by the random forest. This way the resulting classifier is generic enough while still being powerful.

For the feature set used in this work the feature set Kainz et al. [51] used for nuclei detection got used as a basis and was extended by several texture features. Overall it comprises of 43 features: RGB-channels (3), LUV-channels (3), first and second order gradients in x- and y-direction (4), gradient magnitude (1), histogram-equalized gray scale image (1), histogram of oriented gradients (9), local binary pattern of 8 radii (8), and the first 13 Haralick features (13).

Since the texture features local binary pattern (LBP) and the Haralick features are not as straight forward as other features they are given a short introduction hereinafter.

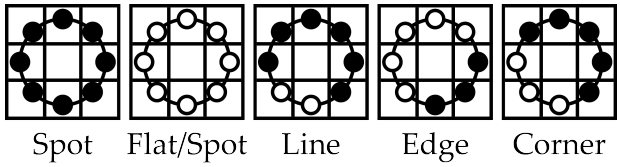


Figure 23: Textural details that can be described using LBP [69].

The LBP texture feature is a 2D texture analysis descriptor that works on gray-level images which summarizes local structures by comparing a single pixel with its surrounding pixels. It was initially proposed by Wang et al. [70] and gained popularity because it is easy to implement and fast to compute while still being a powerful feature for discriminating textures. In its initial version each pixel gets assigned to an 8 bit value that gets computed considering the pixel's 3×3 neighbourhood. Said center pixel gets compared to all its neighbouring pixels; if the center pixel's intensity is greater-equal a 1 gets denoted otherwise a 0. This way a binary number like 01110101 that translates to a decimal value with an intensity between 1 - 255 gets assigned to each pixel. Figure 23 depicts the textural features that can be described using LBP. Further adaptions arose over the years concerning different neighbourhoods and interpolation methods. Most commonly the radius and the number of the sample points are adapted. In the feature set used in this work only the radii are adapted starting at a radius $r = 1$ and going up to a distance of 100 px (see figure 22).

The Haralick features on the other hand are not based on single pixel differences but rather compute statistical features from a gray-level co-occurrence matrix (GLCM) (also "called gray-tone spatial dependence matrix" in Haralick's initial publication in 1973 [71]).

GLCMs are square matrices of size $N_G \times N_G$ with N_G being the number of gray levels in the image to be classified. For a gray-level image patch the GLCM stores at position (i, j) how often a pixel with intensity i is adjacent to a pixel with intensity j. Four different adjacency definitions can be used which, depending on the image lead to different GLCMs: 0°, 90°, 135°, and 45°. Usually only directly neighbouring pixels are taking into account, but just like it happened for LBP, different adjacency definitions were used over the years. In this work a radius $r = 1$ and an angle of 0° was used. When computing a GLCM for an 8 bit image the matrix' size would be 255×255 which is linked to two drawbacks: (a) the

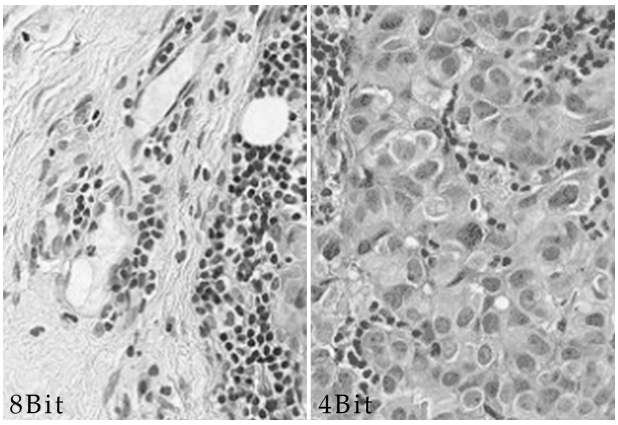


Figure 24: Comparison of an input image at 4 bit and 8 bit. A loss of texture only is visible at a resolution this low. Hence a reduction to 16 gray values is legitimate.

resulting matrix would be relatively sparse and (b) the computation for such a large Matrix would be relatively costly. Hence the input images color range needs to be reduced. A reduction to 4 bits, resulting in a GLCM of size 16×16 was found to be a sufficient tradeoff between the preservation of details and computational complexity. The difference between both color ranges is shown in figure 24. The GLCM gets computed on a patch of defined size around a pixel. The patch size is a globally defined variable which gets also used for the computation of other features.

After the GLCM got computed, it gets used to calculate 14 statistical features, called the Haralick features. Their formal definition is to be found in section A.1. Some of the features are directly linked to visual appearance, e.g. f_1 is a good descriptor for homogeneity, while others can not be described by terms commonly used for texture description.

Especially texture features like LBP tend to result in noisy images. Hence each feature image additionally gets blurred using a Gaussian kernel to generate a larger variety of features the random forest can choose from; 43 features at 6 stages of blurring each results in 258 features. In figure 22 one can see that for instance the feature "LBP1" gets more valuable for the random forest with increasing blur, respectively with the increase of global influence.

Furthermore 22 shows also that some features, like "U" and "F6" are used more often by the random forest than others, which might change for different input images. This also implies that the random forest works as expected and chooses the feature dimensions of interest properly according to the information gain they induce. In addition it shows that the random forest implementation is working well in the high dimensional feature space.

A last bit that needs to be taken into account when designing a set of features, is whether it can be adapted such that it is applicable in a patch processing use-case. Usually WSIs are not evaluated in one piece since they would not fit into the RAM together with all their feature images. Therefore it is important that a feature does not depend on a certain surrounding that should be the same for all tiles. Otherwise joints on patch borders can be unsteady. This for instance goes for the histogram-equalized gray-scale feature image where each pixel's value depends on the whole image. Depending on the pixel intensities within their respective patches, adjacent pixels that join on the patch borders might get assigned different intensities after the histogram equalization, even though they initially had the same intensity. Hence in this case the cumulative distribution function used for the histogram equalization needs to

be computed for the entire WSI and then be applied to each patch. If the cumulative distribution function would be computed patch-wise the outcome for each intensity value would differ for each patch. More on patch-wise evaluation of WSIs is discussed in the next section, section 4.4.3.

4.4.3 Evaluation

This section's topic is the evaluation of the WSI. It is the application of the random forest previously trained to classify the WSI at the resolution the user choose initially. Furthermore different opportunities regarding the technical implementation of this step are discussed.

Regardless of the implementation of the WSI-evaluation it is important to properly fit this step in the interaction pipeline. This is necessary because the user does not have control during the evaluation step any more: One the one hand it is an automated step in which he does not have to control anything. On the other hand, when the evaluation gets executed on the workstation he is working on, the sophisticated operations during the evaluation will impair the computer's performance significantly.

Depending on the user's work station different options exist. Given the fact, that the user has access to any kind of high-performance computer or cluster, the evaluation can be carried out right after the training. This is the most continuous work flow imaginable, since the user would not need to interact with one case twice. The mental workload of recalling a case after a short amount of time would not be an issue.

A second option, that became more popular in the recent past, would be a cloud computing based version of this application. This comes with advantageous possibilities as well as an infrastructural drawback. Before a WSI can be used by a cloud-computing based application it needs to be uploaded. Since file sizes up to 3 GB are common this constitutes a considerable bottleneck in the pipeline. On the other hand once uploaded, a WSI can be used in other use-cases, e.g. in tele-pathology for getting a second opinion. Furthermore once the WSI got labeled it can serve as a training sample for the development of other machine learning approaches or simply can be used for educational purposes [72].

Despite the uploading of the WSIs a cloud-computing based application would result in the same user experience compared to the scenario in which the user has access to a high-performance computer.

The last and most common scenario is that the pathologist has to do the evaluation on his own workstation which usually is consumer-level hardware.

No matter on which computer the evaluation eventually gets carried out, all three versions would feature an implementation in which single patches taken from the WSI are processed independently. Details about the tile-based processing are covered hereinafter.

Before starting with implementational details the output of the training step shall be recalled, which is: (a) a forest f_s that was trained on k section images of the entire WSI (b) k labeled section images, and (c) k specifically trained forests f_i, each belonging to one of the k section images $\mathcal{T}_i$.

Using this as input three use-cases are imaginable. The first would be to evaluate the WSI just using the forest f_s (hereafter referred to as "option A"). The problem with this use-case is that the forest f_s got adapted many times throughout the training procedure without being actually refined. This means that e. g. forest f_3 which was trained on $\mathcal{T}_3$ not necessarily does perform well on previously seen section images ($\mathcal{T}_1$ or $\mathcal{T}_2$) any more. Its performance is strongly dependent from the section image $\mathcal{T}_i$ seen last.

Another option would be to simply concatenate the single trees from each forest and form a new, larger forest comprising of all these trees ("option B"). This would bundle the available knowledge while also creating a lot of redundancy. Furthermore it is fast since it involves not training.

The last option would be to take advantage of all the labeled section images and train a new forest using them ("option C"). This would incorporate all the available knowledge but also means more effort. However compared to the effort it takes to evaluate the entire WSI, the effort for only training a new forest is relatively low.

These three options were compared regarding their coherence with the ground truth using the statistical measure Cohen's kappa, which gets introduced in section 5.1.1. It was found that using option C generates the most accurate classification while using option A was least accurate. Since the performance of the forests is dependent from the section image seen last, a WSI was evaluated for all nine forests to evaluate option A. The best forest only was 11.65% less accurate than option C while the least accurate forest classified the WSI with an accuracy 55.5% lower than option C. The average of all the nine forests was 21.6% below the accuracy provided by option C. Using option B was found to be only 10.59% less accurate than the output from using option C. Hence option C is used for the evaluation of the WSI.

Furthermore accuracy was compared for different kinds of training data input:

(a) The proximity score map comprising of continuous values gets used as training data.

(b) The pixel-wise segmentation determined by the user controlled threshold gets used as training data.

(c) The superpixel-wise segmentation gets used as training data.

This comparison is addressed at the filtering nature of the superpixels: Will the lack of details influence the accuracy in a negative way or will it influence it in a positive way by filtering irrelevant parts from the input image.

The evaluation shows that the highest accuracy was achieved by using the superpixel-wise segmentation as training data. Its results only were 7.1% less accurate than the results achieved by using the ground truth segmentation as input. Second most accurate results are generated by using the proximity score map (21.1% less accurate compared to the ground truth). Least accurate was the classification result generated by the forest trained on a pixel-wise segmentation (28.9% less accurate).

After evaluating the role the superpixels play during the training procedure, its effect on the accuracy during the evaluation step is evaluated. Hence Cohen's kappa is computed for two segmentation results: One is computed using pixel-wise evaluation and a second one using superpixel-wise segmentation. This way it gets evaluated whether the noise cancelling nature of superpixels have a positive or negative effect on the overall classification result. The effect of the superpixels turned out not to bes as strong as for the training process but was still perceptible. The pixel-wise evaluation is 3.78% less accurate compared to the superpixel-wise one. Even though this difference does not seem to be a lot but it was significant across all combinations of evaluation setups tested (ranging from 2.5% to 7.58%).

The last aspects are not concerned with accuracy but are rather related to the tile-wise processing of the WSI itself. As already mentioned in 4.4.2 it is important to ensure steady junctions at the tile borders. The aspect discussed was to choose features whose single pixel values are not dependent from the other pixels of the tile. Furthermore when performing any kind of convolution (e. g. gauss filtering for increasing global influence) it is important to define an overlap of the image tiles that suffices. In the proposed method instead of applying large gauss filter kernels, smaller ones were applied multiple times which has the same effect. Applying a Gaussian kernel with σ_1 and a radius r_1 once has the same effect as applying a smaller Gaussian kernel with σ_2 and a radius r_2 n times, with $\sigma_2 = \sqrt{n * \sigma_1^2}$ and radii being $r_i = 3 * \sigma_i$, with $i \in 1, 2$.

When using overlapping tiles it also is important to choose the overlap wisely since it affects the performance: When using a small tile size single pixel values are computed multiple times which results in higher computational effort. If the tile size

is determined to large, it also will effect the performance in a negative way when the RAM is overflowing and techniques like virtual memory management start to perform.

This section covered parameters that need to be taken into account when implementing an algorithm that evaluates entires WSIs. These parameters concern the random forest implementation itself as well as tile-wise processing in general which is indispensable when evaluating WSIs.

4.4.4 Filter and Visual Clues

To further support the pathologist during examination some few visualizations are available. Their main purpose is to present information to the user in a non obstructive way. When displaying e.g. heat maps as overlay image the underlying structure gets distorted which makes the visual examination more complicated since the user has to switch between two visual representations which increases the mental workload. In the developed application the superpixel segmentation is used for such non obstructive overlays since their boundaries tend to follow edges while not crossing homogeneous areas.

Hereinafter three different ways of displaying context information on grids are presented. In figure 26 the classification result is depicted as a semi transparent overlay over the tissue image. It is easy to see that the original data gets distorted which makes working with it even more demanding. Hence context information like the heat map shown in figure 26 gets displayed on grid structures. Two more regular grids are used to show the advantages of the superpixels for displaying context information. The most trivial grid is a regular grid with orthogonal joints depicted in figure 25. The second grid is arranged in a *chicken-wire* pattern which is also a common pattern to be found in different kinds of pathological images. Amongst them are perivenular fibrosis found in pathological fat liver tissue [73] and stained tissue for HER2 tests in IHC [74]. The parameters were chosen in a way so that the amount of pixels obstructing the image is similar for each image, which is $\sim 18\,\% - 21\,\%$ of the overall amount of pixels.

Figure 25 shows, that regular grids exacerbate the perception of the tissue's texture while the superpixel segmentation on the other hand adapts to the granularity of the texture. Thereby it does not obstruct the tissue too much and allows for displaying non-obstructive overlays like it is depicted in figure 26. In figure 25 the characteristics of the SLIC superpixel algorithm discussed in section 3 can be seen in praxis: Depending on the initial positions of the super pixel centers the superpixel boundaries either closely encompass dense, homogeneous structures like nuclei or avoid them at all. On

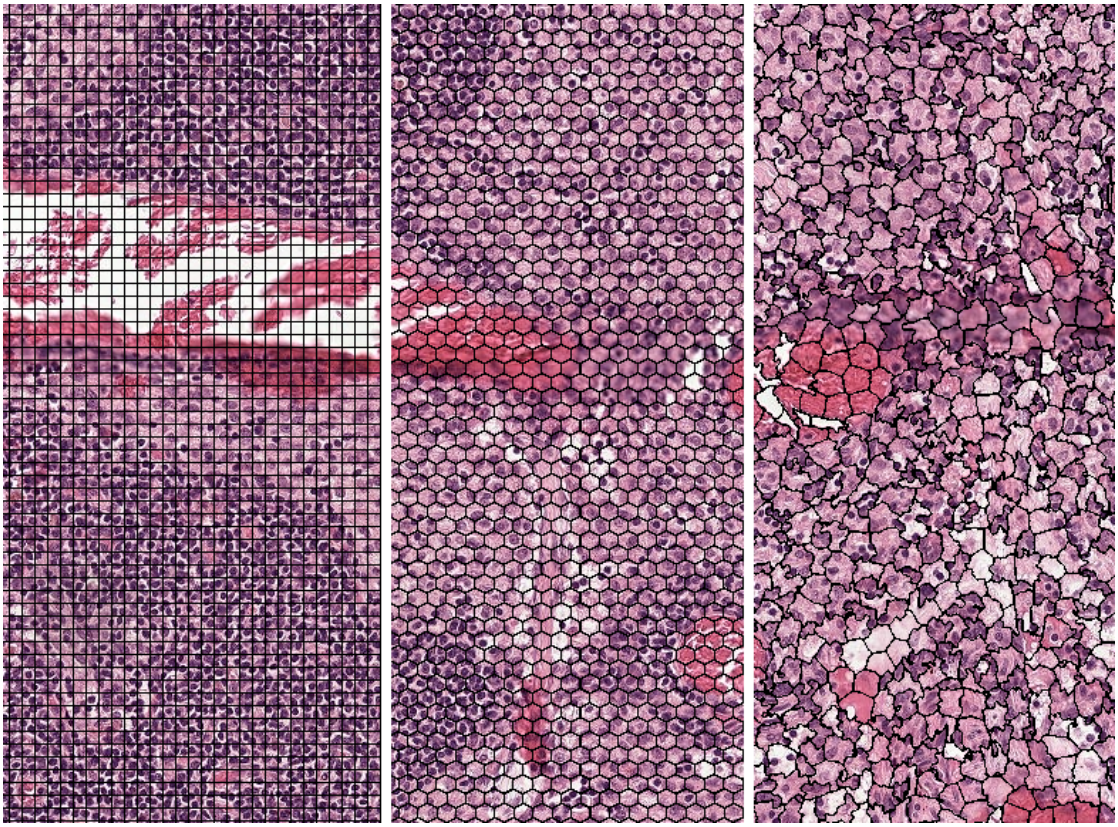


Figure 25: Comparison of different overlay styles: On the left a regular grid is used, in the middle a chicken-wire pattern, and on the right SLIC superpixels. The images depicted are zoomed in from the normal screen resolution since otherwise the effect would not have been perceivable. Figure 26 shows the overlay in normal screen resolution.

the contrary regular grids do cross such regions which results in an obstructed vision of the tissue. Hence the borders of superpixels provide a suitable part of the image in which context information can be displayed.

Depending on personal preferences or special characteristics of a certain dataset the pathologist can choose between several display styles. The most general one is whether he would like to see the superpixel segmentation or not. Both cases have their own advantages according to the mental model the pathologist has formed: Hiding the segmentation decreases the mental effort since the user is not faced with the question which role the superpixels do play in the whole classification procedure. On the other hand for a more technically minded user, displaying the superpixel segmentation might close the gap between visualization and underlying implementation. Hence the user can choose if the border lines are displayed or not.

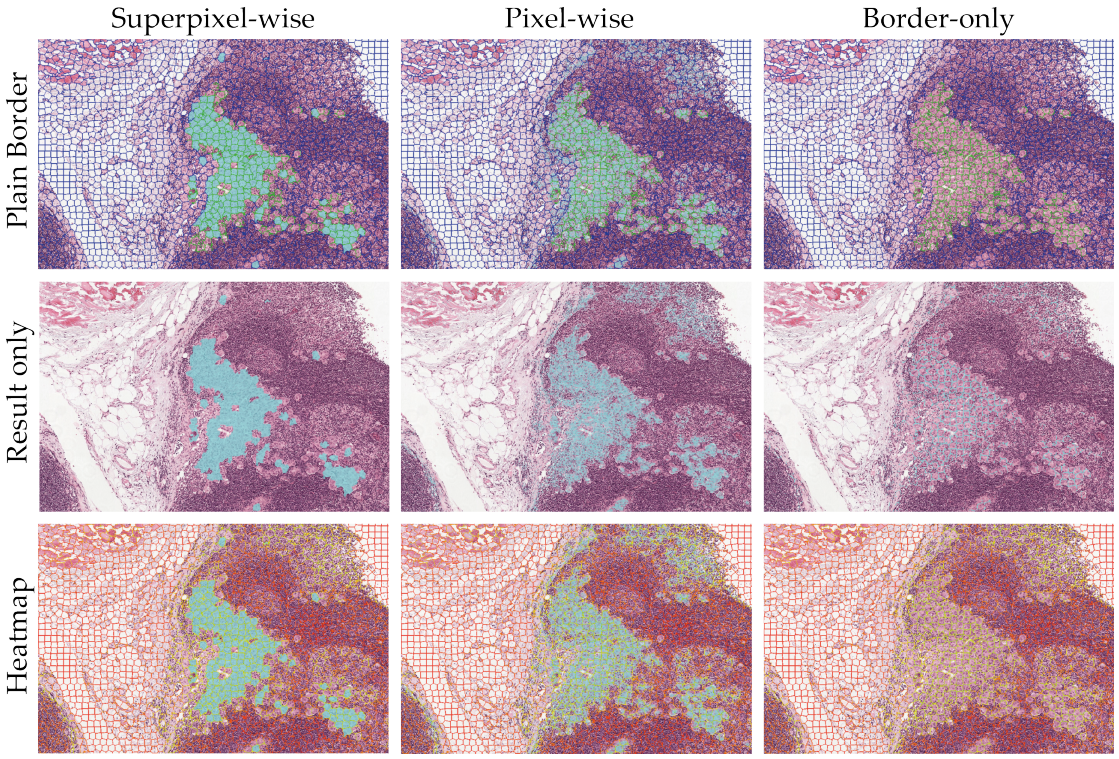


Figure 26: All available visualization styles the pathologist can choose from when training the forest (besides deactivating any overlays).

If he chooses to display the superpixel segmentation he further can select the information that shall be displayed within the supderpixel borders. He can choose to display the segmentation result or a heat map displaying the certainty with which certain types of tissue get classified or no information (see column three in figure 26).

Said heat map codes the certainty with which the random forest classifies the displayed classification result. The color gets chosen by altering the value component H in the HSV color-space while S and V are constant. The highest probability-value p^* for pixel(x, y) gets determined as follows: $p^*_{x,y} = \arg\max_{i \in k} \mathcal{M}^i_{x,y}$. Since this value is within the range $[0, 1]$ and a color scheme between red and green is desirable for a heat map, $p^*_{x,y}$ simply gets multiplied with 120: $H = p^*_{x,y} * 120$. This leads to red for a probability of zero and to green for a probability of one. The application of this heat map is depicted in the last row of figure 26. The part that got stained red was classified as not belonging to the class of interest with high certainty. Whereas the output for the upper right corner which is colored orange is not as certain. This indicates that the random forest needs more training for regions with similar tissue. This way the pathologist sees both, if tissue falsely gets classified with high certainty and also if the

tissue gets correctly classified with high certainty. Using the heat map visualization the pathologist knows about the random forest's performance and is provided with simple yet efficient tools for tuning it.

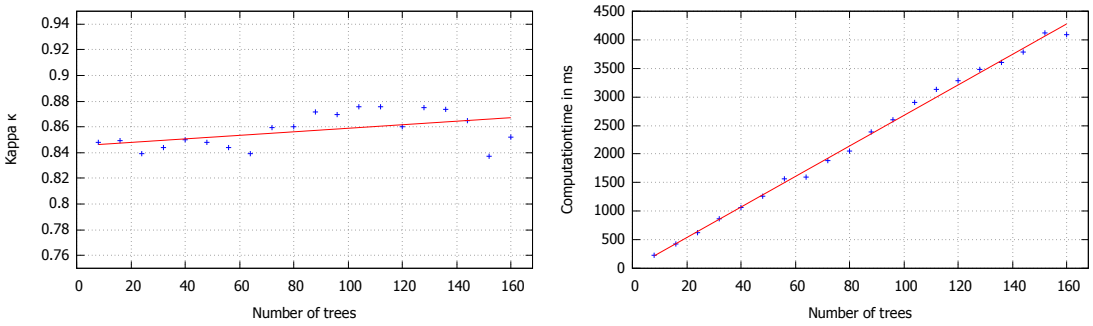
The last option the user can choose from is whether he prefers the classification result displayed as superpixels or per each pixel. When choosing a superpixel-wise visualization this does suppress outliers and small less relevant parts which leads to a better structured classification result. On the other side this kind of noise filtering might also suppress small parts that are of relevance. Hence the user can choose and switch between both visualizations depending on the problem he is working on. The comparison between both visualization styles is shown in figure 26 in row two.

If the pathologist is not in the need of using the heat map either for personal preferences or task-related factors, he might also choose to display the results within the superpixel borders. This is the least obstructive way within this framework to provide the user with a classification result since only a minimum of underlying tissue gets obstructed. Still it is important to mention, that it is also the most imprecise representation with the lowest visual contrast. In figure 26, row two the direct comparison shows that this visualization depicts the least amount of details. Still it too visualizes the uncertain parts in the upper right corner.

In conclusion the superpixels provide a good basis for visualizations that otherwise would obstruct the data that gets examined. Furthermore this section introduced different visualizations from which the pathologist can choose according to his preferences and data characteristics. The following section, section 4.4.5, has no connection to the visualization part but evaluates the set of parameters that essentially determines the online random forest's behaviour.

4.4.5 Parameter Evaluation

Since the random forest implementation used in this work needs to be both fast and easy to modify and yet powerful, the selection of parameters is an essential step for this approach's design. Making tradeoffs between speed and accuracy is inevitable to achieve a seamless user experience. Hence all parameters get evaluated regarding both the agreement with the groundtruth and computation time. This section presents the values chosen for important variables concerning the random forest and backs them up with series of experiments. This section is not part of section 5 since it not directly evaluates this work's approach but only covers the tuning process of the random forests parameters.

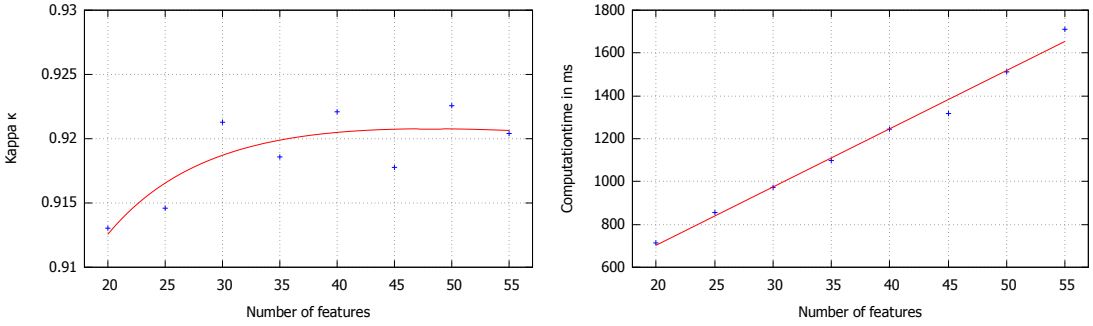


(a) Evaluation of the parameter *number of trees* relative to Cohens kappa κ.

(b) Evaluation of the parameter *number of trees* relative to computation time.

Figure 27: The coherence of forest size and computation time is linear. Accuracy also increases with increasing forest size but meets a limit at a certain point.

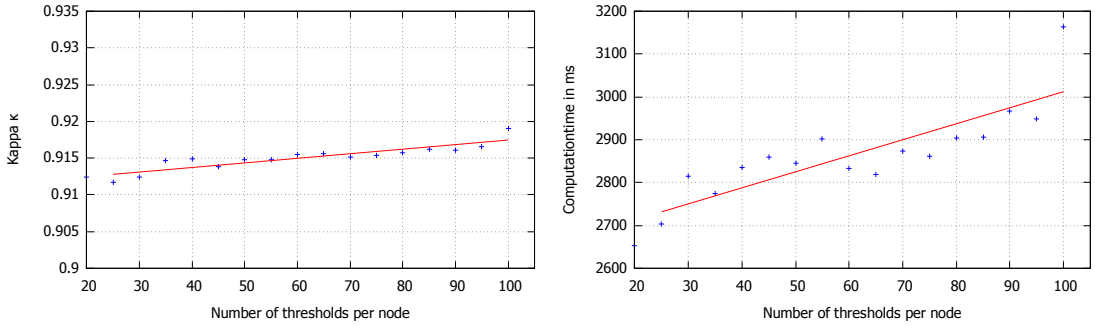
The following parameters are evaluated: *number of features tested for a split*, *number of thresholds tested for a split*, *maximum number of levels*, *minimum number of samples necessary for a split*, and *number of trees*. The aim of the evaluation is to find the best value for each parameter according to Cohen's kappa which is a statistic for measuring consensus between several qualitative items. For each parameter, there is a set of possible values which were chosen by an educated guess. Each of these values gets analysed by evaluating 100 random forest that were generated using this combination of parameters. The same is done for computation time. The result is visualized by plotting each value against computation time and kappa. This way a set of parameters is manually chosen from a practical point of view making tradeoffs between accuracy and speed.



(a) Evaluation of the parameter *number of features* relative to Cohens kappa κ.

(b) Evaluation of the parameter *number of features* relative to computation time.

Figure 28: When increasing the *number of features* that are compared when training a node the accuracy also increases until a certain point. As for the size of the forest the coherence of the number of features and computation time is linear.



(a) Evaluation of the parameter *number of thresholds* relative to Cohens kappa κ.

(b) Evaluation of the parameter *number of thresholds* relative to computation time.

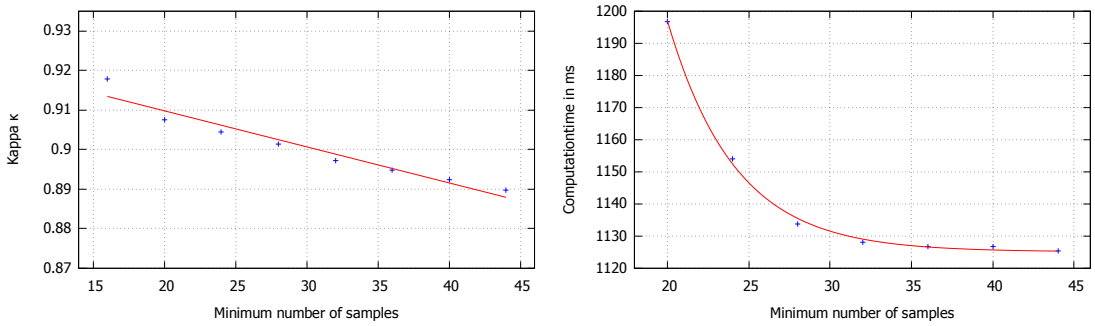
Figure 29: An increase in the number of thresholds implies both an increase in accuracy as well as an increase in computation time. Both relations are close to linear with the rise of the computation time being steeper.

The most characteristic parameter for an ensemble learner like a random forest is the forest size, respectively the number of trees. The influence of this parameters value is discussed when introducing the concept of random forests in section 3.1.1.

Figure 27 shows the evaluation results of the *number of trees* parameter. As expected the coherence of computation time and forest size is linear. Therefore it is important to choose a value that is as low as possible and still provides sufficiently accuracy. Since the performance for small forests is not significantly lower than for forests of size ~ 100, a forest size of eight was found to be sufficient for the online implementation of the random forest. For the evaluation of the WSIs a forest of size 64 is used scince it constitutes the best tradeoff between computation time and accuracy.

The next two parameters *number of features* and *number of thresholds* determine the training process of single nodes. As introduced in section 3.1.1 a split function $\boldsymbol{\theta}$ is defined by the parameters $(\boldsymbol{\phi}, \boldsymbol{\psi}, \boldsymbol{\tau})$. When training a node a set of split functions get defined from which the one with the highest information gain gets assigned to the node. $\boldsymbol{\tau}$ defines a threshold, $\boldsymbol{\phi}$ contains a selection of feature channels, and $\boldsymbol{\psi}$ defines how the split itself is carried out. When the training procedure of the node starts a set containing pairs of $\boldsymbol{\psi}$ and $\boldsymbol{\phi}$ is generated. Said sets size is defined by the *number of features* parameter. Furthermore for each pair a set of thresholds $\boldsymbol{\tau}$ is created; its size is defined by the *number of thresholds* parameter. As a consequence an increase of the value of both parameters has similar characteristics.

The effect of an increase is shown in figure 28 and 29. The impact of the parameter *number of features* is slightly higher since it also affects the *number of thresholds*. In



(a) Evaluation of the parameter *minimum number of samples* relative to Cohens kappa κ.

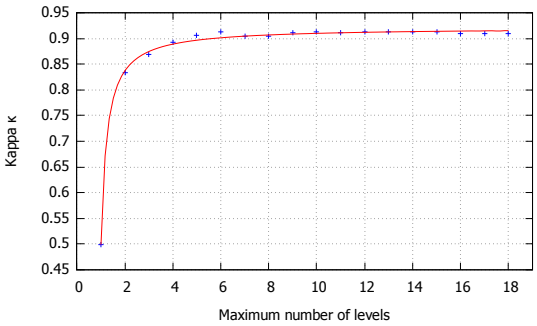
(b) Evaluation of the parameter *minimum number of samples* relative to computation time.

Figure 30: Contrary to all the other parameters both, the accuracy and computation time, decreases when increasing the *minimum number of samples* parameter. When choosing a value it needs to be taken into account that the graph related to κ looks different (less steep) in a cross-validation scenario.

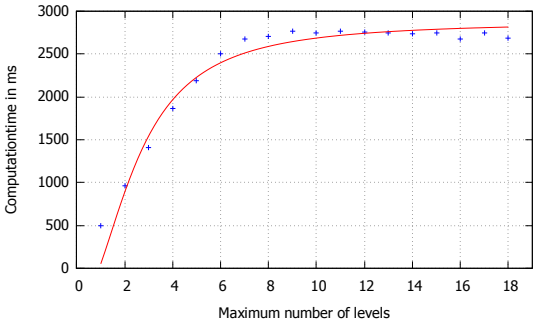
general trying out more parameter combinations when training a single node increases the chance that a combination with a higher information gain gets found. Also their impact on computation time is comparable even though again the impact of the *number of features* parameter is higher.

For the final set of parameters the values of the *number of features* parameter was set to 35 and the *number of thresholds* parameter was set to 50.

The next two parameters *maximum number of levels* and *minimum number of samples* are in fact stopping criteria mentioned in section 3.1.1 and are closely coupled with the overall amount of samples. The *maximum number of levels* parameter restricts the depth of trees by defining a limit independent from the data and the ongoing training process. The *minimum number of samples* parameter on the other hand limits the trees depth by creating a threshold that needs to be exceeded to allow the tree to grow further. Hence when a node is trained, first it is checked whether the size of the Set of training samples that arrived at this node is larger than the threshold defined by the *minimum number of samples* parameter. The intention of this stopping criterion is to avoid overfitting. The term overfitting refers to a scenario in which a classifier is perfectly trained for a certain dataset but performs poorly when applied in a cross validation scenario. Hence the value of Cohens kappa κ decreases with an increase of the value of the parameter *minimum number of samples*. The appearance of the graph in 30b is strongly dependent on the overall amount of samples and also on the *maximum number of levels* parameter.



(a) Evaluation of the parameter *maximum number of levels* relative to Cohens kappa κ.



(b) Evaluation of the parameter *maximum number of levels* relative to computation time.

Figure 31: Both values, computation time and accuracy, converge against a limit value. Preferably a value for the *maximum number of values* parameter is chosen that is larger than the value at which the rate of change of κ is already relatively low and the rate of change of computation time still is large.

When decreasing the *minimum number of samples* parameter value computation time increases exponentially since the tree grows bigger and more split functions need to be generated. This goes on until the stopping criterion defined by *maximum number of levels* is met. Afterwards computation time stays the same which is not captured in the graph in figure 30b since only one criterion should be evaluated at a time.

The *maximum number of levels* only shows an effect if the size of the set of training samples is sufficiently large. In figure 31a it is shown that further increasing the numbers of levels after a certain point only has a marginal impact on the accuracy. The reason for this is simple: The information gain for the split functions close to the root node is larger by tendency. Adding further valuable information after a sample point already had passed many split functions becomes more unlikely. Hence the accuracy only rises very slightly after a certain point. The computation time behaves in a similar way after a certain point, since the number of split functions that need to be generated does not increase further. This point is dependent on the size of the set of sample points. With increasing size of the set of sample points, the *maximum number of levels* also needs to increase in order to make a difference.

Since the amount of sample points remotely stays the same, in this works use case the *maximum number of levels* was set to six. Making the same tradeoff between computation time and accuracy the *minimum number of samples* was set to 25.

This section gave an overview about the parameter values that were chosen for all parameters regarding the random forest implementation. It not only justified the set of parameters used in this work but also showed which facts need to be taken into account when deciding about a set of parameters.

The actual evaluation of this work's method follows in the section 5.

5 Findings and Evaluation

When evaluating the proposed method two essential parts need to be evaluated separately and in interplay: (a) the offline part of the random forest implementation, and (b) the user interaction itself. Therefore a statistical measure that suits the data is needed, as well as a labeled dataset and pathologists taking part in the user test. This section starts with the introduction of Cohen's kappa which gets used for evaluation, presents the dataset and the used hardware, and subsequently presents the results from both of the evaluation steps.

5.1 Preliminary Consideration

This section covers all context information needed to understand and to interpret the evaluation results. It comprises of the introduction of the statistical measure (Cohen's kappa), the dataset used for the evaluation, and the hardware used.

5.1.1 Cohen's Kappa

To measure coherence between two ore more qualitative items a statistical measure is needed that suits the data. Choosing the right statistical measure is of great importance because the ROIs are really sparse within the WSI. Hence naive statistical measures would fail to properly evaluate the segmentation result of a WSI. Such a naive measure simply could compute the percentage of pixels in which the ground truth and the segmentation result match. This would result in concordance rates over 99% even for a segmentation image only filled with zeros, which translates to *belonging to no class*. Therefore Cohen's kappa [75] is used for the evaluation of the proposed method because it corrects for chance expected agreement [7].

To compute Cohen's kappa for a segmentation and the related ground truth a statistic is computed which counts the cases of true positives (a), false positives (b), false negatives (c), and true negatives (d).

The proportion of observed overall agreement p_0 is computed as the fraction of true positives (a) and true negatives (d) of the overall amount of pixels:

$$p_0 = \frac{a+d}{a+b+c+d}. \tag{5.1}$$

Furthermore the overall probability that both the segmentation and the ground truth randomly classify a pixel as belonging to a class is computed as follows:

$$p^+ = \frac{a+b}{a+b+c+d} \cdot \frac{a+c}{a+b+c+d}. \tag{5.2}$$

Similarly p^- is computed:

$$p^+ = \frac{c+d}{a+b+c+d} \cdot \frac{b+d}{a+b+c+d}. \tag{5.3}$$

Therefore the overall probability of random agreement is $p_e = p^+ + p^-$. The coefficient κ now simply is the proportion of agreement after chance agreement is removed from consideration:

$$\kappa = \frac{p_0 - p_e}{1 - p_e}. \tag{5.4}$$

For the interpretation of the resulting kappa different similar guidelines exist. Its values range from -1 to 1; with complete agreement kappa equals 1 and respectively for complete disagreement it equals -1. For agreement greater than chance kappa lies within the range $[0, 1]$; for a agreement less than or equal chance kappa ≤ 0 [7]. The interpretation guideline by Cicchetti et al. [76] assigns the following terms to certain intervals of kappa: "excellent" (kappa 0.75 - 1.00), "good" (kappa 0.60 - 0.74), "fair" (kappa 0.40 - 0.59), "poor" (kappa < 0.40).

5.1.2 Dataset

The dataset used for the evaluation of the proposed method is provided by the Camelyon challenge [77]. The Camelyon Challenge aims at evaluating new and existing approaches for the automated detection and classification of breast cancer metastases in WSIs. The organisers, who mainly are members of Radboud Universities Medical Center, asked the participants to take part in the spirit of cooperative scientific progress.

The dataset comprises of 400 labeled WSIs from two different medical centers in the Netherlands (Utrecht and Nijmegen). They were released in two stages: 270 of them served as training dataset, 130 were used as test dataset. Of the 270 WSIs, 70 contain metastases while the other 100 were taken from healthy lymph nodes. The ground

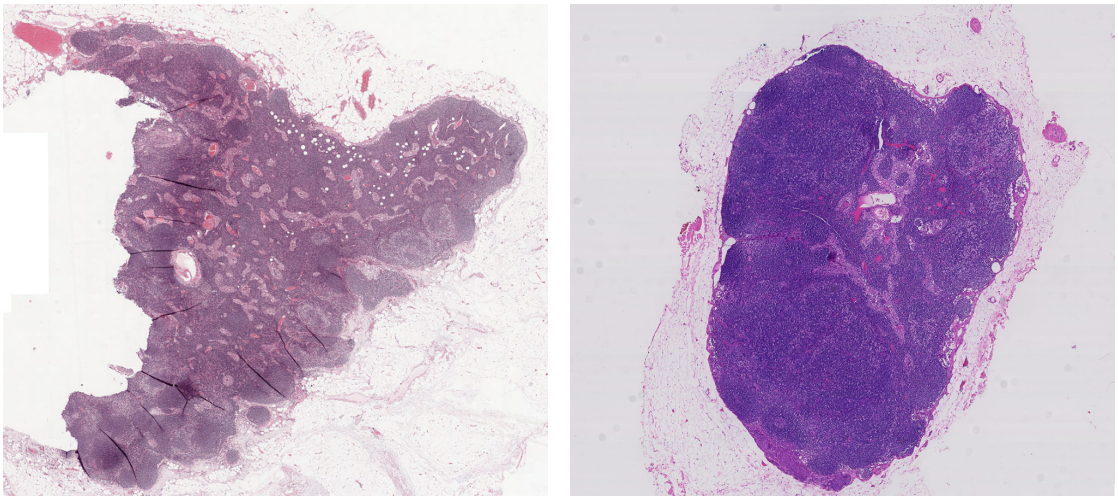


Figure 32: Two WSIs taken from the Camelyon challenge dataset, that were provided by two different labs, show common differences in dye concentration leading to problems with the generalizability of full automatic tissue classification methods [77].

truth contains the outlines of the metastases as binary masks of the size of the WSI. The annotations were prepared under supervision of expert pathologist.

Interestingly enough the submitted methods are challenged with the variability that is introduced when datasets are acquired by multiple different labs. Figure 32 shows the conspicuous differences in dye concentration of H&E.

5.1.3 Hardware

In order to be able to compare different approaches it is necessary to know about the hardware on which the evaluation was carried out. As determined in the requirements in section 4.1, the proposed method was evaluated using consumer-level hardware. The hardware specifications of the used computer alongside with the implications they have on the evaluation can be found in table 5.1.

Most important for the performance of the proposed method are inherently the processor and the hard drive: Since the WSIs are accessed very often, using an HDD would represent a major restriction.

Even though the implementation used for the evaluation of the proposed method uses parallelization it is not realized as a GPU implementation. Therefore the evaluation is not affected by the GPUs specification but might make a major difference for other implementations. For the used implementation only the number of cores

and the number of threads are of importance. Again these facts are specific to the implementation used in this work.

Furthermore a size of at least 16 GBs is recommended for the RAM; the size will not directly affect the computation speed but will result in a buffer overflow if it is too small.

Hardware Component	**Specification**	**Influence on Computation**
Processor	Intel ® Core™ i7-4930K, 3.40 GHz, 12 MB Cache	affects training- and evaluation time, more cores accelerate the parallelization
Memory (RAM)	16 GB DDR3	at least 8 GB recommended, used for storing the entire WSI and associated feature images
Graphics Card	Nvidia GeForce GTX 760	No influence on this implementation, even though a potential for a parallel GPU implementation exists during the training and evaluation
Hard Drive	840 EVO SATA III, 500 GB	Influences the access rate on the WSI, SSDs are faster than HDDs

Table 5.1: Specifications of the computer used for the evaluation.

5.2 Offline Component

The first step of evaluation covers the part of the proposed method in which no user interaction is needed. It simply evaluates the capabilities of the random forest implementation. Therefore a forest gets trained on the ground truth data and gets evaluated in a cross validation using the parameters determined in section 4.4.5. To put the results into perspective they are compared with the results of the Camelyon challenge [77].

5.2.1 Setup

Similar to the evaluation of the approaches submitted to the Camelyon challenge, the forest used for the evaluation gets trained using the ground truth provided by the organisers. However the training- and the test dataset are not used as intended by the organisers. The challenge mainly is addressed at deep learning algorithms which is why such a large dataset is provided. The method proposed in this work on the other hand aims at training a classifier on a relatively small dataset. Hence it is expected that the approaches submitted to the challenge achieve higher accuracies. Therefore they are also associated with a lower level of generalizability.

The classifier is trained, as intended in the real world scenario, on nine section images taken from the WSI and their corresponding ground truth images. Said classifier now is used to classify the whole slide. This way the evaluation is similar to a cross validation since most parts of the WSI remains unseen to the classifier.

5.2.2 Comparison with the Outcome of the Camyleon Challenge

Even though the approaches submitted to the Camyleon challenge pursue a different goal than the proposed method, since they are meant to be used as full automatic approaches, they serve as a good baseline to be compared with. Furthermore they also use Cohen's kappa for the evaluation which further eases the comparison.

The top five teams achieved accuracies between 0.8958 and 0.8244 according to Cohen's kappa. Overall 32 teams participated and achieved an accuracy of 0.3681 on average. The top contribution by Wang et al. [78] generated a training set comprising of $\sim 290K$ 256×256 sample patches. Their network structure of GoogLeNet consists of 27 layers and more than 6 million parameters. Training such a large network takes ~ 2 days without using any kind of high-performance computer; Quincy Wong, third in the overall ranking of the 2016 Camyleon challenge, trained a similar network in ~ 37 hours [79].

The contributors of the Camelyon challenge show, that it is possible to achieve sufficiently accurate classification results using more advanced methods, compared to the proposed method. On the downside they accept higher computational costs which are not in line with the requirements defined for the proposed method.

The evaluation of the proposed method led to the following results: On average the accordance rate between the ground truth and the classification is 0.6367. This result was computed using ten section images from ten different WSIs and their corresponding ground truth. These nine section images that were chosen at a magnification of $\times 19.5$

were used to train ten random forests. After the ten random forests finished their training procedure they were applied to evaluate the entire WSIs which they were trained on. The classification results were compared with the ground truth images using Cohen's kappa (see section 5.1.1).

Furthermore the relation between accuracy, magnification and computation time was examined. While the computation time for the training of the random forests is independent from the magnification, computation time for the evaluation increases exponentially with increasing magnification. At $\times 19.5$ it took the hardware setup described in section 5.1.3 $\sim$ 13 hours to evaluate the whole WSI, whereas the evaluation of an entire WSI at $\times 9.6$ only took 3.5 hours. For the dataset used in the evaluation of these connections, the decrease of detail also meant a decrease of accuracy: The concordance rate with the ground truth dropped from 51.352 to 35.322. On the other no further increase of accuracy for magnifications larger than $\times 20$ could be determined.

5.3 User Interaction

A major part of the proposed method consists of superpixel-based user interaction. One of the aims of the proposed method was to reduce human error during the training procedure of a machine learning algorithm and thereby reducing the variance between different pathologists. Hence the user interaction is evaluated separately as well.

5.3.1 Setup

The evaluation procedure works similar to the evaluation of the offline part. But instead of using the ground truth segmentation, the superpixel-wise segmentation for each of the nine sections gets used to train the classifier. The resulting random forests are used for evaluating the WSIs. Subsequently their concordance rates with both the ground truth and the classification results of other pathologists are computed.

In the evaluation three pathologists took part who used the proposed method on three different WSIs for classifying metastases within the digitized tissue of lymph nodes. An evaluation with such a small group of test subjects resembles the characteristics of an expert review an is relatively common for user studies involving pathologist. Studies by Varga et al. [80] and Gilles et al. [7] which examined similar topics used test groups of size 4-5.

The pathologist were presented with three WSIs which all contained metastases. After a short introduction to the method they started the classification without training.

	Pathologist 1	Pathologist 2	Pathologist 3
Ground truth WSI 1	0.4084	0.7221	0.7161
Ground truth WSI 2	0.3843	0.6982	0.7419
Ground truth WSI 3	0.4067	0.7422	0.7219

Table 5.2: Concordance rate between the classification results generated by three pathologist and the ground truth corresponding to the WSIs that were examined.

Afterwards the classifier which they did train were used to classify the whole WSI. This part was carried without the involvement of the pathologists.

5.3.2 Findings

First the overall accuracy regarding the concordance with the ground truth for each classification are discussed; they are shown in table 5.2. Secondly the inter-rater agreement is discussed by computing the concordance rate between each of the classification results using Cohen's kappa (see section 5.1.1); these results are shown table 5.3.

When comparing both tables it is obvious that the concordance rate between two pathologists is higher than their particular concordance with the ground truth: The average concordance rate with the ground truth amounts to 0.6158, whereas the concordance rate between the classification results generated by the pathologists is 0.859. The reason for this difference is visualized in figure 33. It shows the section of a ground truth image with all three classification results. Since no postprocessing steps, like morphological dilation or erosion, or any kind of noise cancelling filters are applied many falsely as positive classified regions are visible. On the other hand the false-negative rate is relatively low: Parts of the image that are coloured black refer to regions which non of the classification results classified as positive although it is labeled positive by the ground truth. It is characteristically located at the borders of the cancerous areas which indicates that post processing steps could improve the classification result in those areas. Furthermore many small discontiguous parts, mostly coloured in green and turquoise, are visible within the classification result, which easily can be removed by applying a noise filter, e. g. an anisotropic diffusion filter. Besides the green and turquoise parts, grey parts occur with a similar characteristic which

means that all three classification results do match in these areas. This suggests that these misclassifications are not a result of badly trained random forests but rather can not be distinguished from the cancerous regions by the proposed method. Hence the inter-rater concordance is higher than the average concordance with the ground truth. In summary it can be said, that the concordance rate with the ground truth can be improved with simple methods while the inter-rater concordance rate is sufficiently high with 85.9%. For comparison the results from a recent study that was introduced in section 1.1 by Elmore et al. [9] shall be recalled: Their study with 110 pathologists and 6900 cases showed an overall concordance rate of 75.3%. Furthermore the final inter-rater concordance rate exceeds the goal of 80% defined in section 1.5.

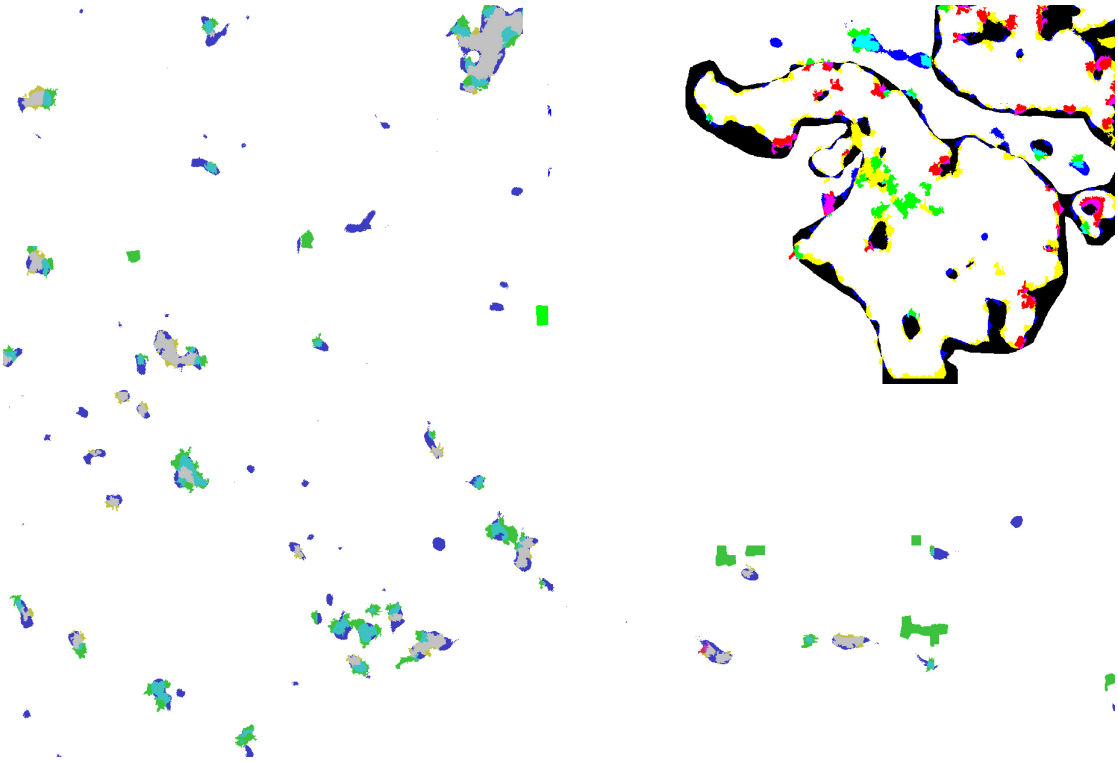


Figure 33: Concordance of all the classification results and the ground truth. The three classification results were compared with each other and the ground truth: The white parts indicate areas in which all three classification results match each other and the ground truth. Parts colored in grey indicate areas in which the classification results conform with each other but differ from the ground truth, whereas in the colored areas at least one classification result deviates from the others.

The qualitative evaluation was done by both observing the pathologists while using the proposed method to classify the tissue and loosely structured interview afterwards.

It was noticeable that the latency times between two sections, which are ~ 10 seconds, disrupted the work flow. Hence the whole examination took ~ 7 minutes per WSI depending on the WSI and influencing variables like, which section was chosen for the initial training of the forest.

The interaction techniques of brushing and erasing first were found unusual by all the participants since they vastly differ from common interaction techniques: When brushing a superpixel this does not mean it gets selected. Furthermore a stroke does not only affect the brushed area but the whole image. Hence the interaction was found demanding at first but all pathologists internalized the principle very quickly. The fast update and immediate feedback also got positive remarks.

One scenario in particular caused confusion during the training process: When the user applies the brush and the eraser alternately, the random forest gets refined until a point where no further improvements in accuracy are possible. Though sometimes for difficult sections it might be that the pathologist is not pleased with the classification result. At this point the user's only option left is to accept the classification result as is.

During the overt observation it was noticeable which visual representations were found most useful for the training procedure. Most often the users used the heat map representation for the borders and the superpixel-wise representation for the classification result. Visibility of both visual elements was often toggled in order to examine the classification result. By contrast the option, where the superpixel borders are used for displaying the classification result, was never used, since the visual contrast is too low. When the pathologists first tried out this setting a phenomenon called change blindness was noticeable. Due to the nature of the interaction technique the whole classification results changes while the user only focuses on one region in particular. When the change of a visual stimulus is too low it goes unnoticed. Hence this representation was found impractical. Another feature that was rarely used is the confidence slider. Even though it works just like intended and the pathologists also acknowledged it when they were told to use it for test purposes, they preferred using the brush and the eraser for the classification task.

In summary it can be said, that the proposed user interaction lead to the desired improvement in inter-rater concordance rate even though the concordance with the ground truth is worthy of improvement; some simple preprocessing steps already could lead to a significant increase of accuracy. Furthermore the superpixel-based interaction technique in interplay with the online random forest implementation was evaluated positively. What applies in this regard too, is that an improvement in accuracy will lead

	Pathologist 1	**Pathologist 2**	**Pathologist 3**
Pathologist 1	–	0.872 (0.89/0.87/0.85)	0.816 (0.79/0.84/0.82)
Pathologist 2	0.872 (0.89/0.87/0.85)	–	0.839 (0.87/0.83/0.82)
Pathologist 3	0.816 (0.79/0.84/0.82)	0.839 (0.87/0.83/0.82)	–

Table 5.3: Inter-rater agreement between three pathologist who examined three WSIs. The top row of each cell states the average agreement over three classification results of the two pathologists, whereas the bottom row states the agreement for each of the WSIs.

to an improvement of user satisfaction. Likewise the usefulness of the superpixel-based visualizations were acknowledged in general by the pathologists that took part in the evaluation.

6 Discussion

Before starting the discussion the major findings of this work shall be recalled. They are listed in a chronological order regarding the interaction pipeline:

(a) Guiding the pathologists awareness to certain ROIs did not attain the intended overvalue. By contrast, the feature created confusion and was therefore merely used.

(b) Letting the user decide, on which level of detail to analyse the WSI, led to resolution values lower than required for the algorithm to provide sufficiently accurate results.

(c) The superpixel-based user interaction for the training of the random forest and thereby generating ground truth data quickly got adopted by the pathologists.

(d) The tradeoff between accuracy and computation time for the algorithm that suggests a new section based on previous segmentations by the pathologist needs to be tuned. Furthermore the visual clues indicating change during this step, need further improvements in order to support the user's cognition.

(e) The introduced random forest implementation is able to compete with more advanced deep learning methods that were submitted to the Camelyon challenge (at least for the use case considered in this work).

(f) Using the borders of superpixels for displaying context information was proven useful for the use case presented in this work.

(g) The inter-rater concordance could be increased using the proposed method.

In general the proposed method suffices since the aim defined in section 1.5 was reached but still many improvements that can be made, manifested during the evaluation. Hence in the following these findings are discussed regarding their meaning and their importance for future work.

The first point, point (a) relates to the topic discussed in section 4.3. The initial intention was to increase the visual contrast by displaying an overlay quasi-segmentation in order to guide the pathologists attention to similar structures. But neither did the visualization increase the pathologists insight nor did it ease the process of finding ROIs. Nonetheless such a visualization would surely prove to be useful if its predictions were based on information the user entered in the system. This aspect gets picked up later when the prediction of new sections is discussed.

In the requirements definition in section 4.1 the second point demands *not to rely on prior knowledge*. This requirement was defined with regards to the other approaches introduced in section 2.1, which all either rely on a pretrained classifier or at least really specific characteristics of the tissue of interest (for instance the approach by Bahlmann et al. [38]). However, if the method is designed in such a way that it not completely relies on a pretrained classifier but enables the user to classify the tissue from scratch, it might be advantageous for the user to use available knowledge.

For instance a hybrid version would be conceivable in which the user selects a pretrained classifier from a library of different classifiers, which generates a heat map like overlay, that guides the user's attention, as it was intended. The main difference to the approaches from section 2.1 would be that said pretrained classifiers are not determining the classification process from start but leave the user in charge while supporting him in what he does. As mentioned earlier the heat map topic is picked up again when discussing the method that suggests the section that gets presented to the user when he clicks on *next*.

The second point (b) is based on the assumption that a user would choose a sufficiently low level for examining the tissue. The intention behind this design choice was to keep the approach as simple as possible and as complex as necessary. For most cases this assumption produced sufficiently accurate output but especially for obvious cases pathologists tend to choose higher levels in the pyramid image. Doing so they rely less on textural features but more on morphological ones. Hence, since the proposed method only incorporates textural features, a seamless classification is complicated, because the details needed are only available on lower levels of the pyramid image. This realisation could be used in future research in the design of feature sets: Hybrid real-time capable machine learning approaches could use different features depending on the factor of magnification. This could improve both the accuracy and the performance as well.

Furthermore the human cognition has a strong influence on choosing the section on which the forest gets trained. For instance, users not knowingly apply gestalt principles as described by Wertheimer [81] when choosing a section of the WSI, often resulting in contradictions between the laws of closure and good continuation [81]. As a consequence users rarely choose sections that crop compact metastases, from which follows that users often zoom out so that the metastases fit into screen space resulting in image resolutions too low for sufficiently accurate texture classification. Therefore an adapted version of the proposed method for choosing the right level needs to be developed, that can cope with the described error.

After the user found a ROI the next step is the superpixel-based tissue classification. The presented approach has a similar goal as the method introduced by Peter et al. [34]: It results in a delineation of the ROI and further refines the classifier. In contrast to the method by Peter et al. [34] the superpixel-based approach replaces the tedious delineation process. The difference to delineation is, that no outline is generated but the ROI is *painted* (see section 4.4.1). Even though the user has to habituate to the novel interaction style the pathologists quickly understood the functionality. In section 5, covering the evaluation, a problem is described which happens during the end of a classification process, when classifier's accuracy can not be increased further by the normal training procedure. To cope with this problem a simple mode could be introduced which sets the label of the superpixels that were brushed as final.

Since the proposed method is not restricted to random forests it is conceivable to include it in any kind of online machine learning method where a delineation is beneficial to the classification accuracy, like the one proposed by Peter et al. [34].

After the user classified a section image he either can request a new section for examination from the framework or continue the examination in a classical way by zooming out and searching for similar regions. The proposed method for suggesting new sections is not based on a region scoring function as proposed by Peter et al. [34], since the complexity for computing all the used features would be to demanding for a real-time application. In consequence a greedy approach was used that chooses the first section that suffices a certain criteria (see section 4). In praxis it turned out that this approach was neither fast nor accurate enough. A recommendation for future research would be to use two types of forests: A more complex one for the final classification of the WSI and another less complex one for all steps that involve user interaction. The feature set for the less complex forest could be chosen by evaluating the frequencies with which the features from the more complex forest are used (see section 4.4.2 and

figure 22). When selecting the features, their complexity also has to be considered; for instance, even though the Haralick feature f6 is the most frequently used feature for this works use case, it is not applicable in a real-time application. Hence it is advisable to choose the features according to a metric in order to sustain real-time capability. Figure 22 shows that a simple feature, whose computation time is negligible, like the U-channel also can contribute to the classification performance significantly.

A further issue with suggesting new sections for evaluation is that the pathologist is not supported in building a mental model of the WSI. When examining the WSI in a classical manner the pathologist navigates through a lot of tissue that holds no valuable information for him, but nevertheless gains insight over the structure of the WSI and the distribution of certain tissue types. The proposed method for suggesting sections of interest can be compared to taking the underground rather than going by bus: Even though it is faster it is more complicated for a passers-by to understand the cities structure and the distribution of points of interest.

Future research could address this issue on how to support the pathologist in building a mental model of the WSI while avoiding him to navigate through valueless tissue. One possible approach for dealing with this topic could be to implement a random forest which is capable of evaluating the entire WSI as an anytime algorithm. Said algorithm could be used to compute a heatmap that shows a preview of the classification result as a heat map-like overlay visualization. Maybe also a visualization similar to the one in figure 8 could help in keeping track of which areas already were examined. Both visualizations could be combined as shown in figure 26. Another approach would be to evaluate how zoom animations, connecting two successive sections, do work in practice. This way it could be ensured that the user less likely looses track of the relation between focus (each section image) and its context (the entire WSI).

The next step, after finishing the training procedure is the evaluation of the whole WSI. Even though the evaluation was carried out differently compared to the Camelyon challenge it is conceivable that the classification results generated by the proposed method results in accuracy rates slightly higher than the average classification result computed by the participating teams of the Camelyon challenge. Again, the results are not directly comparable: The proposed method is a semi-automatic CAD application that could not work without the supervision of a pathologist, whereas the participants of the Camelyon challenge were designing automated approaches. The results of the Camelyon challenge only serve as a baseline to show that the proposed method is able to produce classification results with accuracies of the same order of magnitude.

More important for this work are the user interaction and the visual clues relating thereto: It was shown that the superpixel-based visualizations are appropriate for visualizing context information without obstructing the tissue and without impeding the pathologist in examining the WSI. Furthermore they were shown to perform well as a link between the domain of pathology and machine learning: Due to their appearance, which is intensely intertwined with the tissue structure, they are merely questioned and taken as a natural part of the user interaction. Hence it was suggested to use the same visualization techniques on higher levels which only is possible with a random forest implementation that is capable of evaluating entire WSIs at low resolutions.

Finally it was also shown that the inter-observer agreement could be improved using the proposed method. Even though the pathologist were not given any training they quickly understood the principle and started working without supervision. The increase is presumably attributable to mainly three things: (a) Unlike other methods that were introduced in section 2.1 the proposed method uses completely labeled mask images as input for the machine learning algorithm, (b) the visualization enabled the pathologists to properly train the forest, and (c) the pathologists were presented with similar looking sections of the WSI.

This short discussion showed which aspects of the proposed method can be utilized for future research or can be integrated into other state of the art methods. This mainly refers to the superpixel-based interaction technique as well as the superpixel-based visualization. Both are sufficiently generic so that they can be used for many online machine learning applications across different domains.

One shortcoming of the evaluation is the size of the group of participating pathologists as well as the amount of cases that were looked into. For instance none of the WSIs that were evaluated contained metastases since the proposed method only provides pathologist with little support for this use case. On the other hand, when looking at figure 5 it can be seen that the concordance for benign without atypia (atypical hyperplasia) breast biopsies is relatively high compared to, for instance, breast biopsies with atypia (diagnostic classification with only 48% concordance rate) [9]. Hence a larger study with a refined version of the proposed method, in which the suggested improvements got incorporated, would lead to more significant and more interesting results.

Furthermore the study could not investigate the influence of all variables that affect the outcome. Two variables were found especially important whose influence on the classification result needs to be evaluated but were not: (a) the first section image

on which the initial random forest is trained, and (b) the classification results that the pathologist is presented with during the training procedure. The structure of the random forest is defined during the initial training step using the section image that was chosen by the pathologist. Even though the random forest is altered during the training procedure, this implies that the structure and the textural composition of the first section image influences the performance of the classifier significantly. Secondly when switching to a new section the user is presented with a classification result that was generated by applying the trained classifier. Depending on the progress of the training procedure many false-positives and false-negatives will be part of the classification result. Hence for further development it would be important to investigate if the pathologist is biased by the classification result and tends to trust the output of the algorithm or if this type of interference is not influencing the pathologists' performance.

The clinical relevance of the proposed method in its current state is relatively low, but it holds great promise for future developments to reduce both the pathologists' workload and the subjectivity in diagnosis. The proposed method could either be refined further or parts of it that were found useful could be integrated in more established methods. For instance one state of the art method introduced in section 2.1 by Peter et al. [34] would benefit from an easy method that allows a pathologist to easily generate a (super-) pixel-wise classification result.

The possibilities for future research are broad and many were already mentioned in the discussion. The most essential research would be to investigate a refined version of the proposed method that incorporates the improvements that were suggested in this section. A next step could be to further improve the proposed method or to specialize it for certain use cases: Figure 34 shows how the proposed method was used to classify different parts of an MR scan of the brain. Said image furthermore shows that the used feature set only is capable of distinguishing between different texture features; features describing locations ore morphological attributes could easily improve the segmentation.

This is closely linked to applying the proposed method to problems and tasks which are not part of the medical domain. For instance it could be used in graphics editing tools for the selection of certain image contents. Many more application areas arise when thinking about changing the feature set: For instance morphological features could be incorporated which opens up new fields of application.

This section gave a conclusion of the advantages and drawbacks of this method. For many drawbacks simple improvements were discussed that could lead to significant

increases in user satisfaction and classification accuracy. The advantages were discussed as well with a view to possible alternative explanations of the findings. The most important findings that were discussed are the superpixel-based visualization and -interaction techniques that were introduced in this work. Both are recommended to be used and adapted for ongoing research.

7 Summary and Future Work

In the analysis of medical images many different tasks entail the classification of ROIs. Especially the analysis of WSIs has proven to be challenging due to their size and their high information density. This works case of application was the detection of metastases in WSIs of lymph node sections taken from breast cancer patients. This application is of high clinical relevance since the diagnosis requires a large amount of reading time from the pathologist. Furthermore recent studies showed that the diagnosis of WSIs in general and tissue classification for breast cancer staging in particular is prone to high inter-rater variability. In order to motivate for this work not only the incidence, prevalence and mortality of breast cancer in general was presented but also the current state of diagnosis accuracy. It was discussed which aspects, like cost efficiency, do influence the performance of pathologists and therefore lead to low inter-rater concordance rates.

Accordingly a successful solution would hold the great promise to reduce the pathologists' workload while increasing the inter-rater concordance rate. Hence the main premise of this work is that a reduction of diagnosis subjectivity could be reached by mainly two things: (a) by presenting the pathologists with similar sections of the entire WSI, and (b) by supporting him during the training procedure with visual feedback that is suitable for the domain and the prior knowledge of the pathologist.

The main goal of this work was to develop a method that generates tissue classifications with a higher inter-rater concordance rate than the one found by the study of Elmore et al. [9], which was 75.0%. The goal was set to improve it by 5% which was exceeded by further 5%.

Existing approaches and other promising techniques, especially full automatic approaches and deep learning methods, were discussed regarding their applicability. It was found that a semi-automatic approach is most suitable for pathology in general, since the whole field is undergoing a large shift from analogue to digital pathology. Comprehensive automation first is applicable after digital pathology became an established part in daily clinical routines. Furthermore this method is designed for use cases in which no labeled training data is available. Hence methods that require thousands of

samples, like neural networks, are not applicable, since such methods usually require thousands of samples for training.

The proposed method comprises of a superpixel-based training procedure that allows the pathologist to easily train an online random forest for tissue classification. Said procedure is backed by adequate visualizations that provides immediate feedback and supports the pathologist while keeping the additional mental workload for the pathologist as minimal as possible. The fact that the classifier is not trained on any kind of prior knowledge is different from many other state of the art approaches. Hence it allows the pathologist to search for any kind of tissue, and not only for certain types the pre-trained classifier might be specialized on.

The evaluation was carried out by three pathologist. Each on evaluated the same three WSIs using the proposed method. The qualitative evaluation was done by observation and unstructured interviews afterwards. It has been found that the pathologists quickly adopted the novel user interaction style after a short period of familiarisation. Furthermore the fast feedback was positively acknowledged as well es the heat map visualisation which is displayed within the superpixel borders.

The quantitative evaluation computed both the concordance rate of the classification results generated by the pathologists with the ground truth and the average concordance rate between the generated classifications. While the concordance rate with the ground truth was found to be average compared to the results from the Camelyon challenge, the inter-observer concordance rate was higher than 75% which was the value for inter-rater concordance found in a study with 110 pathologists by Elmore et al. [9] in a real world scenario. The inter-rater concordance rate achieved by using the proposed method is 85.9%. When comparing these results it has to be considered that the sizes of both groups of subjects differed vastly (3/110) as well as the amount of classified WSI (3/240). Therefore the evaluation of the proposed method is not as significant but still shows a tendency towards a higher inter-rater concordance rate.

Since the proposed method is kept as generalizable as possible further applications for different medical imaging modalities are conceivable. Little adaptions regarding the set of features might be necessary in order to classify objects instead of materials (see section 4.4.2). Without adapting the feature set the proposed method was applied to a segmentation task of the brain. The result of the segmentation of a publicly available IBSR2-18 brain Dataset of a brain MR scan [82] is shown in figure 34. The region highlighted in green is one of 32 labeled ROIs. It is easy to see, that both symmetrical regions get classified, whereas the ROI only covers the left hemisphere. This can easily

be avoided by including features into the set of features that are able to distinguish shape information or that incorporate any kind of location information.

Furthermore the proposed method is applicable for segmentation task outside the medical imaging domain. Especially outlining a ROI during a segmentation task is a tedious but necessary step for producing detailed segmentations. This is made easy using the proposed method such that a use by novice users is conceivable. In general the presented user-interaction can be used in a wide variety of applications. For instance, Peter et al. [34] proposed two versions of their method: one that includes a tedious delineation task for generating input for their machine learning approach and a second method which generates rougher input leading to lower accuracy. Incorporating the superpixel-based user-interaction presented in this work would embody a suitable tradeoff between low accuracy and tedious user interactions.

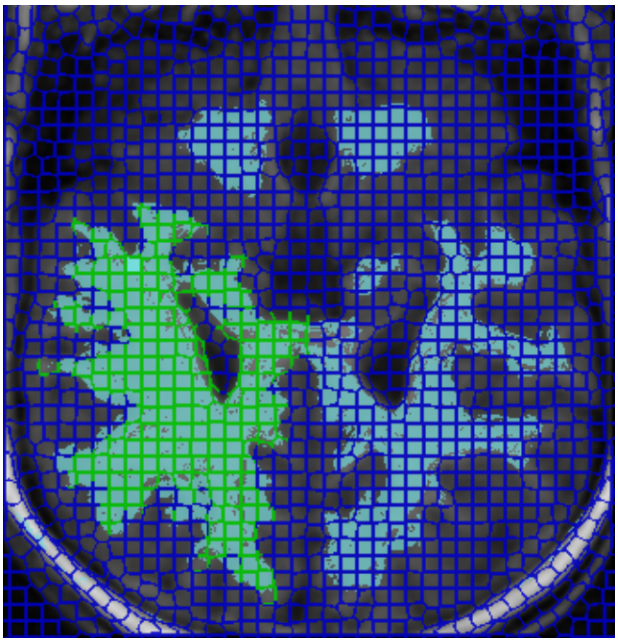


Figure 34: Application of the proposed method to an MR dataset of the brain (publicly available IBSR2-18 Brain Dataset [82]). The proposed method was not adapted to cope with different kind of data.

When developing CAD applications for pathologists who work with WSIs it is important to consider the challenging characteristics of the image modality itself. Especially when designing machine learning algorithms characteristics, which occur when working in a cross-domain context, need to be taken into account: It must not be asked of the pathologist to understand the algorithm, the algorithm needs to adapt to the capabilities of the pathologist and the context of his daily routines. This way the mental workload can be reduced so that the pathologist can focus on his actual task. Particular attention should be paid to the process of building a mental model. As it was shown in this work it is important to make sure that the pathologist is fully aware of the system's state. This can be achieved by providing fast and adequate feedback visualisations and, especially important for WSIs, context visualisations

when navigating through the slide. Otherwise the pathologist will start to doubt the algorithm which impairs the overall performance. Many aspects regarding the reduction of the pathologists mental workload that need consideration when designing machine learning algorithms for the analysis of WSIs were presented in this work. Some inconsistencies that were found during the evaluation are still part of the proposed interaction framework and may serve as bad examples.

Even though the proposed method is not as robust and fail-safe as it would need to be, in order to be established in a daily, clinical routine, it can serve as starting point for further research. It could either be further developed and refined according to the discussed suggestions of improvement or individual aspects can be extracted in order to use them in other methods. But most importantly this work presented many user interaction related aspects that need to be considered when developing machine learning based CAD applications for the analysis of WSIs. These are universally applicable in many scenarios and help to improve overall performance, by reducing the human error.

Bibliography

[1] Jacques Ferlay et al. "Cancer incidence and mortality worldwide: sources, methods and major patterns in GLOBOCAN 2012". In: *International journal of cancer* 136.5 (2015).

[2] Lindsey A Torre et al. "Global cancer incidence and mortality rates and trends—an update". In: *Cancer Epidemiology and Prevention Biomarkers* 25.1 (2016), pp. 16–27.

[3] Drazen M Jukić et al. "Clinical examination and validation of primary diagnosis in anatomic pathology using whole slide digital images". In: *Archives of pathology & laboratory medicine* 135.3 (2011), pp. 372–378.

[4] Thomas W Bauer et al. "Validation of whole slide imaging for primary diagnosis in surgical pathology". In: *Archives of pathology & laboratory medicine* 137.4 (2013), pp. 518–524.

[5] John S Meyer et al. "Breast carcinoma malignancy grading by Bloom–Richardson system vs proliferation index: reproducibility of grade and advantages of proliferation index". In: *Modern pathology* 18.8 (2005), pp. 1067–1078.

[6] İpek Işık Gönül et al. "Comparison of 1998 WHO/ISUP and 1973 WHO classifications for interobserver variability in grading of papillary urothelial neoplasms of the bladder". In: *Urologia internationalis* 78.4 (2007), pp. 338–344.

[7] Floyd H Gilles et al. "Pathologist interobserver variability of histologic features in childhood brain tumors: results from the CCG-945 study". In: *Pediatric and Developmental Pathology* 11.2 (2008), pp. 108–117.

[8] Daniël Eefting et al. "Assessment of interobserver variability and histologic parameters to improve reliability in classification and grading of central cartilaginous tumors". In: *The American journal of surgical pathology* 33.1 (2009), pp. 50–57.

[9] Joann G Elmore et al. "Diagnostic concordance among pathologists interpreting breast biopsy specimens". In: *Jama* 313.11 (2015), pp. 1122–1132.

[10] Metin N Gurcan et al. "Histopathological image analysis: A review". In: *IEEE reviews in biomedical engineering* 2 (2009), pp. 147–171.

[11] Maria-Paula Curado et al. *Cancer incidence in five continents, Volume IX*. IARC Press, International Agency for Research on Cancer, 2007.

[12] Rebecca L Siegel, Kimberly D Miller, and Ahmedin Jemal. "Cancer statistics, 2016". In: *CA: a cancer journal for clinicians* 66.1 (2016), pp. 7–30.

[13] Michelle D Althuis et al. "Global trends in breast cancer incidence and mortality 1973–1997". In: *International journal of epidemiology* 34.2 (2005), pp. 405–412.

[14] L Chatenoud et al. "Trends in mortality from major cancers in the Americas: 1980–2010". In: *Annals of oncology* 25.9 (2014), pp. 1843–1853.

[15] Freddie Bray et al. "Global cancer transitions according to the Human Development Index (2008–2030): a population-based study". In: *The lancet oncology* 13.8 (2012), pp. 790–801.

[16] Danny R Youlden et al. "Incidence and mortality of female breast cancer in the Asia-Pacific region". In: *Cancer biology & medicine* 11.2 (2014), p. 101.

[17] Abhishek Chatterjee, Nicholas Serniak, and Brian J Czerniecki. "Sentinel lymph node biopsy in breast cancer: A work in progress". In: *Cancer journal (Sudbury, Mass.)* 21.1 (2015), p. 7.

[18] National Cancer Institute LLC. *Sentinel Lymph Node Biopsy*. 2011. URL: https://www.cancer.gov/about-cancer/diagnosis-staging/staging/sentinel-node-biopsy-fact-sheet (visited on 09/14/2017).

[19] Raphael Rubin, David S Strayer, Emanuel Rubin, et al. *Rubin's pathology: clinicopathologic foundations of medicine*. Lippincott Williams & Wilkins, 2008.

[20] Wendie A Berg et al. "Diagnostic accuracy of mammography, clinical examination, US, and MR imaging in preoperative assessment of breast cancer". In: *Radiology* 233.3 (2004), pp. 830–849.

[21] Stuart J Schnitt. "Classification and prognosis of invasive breast cancer: from morphology to molecular taxonomy". In: *Modern Pathology* 23.S2 (2010), S60.

[22] Mitko Veta et al. "Breast cancer histopathology image analysis: A review". In: *IEEE Transactions on Biomedical Engineering* 61.5 (2014), pp. 1400–1411.

[23] Nikolas Stathonikos et al. "Going fully digital: Perspective of a Dutch academic pathology lab". In: *Journal of pathology informatics* 4 (2013).

[24] Kunio Doi. "Computer-aided diagnosis in medical imaging: historical review, current status and future potential". In: *Computerized medical imaging and graphics* 31.4 (2007), pp. 198–211.

[25] Navid Farahani, Anil V Parwani, and Liron Pantanowitz. "Whole slide imaging in pathology: advantages, limitations, and emerging perspectives". In: *Pathol Lab Med Int* 7 (2015), pp. 23–33.

[26] Christopher R King and John P Long. "Prostate biopsy grading errors: a sampling problem?" In: *International journal of cancer* 90.6 (2000), pp. 326–330.

[27] Morten Ladekarl. "Objective malignancy grading: a review emphasizing unbiased stereology applied to breast tumors". In: *Apmis* 106.S79 (1998), pp. 5–34.

[28] Farzad Ghaznavi et al. "Digital imaging in pathology: whole-slide imaging and beyond". In: *Annual Review of Pathology: Mechanisms of Disease* 8 (2013), pp. 331–359.

[29] Magdalene Merk, Ruth Knuechel, and Alberto Perez-Bouza. "Web-based virtual microscopy at the RWTH Aachen University: Didactic concept, methods and analysis of acceptance by the students". In: *Annals of Anatomy-Anatomischer Anzeiger* 192.6 (2010), pp. 383–387.

[30] Michelle D Reid et al. "Calculation of the Ki67 index in pancreatic neuroendocrine tumors: a comparative analysis of four counting methodologies". In: *Modern pathology: an official journal of the United States and Canadian Academy of Pathology, Inc* 28.5 (2015), p. 686.

[31] Guido Rindi et al. "TNM staging of foregut (neuro) endocrine tumors: a consensus proposal including a grading system". In: *Virchows Archiv* 449.4 (2006), pp. 395–401.

[32] Lowell B Anthony et al. "The NANETS consensus guidelines for the diagnosis and management of gastrointestinal neuroendocrine tumors (nets): well-differentiated nets of the distal colon and rectum". In: *Pancreas* 39.6 (2010), pp. 767–774.

[33] Anne E Carpenter et al. "CellProfiler: image analysis software for identifying and quantifying cell phenotypes". In: *Genome biology* 7.10 (2006), R100.

[34] Loïc Peter et al. "Assisting the examination of large histopathological slides with adaptive forests". In: *Medical image analysis* 35 (2017), pp. 655–668.

[35] Ezgi Mercan et al. "Localization of diagnostically relevant regions of interest in whole slide images". In: *Pattern Recognition (ICPR), 2014 22nd International Conference on*. IEEE. 2014, pp. 1179–1184.

[36] Arnout C Ruifrok, Dennis A Johnston, et al. "Quantification of histochemical staining by color deconvolution". In: *Analytical and quantitative cytology and histology* 23.4 (2001), pp. 291–299.

[37] Ezgi Mercan et al. "Localization of diagnostically relevant regions of interest in whole slide images: A comparative study". In: *Journal of digital imaging* 29.4 (2016), pp. 496–506.

[38] Claus Bahlmann et al. "Automated detection of diagnostically relevant regions in H&E stained digital pathology slides". In: *Proc. SPIE*. Vol. 8315. 2012.

[39] Andrew H Beck et al. "Systematic analysis of breast cancer morphology uncovers stromal features associated with survival". In: *Science translational medicine* 3.108 (2011), 108ra113–108ra113.

[40] Babak Ehteshami Bejnordi et al. "A multi-scale superpixel classification approach to the detection of regions of interest in whole slide histopathology images". In: *SPIE Medical Imaging*. International Society for Optics and Photonics. 2015, 94200H–94200H.

[41] Jun Kong et al. "Computer-aided evaluation of neuroblastoma on whole-slide histology images: Classifying grade of neuroblastic differentiation". In: *Pattern Recognition* 42.6 (2009), pp. 1080–1092.

[42] Jun Kong et al. "Image analysis for automated assessment of grade of neuroblastic differentiation". In: *Biomedical Imaging: From Nano to Macro, 2007. ISBI 2007. 4th IEEE International Symposium on*. IEEE. 2007, pp. 61–64.

[43] Pavel Pudil, Petr Somol, and Michal Haindl. *Introduction to statistical pattern recognition*. 1990.

[44] Pavel Pudil et al. "Floating search methods for feature selection with nonmonotonic criterion functions". In: *Pattern Recognition, 1994. Vol. 2-Conference B: Computer Vision & Image Processing., Proceedings of the 12th IAPR International. Conference on*. Vol. 2. IEEE. 1994, pp. 279–283.

[45] Leo Breiman. "Random forests". In: *Machine learning* 45.1 (2001), pp. 5–32.

[46] Radhakrishna Achanta et al. "SLIC superpixels compared to state-of-the-art superpixel methods". In: *IEEE transactions on pattern analysis and machine intelligence* 34.11 (2012), pp. 2274–2282.

[47] Gérard Biau and Erwan Scornet. "A random forest guided tour". In: *Test* 25.2 (2016), pp. 197–227.

[48] Amir Saffari et al. "On-line random forests". In: *Computer Vision Workshops (ICCV Workshops), 2009 IEEE 12th International Conference on*. IEEE. 2009, pp. 1393–1400.

[49] Richard E Bellman. *Adaptive control processes: a guided tour*. Princeton university press, 2015.

[50] Antonio Criminisi, Jamie Shotton, Ender Konukoglu, et al. "Decision forests: A unified framework for classification, regression, density estimation, manifold learning and semi-supervised learning". In: *Foundations and Trends® in Computer Graphics and Vision* 7.2–3 (2012), pp. 81–227.

[51] Philipp Kainz et al. "You should use regression to detect cells". In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer. 2015, pp. 276–283.

[52] Franklin C Crow. "Summed-area tables for texture mapping". In: *ACM SIGGRAPH computer graphics* 18.3 (1984), pp. 207–212.

[53] Roy D Pea. "User centered system design: new perspectives on human-computer interaction". In: *Journal educational computing research* 3 (1987), pp. 129–134.

[54] Don Norman. *The design of everyday things: Revised and expanded edition*. Basic Books (AZ), 2013.

[55] Mica R Endsley. *Designing for situation awareness: An approach to user-centered design*. CRC press, 2016.

[56] Gary A Klein. *A recognition-primed decision (RPD) model of rapid decision making*. Ablex Publishing Corporation New York, 1993.

[57] R Lipshitz. "Decision making in the real world: Developing descriptions and prescriptions from decision maker's retrospective accounts". In: *Boston, MA: Boston University Center for Applied Sciences* (1987).

[58] David Noble, Carla Grosz, and Deborah Boehm-Davis. *Rules, Schema, and Decision Making.* Tech. rep. ENGINEERING RESEARCH ASSOCIATES INC VIENNA VA, 1987.

[59] Ben Shneiderman. "The eyes have it: A task by data type taxonomy for information visualizations". In: *Visual Languages, 1996. Proceedings., IEEE Symposium on*. IEEE. 1996, pp. 336–343.

[60] Brock Craft and Paul Cairns. "Beyond guidelines: what can we learn from the visual information seeking mantra?" In: *Information Visualisation, 2005. Proceedings. Ninth International Conference on*. IEEE. 2005, pp. 110–118.

[61] Christopher G Healey. "Choosing effective colours for data visualization". In: *Visualization'96. Proceedings.* IEEE. 1996, pp. 263–270.

[62] ME Birch and RA Cary. "Elemental carbon-based method for monitoring occupational exposures to particulate diesel exhaust". In: *Aerosol Science and Technology* 25.3 (1996), pp. 221–241.

[63] Wassily Hoeffding. "Probability inequalities for sums of bounded random variables". In: *Journal of the American statistical association* 58.301 (1963), pp. 13–30.

[64] Pedro Domingos and Geoff Hulten. "Mining high-speed data streams". In: *Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM. 2000, pp. 71–80.

[65] Pawel Matuszyk, Georg Krempl, and Myra Spiliopoulou. "Correcting the usage of the hoeffding inequality in stream mining". In: *International Symposium on Intelligent Data Analysis*. Springer. 2013, pp. 298–309.

[66] Jean Ponce and Andrew Zisserman. *Object Representation in Computer Vision II: ECCV'96 International Workshop, Cambridge, UK, April 13-14, 1996. Proceedings*. Springer Science & Business Media, 1996.

[67] Keinosuke Fukunaga. *Introduction to statistical pattern recognition*. Academic press, 2013.

[68] Inge M Ambros et al. "Morphologic features of neuroblastoma (Schwannian stroma-poor tumors) in clinically favorable and unfavorable groups". In: *Cancer* 94.5 (2002), pp. 1574–1583.

[69] OpenCV 2.4.13.4 documentation. 2017. URL: `https://docs.opencv.org/2.4/modules/contrib/doc/facerec/facerec_tutorial.html` (visited on 11/07/2017).

[70] Li Wang and Dong-Chen He. "Texture classification using texture spectrum". In: *Pattern Recognition* 23.8 (1990), pp. 905–910.

[71] Robert M Haralick, Karthikeyan Shanmugam, et al. "Textural features for image classification". In: *IEEE Transactions on systems, man, and cybernetics* 6 (1973), pp. 610–621.

[72] Antje Reich. *New Image Sharting Platform Anyslide Opens up Digital Pathology to Everyone*. 2011. URL: https://micro-dimensions.com/news/2017/8/31/new-image-sharing-platform-anyslide-opens-up-digital-pathology-to-everyone (visited on 11/06/2017).

[73] Jay H Lefkowitch. "Morphology of alcoholic liver disease". In: *Clinics in liver disease* 9.1 (2005), pp. 37–53.

[74] Wedad M Hanna et al. "Letter to the Editor". In: *Modern Pathology* 21.10 (2008), pp. 1278–1280.

[75] Jacob Cohen. "A coefficient of agreement for nominal scales". In: *Educational and psychological measurement* 20.1 (1960), pp. 37–46.

[76] Domenic V Cicchetti. "Methodological commentary the precision of reliability and validity estimates re-visited: distinguishing between clinical and statistical significance of sample size requirements". In: *Journal of Clinical and Experimental Neuropsychology* 23.5 (2001), pp. 695–700.

[77] Consortium for Open Medical Image Computing. 2017. URL: https://camelyon17.grand-challenge.org/data/ (visited on 11/17/2017).

[78] Dayong Wang et al. "Deep learning for identifying metastatic breast cancer". In: *arXiv preprint arXiv:1606.05718* (2016).

[79] Quincy Wong. "Perceptor: Under the Microscope with Machine Learning". available on https://camelyon16.grand-challenge.org/results/. (Visited on 11/17/2016).

[80] Zsuzsanna Varga et al. "How reliable is Ki-67 immunohistochemistry in grade 2 breast carcinomas? A QA study of the Swiss Working Group of Breast-and Gynecopathologists". In: *PloS one* 7.5 (2012), e37379.

[81] Max Wertheimer. "Untersuchungen zur Lehre von der Gestalt. II". In: *Psychologische forschung* 4.1 (1923), pp. 301–350.

[82] Neuroimaging Informatics Tools and Resources Clearinghouse. 2007. URL: https://www.nitrc.org/projects/ibsr (visited on 11/17/2017).

[83] Eizan Miyamoto and Thomas Merryman. “Fast calculation of Haralick texture features”. In: *Human computer interaction institute, Carnegie Mellon University, Pittsburgh, USA. Japanese restaurant office* (2005).

List of Figures

List of Tables

A Appendix

A.1 Haralick Features

Haralick et al. [71] defined 14 equations that use gray-Level co-occurrence spatial dependency matrices to define textural features. First statistical properties of co-occurrence matrices that are used in the actual computation of the Haralick features are defined, followed by the equations of the Haralick features them self. The source for these equations is [83].

Statistical Properties of Co-occurrence Matrices

$$R = \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} P(i,j) \tag{A.1}$$

$$p(i,j) = \frac{P(i,j)}{R} \tag{A.2}$$

$$p_x(i) = \sum_{j=1}^{N_g} P(i,j) \tag{A.3}$$

$$p_x(j) = \sum_{i=1}^{N_g} P(i,j) \tag{A.4}$$

$$p_{x+y}(k) = \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} p(i,j), i+j = k \text{ and } k = 2,3,\ldots,2N_g \tag{A.5}$$

$$p_{x-y}(k) = \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} p(i,j), |i-j| = k \text{ and } k = 0,1,\ldots,N_g - 1 \tag{A.6}$$

$$HXY1 = \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} p_x(i) P_y(j) \log\big(p_x(i) p_y(j)\big) \tag{A.7}$$

Description of 14 Haralick Features

$$f_1 = \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} p(i,j)^2 \tag{A.8}$$

$$f_2 = \sum_{k=0}^{N_g-1} k^2 p_{x-y}(k) \tag{A.9}$$

$$f_3 = \frac{\sum_{i=1}^{N_g} \sum_{j=1}^{N_g} (i,j) p(i,j) - \mu_x \mu_y}{\sigma_x \sigma_y} \tag{A.10}$$

$$f_4 = \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} (i-\mu)^2 p(i,j) \tag{A.11}$$

$$f_5 = \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} \frac{1}{1+(i-j)^2} p(i,j) \tag{A.12}$$

$$f_6 = \sum_{i=2}^{2N_g} i p_{x+y}(i) \tag{A.13}$$

$$f_7 = \sum_{i=2}^{2N_g} (i - f_8)^2 p_{x+y}(i) \tag{A.14}$$

$$f_8 = -\sum_{i=2}^{2N_g} p_{x+y}(i) \log(p_{x+y}(i)) \tag{A.15}$$

$$f_9 = \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} p(i,j) \log(p(i,j)) \tag{A.16}$$

$$f_{10} = \text{variance of } p_{x-y} \tag{A.17}$$

$$f_{11} = -\sum_{i=1}^{N_g-1} p_{x-y}(i) \log(p_{x-y}(i)) \tag{A.18}$$

$$f_{12} = \frac{f_9 - HXY1}{\max(HX, HY)} \tag{A.19}$$

$$f_{13} = \sqrt{1 - \exp^{-2(HXY2 - f_9)}} \tag{A.20}$$

$$f_{14} = (\text{second largest Eigenvalue of } Q)^{1/2} \tag{A.21}$$

$$\text{where } Q(i,j) = \sum_k \frac{P(i,k) p(j,k)}{p_x(i) p_y(k)}$$

A.2 Features Used by Pathologists

Features	Description	Category
Neuropil	The degree of presence of neuropil	No/minimal/sparse/ moderate/prominent
Cell cellularity	Number of cells per HPF (high power field)	Low/intermixed/ high/intermediate
Nuclear size	Variation of nuclear size	Variable/uniform/ pleomorphic
Nuclear shape	Shape regularity	Round-to-oval/ pleomorphic
Mitotic karyorrhectic index	Number of tumor cells in mitosis and karyorrhexis	Low/intermediate/high
Mitotic rate	Number of mitoses in 10 contiguous HPFs at 400× magnification	Low/high
Calcification	Presence of dense basophilic clumps or amorphous granular material	Yes/no

Table A.1: Features used by pathologists in the visual grading process [41].