Assisting the Examination of Large Histopathological Slides with Adaptive Forests

Loïc Peter[a], Diana Mateus[a,b], Pierre Chatelain[a,c], Denis Declara[a], Noemi Schworm[d], Stefan Stangl[d], Gabriele Multhoff[d,e], Nassir Navab[a,f]

[a]*Computer Aided Medical Procedures, Technische Universität München, Germany*
[b]*Institute of Computational Biology, Helmholtz Zentrum München, Germany*
[c]*Université de Rennes 1, IRISA, France*
[d]*Department of Radiation Oncology, Technische Universität München, Germany*
[e]*Institute of Innovative Radiotherapy (iRT), Department of Radiation Sciences, Helmholtz Zentrum München, Germany*
[f]*Computer Aided Medical Procedures, Johns Hopkins University, USA*

Abstract

The examination of biopsy samples plays a central role in the diagnosis and staging of numerous diseases, including most cancer types. However, because of the large size of the acquired images, the localization and quantification of diseased portions of a tissue is usually time-consuming, as pathologists must scroll through the whole slide to look for objects of interest which are often only scarcely distributed. In this work, we introduce an approach to facilitate the visual inspection of large digital histopathological slides. Our method builds on a random forest classifier trained to segment the structures sought by the pathologist. However, moving beyond the pixelwise segmentation task, our main contribution is an interactive exploration framework including: (i) a region scoring function which is used to rank and sequentially display regions of interest to the user, and (ii) a relevance feedback capability which leverages human annotations collected on each suggested region. Thereby, an online domain adaptation of the learned pixelwise segmentation model is performed, so that the region scores adapt on-the-fly to possible discrepancies between the original training data and the slide at hand. Three real-time update strategies are compared, including a novel approach based on online gradient descent which supports faster user interaction than an accurate delineation of objects. Our method is evaluated on the task of extramedullary hematopoiesis quantification within mouse liver slides. We assess quantitatively the retrieval abilities of our approach and the benefit of the interactive adaptation scheme. Moreover, we demonstrate the possibility of extrapolating, after a partial exploration of the slide, the surface covered by hematopoietic cells within the whole tissue.

Keywords: Random Forests, Histopathology, Online Learning, Active Learning, Domain Adaptation

1. Introduction

Histopathology is a crucial tool in modern clinical practice. It consists in the microscopic observation of biological tissues surgically extracted from a patient, in order to collect information regarding the presence or extent of a particular disease in the sample. In particular, it is part of the standard experimental protocol for the definitive diagnosis, grading and staging of most cancer types and plays an essential role in the design of appropriate patient-specific treatments. Histopathological examinations usually aim at searching for a certain kind of anatomical structure, like biomarkers, cancer cells or necrotic areas, whose presence or proportion within the tissue has to be quantitatively estimated. Although this procedure is traditionally conducted under a standard optical microscope, digital acquisitions of entire slices can be performed at comparable resolutions and are increasingly used by pathologists in their clinical workflow as well as for educational and research purposes (Farahani et al., 2015). Moving from optical to digital examinations has been shown to maintain similar diagnosis performances (Jukić et al., 2011; Bauer et al., 2013) and offers numerous additional advantages such as the applicability of image analysis algorithms, easier recordings and safer storage of patient data, and the possibility of displaying the scanned tissue to several examiners simultaneously (Al-Janabi et al., 2012). However, because of their high resolution, the size of digitally acquired images is very large and commonly reaches the order of a billion of pixels (Cooper et al., 2012). This increases the time required for manual quantification procedures: beyond the tediousness of annotating objects in images, a pathologist also spends a lot of time navigating through the large slide looking for evidence of the disease of interest. Moreover, the objects to localize may only be scarcely distributed, for instance at early stages of diseases or after a treatment has been applied. In such situations, the exploration phase even becomes the bottleneck of the process, since most of the time of the pathologist is spent scrolling through uninformative areas (Fig. 1).

Some characteristics of the field of histopathology bring specific challenges for an automated analysis of the acquired images. First, the accurate identification of diseased areas based on their visual appearance can be a very difficult task requiring a lot of expertise. Pathologists are typically trained several years before reaching satisfactory diagnosis abilities (Jaarsma et al., 2014), and the variability between experts remains nevertheless significant for several applications (Meyer et al., 2005; Gonul et al., 2006; Gilles et al., 2007; Eefting et al., 2009). An-

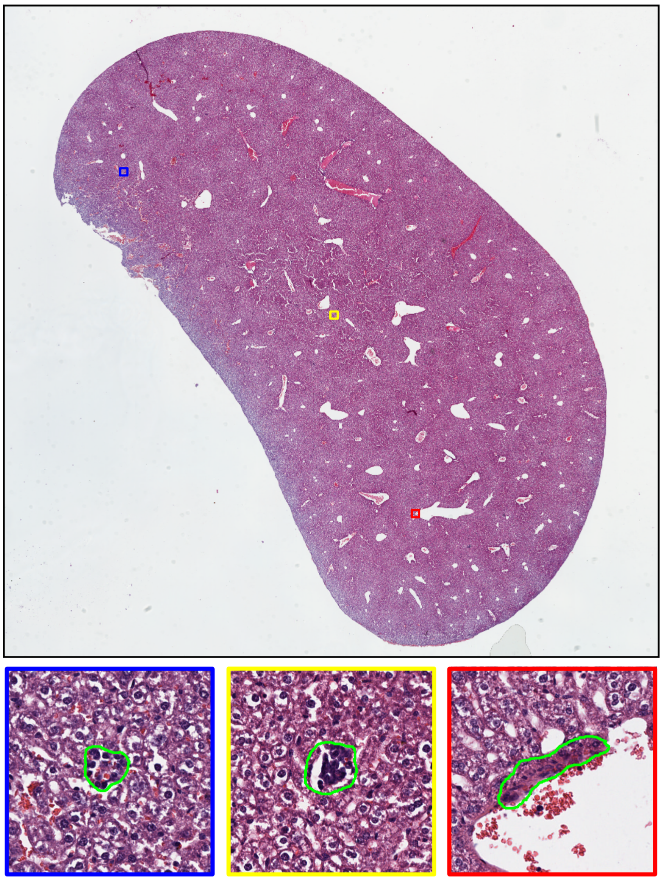


Figure 1: **Typical histopathological slide.** Three regions (blue, yellow and red squares) containing a cluster of hematopoietic cells are highlighted. Our method aims at retrieving such regions of interest within a large slide which mostly contains irrelevant background areas. Note the absence of large-scale context to guide the visual search, which would require an exhaustive screening of the slide in the case of a manual examination.

other common challenge in histopathological image analysis is the visual variability between two acquisitions. In particular, the consistency of the staining procedure is difficult to control experimentally leading to variations in terms of dye concentration (Fig. 2). Therefore, an algorithm that has been trained or designed on labeled data may not generalize well to newly acquired samples. To mitigate this source of inaccuracies, color normalization can be performed as a preprocessing step and remains an active field of research (Rabinovich et al., 2003; Macenko et al., 2009; Khan et al., 2014; Onder et al., 2014; Bautista and Yagi, 2015; Vahadane et al., 2015), together with generic techniques for online domain adaptation (Sec. 3.3). Finally, a tissue extracted surgically and observed under a microscope is less structured than other kinds of medical data such as body scans, while being of a much larger size. Objects of interest are expected to appear anywhere within the tissue, so that location or connectivity priors are rarely available.

In this work, we introduce an interactive method to assist a pathologist in exploring and quantifying large histological slides (Fig. 3). Our approach builds on a *pixelwise segmentation model* provided by a pre-trained random forest classifier (Sec. 4.1), and uses its output to perform an *interactive slide exploration* by suggesting, in a sequential manner, a series of candidate regions of interest (Sec. 4.2). This interactive navigation framework includes a component which allows the pathologist to provide, after each suggestion, some feedback about the actual relevance of the proposed region. From these user inputs, the underlying forest-based model is modified on-the-fly via a real-time online adaptation framework. This enables a progressive adjustment to the characteristics of the data at hand and compensates for possible mismatches with the original training set without specific assumptions about their nature, in contrast to the aforementioned explicit stain normalizations. Finally, we also demonstrate how a *whole-slide quantification* can be inferred after a partial exploration of the slide (Sec 4.3). The experimental evaluation of our approach was conducted in the context of extramedullary hematopoiesis quantification within mouse liver slides. The results demonstrate the ability of our method to quickly retrieve regions of interest and confirm the benefit of the interactive online adaptation scheme. The whole-slide quantification capabilities of our approach are also evaluated depending on the duration of the exploration stage. Finally, we demonstrate how one of our update strategies can be used with one-click inputs for faster interaction without decreasing its performance (Sec. 6).

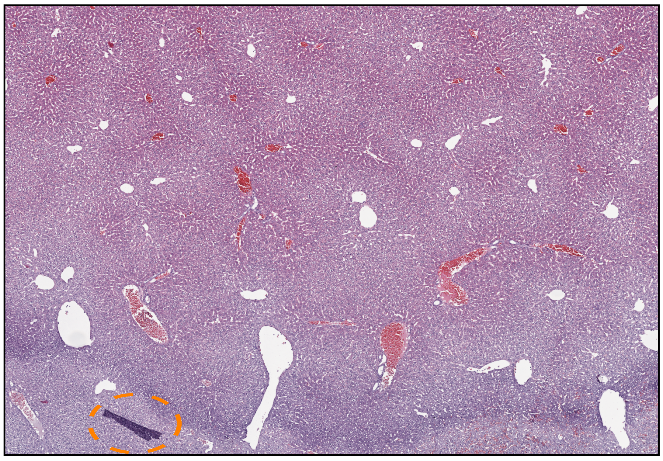


Figure 2: **Examples of visual artifacts.** A portion of a slide is displayed here. The staining itself is inhomogeneous and presents a vertical shading. Moreover, a dark artifact is present (circled in orange). Such a visual variability between and within slides complicates the application of supervised learning techniques and prompts an adaptation at prediction time.

2. Contributions

We propose an interactive framework using a forest-based pixelwise segmentation to explore large digital slides according to a predefined quantification task. In our application case, this clinical objective is the assessment of the surface covered by hematopoietic cells within mouse liver slides. Two main methodological contributions are introduced:

- The design of a region scoring function to convert pixelwise predictions into a score for each region of the slide.

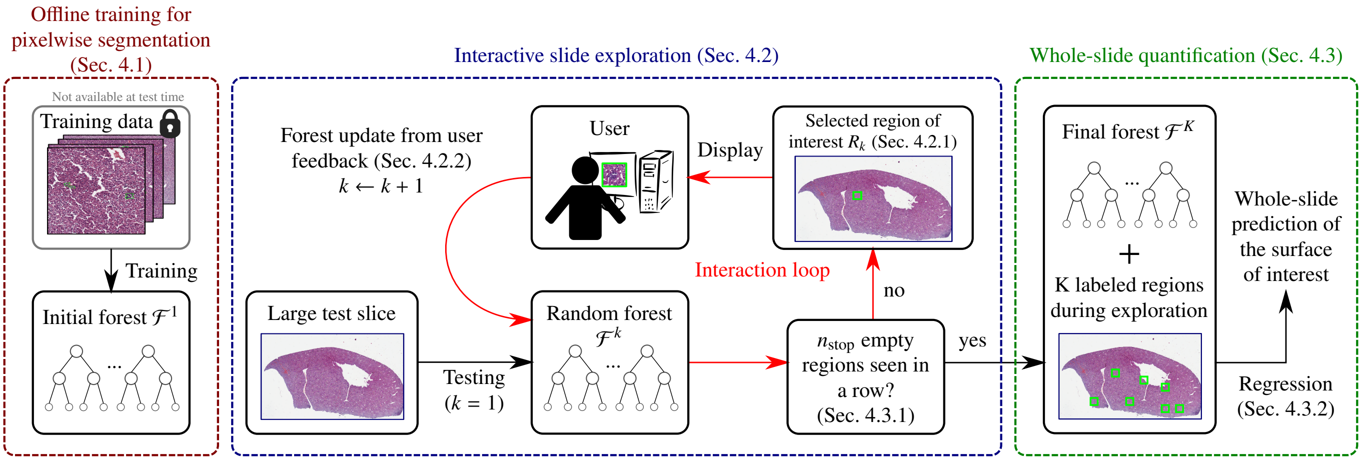

Offline training for pixelwise segmentation (Sec. 4.1)
Not available at test time
Training data
Training
Initial forest $\mathcal{F}^1$
Interactive slide exploration (Sec. 4.2)
Forest update from user feedback (Sec. 4.2.2)
$k \leftarrow k + 1$
User
Display
Selected region of interest R_k (Sec. 4.2.1)
Interaction loop
no
Large test slice
Testing
$(k = 1)$
Random forest $\mathcal{F}^k$
n_{stop} empty regions seen in a row? (Sec. 4.3.1)
yes
Whole-slide quantification (Sec. 4.3)
Final forest $\mathcal{F}^K$
+
K labeled regions during exploration
Whole-slide prediction of the surface of interest
Regression (Sec. 4.3.2)

Figure 3: **Overview of our scenario.** Initially, a classification forest $\mathcal{F}^1$ is trained offline at pixel level to segment the structures of interest (in our case, clusters of hematopietic cells). Given a large new slide to analyze, our method sequentially displays to the pathologist regions that are likely to contain these objects, thereby alleviating a tedious manual navigation through the slide. After the suggestion of a region R_k ($k \geq 1$), chosen as to maximize a forest-based scoring function $\phi(.|\mathcal{F}^k)$, the user provides a relevance feedback about its actual content. From this input, the current forest $\mathcal{F}^k$ is updated in real time leading to a new forest $\mathcal{F}^{k+1}$. By doing so, the visual specificities of the test slide are progressively incorporated into the decision model so that upcoming region suggestions can be reconsidered. The exploration is stopped after seeing a certain number n_{stop} of negative regions suggestions in a row. The updated forest and the user labels collected during the exploration can then be combined in a regression framework to predict the total surface covered by hematopietic cells in the tissue, including the areas not observed by the pathologist.

- An online domain adaptation scheme based on interactions with the user. Three real-time strategies are compared, including a novel approach based on online gradient descent which is compatible with lighter kinds of annotations.

By means of our experimental validation, we were able:

- To demonstrate the exploration abilities of our method and the benefit of the online adaptation.
- To show how, as a by-product of an only partial exploration of the slide, a whole-slide estimate of the surface covered by hematopietic cells can be predicted.
- To study experimentally the use of discretized user inputs during adaptation, and demonstrate how one-click inputs can be effectively used instead of accurate annotations.

The present manuscript builds on an earlier version presented at a conference (Peter et al., 2014) and includes the following extensions. Our original forest update scheme based on online gradient descent has been improved by incorporating theoretical results from the online learning literature. In addition, two alternative forest update methods are considered and studied in our experiments. The size of the dataset has been doubled and now consists of 70 fully labeled high-resolution images extracted from 16 different mouse slides. The cross-validation used in our experiments now includes the optimization of the update-related hyperparameter λ on a validation set. We also demonstrate how a whole-slide quantification can be performed based on an only partial exploration of the slide. Finally, we extended the validation regarding the discretization of user inputs. The impact of the amount of discretization is studied experimentally and we introduce and evaluate an alternative quantization procedure showing a more robust behavior.

3. Related Work

3.1. Learning-Based Image Segmentation in Histopathology

A large number of methods have been introduced for the automated analysis of histological slides and are progressively put into practice, as demonstrated by the development of general-purpose toolboxes such as Ilastik (Sommer et al., 2011), CellProfiler (Carpenter et al., 2006) and CellCognition (Held et al., 2010). We refer to surveys (Gurcan et al., 2009; Veta et al., 2014) for a broader overview of the field and focus here more specifically on the case of segmentation of histological images. In this context, learning-based techniques have been successfully applied for different tasks, including cell segmentation within follicular lymphoma images (Kong et al., 2011) and within lung and brain tumor samples (Su et al., 2015), segmentation of cancer tissue within colon images (Xu et al., 2014), and whole-slide segmentation of necrotic areas (Homeyer et al., 2013). The prediction of the Gleason grading, which is one of the most important quantitative measures for prostate cancer staging, has also gathered a particular interest in the field. Several statistical learning methods such as support vector machines (Nguyen et al., 2014), AdaBoost (Gorelick et al., 2013) and randomized forests (Khurd et al., 2010) were used towards an automatic prediction of this score. To overcome the difficulty of efficiently processing large whole-slide images, multi-resolution approaches were designed to find and segment regions of interest in a hierarchical manner (Sertel et al., 2009; Roullier et al., 2011; Huang et al., 2011; Doyle et al., 2012). These approaches simulate the behavior of a pathologist, starting from the lowest resolution and progressively refining the analysis towards presumably interesting areas. In this work, our segmentation model is a pixelwise forest classifier trained with Haar-like features (Sec. 4.1), whose efficiency allows us to operate directly at the highest resolution instead.

3.2. Assisted Navigation within Large Digital Slides

While the aforementioned methods focus on the segmentation task, a few other approaches aim at identifying regions of interest within histological data. A method for classifying regions as relevant or irrelevant using support vector machines was introduced (Bahlmann et al., 2012), and extended to a scenario where the ground truth is generated by analyzing the actual behavior of a pathologist with viewport tracking data (Mercan et al., 2014). These classification techniques are closer to our goal but differ methodologically in two aspects. First, the methods above model the region retrieval task as a classification problem, ignoring the differences between positive regions. In particular, one may desire to display in priority regions containing larger structures of interest. Building our region scoring scheme on an underlying segmentation model naturally provides such a ranking of regions and gives the opportunity to extrapolate the quantification estimate to unobserved areas. Secondly, our method is more flexible, as it includes the ability to update the region selection rule from user annotations collected after each suggestion, in an online domain adaptation fashion.

3.3. Online Domain Adaptation

Experimental constraints during the preparation of a tissue may induce inconsistencies in terms of visual aspect between acquisitions. In particular, a newly acquired sample may differ from the data used to train the initial classifier. The problem of domain adaptation consists in the correction of such a shift between the distributions of the training and testing data. Most domain adaptation strategies retrain a new classifier once samples from the target domain have been observed. In our case, new samples are collected every time a suggested region is labeled by the user, after which the current classifier is accordingly adapted. To keep this interaction loop tractable in practice, the updates must be performed in real time, which excludes a retraining of the classifier between two suggestions. Because of this constraint, our scenario is more precisely an *online domain adaptation* task. This relatively recent paradigm has been addressed in a few works only. A generic unsupervised method based on Gaussian process regression was introduced to adapt the decision boundary of any black-box classifier to a target image (Jain and Learned-Miller, 2011). Other approaches combined a classifier trained on the source data with an online classifier continuously updated from the target data (Zhao and Hoi, 2010; Tommasi et al., 2012). Originally designed with kernel-based classifiers, transferring this technique to forest models poses some challenges. First, since it treats the initial classifier as a whole, applying this procedure to a forest would result in entire trees being discarded if they do not suit the testing distribution anymore. Thereby, one would ignore the fact that some areas of the feature space, i.e. some tree leaves, may remain valid. Moreover, it requires a strategy to build decision trees online which is not straightforward. One of the most popular strategies for online forest training is to grow trees progressively, starting from a root node, by turning a leaf node into an internal node as soon as a split of sufficient quality can be found, both in terms of information gain and statistical representativity (Saffari et al., 2009). Moreover, this approach includes, in the context of tracking, the idea of online domain adaptation by discarding trees when they no longer match the distribution of the arriving samples. Another recent work models trees as samples from Mondrian distributions (Lakshminarayanan et al., 2014) and improves over existing online approaches, yet at the cost of losing the compatibility of forests with high-dimensional feature spaces. In our case, the large size of histological data and the fact that online forest updates take place between two human interactions impose strict computational constraints to ensure the practical applicability of our method. In particular, modifications of the structure of the trees, such as in the two aforementioned approaches, are compromised. By acting on the leaf probabilities only, we achieve real-time updates between two interactions (Sec. 4.2.2).

3.4. Active Learning

Finally, querying user labels in order to improve a classifier can be seen as a form of active learning (Settles, 2010), which inspired a few approaches in the context of histopathology (Homeyer et al., 2011). In general, active learning algorithms query the label of the most uncertain samples given the knowledge of the current classifier, in order to minimize the labeling effort from the user. The spirit of our approach is different: from a clinical perspective, a pathologist is only interested in seeing positive examples in a short amount of time. This asymmetry between positive and negative observations leads us to focus on finding and displaying positive regions as quickly as possible, so that they can be visually inspected and validated by the user. The annotations obtained during the process are used to assess the accuracy of the initial model and correct it if necessary. Moreover, by doing so, any erroneous region suggestion naturally provides a challenging negative example to include in the online adaptation process.

4. Methods

This section exposes our methodology. Our slide examination method is based on an initial forest-based model whose goal is to segment the objects of interest within the tissue. This initial forest, denoted $\mathcal{F}^1$, is trained offline on some labeled examples and encodes the available prior knowledge before observing the test data. After training, the original training data are no longer considered available. This assumption is driven by practical aspects: while sharing and transferring a classifier from a machine to another is straightforward, this is usually less feasible with patient data which are of larger size and subject to ethical considerations. The training mechanism generating $\mathcal{F}^1$ from labeled images is conducted in a standard way (Sec. 4.1).

In addition to its use as a navigation tool (Sec. 4.2), we also discuss whole-slide quantification abilities for our method. Indeed, after the exploration phase has been completed, it can be of interest to estimate the total amount of cells in the slide, including in the areas that have not been seen by the pathologist. After a discussion on a relevant definition of a stopping criterion for the exploration stage in Sec. 4.3.1, we expose our regression-based strategy for this task in Sec. 4.3.2.

4.1. Forest-Based Segmentation Model

Our slide analysis method builds on a segmentation model assessing the surface covered by a certain type of anatomical structure within the acquired tissue. In this work, we follow a pixelwise classification approach by modeling the segmentation process as a series of independent decisions for each pixel $\mathbf{x}$ in an image. Each decision is conducted by a random forest classifier $\mathcal{F}$ which outputs a conditional probability $\mathbb{P}(\mathbf{x} \in \mathcal{P}|\mathcal{F})$ that the true label $y(\mathbf{x})$ is 1 given the classifier $\mathcal{F}$, with $\mathcal{P} = \{\mathbf{x} \in \mathcal{X} | y(\mathbf{x}) = 1\}$ denoting the set of positive samples. $\mathcal{X}$ denotes the theoretical set of observable samples.

The visual aspect around each pixel has to be quantitatively modeled by a set of features on which the node splitting functions are based. In this work, we use Haar-like features which describe each pixel by its visual content at offset locations. The precomputation of an integral image for each color channel allows a fast access to any of these feature values, so that this large set of descriptors can be efficiently handled at a low memory footprint. These generic and computationally efficient features were originally used in combination with a boosting classifier, first in the context of face detection (Viola and Jones, 2004) and later extended to object recognition and segmentation (Shotton et al., 2006). Their use within the random forest framework (Criminisi et al., 2009) has been successfully applied to a great variety of tasks and imaging modalities (Pauly et al. (2011); Montillo et al. (2011); Chatelain et al. (2013); Gauriau et al. (2014); Ebner et al. (2014); Kontschieder et al. (2014); Zikic et al. (2014)).

The forest training was conducted using axis-aligned splitting functions and the Gini index (Breiman et al., 1984) as node purity measure. More precisely, n_{tries} candidate Haar-like features are randomly drawn at each node and $n_{\text{thresholds}}$ thresholds are tried for each of them. The candidate features are drawn in a fine-to-coarse fashion instead of the standard uniform sampling over a patch, so that an appropriate visual scale can be automatically inferred at each node (Peter et al., 2015). Among all tried splits, the best one is retained and the procedure recursively repeated until purity, or until none of the n_{tries} candidate splits send at least $n_{\text{samples/leaf}}$ training samples to the two child nodes. As soon as one of these stopping criteria is satisfied, a leaf L is created. A probability π_L is computed from the class histogram of training samples reaching L and stored at this leaf.

4.2. Interactive Slide Exploration

Considering a new test slide, we partition it into a predefined set $\mathcal{R}$ of non-overlapping regions of fixed size $\delta \times \delta$. The first step of our algorithm consists in retrieving the region $R_1 \in \mathcal{R}$ of highest interest to the user. Since $\mathcal{F}^1$ provides a pixelwise estimate, this choice of region is made according to a region scoring function $\phi(R|\mathcal{F}^1)$, which predicts the expected interest of a region R given the knowledge carried by the forest model $\mathcal{F}^1$. The first region displayed to the pathologist is $R_1 = \text{argmax}_{R \in \mathcal{R}} \, \phi(R|\mathcal{F}^1)$. Once R_1 has been shown, the pathologist reports the actual relevance of its content. To do so, two possibilities of user labelings are considered in this work: either a full delineation of the object of interest in R_1, which is accurate but time-consuming, or a one-click input obtained by discretization, which is faster to provide but more ambiguous (Sec. 6). Using the input of the pathologist on R_1, the forest $\mathcal{F}^1$ is accordingly modified, leading to a new forest $\mathcal{F}^2$. This procedure is repeated several times by showing, at each iteration, the region $R_k = \text{argmax}_{R \in \mathcal{R} \setminus \{R_1, \ldots, R_{k-1}\}} \, \phi(R|\mathcal{F}^k)$. In Sec. 4.2.1, we describe in more details our choice of scoring function ϕ. Section 4.2.2 is dedicated to the techniques for online domain adaptation, where three real-time alternatives are described including a novel approach based on online gradient descent.

4.2.1. Region Scoring Function

After training on a set of labeled images, we obtain a random forest classifier $\mathcal{F}^1$ which outputs, for every pixel $\mathbf{x}$ in an image, the probabilistic estimate $\mathbb{P}(\mathbf{x} \in \mathcal{P}|\mathcal{F}^1) \in [0, 1]$ that $\mathbf{x}$ is a positive instance, i.e. belongs to one of the sought structures. Since the goal of our approach is to display regions of interest to a pathologist, we use these pixelwise forest predictions to build a region scoring function ϕ. We propose to define the score of a region R as

$$\phi(R|\mathcal{F}) = \sum_{\mathbf{x} \in R} \mathbb{P}(\mathbf{x} \in \mathcal{P}|\mathcal{F}) \tag{1}$$

given a pixelwise classification forest $\mathcal{F}$. This scoring function can be interpreted as the mathematical expectation of a random variable counting the number of positive pixels in the region R. In particular, regions containing larger objects obtain a higher score. Since the quantification task consists in the estimation of the total surface covered by the structures of interest within the slide, this amounts to showing first regions which have a greater contribution to this quantity. Due to its simplicity, our scoring function possesses important properties in the context of forest updates (computational efficiency and convexity) which will be detailed in Sec. 4.2.2. These advantages result from the fact that $\phi(R|\mathcal{F})$ can be rewritten as a scalar product between the vector of leaf models of the forest $\mathcal{F}$ and a sparse vector characterizing the region R. The derivation of this equivalent formulation is exposed in the next paragraph. Finally, although the leaf models are probabilistic estimates of a classification task in this work, a similar scoring function could be equivalently used for a regression output. For instance, if the forest predicts a density of objects (Fiaschi et al., 2012), this scoring function would count a number of objects in the region instead of a surface.

Expressing $\phi(R|\mathcal{F})$ as a scalar product. Let us first introduce some notations. The set of leaf nodes belonging to the t^{th} tree is denoted $\mathcal{L}_t$. $\mathcal{L} = \cup_{1 \le t \le n_{\text{trees}}} \mathcal{L}_t$ is the set of all leaf nodes contained in the forest $\mathcal{F}$, and we denote $\text{tree}(L) \in \{1, \ldots, n_{\text{trees}}\}$ the index of the tree to which a leaf $L \in \mathcal{L}$ belongs. We arbitrarily order the finite set $\mathcal{L}$ and consider the leaf probabilities jointly as a (finite-dimensional) vector $\boldsymbol{\pi} = (\pi_L)_{L \in \mathcal{L}}$. We denote $\Sigma = (\sigma_t)_{1 \le t \le n_{\text{trees}}}$ the list of routing functions $\sigma_t : \mathcal{X} \to \mathcal{L}_t$, which assign to each sample $\mathbf{x} \in \mathcal{X}$ the leaf $\sigma_t(\mathbf{x})$ that it reaches when passed through the t^{th} tree (Fig. 4). Intuitively, Σ encodes

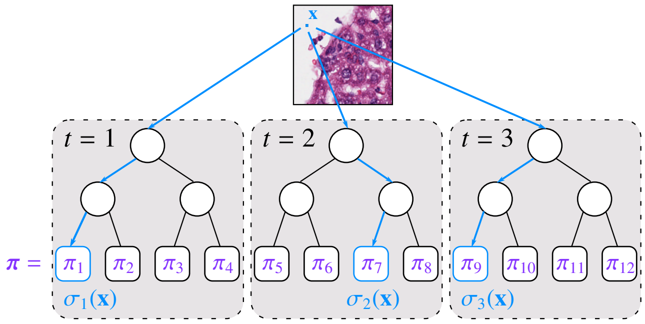


Figure 4: **Notations.** This figure illustrates our notations in the simplified case of a small forest (3 trees of depth 2). The leaves are arbitrarily ordered and the leaf models are jointly considered as a vector $\boldsymbol{\pi}$. For each pixel $\mathbf{x}$, $\sigma_t(\mathbf{x})$ denotes the leaf reached by $\mathbf{x}$ when passed through the t^{th} tree. The list of functions $\Sigma = (\sigma_t)_{1\leq t\leq n_{\text{trees}}}$ encompasses information about the arrangement of the trees and the node splitting functions.

the structure of the forest determined by the arrangement of the nodes and the splitting functions, i.e. the way the forest partitions the space of observations $\mathcal{X}$, while the vector $\boldsymbol{\pi}$ defines the label predictions stored in the terminal nodes. Σ and $\boldsymbol{\pi}$ fully determine the forest decision rule, defined as

$$\mathbb{P}(\mathbf{x} \in \mathcal{P}|\mathcal{F}) = \frac{1}{n_{\text{trees}}} \sum_{t=1}^{n_{\text{trees}}} \pi_{\sigma_t(\mathbf{x})}. \tag{2}$$

By incorporating Eq. 2 into the definition of the scoring function (Eq. 1) and rearranging the sum signs (see Appendix A for details), we obtain the identity

$$\phi(R|\mathcal{F}) = \phi(R|\Sigma, \boldsymbol{\pi}) = \langle \boldsymbol{\rho}(R|\Sigma), \boldsymbol{\pi} \rangle\,, \tag{3}$$

where $\boldsymbol{\rho}(R|\Sigma) = (\rho_L(R|\Sigma))_{L\in\mathcal{L}}$ is a vector of dimension n_{leaves} characterizing the region R and defined as

$$\rho_L(R|\Sigma) = \frac{1}{n_{\text{trees}}} \underbrace{\#\{\mathbf{x} \in R \mid \sigma_{\text{tree}(L)}(\mathbf{x}) = L\}}_{\text{number of pixels in } R \text{ falling in the leaf } L}\,. \tag{4}$$

Hence, the scoring function of a region R appears as a scalar product between the vector of leaf models $\boldsymbol{\pi}$ and a vector $\boldsymbol{\rho}(R|\Sigma)$, which only depends on how the samples from the region R are sent to the leaves (Fig. 5). Moreover, since every pixel $\mathbf{x}$ of a region R falls in exactly n_{trees} leaves, each vector $\boldsymbol{\rho}(R|\Sigma)$ is sparse (or of small size) with at most $n_{\text{trees}}\,|R|$ non-zero elements.

4.2.2. Interactive Forest Adaptation

In Sec. 4.2.1, we described how to score regions of a large histological slide so that they can be ranked and displayed in decreasing order of interest to a pathologist. This ranking is based on the output of a pixelwise classification forest learned on labeled data. If the data at hand differs from the training images, for instance because of variations in terms of dye concentration or because of the presence of artifacts, this initial forest model can be prone to errors (Fig. 6). However, the fact that regions of interest are shown sequentially to the human expert offers the opportunity to let the user report the actual validity of the suggestions and, thereby, to recalibrate the forest model to take into account the characteristics of the slide to analyze. This scenario corresponds to an online domain adaptation problem, for which we consider three different strategies. The first two require accurate delineations of the objects of interest by the user, whereas the third approach only requires a weaker form of labeling stating the actual surface covered by such objects within a suggested region. By discretizing this quantity, faster user interactions can be performed (see Sec. 6).

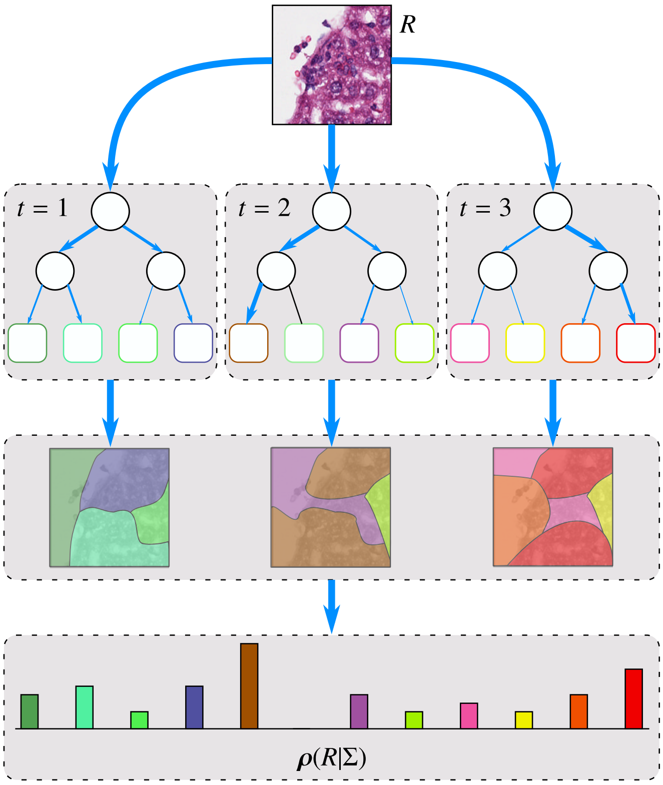


Figure 5: **Characteristic vector of a region R.** Applying a forest on all the pixels of a region R leads to n_{trees} different partitions of R, defined by the leaves reached by the sent pixels. By counting and concatenating the leaf occurrences, one obtains a characteristic vector $\boldsymbol{\rho}(R|\Sigma)$ of the region R which only depends on the structure Σ of the forest (Eq. 4). Consequently, the score of any region R can be written as $\phi(R|\mathcal{F}) = \langle\boldsymbol{\rho}(R|\Sigma), \boldsymbol{\pi}\rangle$.

As exposed at the beginning of Sec. 4, the adaptation procedure generates, starting from a forest $\mathcal{F}^1$, a series of forests $\mathcal{F}^2, \mathcal{F}^3, \ldots$ where each forest $\mathcal{F}^{k+1}$ is created after k regions have been observed by the pathologist and the k corresponding inputs collected. At each iteration k, the region R_k is chosen as the one maximizing the scoring function $\phi(.|\mathcal{F}^k)$ over the set of remaining regions. The three alternative strategies described below are based on an assumption of fixed structure for all the forests $\mathcal{F}^k$, so that only the leaf probabilities are modified. This assumption offers the following computational advantage. For all $k \geq 1$, the structure Σ^k of the forest $\mathcal{F}^k$ is equal to the structure Σ^1 of the initial forest $\mathcal{F}^1$. In particular, the vectors $\boldsymbol{\rho}(R|\Sigma^k), R \in \mathcal{R}$ are now kept unchanged during the whole exploration process, so that they can be precomputed once for all at the first iteration and compactly stored in memory due to their sparsity. For simplicity, we omit their dependency in Σ^1 and denote these vectors $\boldsymbol{\rho}(R), R \in \mathcal{R}$. The score

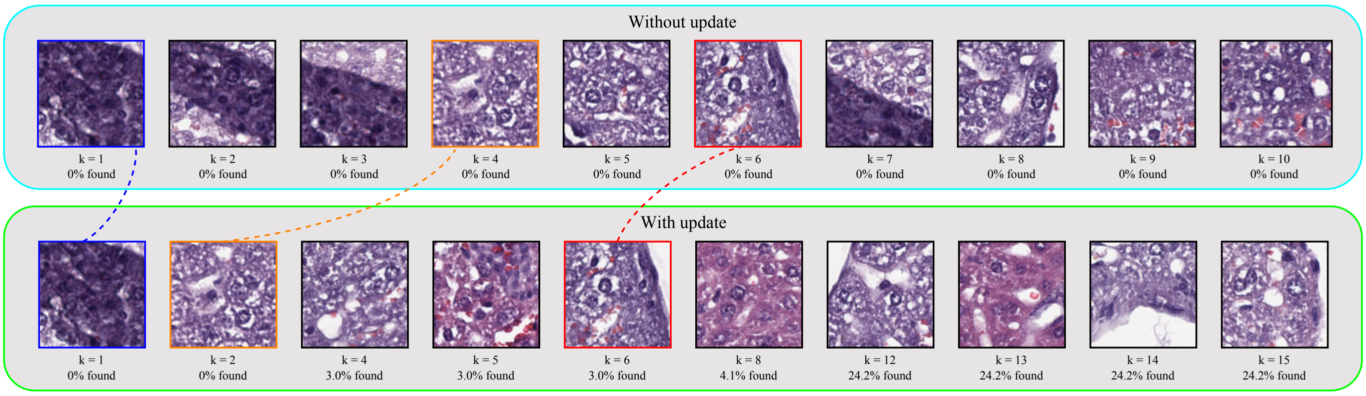


Figure 6: **Benefit of the interactive adaptation illustrated by the negative region suggestions.** In challenging cases (here the slide containing the portion shown in Fig. 2), a direct application of the pre-trained forest model leads to difficulties in the exploration. The first 10 regions that are suggested are in fact negative (first row). These confusions are due to the variation of staining and the presence of an artifact shown in Fig. 2, which confirms the necessity of adapting the initial classifier. When interactive updates are performed, each negative suggestion is signaled by the user and the underlying model is accordingly calibrated. The benefit of this adaptation scheme can be illustrated by looking at the first 10 negative suggestions (second row, using the ALM update). First, we can see that positive suggestions occur earlier in the exploration: the 3rd, 7th, 9th, 10th and 11th suggestions were positive and allowed the pathologist to localize already 24.2% of the hematopoietic patterns present in the slide. This demonstrates that adapting the classifier clearly improves the quality of the exploration. Moreover, we can see that the adaptive approach 'learns from its mistakes' through the greater diversity of its negative suggestions: regions with different shades of stain are proposed and multiple suggestions within the dark artifacts are avoided. Among their 10 first negatives, only 3 suggestions are shared by the two approaches (emphasized with colors), which further illustrates their difference.

$\phi(R|\mathcal{F}^k) = \left\langle \boldsymbol{\rho}(R|\Sigma^k), \boldsymbol{\pi}^k \right\rangle$ of a region R described in Eq. 3 can be rewritten $\phi(R|\mathcal{F}^k) = \left\langle \boldsymbol{\rho}(R), \boldsymbol{\pi}^k \right\rangle$. Hence, to obtain the updated scoring functions $\phi(R|\mathcal{F}^k)$ for a new forest $\mathcal{F}^k$, one only needs to recompute the sparse scalar products $\left\langle \boldsymbol{\rho}(R), \boldsymbol{\pi}^k \right\rangle$ with the new leaf models $\boldsymbol{\pi}^k$. The efficiency of this operation yields real-time updates between two region suggestions, and the histological slide has to be passed only once through a forest (the initial $\mathcal{F}^1$) as a preliminary step before starting the exploration.

We expose now our three alternative real-time update strategies. They are all equivalent from a computational point of view, with a worst-case complexity of $O(n_{\text{trees}}\,|R|)$, and depend on one hyperparameter $\lambda > 0$ weighting the importance of the prior knowledge in comparison to the newly observed samples.

Update of Leaf Statistics (ULS) If the input provided by the pathologist is a full object delineation in the displayed region R_k, the pixels $\mathbf{x} \in R_k$ can be seen as new training samples whose label is known. Therefore, the leaf statistics can be updated (Criminisi et al., 2012). We denote $N_L^{1,+}$ (resp. $N_L^{1,-}$) the number of positive (resp. negative) samples which arrived in the leaf L during the training of the initial forest $\mathcal{F}^1$, leading to the leaf models $\pi_L^1 = \frac{N_L^{1,+}}{N_L^{1,-}+N_L^{1,+}}$. We also denote $N_L^{k+1,+}$ and $N_L^{k+1,-}$ the total number of positive and negative samples collected in the regions $R_1, \ldots, R_k$. Given these quantities, the ULS strategy updates the probability of each leaf L after labeling the region R_k as

$$\pi_L^{k+1} = \frac{N_L^{k+1,+} + \lambda N_L^{1,+}}{N_L^{k+1,+} + N_L^{k+1,-} + \lambda \left(N_L^{1,+} + N_L^{1,-} \right)}. \tag{5}$$

Average of the Leaf Models (ALM) In the same conditions than the update described above, we propose an alternative leaf update consisting in computing a separate probability π_L^{new} based on the pixels observed in $R_1, \ldots, R_k$ only (i.e. originating from the test slide) and averaging it with the initial probability π_L^1. By doing so, the choice of λ is made independent of the initial number of samples in the leaf. This update can be written in vectorial form as

$$\boldsymbol{\pi}^{k+1} = \frac{1}{1+\lambda} \left(\boldsymbol{\pi}^{\text{new}} + \lambda \boldsymbol{\pi}^1 \right) \tag{6}$$

where, for each leaf L, $\pi_L^{\text{new}} = \frac{N_L^{k+1,+}}{N_L^{k+1,+}+N_L^{k+1,-}}$ if some new samples have been observed in the leaf L (i.e. if $N_L^{k+1,+} + N_L^{k+1,-} > 0$). Otherwise, we define $\pi_L^{\text{new}} = \pi_L^1$.

Online Gradient Descent (OGD) The two previous updates require a pixelwise labeling provided by the user. Instead, this last update method uses the (weaker) information $Q(R_k)$ stating the amount of positive pixels located in the region R_k. This quantity is, in fact, what the score $\phi(R_k|\mathcal{F}^k) = \left\langle \boldsymbol{\rho}(R_k), \boldsymbol{\pi}^k \right\rangle$ used to assess the relevance of the region R_k estimates (Sec. 4.2.1). We propose to measure the discrepancy between the true value $Q(R_k)$ revealed by the user and the prediction from the set of leaf models $\boldsymbol{\pi}$ with the squared loss $l_k(\boldsymbol{\pi}) = (\langle \boldsymbol{\rho}(R_k), \boldsymbol{\pi} \rangle - Q(R_k))^2$. Hence, at iteration k, the incurred loss is $l_k(\boldsymbol{\pi}^k)$. The convexity of the loss function l_k, which directly results from the linear rewriting of our scoring function (Eq. 3), allows us to see the update problem as an online convex optimization scenario (Shalev-Shwartz, 2012). We solve this problem via an online gradient descent strategy (Zinkevich, 2003), which leads to the update rule

$$\begin{aligned} \boldsymbol{\pi}^{k+1} &= \Pi_{[0,1]^{|\mathcal{L}|}} \left[\boldsymbol{\pi}^k - \eta \vec{\nabla} l_k(\boldsymbol{\pi}^k) \right] && (7) \\ &= \Pi_{[0,1]^{|\mathcal{L}|}} \left[\boldsymbol{\pi}^k - 2\eta \left(\left\langle \boldsymbol{\rho}(R_k), \boldsymbol{\pi}^k \right\rangle - Q(R_k) \right) \boldsymbol{\rho}(R_k) \right], && (8) \end{aligned}$$

where η is a learning rate. $\Pi_{[0,1]^{|\mathcal{L}|}} : \mathbb{R}^{|\mathcal{L}|} \to [0,1]^{|\mathcal{L}|}$ is the projection operator on $[0,1]^{|\mathcal{L}|}$ which projects each individual component π_l of a vector $\boldsymbol{\pi} \in \mathbb{R}^{|\mathcal{L}|}$ onto the set $[0,1]$, ensuring that the leaf probabilities stay in $[0,1]$ after each update. By transferring generic considerations on online gradient descent to our forest-based scenario (see Appendix B), we choose a learning rate η of the form

$$\eta = \frac{1}{2\lambda\delta^4}\sqrt{\frac{n_{\text{trees}}\,|\mathcal{L}|}{|\mathcal{R}|}}, \tag{9}$$

where λ is a positive hyperparameter, $|\mathcal{L}|$ (resp. n_{trees}) is the number of leaves (resp. trees) in the forest, $|\mathcal{R}|$ is the number of regions in the slide and $\delta \times \delta$ is the predefined region size.

The difference between ALM and ULS can be seen by considering their respective asymptotic behavior when the number of new samples increases. In Eq. 5, we have $\boldsymbol{\pi}^k \sim \boldsymbol{\pi}^{\text{new}}$, whereas, in Eq. 6, $\boldsymbol{\pi}^k$ always includes a fixed contribution from $\boldsymbol{\pi}^1$ regardless of the number of new samples which have been collected. These two variants correspond to the simplest way to update an existing tree (Criminisi et al., 2012), where we introduce a parameter weighting old and new training data. More sophisticated but computationally costly strategies would involve further splitting or the replacement of old trees by new ones (Saffari et al., 2009). In our case, since the original training data used to train $\mathcal{F}^1$ are no longer available at testing time, we keep the structure of the old trees which represent the only available prior knowledge about the quantification task. This emphasizes the scope of our adaptation procedure, which should be seen as adjusting a known supervised segmentation task (e.g. within a same clinical study) to the variations of visual appearances that may occur experimentally. However, a different task cannot be accommodated a priori and would first require the training of a new segmentation model. Finally, due to the nature of the required input, OGD can be used for a lighter kind of user interaction (see Sec. 6), which is not supported by existing forest online learning algorithms.

4.3. From Partial Exploration to Whole-Slide Quantification

4.3.1. Stopping the Exploration Stage

In practice, the exploration process is meant to be interrupted before seeing the whole slide. Since the amount of hematopoietic cell clusters is variable from a slide to another, some slides intrinsically require more time from the pathologist than others. Therefore, fixing in advance the number of iterations for the exploration would be inappropriate. Instead, we propose a stopping criterion based on the density of positive suggestions, and interrupt the exploration as soon as n_{stop} negative regions were suggested in a row, i.e. when most positive regions were presumably seen. Once the stopping criterion is reached, a whole-slide prediction can be made (Sec. 4.3.2).

The chosen value for n_{stop} is directly depending on the amount of time that the pathologist is ready to spend for the analysis. Strictly reasoning in terms of accuracy, it is always preferable to let the pathologist see a maximum number of regions. Defining a recommended value for n_{stop} is hence subjective and result from a tradeoff between accuracy and human effort. Our experiments regarding the whole-slide quantification were conducted for several values of n_{stop}, encoding different amounts of effort that the pathologist is ready to invest.

4.3.2. Whole-Slide Quantification via Regression

Once the stopping criterion has been reached, we predict an estimate $\hat{q}$ of the surface covered by hematopoietic cells within the whole slide with linear regression. Denoting K the total number of regions that have been seen during the exploration stage, a partial knowledge on $\hat{q}$ is available via the quantity $q^{\text{labeled}} = \sum_{k=1}^{K} Q(R_k)$ obtained as the user annotated the regions $R_1, \ldots, R_K$ during the exploration. In addition, the updated forest model $\mathcal{F}^{K+1}$ obtained at the end of the exploration phase provides a prediction $\phi(R|\mathcal{F}^{K+1})$ of the quantity of positive pixels in each region $R \in \mathcal{R}$ of the slide. In particular, this gives a total prediction $\Phi^{\text{total}} = \sum_{R\in\mathcal{R}} \phi(R|\mathcal{F}^{K+1})$ and a prediction on the labeled regions $\Phi^{\text{labeled}} = \sum_{k=1}^{K} \phi(R_k|\mathcal{F}^{K+1})$. We formalize our regression problem by considering that the relative change between the total quantity $\hat{q}$ and the partial quantity q^{labeled} is proportional to the relative change between total prediction and the partial prediction, i.e.

$$\frac{\hat{q} - q^{\text{labeled}}}{q^{\text{labeled}}} \propto \frac{\Phi^{\text{total}} - \Phi^{\text{labeled}}}{\Phi^{\text{labeled}}}. \tag{10}$$

This corresponds to a prediction rule of the form

$$\hat{q} = q^{\text{labeled}} + a\frac{\Phi^{\text{total}} - \Phi^{\text{labeled}}}{\Phi^{\text{labeled}}}q^{\text{labeled}}. \tag{11}$$

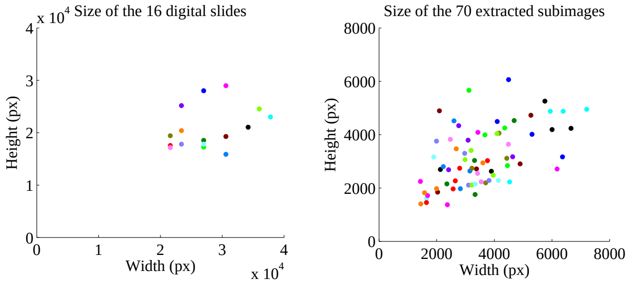
The regression parameter $a \in \mathbb{R}$ is learned on a validation set.

5. Experiments

5.1. Dataset and Medical Motivation

The presence of hematopoietic cells outside the bone marrow, also called extramedullary hematopoiesis, is a marker of an extensive stimulation of the immune system (Tao et al., 2008). There is accumulating evidence that the amount of infiltrating immune cells such as cytotoxic CD8-positive T-lymphocytes into the tumor can be considered as a tumor biomarker for measuring clinical outcome (Balermpas et al., 2016). We evaluated our approach in this clinical context on a dataset addressing the aspect of lymphocytic infiltration into mouse liver tissues, for which the amount of these cells within histological samples must be estimated.

Slides from 16 mice were digitally acquired at the resolution 0.5 µm per pixel and downsized by 2 to speed up the training and testing steps. 70 large representative subimages were extracted from these slides and fully segmented, covering approximately 20% of the total tissue (Fig. 7). Resorting to a set of subimages follows the clinical practice and was necessary to obtain accurate labels for a sufficient number of different slides. This is particularly important in our study which



(a) (b)

Figure 7: **Dataset dimensions.** Our dataset consists of 16 digital slides, an example of which is provided in Fig. 1. Figure 7a shows their dimensions, where each colored point corresponds to one slide. Instead of working on the slide, we extracted and labeled entirely a total of 70 subregions such as the one shown in Fig. 2. The dimensions of these subregions are reported in Fig. 7b. The color code of Fig. 7a is respected so that the slide from which each subimage is extracted can be identified by its color. In total, the extracted subimages cover around 20% of the acquired tissues.

focuses on issues arising from the visual variability between acquisitions. Yet, it comes at the cost of possibly introducing a few natural biases such as underrepresentation of border areas or of straightforwardly negative objects (e.g. large white parts).

5.2. Experimental Settings

We randomly split our dataset $\mathcal{D}$ into 4 sets $\mathcal{D}_i$ of 4 slides each and we performed a 4-fold nested cross-validation. The goal of this procedure is to optimize the update-related hyperparameter λ independently of the test data to avoid overfitting. Let Λ be a set of candidate values for λ and $l^{\mathrm{cv}}(\lambda, \mathcal{F}^1, I)$ a loss measuring the error of our method on the slide I when using the hyperparameter λ and an initial forest $\mathcal{F}^1$. The nested cross-validation consists of 4 runs, each run corresponding to a set $\mathcal{D}_{i_{\mathrm{out}}}$ (with $i_{\mathrm{out}} \in \{1, 2, 3, 4\}$) left out for testing. For each run, a second cross-validation (called inner cross-validation) is performed over the 3 remaining sets $(\mathcal{D}_i)_{i \neq i_{\mathrm{out}}}$, where 2 sets are used to train the forest and the remaining one is used as a validation set. At the end of the inner cross-validation, i.e. when 3 forests have been trained and each of the 3 sets has been used as a validation set, we define the optimal hyperparameter $\lambda_{i_{\mathrm{out}}}$ of this run as the one minimizing the total loss over the 3 bags, i.e.

$$\lambda_{i_{\mathrm{out}}} = \operatorname*{argmin}_{\lambda \in \Lambda} \sum_{i \neq i_{\mathrm{out}}} \sum_{I \in \mathcal{D}_i} l^{\mathrm{cv}}(\lambda, \mathcal{F}_{\mathcal{D} \setminus (\mathcal{D}_i \cup \mathcal{D}_{i_{\mathrm{out}}})}, I). \qquad (12)$$

$\mathcal{F}_{\mathcal{D} \setminus (\mathcal{D}_i \cup \mathcal{D}_{i_{\mathrm{out}}})}$ denotes the forest obtained by training on the two remaining sets after excluding $\mathcal{D}_i$ and $\mathcal{D}_{i_{\mathrm{out}}}$. Using the hyperparameter value $\lambda_{i_{\mathrm{out}}}$, we then report independently the prediction of each of the 3 forests $\left(\mathcal{F}_{\mathcal{D} \setminus (\mathcal{D}_i \cup \mathcal{D}_{i_{\mathrm{out}}})}\right)_{i \neq i_{\mathrm{out}}}$ on the left-out set $\mathcal{D}_{i_{\mathrm{out}}}$. This procedure allows us to learn automatically the hyperparameter independently of the testing set, and outputs 3 different predictions for each test slide which gives an idea of their dependency on the original training data. This results in a total of 48 predictions. Note that, to conduct the entire nested cross-validation, only 6 forests $\left(\mathcal{F}_{\mathcal{D}_i \cup \mathcal{D}_j}\right)_{1 \leq i < j \leq 4}$ have to be trained. The user interaction was automatically simulated from the ground truth delineations. Regions were chosen of size $\delta \times \delta$ with $\delta = 60$ µm. Every time a region is displayed, the user can easily extend the field of view around it if necessary. We simulated this behavior automatically by showing the neighboring positive region(s) in the case where an object of interest is not fully included in the displayed region.

The forests were initially trained on labeled pixels which were densely collected every 8 µm in the two directions. Parallelized on 10 threads, this training step took between 3 and 6 hours depending on the cross-validation run (with a corresponding number of training samples comprised between 5×10^5 and 10^6). Given an incoming slide, testing was performed on all pixels, which is tractable since it has to be done only once at the beginning of the process (see Sec. 4.2.2). This preliminary step took around 1 minute, after which the interaction loop could take place in real-time conditions. More precisely, the update of forest leaf models and recomputation of box scores between two iterations took between 10 and 100 ms without any parallelization. The visual features were computed on the Lab color space. The following forest parameters were used: $n_{\mathrm{samples/leaf}} = 10$, $n_{\mathrm{trees}} = 30$, $n_{\mathrm{tries}} = 500$, $n_{\mathrm{thresholds}} = 10$, and the bagging rate was 0.5.

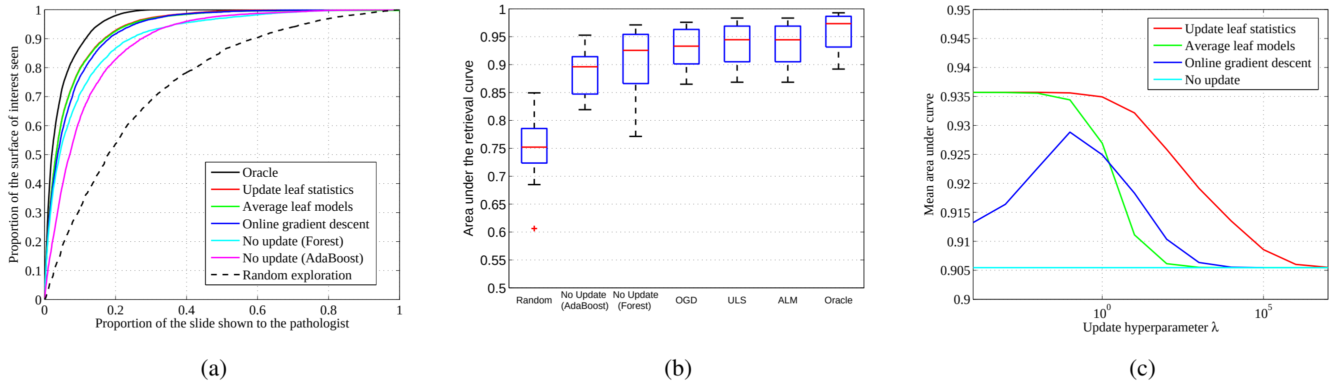
Due to the efficiency of Haar-like features, our segmentation algorithm is able to work directly at the highest level of magnification, processing approximately 2.0×10^7 pixels per minute. This order of magnitude is, for instance, the same as in a recent boosting-based hierarchical segmentation approach (Doyle et al., 2012) which analyzes around $1.4{\times}10^7$ pixels in less than 3 minutes (with parallelization on 2 threads instead of 10). However, this latter work used more complex features for their application, hence justifying a hierarchical strategy.

5.3. Evaluation of the Exploration Stage

We studied the ability of our approach to retrieve regions of interest as quickly as possible within large slides. The following alternatives were compared:

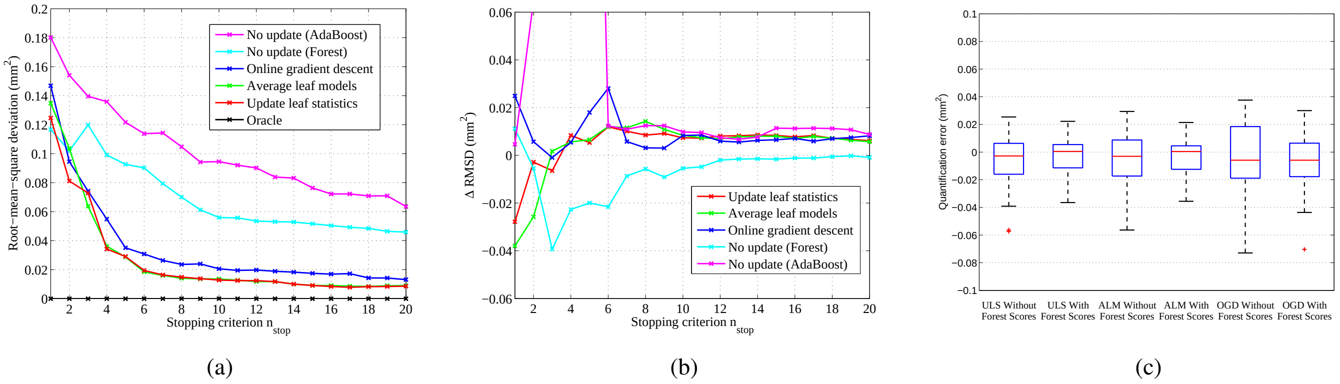
- the simplest forest-based exploration approach without update from the pathologist (`No Update (Forest)`), i.e. only relying on the pre-trained forest $\mathcal{F}^1$,
- the three update strategies (`ULS`, `ALM` and `OGD`) exposed in Sec. 4.2.2,
- a baseline showing, at each iteration, a region randomly (uniformly) drawn among the remaining regions (`Random exploration`),
- an oracle whose scoring function is extracted from the ground truth, hence serving as a gold standard showing the highest achievable performance (`Oracle`).

In addition, to position the forest-based performance among other classification methods, we trained an AdaBoost classifier and used it as segmentation model instead of the forest (`No Update (AdaBoost)`). The chosen weak classifiers were decision stumps based on Haar-like features such as the splitting functions stored in tree nodes. 100 boosting iterations were conducted, and 500 stumps tried at each iteration. These design choices led to a training time similar to the forest one.



(a) (b) (c)

Figure 8: **Experimental validation of the slide exploration stage.** (a) For each method, we plot the mean retrieval curve which shows the proportion of positive pixels seen by the pathologist after having seen a certain proportion of the slide. We use the area under these curves to measure quantitatively the slide exploration abilities. (b) Statistical distribution of the area under the curve for each method. Each box plot is computed over the 48 measurements obtained during the nested cross-validation. (c) Influence of the hyperparameter λ on the performance. For each of the three update strategies, we studied the behavior of the mean area under the curve when λ varies, i.e. without optimization on a validation set.



(a) (b) (c)

Figure 9: **Experimental validation of the whole-slide quantification stage.** (a) We report the root-mean-square deviation for several values of the stopping criterion n_{stop} which represents the amount of interaction provided by the pathologist. (b) Difference ΔRMSD between the root-mean-square deviation obtained when using a regression of the form $\hat{q} = (1 + a)q^{\text{labeled}}$ based on the user inputs alone, and the one obtained with the regression of Eq. 11 whose results are reported in Fig. 9a. Positive values of ΔRMSD correspond to a gain of accuracy when including the forest scores Φ^{total} and Φ^{labeled} in the regression task. As soon as n_{stop} is large enough to obtain stable results, the information carried by an updated forest improves the performance. (c) Distribution of the signed quantification error over slides for $n_{\text{stop}} = 10$, with and without including the forest scores in the regression.

To assess quantitatively the performance of each method, we consider the curve showing the proportion of positive pixels that have been displayed after having shown a certain percentage of the slide to the pathologist (Fig. 8a). A good exploration method is expected to lead to a curve converging quickly towards 1. We summarize quantitatively the performance on a slide I by computing the area $A(\lambda, \mathcal{F}^0, I)$ under this curve. The nested cross-validation procedure described in Sec. 5.2 was accordingly performed using the loss function $l^{\text{cv}}(\lambda, \mathcal{F}^1, I) = 1 - A(\lambda, \mathcal{F}^1, I)$ and optimizing λ over a logarithmic grid. The statistical distribution of the area under curves obtained at prediction time for each method are shown in Fig. 8b. We performed statistical pairwise comparisons between methods by conducting paired Wilcoxon's signed-rank tests over these values. To maintain the independence between samples, we repeated each test 100 times retaining at random one of the 3 runs for each slide and considered the median p-value over these 100 runs. Denoting Method 1 $\prec$ Method 2 the fact that Method 2 is significantly better than Method 1 and Method 1 $\approx$ Method 2 the absence of demonstrated statistical difference between the two methods, the series of tests provided the following ranking:

Random $\prec$ No Update (AdaBoost) $\prec$ No Update (Forest) $\prec$ OGD $\prec$ ULS $\approx$ ALM $\prec$ Oracle.

All p-values showing statistical difference were lower than 10^{-3}, and the p-value obtained when comparing ULS and ALM was 0.5. This ranking confirms what was intuitively expected. The three methods proposing a model update from the user inputs improve over a non-interactive exploration, and the two methods using accurate pixelwise labelings outperform the online gradient descent technique which is based on a weaker but lighter type of information. We also see, from the performance of a random exploration, that using a pre-trained forest drastically helps finding relevant objects more quickly. Note that, in theory, one might have expected a straight '$y = x$' line for the random exploration. In fact, when a suggested region belongs to a larger object, the user extends the field of view to see the object in its totality. Hence, a positive suggestion may be immediately followed by other positive ones due to the user intervention. This bias explains why, in spite of a random exploration, one obtains a slightly 'better than random' curve.

The impact of the update-related parameter λ was assessed experimentally (Fig. 8c). As expected, when $\lambda \to \infty$, the three methods converge towards the method without update. The choice $\lambda = 0$ leads to a nearly maximal performance for the two update strategies based on pixelwise labelings (ULS and ALM). Since, moreover, these two methods are equivalent for $\lambda = 0$, they behave very similarly after optimization on a validation set, as observed on Fig. 8a and Fig. 8b. Note that choosing $\lambda = 0$ does not mean that the initial forest $\mathcal{F}^1$ is ignored. The prior knowledge contained in $\mathcal{F}^1$ is used through both the tree structures and their leaf models. Moreover, a leaf remains unchanged as long as it does not appear in a selected region, which may happen if it accurately predicts background areas.

5.4. Evaluation of the Whole-Slide Quantification Stage

To assess, at prediction time, the accuracy of a list of n_{pred} estimates $(\hat{q}_i)_{1\leq i\leq n_{\text{pred}}}$ given the corresponding true quantities $(q_i)_{1\leq i\leq n_{\text{pred}}}$, we use the root-mean-square deviation

$$\text{RMSD} = \sqrt{\frac{1}{n_{\text{pred}}}\sum_{i=1}^{n_{\text{pred}}}(q_i - \hat{q}_i)^2}. \qquad (13)$$

In our case, $n_{\text{pred}} = 48$. This corresponds to the 3 predictions obtained for each of the 16 slides during the cross-validation.

The experimental evaluation of the whole-slide quantification abilities was conducted as follows. We kept the cross-validation setup described in Sec. 5.2 and learned the hyperparameter a on a validation set, as was done for the update parameter λ, using here a squared loss $l^{\text{cv}}(a, \mathcal{F}^1, I) = (q_I - \hat{q}_I)^2$ between true and predicted whole-slide estimates. This procedure was performed independently for several values of the stopping criterion n_{stop}. We report the resulting curves in Fig. 9a and an example of the correspondence between estimates and true quantities in Fig. 10. The ranking of methods obtained while studying the exploration abilities is preserved, due to the fact that the quality of the exploration phase is directly linked to the amount of regions which are eventually labeled. Asymptotically, if all regions containing positive samples are labeled, choosing $a = 0$ provides a perfect prediction.

During our experiments, we observed that the sum of segmentation probabilities over the whole slide, i.e. predicting $\hat{q} = \Phi^{\text{total}}$ (with the notation of Eq. 11), does not form a reliable whole-slide quantification and overestimates the quantity of positive pixels due to two effects. First, the random nature of trees leads in general to small nonzero probabilities on negative pixels. When summed over all pixels, these small errors aggregate. Moreover, since we mainly retrieve positive examples during the slide exploration, the distribution of incoming samples for the update is strongly biased towards positive instances. These difficulties motivate the use of a regression. In Fig. 9b and Fig. 9c, we show that our regression approach (Eq. 11) outperforms a regression based on the user inputs alone of the form $\hat{q} = (1 + a)q^{\text{labeled}}$. This demonstrates that, in spite of its global overestimation, the forest estimate can be effectively exploited by a regression procedure.

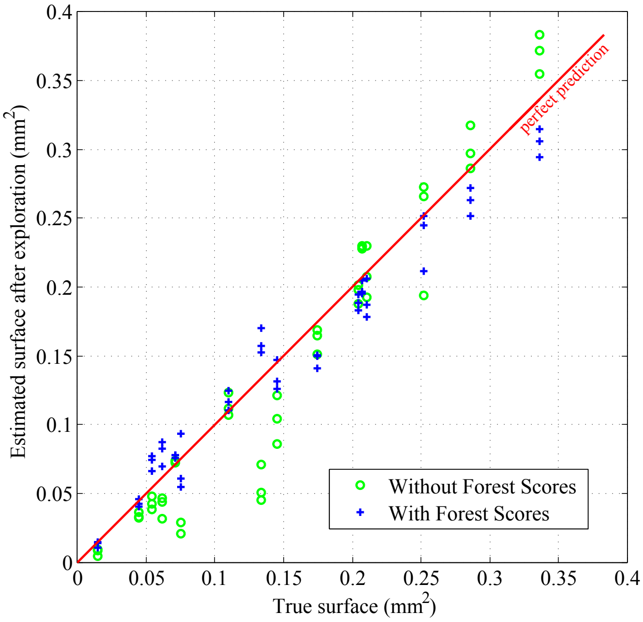


Figure 10: **Correlation between estimated and true hematopoietic surface within whole slides.** We show an overview on the whole dataset of the whole-slide estimates after exploration. Each slide tested during the cross-validation appears 3 times corresponding to different initial forests (Sec. 5.2). Perfect predictions would lie on the red line. This example was obtained using the ULS adaptation for $n_{\text{stop}} = 6$, leading to RMSD $= 1.9 \times 10^{-2}$ mm^2. The median proportion of the slide which was seen during the exploration was 5.2% in this case. The predictions obtained without using the final forest scores (see Fig. 9b and Fig. 9c) are also reported.

6. Input Discretization for Lighter Interactions

In the experiments presented in Sec. 5, the forest adaptation techniques assumed that the pathologist provides a full pixelwise labeling of the objects of interest in the displayed regions. In this section, we demonstrate how our OGD scheme can be used with one-click inputs instead without decreasing its performance, thereby allowing faster user interaction.

6.1. One-Click User Inputs

Unlike the two other techniques based on individual labels for each pixel in a region, the OGD forest update employs as user input for a region R_k the amount of positive pixels $Q(R_k)$ contained in R_k. This is a different kind of input, which can be infered from a delineation or communicated directly instead. Here, we propose to discretize the input values into bins and ask the user to select the bin to which the proportion of positive pixels belongs. This interaction is performed with only one click, or possibly without a mouse (e.g. via voice recognition).

Formally, the user annotations are discretized as follows. Instead of providing the exact quantity $Q(R_k)$, the user simply indicates an interval within which the proportion $\tilde{Q}(R_k) = \frac{Q(R_k)}{|R_k|}$ lies. The list of available ranges is predefined as $\{0\}$, $]0; \frac{1}{m}]$,

$\ldots, \left]\frac{m-2}{m}; \frac{m-1}{m}\right], \left]\frac{m-1}{m}; 1\right[, \{1\}$, where m is a positive integer which encodes the fineness of the quantization. Accordingly, by taking the middle-value of each bin, the proportions $\tilde{Q}(R_k)$ provided by the user take their values in the finite set $D_m = \left\{0, \frac{1}{2m}, \frac{3}{2m}, \ldots, \frac{2m-1}{2m}, 1\right\}$. This gives in total $m + 2$ input possibilities for the discrete input $\tilde{Q}(R_k)$, including the 2 trivial ones corresponding to an empty ($\tilde{Q}(R_k) = 0$) or a full ($\tilde{Q}(R_k) = 1$) region . By doing so, only one click per region is required from the pathologist, resulting in a lesser amount of interactions.

Given a discrete region label $\tilde{Q}(R_k) \in D_m$ provided by the user, we have to compute the actual quantity $Q(R_k)$ eventually used in the adaptation process (Eq. 8) and recorded for an eventual whole-slide quantification (Eq. 11). The simplest idea consists in directly taking $Q(R_k) = |R_k|\,\tilde{Q}(R_k)$, but has the drawback of losing information due to the discretization. To attenuate this aspect, we propose to perform updates only if the forest estimate $\phi(R_k|\mathcal{F}^k)$ (whose objective is to predict the quantity $Q(R_k)$) deviates too strongly from the user label. More precisely, we define $Q(R_k) = |R_k|\,\tilde{Q}(R_k)$ if $\left|\frac{\phi(R_k|\mathcal{F}^k)}{|R_k|} - \tilde{Q}(R_k)\right| \geq \frac{1}{2m}$, and $Q(R_k) = \phi(R_k|\mathcal{F}^k)$ otherwise. In other words, we fully trust the forest estimate as long as it leads to the same bin as the one indicated by the user. This distinction is only made if the label is ambiguous, i.e. different than 0 and 1. Otherwise, it corresponds in fact to an exact labeling and is treated as such.

6.2. Evaluation

The experiments involving online gradient descent presented in Sec. 5.3 and Sec. 5.4 were repeated using these discretized inputs instead of the exact ones, for every level of quantization $m \in \{1, \ldots, 5\}$. In terms of retrieval performance, discretizing the inputs does not show any clear difference in comparison to the use of exact user inputs, and this from $m = 1$ on (Fig. 11). For $m = 1$, the paired Wilcoxon's signed-rank test leads to a p-value of 0.65. Additionally, if we assume the differences to be normally distributed, the confidence interval for the mean difference is $\left[-4.8 \times 10^{-3}, 2.3 \times 10^{-3}\right]$. Hence, by making available to the user 3 buttons (corresponding to the choice $m = 1$) stating respectively whether a region is empty, full of hematopoietic cells or partially covered, the exploration phase is of equivalent quality as the one provided by the online gradient descent method with accurate user labelings.

Since the task of whole-slide quantification from the exploration phase (Sec. 5.4) relies strongly on the user inputs $\tilde{Q}(R_k)$ (see Eq. 11), obtaining satisfactory results for this task with discretized inputs requires a more accurate quantization. This minimum level was experimentally found to be $m = 3$, which remains nevertheless tractable in practice (Fig. 12).

7. Conclusion

We introduced an interactive framework able to help a pathologist to navigate efficiently through large digital slides. Our approach is based on a pixelwise random forest classifier pre-trained to segment objects of interest within the tissue whose predictions are used to score, rank and display regions according to their expected interest. By allowing the user to provide labels on each suggested region, the leaf nodes of the forest model are adjusted in real time during the exploration procedure so that visual specificities of the data at hand can be gradually incorporated into the region selection process. For this purpose, in addition to two standard leaf update techniques, we introduced a novel adaptation scheme based on online gradient descent which supports one-click inputs from the pathologist instead of more tedious accurate object delineations. Experimental validation was conducted on the task of extramedullary hematopoiesis quantification within mouse liver slides. Beyond its slide exploration abilities, we demonstrated how our method can successfully exploit both the forest segmentation output and the labels collected during the exploration stage to provide accurate estimates of the surface covered by hematopietic cells in the whole slide.

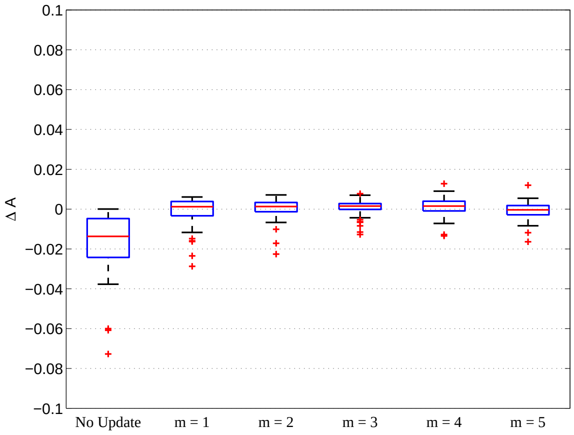


Figure 11: **Impact of input quantization on the exploration phase.** We consider the difference ΔA of area under the curve observed when using discretized user inputs instead of exact ones in the OGD adaptation. The parameter m encodes the fineness of the quantization. No statistical difference is observed from $m = 1$ on. To put the variations in perspective, we also report the difference obtained when no updates are performed.

Acknowledgments

This work was partially supported by the Collaborative Research Center 824: "Imaging for Selection, Monitoring and Individualization of Cancer Therapies". We are also very grateful to the pathologist Dr. Katja Steiger for supervising the acquisition and labeling of the data used in this study.

References

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

Bahlmann, C., Patel, A., Johnson, J., Ni, J., Chekkoury, A., Khurd, P., Kamen, A., Grady, L., Krupinski, E., Graham, A., et al., 2012. Automated detection of diagnostically relevant regions in H&E stained digital pathology slides, in: SPIE Medical Imaging, International Society for Optics and Photonics. pp. 831504–831504–8.

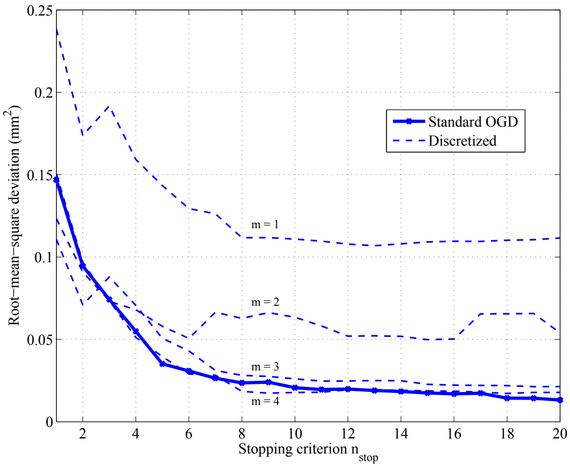


Figure 12: **Whole-slide quantification from discrete inputs.** From $m \geq 3$, the discretized approach reaches similar quantification performances as an OGD update with exact inputs.

Balermpas, P., Rödel, F., Rödel, C., Krause, M., Linge, A., Lohaus, F., Baumann, M., Tinhofer, I., Budach, V., Gkika, E., Stuschke, M., Avlar, M., Grosu, A.L., Abdollahi, A., Debus, J., Bayer, C., Stangl, S., Belka, C., Pigorsch, S., Multhoff, G., Combs, S.E., Mönnich, D., Zips, D., Fokas, E., 2016. CD8+ tumour-infiltrating lymphocytes in relation to HPV status and clinical outcome in patients with head and neck cancer after postoperative chemoradiotherapy: A multicentre study of the german cancer consortium radiation oncology group (DKTK-ROG). International Journal of Cancer 138, 171–181.

Bauer, T.W., Schoenfield, L., Slaw, R.J., Yerian, L., Sun, Z., Henricks, W.H., 2013. Validation of whole slide imaging for primary diagnosis in surgical pathology. Archives of Pathology & Laboratory Medicine 137, 518–524.

Bautista, P.A., Yagi, Y., 2015. Staining correction in digital pathology by utilizing a dye amount table. Journal of Digital imaging , 1–12.

Breiman, L., Friedman, J., Stone, C., Olshen, R., 1984. Classification and regression trees.

Carpenter, A.E., Jones, T.R., Lamprecht, M.R., Clarke, C., Kang, I.H., Friman, O., Guertin, D.A., Chang, J.H., Lindquist, R.A., Moffat, J., et al., 2006. Cellprofiler: image analysis software for identifying and quantifying cell phenotypes. Genome biology 7, R100.

Chatelain, P., Pauly, O., Peter, L., Ahmadi, S.A., Plate, A., Bötzel, K., Navab, N., 2013. Learning from multiple experts with random forests: Application to the segmentation of the midbrain in 3D ultrasound, in: Medical Image Computing and Computer Assisted Intervention (MICCAI). Springer, pp. 230–237.

Cooper, L.A., Carter, A.B., Farris, A.B., Wang, F., Kong, J., Gutman, D.A., Widener, P., Pan, T.C., Cholleti, S.R., Sharma, A., et al., 2012. Digital pathology: Data-intensive frontier in medical imaging. Proceedings of the IEEE 100, 991–1003.

Criminisi, A., Shotton, J., Bucciarelli, S., 2009. Decision forests with long-range spatial context for organ localization in CT volumes, in: MICCAI Workshop on Probabilistic Models for Medical Image Analysis.

Criminisi, A., Shotton, J., Konukoglu, E., 2012. Decision forests: A unified framework for classification, regression, density estimation, manifold learning and semi-supervised learning. Foundations and Trends® in Computer Graphics and Vision 7, 81–227.

Doyle, S., Feldman, M., Tomaszewski, J., Madabhushi, A., 2012. A boosted bayesian multiresolution classifier for prostate cancer detection from digitized needle biopsies. IEEE Transactions on Biomedical Engineering 59, 1205–1218.

Ebner, T., Stern, D., Donner, R., Bischof, H., Urschler, M., 2014. Towards automatic bone age estimation from MRI: Localization of 3D anatomical landmarks, in: Medical Image Computing and Computer Assisted Intervention (MICCAI). Springer, pp. 429–437.

Eefting, D., Schrage, Y.M., Geirnaerdt, M.J., Le Cessie, S., Taminiau, A.H., Bovée, J.V., Hogendoorn, P.C., et al., 2009. Assessment of interobserver variability and histologic parameters to improve reliability in classification and grading of central cartilaginous tumors. The American Journal of Surgical Pathology 33, 50–57.

Farahani, N., Parwani, A., Pantanowitz, L., 2015. Whole slide imaging in pathology: advantages, limitations, and emerging perspectives. Pathology and Laboratory Medicine International 7, 23–33.

Fiaschi, L., Köthe, U., Nair, R., Hamprecht, F.A., 2012. Learning to count with regression forest and structured labels, in: International Conference on Pattern Recognition (ICPR), IEEE. pp. 2685–2688.

Gauriau, R., Cuingnet, R., Lesage, D., Bloch, I., 2014. Multi-organ localization combining global-to-local regression and confidence maps, in: Medical Image Computing and Computer Assisted Intervention (MICCAI). Springer, pp. 337–344.

Gilles, F.H., Tavaré, C.J., Becker, L.E., Burger, P.C., Yates, A.J., Pollack, I.F., Finlay, J.L., 2007. Pathologist interobserver variability of histologic features in childhood brain tumors: results from the CCG-945 study. Pediatric and Developmental Pathology 11, 108–117.

Gonul, I.I., Poyraz, A., Unsal, C., Acar, C., Alkibay, T., 2006. Comparison of 1998 W HO/ISUP and 1973 WHO classifications for interobserver variability in grading of papillary urothelial neoplasms of the bladder. Pathological evaluation of 258 cases. Urologia Internationalis 78, 338–344.

Gorelick, L., Veksler, O., Gaed, M., Gomez, J., Moussa, M., Bauman, G., Fenster, A., Ward, A., 2013. Prostate histopathology: Learning tissue component histograms for cancer detection and classification. IEEE Transactions on Medical Imaging 32, 1804–1818.

Gurcan, M., Boucheron, L., Can, A., Madabhushi, A., Rajpoot, N., Yener, B., 2009. Histopathological image analysis: A review. IEEE Reviews in Biomedical Engineering 2, 147–171.

Held, M., Schmitz, M.H., Fischer, B., Walter, T., Neumann, B., Olma, M.H., Peter, M., Ellenberg, J., Gerlich, D.W., 2010. Cellcognition: time-resolved phenotype annotation in high-throughput live cell imaging. Nature Methods 7, 747–754.

Homeyer, A., Schenk, A., Arlt, J., Dahmen, U., Dirsch, O., Hahn, H.K., 2013. Practical quantification of necrosis in histological whole-slide images. Computerized Medical Imaging and Graphics 37, 313–322.

Homeyer, A., Schenk, A., Dahmen, U., Dirsch, O., Huang, H., Hahn, H.K., 2011. A comparison of sampling strategies for histological image analysis. Journal of Pathology Informatics 2.

Huang, C.H., Veillard, A., Roux, L., Loménie, N., Racoceanu, D., 2011. Time-efficient sparse analysis of histopathological whole slide images. Computerized Medical Imaging and Graphics 35, 579–591.

Jaarsma, T., Jarodzka, H., Nap, M., Merrienboer, J.J., Boshuizen, H., 2014. Expertise under the microscope: processing histopathological slides. Medical Education 48, 292–300.

Jain, V., Learned-Miller, E., 2011. Online domain adaptation of a pre-trained cascade of classifiers, in: IEEE Computer Vision and Pattern Recognition (CVPR), pp. 577–584.

Jukić, D.M., Drogowski, L.M., Martina, J., Parwani, A.V., 2011. Clinical examination and validation of primary diagnosis in anatomic pathology using whole slide digital images. Archives of Pathology & Laboratory Medicine 135, 372–378.

Khan, A.M., Rajpoot, N., Treanor, D., Magee, D., 2014. A non-linear mapping approach to stain normalisation in digital histopathology images using image-specific colour deconvolution. IEEE Transactions on Biomedical Engineering .

Khurd, P., Bahlmann, C., Maday, P., Kamen, A., Gibbs-Strauss, S., Genega, E., Frangioni, J., 2010. Computer-aided gleason grading of prostate cancer histopathological images using texton forests, in: IEEE International Symposium on Biomedical Imaging (ISBI), pp. 636–639.

Kong, H., Gurcan, M., Belkacem-Boussaid, K., 2011. Partitioning histopathological images: An integrated framework for supervised color-texture segmentation and cell splitting. IEEE Transactions on Medical Imaging 30, 1661–1677.

Kontschieder, P., Dorn, J., Morrison, C., Corish, R., Zikic, D., Sellen, A., DSouza, M., Kamm, C.P., Burggraaff, J., Tewarie, P., et al., 2014. Quantifying progression of multiple sclerosis via classification of depth videos, in: Medical Image Computing and Computer Assisted Intervention (MICCAI), Springer. pp. 429–437.

Lakshminarayanan, B., Roy, D.M., Teh, Y.W., 2014. Mondrian forests: Efficient online random forests, in: Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N., Weinberger, K. (Eds.), Advances in Neural Information Processing Systems (NIPS), pp. 3140–3148.

Macenko, M., Niethammer, M., Marron, J., Borland, D., Woosley, J., Guan, X., Schmitt, C., Thomas, N., 2009. A method for normalizing histology slides for quantitative analysis, in: IEEE International Symposium on Biomedical Imaging (ISBI), pp. 1107–1110.

Mercan, E., Aksoy, S., Shapiro, L.G., Weaver, D.L., Brunye, T., Elmore, J.G., 2014. Localization of diagnostically relevant regions of interest in whole slide images, in: IEEE International Conference on Pattern Recognition (ICPR), pp. 1179–1184.

Meyer, J.S., Alvarez, C., Milikowski, C., Olson, N., Russo, I., Russo, J., Glass, A., Zehnbauer, B.A., Lister, K., Parwaresch, R., 2005. Breast carcinoma malignancy grading by bloom–richardson system vs proliferation index: reproducibility of grade and advantages of proliferation index. Modern Pathology 18, 1067–1078.

Montillo, A., Shotton, J., Winn, J., Iglesias, J., Metaxas, D., Criminisi, A., 2011. Entangled decision forests and their application for semantic segmentation of CT images, in: Information Processing in Medical Imaging (IPMI). Springer, pp. 184–196.

Nemirovski, A., Juditsky, A., Lan, G., Shapiro, A., 2009. Robust stochastic approximation approach to stochastic programming. SIAM Journal on Optimization 19, 1574–1609.

Nguyen, K., Sarkar, A., Jain, A., 2014. Prostate cancer grading: use of graph cut and spatial arrangement of nuclei. IEEE Transactions on Medical Imaging 33, 2254.

Onder, D., Zengin, S., Sarioglu, S., 2014. A review on color normalization and color deconvolution methods in histopathology. Applied Immunohistochemistry & Molecular Morphology 22, 713–719.

Pauly, O., Glocker, B., Criminisi, A., Mateus, D., Möller, A., Nekolla, S., Navab, N., 2011. Fast multiple organ detection and localization in whole-body MR dixon sequences, in: Medical Image Computing and Computer Assisted Intervention (MICCAI). Springer, pp. 239–247.

Peter, L., Mateus, D., Chatelain, P., Schworm, N., Stangl, S., Multhoff, G., Navab, N., 2014. Leveraging random forests for interactive exploration of large histological images, in: Medical Image Computing and Computer Assisted Intervention (MICCAI). Springer, pp. 1–8.

Peter, L., Pauly, O., Chatelain, P., Mateus, D., Navab, N., 2015. Scale-adaptive forest training via an efficient feature sampling scheme, in: Medical Image Computing and Computer Assisted Intervention (MICCAI). Springer, pp. 637–644.

Rabinovich, A., Agarwal, S., Laris, C., Price, J.H., Belongie, S.J., 2003. Unsupervised color decomposition of histologically stained tissue samples, in: Advances in Neural Information Processing Systems (NIPS), pp. 667–674.

Roullier, V., Lézoray, O., Ta, V.T., Elmoataz, A., 2011. Multi-resolution graph-based analysis of histopathological whole slide images: Application to mitotic cell extraction and visualization. Computerized Medical Imaging and Graphics 35, 603–615.

Saffari, A., Leistner, C., Santner, J., Godec, M., Bischof, H., 2009. On-line random forests, in: IEEE International Conference on Computer Vision Workshops, pp. 1393–1400.

Sertel, O., Kong, J., Shimada, H., Catalyurek, U., Saltz, J.H., Gurcan, M.N., 2009. Computer-aided prognosis of neuroblastoma on whole-slide images: Classification of stromal development. Pattern Recognition 42, 1093–1103.

Settles, B., 2010. Active learning literature survey. Computer Sciences Technical Report 1648.

Shalev-Shwartz, S., 2012. Online learning and online convex optimization. Foundations and Trends® in Machine Learning 4, 107–194.

Shotton, J., Winn, J., Rother, C., Criminisi, A., 2006. Textonboost: Joint appearance, shape and context modeling for multi-class object recognition and segmentation, in: European Conference on Computer Vision (ECCV). Springer, pp. 1–15.

Sommer, C., Straehle, C., Köthe, U., Hamprecht, F.A., 2011. Ilastik: Interactive learning and segmentation toolkit, in: 2011 IEEE International Symposium on Biomedical Imaging, pp. 230–233.

Su, H., Xing, F., Kong, X., Xie, Y., Zhang, S., Yang, L., 2015. Robust cell detection and segmentation in histopathological images using sparse reconstruction and stacked denoising autoencoders, in: Medical Image Computing and Computer-Assisted Intervention (MICCAI). Springer, pp. 383–390.

Tao, K., Fang, M., Alroy, J., Sahagian, G., 2008. Imagable 4T1 model for the study of late stage breast cancer. BMC Cancer 8.

Tommasi, T., Orabona, F., Kaboli, M., Caputo, B., 2012. Leveraging over prior knowledge for online learning of visual categories, in: British Machine Vision Conference, pp. 87.1–87.11.

Vahadane, A., Peng, T., Albarqouni, S., Baust, M., Steiger, K., Schlitter, A., Sethi, A., Esposito, I., Navab, N., 2015. Structure-preserved color normalization for histological images, in: IEEE International Symposium on Biomedical Imaging (ISBI), pp. 1012–1015.

Veta, M., Pluim, J., van Diest, P., Viergever, M., 2014. Breast cancer histopathology image analysis: A review. IEEE Transactions on Biomedical Engineering .

Viola, P., Jones, M.J., 2004. Robust real-time face detection. International Journal of Computer Vision .

Xu, Y., Zhu, J.Y., Eric, I., Chang, C., Lai, M., Tu, Z., 2014. Weakly supervised histopathology cancer image segmentation and classification. Medical Image Analysis 18, 591–604.

Zhao, P., Hoi, S.C., 2010. OTL: A framework of online transfer learning, in: International Conference on Machine Learning, pp. 1231–1238.

Zikic, D., Glocker, B., Criminisi, A., 2014. Encoding atlases by randomized classification forests for efficient multi-atlas label propagation. Medical Image Analysis 18, 1262–1273.

Zinkevich, M., 2003. Online convex programming and generalized infinitesimal gradient ascent. International Conference on Machine Learning .

Appendix A. Rewriting the Scoring Function $\phi(R|\mathcal{F})$ as a Scalar Product (Eq. 3)

We expose here in details how Eq. 3 can be derived. This equality mainly comes from the linearity of both the scoring function ϕ and the aggregation of the tree predictions. Combining Eq. 2 and Eq. 1, one obtains

$$\begin{aligned} \phi(R|\mathcal{F}) &= \frac{1}{n_{\text{trees}}} \sum_{\mathbf{x}\in R} \sum_{t=1}^{n_{\text{trees}}} \pi_{\sigma_t(\mathbf{x})} && \text{(A.1)} \\ &= \frac{1}{n_{\text{trees}}} \sum_{\mathbf{x}\in R} \sum_{t=1}^{n_{\text{trees}}} \sum_{L\in\mathcal{L}_t} \pi_L \mathbf{1}_{\{\sigma_t(\mathbf{x})=L\}}. && \text{(A.2)} \end{aligned}$$

We defined in Sec. 4.2.1 the quantity $\text{tree}(L) \in \{1,\ldots,n_{\text{trees}}\}$ as the index of the tree to which a leaf $L \in \mathcal{L}$ belongs. Following this definition, we have for all trees $t \in \{1,\ldots,n_{\text{trees}}\}$ and leaves $L \in \mathcal{L}_t$ the equality $t = \text{tree}(L)$. Thus

$$\begin{aligned} \sum_{t=1}^{n_{\text{trees}}} \sum_{L\in\mathcal{L}_t} \pi_L \mathbf{1}_{\{\sigma_t(\mathbf{x})=L\}} &= \sum_{t=1}^{n_{\text{trees}}} \sum_{L\in\mathcal{L}_t} \pi_L \mathbf{1}_{\{\sigma_{\text{tree}(L)}(\mathbf{x})=L\}} && \text{(A.3)} \\ &= \sum_{L\in\mathcal{L}} \pi_L \mathbf{1}_{\{\sigma_{\text{tree}(L)}(\mathbf{x})=L\}} && \text{(A.4)} \end{aligned}$$

since the double sum $\sum_{t=1}^{n_{\text{trees}}} \sum_{L\in\mathcal{L}_t}$ amounts to summing over all leaves in the forest. Finally, by incorporating Eq. A.4 in Eq. A.2, we obtain

$$\begin{aligned} \phi(R|\mathcal{F}) &= \frac{1}{n_{\text{trees}}} \sum_{\mathbf{x}\in R} \sum_{L\in\mathcal{L}} \pi_L \mathbf{1}_{\{\sigma_{\text{tree}(L)}(\mathbf{x})=L\}} && \text{(A.5)} \\ &= \sum_{L\in\mathcal{L}} \pi_L \left(\frac{1}{n_{\text{trees}}} \sum_{\mathbf{x}\in R} \mathbf{1}_{\{\sigma_{\text{tree}(L)}(\mathbf{x})=L\}} \right) && \text{(A.6)} \\ &= \sum_{L\in\mathcal{L}} \pi_L \rho_L(R|\Sigma) && \text{(A.7)} \\ &= \langle \boldsymbol{\rho}(R|\Sigma), \boldsymbol{\pi} \rangle && \text{(A.8)} \end{aligned}$$

using the definition of $\boldsymbol{\rho}(R|\Sigma)$ given in Eq. 4.

Appendix B. Choice of Learning Rate (Eq. 9)

We expose here the theoretical considerations leading to the form of the learning rate exposed in Eq. 9. We follow a classical reasoning inspired from the online learning literature (Nemirovski et al., 2009; Shalev-Shwartz, 2012) and show how it relates to our scenario by expressing bounds in terms of the parameters of our method. As detailed in Sec. 4.2.2, at each iteration $k \geq 1$, the current set of leaf models $\boldsymbol{\pi}^k$ suffers the loss $l_k(\boldsymbol{\pi}^k) = \left(\left\langle \boldsymbol{\rho}(R_k), \boldsymbol{\pi}^k \right\rangle - Q(R_k)\right)^2$, after which a new vector of leaf models $\boldsymbol{\pi}^{k+1}$ is chosen according to the online gradient descent update rule (Eq. 7). After T region suggestions ($T \geq 1$), the cumulated regret of having used the series of models $\boldsymbol{\pi}^1, \ldots, \boldsymbol{\pi}^T$ is defined as

$$\text{Regret}_T = \sum_{k=1}^{T} \left(l_k(\boldsymbol{\pi}^k) - l_k(\boldsymbol{\pi}^*) \right), \tag{B.1}$$

where

$$\boldsymbol{\pi}^* = \underset{\boldsymbol{\pi} \in [0,1]^{|\mathcal{L}|}}{\text{argmin}} \sum_{k=1}^{T} l_k(\boldsymbol{\pi}) \tag{B.2}$$

corresponds to the set of leaf models which would have incurred the smallest loss over the T iterations. The reasoning consists in computing an upper bound of Regret_T depending on the learning rate η. To do so, we use the fact that the functions l_k are convex, so that, for all k, we have $l_k(\boldsymbol{\pi}^k) \leq l_k(\boldsymbol{\pi}^*) + \left\langle \boldsymbol{\pi}^k - \boldsymbol{\pi}^*, \vec{\nabla} l_k(\boldsymbol{\pi}^k) \right\rangle$ and thus

$$\text{Regret}_T \leq \sum_{k=1}^{T} \left\langle \boldsymbol{\pi}^k - \boldsymbol{\pi}^*, \vec{\nabla} l_k(\boldsymbol{\pi}^k) \right\rangle. \tag{B.3}$$

To find an upper bound of $A_k = \left\langle \boldsymbol{\pi}^k - \boldsymbol{\pi}^*, \vec{\nabla} l_k(\boldsymbol{\pi}^k) \right\rangle$, we use the update rule of Eq. 7 as follows. For all k, denoting $\Pi = \Pi_{[0,1]^{|\mathcal{L}|}}$ and $D_k = \left\| \boldsymbol{\pi}^k - \boldsymbol{\pi}^* \right\|$, we have

$$\begin{aligned} D_{k+1}^2 &= \left\| \boldsymbol{\pi}^{k+1} - \boldsymbol{\pi}^* \right\|^2 && \text{(B.4)} \\ &= \left\| \Pi\left[\boldsymbol{\pi}^k - \eta \vec{\nabla} l_k(\boldsymbol{\pi}^k) \right] - \boldsymbol{\pi}^* \right\|^2 && \text{(B.5)} \\ &= \left\| \Pi\left[\boldsymbol{\pi}^k - \eta \vec{\nabla} l_k(\boldsymbol{\pi}^k) \right] - \Pi\left[\boldsymbol{\pi}^* \right] \right\|^2 && \text{(B.6)} \\ &\leq \left\| \boldsymbol{\pi}^k - \eta \vec{\nabla} l_k(\boldsymbol{\pi}^k) - \boldsymbol{\pi}^* \right\|^2 && \text{(B.7)} \\ &= D_k^2 - 2\eta A_k + \eta^2 \left\| \vec{\nabla} l_k(\boldsymbol{\pi}^k) \right\|^2, && \text{(B.8)} \end{aligned}$$

which leads to the inequality

$$A_k \leq \frac{1}{2\eta} \left(D_k^2 - D_{k+1}^2 + \eta^2 \left\| \vec{\nabla} l_k(\boldsymbol{\pi}^k) \right\|^2 \right). \tag{B.9}$$

The inequality between Eq. B.6 and Eq. B.7 results from the fact that, in the Hilbert space $\mathbb{R}^{|\mathcal{L}|}$, performing a projection on the closed convex set $[0,1]^{|\mathcal{L}|}$ does not increase the distance between two points. Using Eq. B.9 in Eq. B.3, we obtain

$$\begin{aligned} \text{Regret}_T &\leq \frac{1}{2\eta} \sum_{k=1}^{T} \left(D_k^2 - D_{k+1}^2 + \eta^2 \left\| \vec{\nabla} l_k(\boldsymbol{\pi}^k) \right\|^2 \right) && \text{(B.10)} \\ &= \frac{1}{2\eta} \left[D_1^2 - D_{T+1}^2 + \eta^2 \sum_{k=1}^{T} \left\| \vec{\nabla} l_k(\boldsymbol{\pi}^k) \right\|^2 \right] && \text{(B.11)} \\ &\leq \frac{1}{2\eta} D_1^2 + \frac{\eta}{2} \sum_{k=1}^{T} \left\| \vec{\nabla} l_k(\boldsymbol{\pi}^k) \right\|^2. && \text{(B.12)} \end{aligned}$$

To obtain a final bound on the regret, we need to find an upper bound of D_1^2 and of the norm of the gradient $\left\| \vec{\nabla} l_k(\boldsymbol{\pi}^k) \right\|^2$. First, since both $\boldsymbol{\pi}^1$ and $\boldsymbol{\pi}^*$ belong to $[0,1]^{|\mathcal{L}|}$, we have

$$D_1^2 = \left\| \boldsymbol{\pi}^1 - \boldsymbol{\pi}^* \right\|^2 \leq |\mathcal{L}|. \tag{B.13}$$

Secondly, for all k and $\boldsymbol{\pi}$, we have

$$\vec{\nabla} l_k(\boldsymbol{\pi}) = 2 \left(\langle \boldsymbol{\rho}(R_k), \boldsymbol{\pi} \rangle - Q(R_k) \right) \boldsymbol{\rho}(R_k). \tag{B.14}$$

The quantity $\langle \boldsymbol{\rho}(R_k), \boldsymbol{\pi} \rangle$ estimates the surface covered by positive pixels in the region R_k, while $Q(R_k)$ is the actual value of this surface revealed by the user. Since, by definition, both $\left\langle \boldsymbol{\rho}(R_k), \boldsymbol{\pi}^k \right\rangle$ and $Q(R_k)$ are comprised between 0 and the size δ^2 of the region, we have $\left| \left\langle \boldsymbol{\rho}(R_k), \boldsymbol{\pi}^k \right\rangle - Q(R_k) \right| \leq \delta^2$. By definition of $\boldsymbol{\rho}$ (see Eq. 4), we also know that each individual component ρ_L does not exceed $\frac{\delta^2}{n_{\text{trees}}}$ (since at most the number of pixels in the region δ^2 can fall in a leaf L), and, moreover, that these components sum to δ^2. Thus

$$\begin{aligned} \|\boldsymbol{\rho}(R_k)\|^2 &= \sum_{L \in \mathcal{L}} \rho_L^2(R_k) && \text{(B.15)} \\ &\leq \frac{\delta^2}{n_{\text{trees}}} \sum_{L \in \mathcal{L}} \rho_L(R_k) && \text{(B.16)} \\ &= \frac{\delta^4}{n_{\text{trees}}}, && \text{(B.17)} \end{aligned}$$

hence the following upper bound on the gradient:

$$\left\| \vec{\nabla} l_k(\boldsymbol{\pi}^k) \right\|^2 \leq 4 \frac{\delta^8}{n_{\text{trees}}}. \tag{B.18}$$

Finally, including Eq. B.13 and Eq. B.18 in Eq. B.12 gives the bound

$$\text{Regret}_T \leq \frac{|\mathcal{L}|}{2\eta} + \frac{2\eta T \delta^8}{n_{\text{trees}}}. \tag{B.19}$$

We choose the value of η providing the best regret bound, i.e. minimizing the right side of Eq. B.19. This is obtained for

$$\eta = \frac{1}{2\delta^4} \sqrt{\frac{n_{\text{trees}} |\mathcal{L}|}{T}}. \tag{B.20}$$

While the relevant number of iterations T for the practical applicability of our scenario is unknown, it should at least be proportional to the size of the test slide, and thus to the number of regions $|\mathcal{R}|$. This leads us to define $T = \lambda^2 |\mathcal{R}|$ as proportional to this quantity, resulting in Eq. 9, and learn the hyperparameter λ on a validation set.