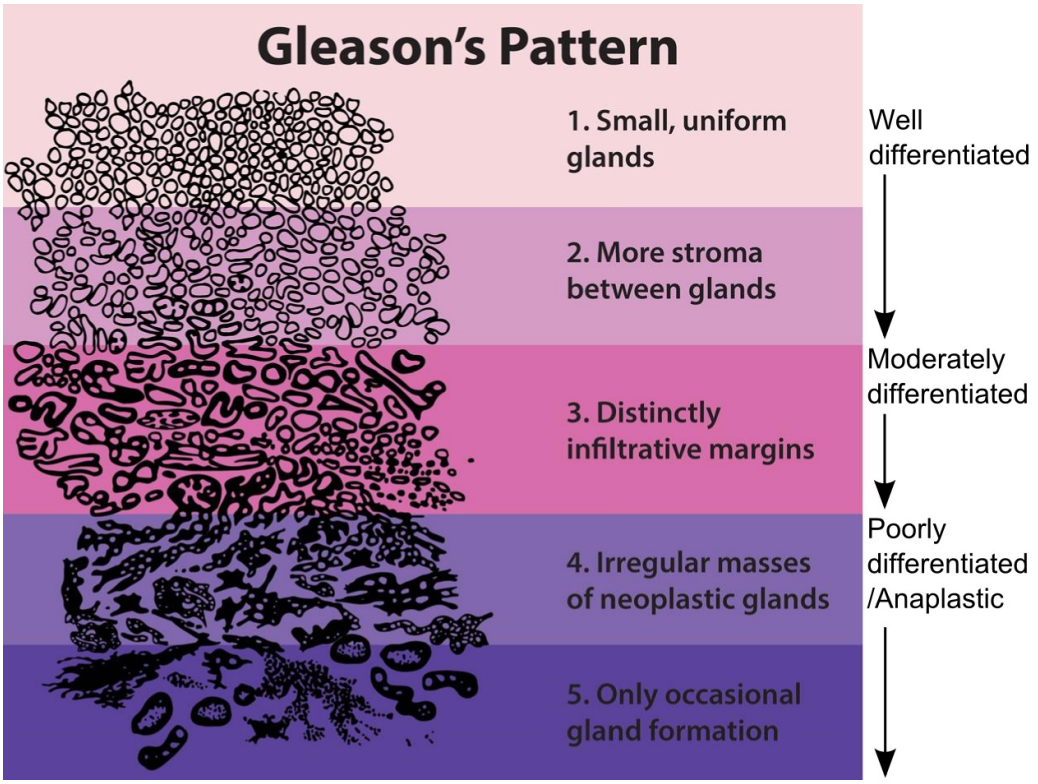
Introduction

This thesis will focus on the most common form of prostate cancer, prostatic adenocarcinoma. For simplicity, we will refer to this type as just 'prostate cancer.' This disease affects one in 1.4 million men every year, making it the most prevalent common type of cancer in men (excluding skin cancers).**Sung2021-iz?** Prostate cancer is a disease of the epithelial cells of the prostate. Epithelial cells line our body cavities, hollow organs, and glands. They undergo rapid proliferation, primarily due to damage. This proliferation increases the risk of genetic mutations, ultimately increasing the risk of a cell uncontrollably dividing. Together with enabling factors of its tissue environment, this can give rise to cancer.

In general, the more aggressive cancerous cells are, the less they will behave and morphologically appear like their original function. The prostate is a gland that produces prostatic fluid. The fluid is transported to the urethra by small 0ubes. These tubes, called prostatic glands, are lined with epithelium. Low-grade cancer will thus mimic those gland structures. High-grade prostate cancer loses its structural morphology, forming sheets of cells or even quasi-randomly dispersed individual cancerous cells.

American pathologist Donald Floyd Gleason systematically wrote down the correlation between growth patterns and prognosis in prostate cancer in the 1960s**cite_original?**. Pathologists still use this Gleason grading, albeit several revisions later[1], to classify prostate cancer.



Gleason's growth patterns. Image of the Gleason score for prostate cancer grading based on the original description in 1977. From: Morphology & Grade. ICD-O-3 Morphology Codes. National Institutes of Health.[2]

Prognostic biomarkers

To decide on a treatment plan, clinicians divide patients into risk groups according to traditional baseline characteristics, such as PSA blood level, Gleason grade, tumor location, tumor size, and lymph node status.[lam2019] This information is gathered from histopathological, radiological assessment, and lab assessments. These assessments can be considered biomarkers as they indicate the prognosis of a patient[3]. The more precise these assessments are, the better we can tailor the treatment to the specific patient; this is known as personalized medicine.

To make treatment more tailored to the patient, researchers try to develop new biomarkers. There is a demand for new biomarkers because most

prostate cancers progress so slowly that they are unlikely to threaten the affected individual's survival, and patients with the same histological and clinical characteristics, can have strikingly different outcomes[4]. Being able to pick out patients with good prognoses would improve their quality of life since treatments for prostate cancer obviously have adverse effects (#tab:adverse){reference-type="ref" reference="tab:adverse"}). Equally so for patients for which we can find out the treatment will not contribute to their health. To prevent adverse effects and increase treatment response, researchers are developing new markers in genomics[4], radiology[5], and pathology, the latter of which is the subject of this thesis.

Table 1: Common Prostate Cancer Treatment Options and Potential Adverse Effects, reproduced from Dunn et al.[6]

Treatment Option	**Disease Progression**	**Potential Adverse Effects**
Active surveillance	Localized	Illness uncertainty
Radical prostatectomy	Localized	Erectile dysfunction
		Urinary incontinence
External beam radiation	Localized and advanced disease	Urinary urgency and frequency
		Dysuria, diarrhea and proctitis
		Erectile dysfunction
		Urinary incontinence
Brachytherapy	Localized	Urinary urgency and frequency
		Dysuria, diarrhea and proctitis
		Erectile dysfunction
		Urinary incontinence
Cryotherapy	Localized	Erectile dysfunction
		Urinary incontinence and retention
		Rectal pain and fistula
Hormone therapy	Advanced	Fatigue
		Hot flashes, and flare effect
		Hyperlipidemia
		Insulin resistance
		Cardiovascular disease
		Anemia
		Osteoporosis
		Erectile dysfunction
		Cognitive deficits
Chemotherapy	Advanced	Myelosuppression
		Hypersensitivity reaction
		Gastrointestinal upset
		Peripheral neuropathy

Biomarkers based on histopathology

We know that histopathology holds prognostic information. Commonly, pathologist also report extra-capsular extension of the tumor and perineural invasion, both signs of poor prognosis. As mentioned earlier, the Gleason patterns were discovered by recording patient prognosis. Gleason growth patterns are grouped into five different groups, of which current pathologists mainly use the last three. It's not hard to imagine there being more clues in the morphology of the behavior of the tumor, if only because the landscape of prostate cancer growth patterns is certainly more complex than the three groups we divide them in. Of note, recently, the 'subpattern' cribriform-like growth was discovered to be an aggressive pattern.

However, these visual biomarkers are hard to explicitly specify and quantify manually. Luckily, machine learning can help. The first chapter will discuss this approach further. However, it makes sense to introduce this research field, computational pathology, first.

Computation Pathology

Pathology is undergoing a digital revolution. More and more labs are purchasing whole-slide scanners, with some already reading most slides digitally. Glass slides are digitized, resulting in gigapixel digital images, commonly referred to as whole-slide images (WSIs). Once the data is digital, opportunities for computational analysis and assistance arise.

Litjens et al. [1] gave an overview of deep learning applications in computational pathology up to 2016. Some early successes in the field focused on segmentation, tissue classification, and disease classification. Often reaching comparable results on the manual performance of the tasks by pathologists. Notably, the vast majority of these tasks are not on prognosis or treatment response prediction. Likely due to the fact these tasks are relatively easier and the kind of data needed is relatively cheap to obtain compared to survival data.

All state-of-the-art methods use some flavor of deep learning. A method where we train a model with multiple layers of computations, interwoven with non-linearities. A decade ago, optimizing these neural networks on GPU accelerators became common. The use of GPU made us able to develop models with a lot of layers (hence 'deep' learning) on large datasets. From

the start, we have been using a type of neural network, termed convolutional neural networks, in vision applications.

Convolutional neural networks

Convolutional neural networks (CNNs) have emerged among the state-of-the-art machine learning algorithms for various computer vision tasks, such as image classification and segmentation.

The central component of a convolutional neural network is often represented as a sliding kernel (or filter) over an input matrix, producing an output matrix. See Figure 1. This has several advantages; we can use a smaller kernel than the whole making the network less complex while exploiting the fact that objects in the image are translation invariant. A cat in the upper-left corner is still a cat in the lower-right corner. We introduce this inductive bias to the network by using convolutions.

TODO: Figure 1. Most convolutional neural network architectures have alternating blocks of layers consisting of a convolutional operation, a non-linear activation function, and often a normalization operation. The non-linearities are essential, as they make the networks able to represent more complex (non-linear) functions. Normalization layers bound the output of the block to be within a specific range which helps during the optimization of the network.

Even though sliding kernels are less complex than having one parameter per input value, the network architectures have evolved to become deeper and wider to enhance their accuracy further. Training larger CNNs demands larger amounts of computer memory, which increases exponentially with the size of input images. Consequently, most natural image datasets in computer vision, such as ImageNet and CIFAR-10, contain sub-megapixel images to circumvent memory limitations.

TODO: Overview of a whole CNN In specific domains like remote sensing and medical imaging, there is a need to train CNNs in high-resolution, where most of the information is contained. Ideally, we want to combine the high-resolution information with a more global context, as pathologists can do during daily practice. However, computer memory becomes a limiting

factor. The memory requirements of CNNs increase proportionally to the input size of the network, quickly filling up memory with multi-megapixel images. As a result, only small CNNs can be trained with such images, rendering state-of-the-art architectures unattainable even on large computing clusters.

Weakly supervised methods

For others, several authors have suggested approaches to train convolutional neural networks (CNNs) with large whole-slide images while preventing memory bottlenecks.

The most common solution is to train on high-resolution, but smaller regions of the slide. These patches are combined with annotations, and this reduces the need for the whole slide to be in memory. While this reduces the context of the whole slide down to what's contained in a small patch, for common problems, this is context enough. For example, tumor classification or segmentation don't require the context of the whole slide.

It is possible to train with the slide-level label and patches, this approach is called Multiple-Instance Learning. Here we assume that one or a few patches are enough the predict the label. In a binary classification setting, a positive slide contains at least one positive patch and a negative slide none. Only the most informative patches per slide are used for backpropagation. [cite MIL]

Another weakly supervised approach is to train a model to compress the WSI into a lower-dimensional latent space. This model is often trained on patches, in a generative or self-supervised way. This allows us to embed the whole slide, patch-per-patch into a smaller matrix, and train a supervised network on the compression. [cite NIC/CLAM]

There are other, even more engineering-heavy, approaches to dealing with the high-resolution of slides. Such as using reinforcement learning, …

In this thesis, a novel streaming method is proposed to train CNNs end-to-end on entire WSIs with slide-level labels. By reconstructing activations and gradients tile-by-tile, we can develop a memory-efficient implementation of the convolutional layers in a CNN. This way, a CNN can learn from full contextual information at high resolution, without relying on patches. Ex-

periments show streaming reaches performance on par with or improving on patch-based methods needing more supervision. Thus, streaming enables direct learning from morphology to aid histopathology analysis using readily available slide-level labels.

Thesis overview

Chapter 2 proposes a method called "streaming" to train convolutional neural networks end-to-end on multi-megapixel histopathology images, circumventing memory limitations. We tile the input image and reconstruct activations and gradients, allowing the use of entire high-resolution images during training without cropping.

Chapter 3 applies streaming to train models on whole prostate biopsy images using only slide-level labels from pathology reports. It shows a modern CNN can learn from high-resolution images without patch-level annotations. The method reaches similar performance to state-of-the-art patch-based and multiple instance learning techniques.

Chapter 4 demonstrates a deep learning system to predict the biochemical recurrence of prostate cancer using tissue morphology. Trained on a nested case-control study and validated on an independent cohort, the system finds patterns predictive of recurrence beyond standard Gleason grading. Concept-based explanations show tissue features aligned with pathologist interpretation.

Furthermore, we will show the preliminary results of streaming on a prognostic task in the discussion. In summary, the thesis explores computational pathology methods to analyze entire high-resolution histopathology images despite memory constraints. It shows neural networks can learn from morphology to aid prostate cancer diagnosis and prognosis when trained end-to-end on whole images using readily available slide-level labels.

1. Epstein JI, Egevad L, Amin MB, Delahunt B, Srigley JR, Humphrey PA. The 2014 international society of urological pathology (ISUP) consensus conference on gleason grading of prostatic carcinoma. 2016;40:244-252.
2. Morphology & Grade | SEER Training.
3. Chen XH, Huang S, Kerr D. Biomarkers in clinical medicine. Published online 2011.
4. Cucchiara V, Cooperberg MR, Dall'Era M, et al. Genomic Markers in Prostate Cancer Decision Making. *European Urology*. 2018;73(4):572-582. doi:10.1016/j.eururo.2017.10.036
5. Roest C, Kwee TC, Saha A, Fütterer JJ, Yakar D, Huisman H. AI-assisted biparametric MRI surveillance of prostate cancer: Feasibility study. *European Radiology*. 2023;33(1):89-96. doi:10.1007/s00330-022-09032-7
6. Dunn MW, Kazer MW. Prostate Cancer Overview. *Seminars in Oncology Nursing*. 2011;27(4):241-250. doi:10.1016/j.soncn.2011.07.002

Streaming convolutional neural networks for end-to-end learning with multi-megapixel images

Hans Pinckaers, Bram van Ginneken, Geert Litjens

Introduction

Convolutional neural networks (CNN) are the current state-of-the-art machine learning algorithms for many computer vision tasks, such as classification or segmentation. Ever since Krizhevsky et al. won ImageNet[1] with a CNN[2] in 2012, these networks have become deeper[3] and wider[4] to further improve accuracy. Training these larger networks requires large amounts of computer memory, which increases exponentially with increasing image size. To avoid shortcomings in memory, most natural image datasets in computer vision contain sub-megapixel images: 0.09 megapixel for ImageNet[1] and 0.001 megapixel for CIFAR-10[5]. In several domains such as remote sensing or medical imaging, there is a need for training CNNs with multi-megapixel-sized images – containing both global contextual and local textural information – to obtain accurate models.

Computer memory becomes a limiting factor because the conventional backpropagation algorithm for optimizing deep neural networks requires the storage of intermediate activations. Since the size of these intermediate activations in a convolutional neural network increases proportionally to the input size of the network, memory quickly fills up with images of multiple megapixels. As such, only small CNNs could be trained with these images and state-of-the-art architectures would be out of reach, even on large computing clusters.

In this paper, we propose a novel method to directly train state-of-the-art

convolutional neural networks using any input image size end-to-end. This method exploits the locality of most operations in modern convolutional neural networks by tiling the forward and backward pass in combination with gradient checkpointing. Gradient checkpointing is a technique where instead of keeping all intermediate feature maps in memory to calculate the gradients, we save some specific feature maps (checkpoints). We recalculate the others by performing partial forward passes starting from the saved feature maps during backpropagation, once they are needed for gradient calculation[6]. We first empirically established equivalence between our tile-based approach and an unmodified convolutional neural network on a subset of ImageNet, ImageNette[7]. Then we applied this method to two public datasets: the CAMELYON17 dataset[8] for metastases detection in lymph nodes, and the TUPAC16 dataset[9] for predicting a proliferation score based on gene expression. In both cases, task-specific performance increased with larger input image sizes.

Related work

Several authors have suggested approaches to train convolutional neural networks (CNNs) with large input images while preventing memory bottlenecks. Their methods can be roughly grouped into three categories: (A) altering the dataset, (B) altering usage of the dataset, and (C) altering the network or underlying implementations.

Altering the dataset

If images are too large to fit in the memory of the processing unit, we could downsample the image or divide the image into smaller parts, i.e., patches. The latter approach has been prevalent in both remote sensing and medical imaging[10,11]. However, both approaches have significant drawbacks: the former results in a loss of local details, whereas the latter results in losing global contextual information.

The common approach of training on patches typically involves creating labels for every patch, which can be time- and cost-intensive. It is sometimes not even possible to produce patch-level labels: if a hypothetical task is to predict whether an aerial image shows a city or a village, it is impossible

to create informative labels for individual patches only containing several houses.

Altering usage of the dataset

When we can assume that individual patches contain enough information to predict the image-level label, the classification can be formalized under the classic multiple-instance-learning (MIL) paradigm. In MIL, each image is considered a bag consisting of patches where a positive bag has at least one positive patch and a negative bag none. In the deep learning case, a model is trained in a weakly supervised manner on patches, where the patch with the highest predicted probability is used for backpropagation[12,13]. Other approaches involve taking the average of the patch predictions or a learned weighted average from low-dimensional patch embeddings[14–18].

In this approach, the receptive field of a network is always at most the size of the patch. The model disregards spatial relationships between patches, limiting the incorporation of contextual information.

By first learning to decide which regions should be analyzed at a higher resolution, the problem that a full image cannot be used can also be circumvented[19–22]. Since these methods use a low-resolution version of the image to decide which parts need to be analyzed at a higher resolution, the low-resolution image needs to have enough information to localize the area that needs to be classified. Additionally, for analysis of the selected areas, these methods still use patch-based analysis with the same caveats as mentioned before.

Another way to utilize datasets with large images is proposed by Tellez et al.[23]. To compress the image to a lower-dimensional space, they proposed unsupervised learning. The model is trained patch-by-patch to reconstruct the original patch. An intermediate feature map of the model (i.e., the embedding) can subsequently be used as a lower-dimensional representation per patch. After training, the whole image is compressed patch-by-patch. A model is subsequently trained on these embeddings, having the receptive field of the whole image while requiring less memory.

Since the compression network is trained by reconstruction, the same compression network can be used for different tasks. However, this means that the low-dimensional embedding is not meant for a specific task and may

have compressed away useful information. Our approach involves one network which learns to compress task-relevant information.

Altering the network or underlying implementations

The memory bottleneck can also be circumvented with memory-efficient architectures or memory-efficient implementations of existing architectures. Recently, Gomez et al.[24] published a method to train deep residual neural networks using less memory, termed the Reversible Residual Network. With these networks, some layer activations are recomputed from others on demand, reducing the total memory required. Network architectures can also be altered to utilize cheaper computational operation, such as depthwise separable convolutions[25] or fewer parameters[26]. Our method does not require reducing the number of parameters and works with most types of layers. Another method to reduce memory usage is to recover intermediate activations by doing partial forward passes during backpropagation, termed gradient checkpointing[6]. This method is similar to our approach, but the whole activation feature map of some layers still need to be stored in memory, limiting the use of multi-megapixel images.

Another memory-saving approach is to share memory between tensors with duplicate or recomputable values[27,28], to develop neural networks with reduced precision using half-precision or mixed precision[29], or to swap data between random access memory (RAM) and graphics processing unit (GPU) memory[30]. These methods are usually insufficient for training with large multi-megapixel images; our proposed method can work orthogonally to them.

Methods

To achieve our goal of training CNNs with multi-megapixel images, we significantly reduce the memory requirements. Memory demand is typically highest in the first few layers of state-of-the-art CNNs before several pooling layers are applied because the intermediate activation maps are large. These activation maps require much less memory in subsequent layers. We propose to construct these later activations by streaming the input image through the CNN in a tiled fashion, changing the memory requirement of the CNN

to be based on the size of the tile and not the input image. This method allows the processing of input images of any size.

Several problems arise when trying to reconstruct the later activation map tile-by-tile. Firstly, convolutional layers handle image borders in different ways, either by padding zeros to perform a "same" convolution or by reducing the image size to perform a "valid" convolution. Secondly, in tile-based processing, border effects occur at both the image borders and the tile borders; naive tiling of the input image would thus result in incomplete activation maps and gradients for backpropagation. Lastly, intermediate feature maps of the tiles still need to be stored in memory for backpropagation, which would counteract the streaming of tiles. We solve these problems by developing a principled method to calculate the required tile overlap throughout the network in both the forward and backward pass and by using gradient checkpointing.

We first explain the reconstruction of the intermediate activation map in the forward pass in section 3.1, then describe the backward pass in section 3.2, elaborate on how to calculate the tile overlap in section 3.3, and finish with the limitations of this method in section 3.4. See Figure [figure:streamingSGD] for a graphical representation of the method.

Streaming during the forward pass

Without loss of generality, we explain the method in the discrete one-dimensional case. Let us define $x \in \mathbb{R}^N$ as the one-dimensional real-valued vector with N elements. In discrete one-dimensional space, a "valid" convolution[1] $(*)$ with a kernel with n weights $w \in \mathbb{R}^n$, and stride 1, is defined as:

$$(x * w)_k = \sum_{i=0}^{n} w_i x_{k+i}$$

where $k \in \{0, \dots, f\}$ and $f = N - n$, for any kernel with length $n \leq N$ (for clarity, we will start all indices from 0). Our goal is to decrease the memory

[1]By convention we used the term *convolution* although the mathematical operation implemented in most machine learning frameworks (e.g., TensorFlow, PyTorch) is a cross-correlation.

load of an individual convolution by tiling the input. Following [eq:1], we can achieve the same result as $x * w$, by doing \ two convolutions on the input:

$$a = \{(x * w)_0, \dots, (x * w)_{f//2}\} b = \{(x * w)_{f//2+1}, \dots, (x * w)_f\}$$

where // denotes a divide and floor operation.

By definition of concatenation ($\frown$):

$$\{(x * w)_0, \dots, (x * w)_f\} = a \frown b$$

To ensure that the concatenation of both tiles results in the same output as for the full vector, we need to increase the size of the tiles, resulting $o = n-1$ overlapping values. The values $\{x_0, \dots, x_{f//2+o}\}$ are required to calculate a, and $\{x_{f//2+1-o}, \dots, x_N\}$ for b.

Since the tiles are smaller than the original vector, these separate convolutions require less memory when executed in series. By increasing the number of tiles, memory requirements for individual convolution operations can be reduced even further.

Without loss of generality, the above can also be extended to multiple layers in succession including layers with stride > 1 (e.g., strided convolutions and pooling layers) which are also commonly used in state-of-the-art networks.

When one applies this tiling strategy naively, no memory benefit is obtained as each tile's intermediate activation would still be stored in memory to allow for backpropagation. We use gradient checkpointing to resolve this: We only store the activations after the concatenation of the tiles – where the memory burden is small. This does require recalculation of all intermediate activations for all tiles during backpropagation, but again, only has a memory requirement of processing of a single tile. The trade-off between memory use and re-computation can be controlled through the selection of the concatenation point in the network.

From this point onward, the term *streaming* refers to the tiling of a vector, applying kernel operations, and concatenating the results.

$o \leftarrow []$ $stream\_o \leftarrow \texttt{concat}(o[0..m])$ $pred \leftarrow \texttt{forward}(layers[i..n], stream\_o)$
$loss \leftarrow \texttt{criterion}(pred)$ $g \leftarrow \texttt{backward}(layers[n..i], loss)$ $filled \leftarrow []$

Streaming during backpropagation

The backward pass of multiple convolutions can also be calculated by utilizing the tiles. To start, let us define p as the output after streaming. The derivative of a weight in a convolutional kernel is defined as:

$$\Delta w_j = \sum_{i=0}^{|p|-1} \begin{cases} \Delta p_i x_{i+j}, & \text{if } i-j \geq 0 \text{ and } i-j < |p| \\ 0, & \text{otherwise} \end{cases}$$

where $|\cdot|$ denotes the length of a vector.

While streaming, this sum has to be reconstructed through the summation of the gradients of all tiles, which will result in the same gradient again:

$$\Delta w_j = \sum_{i=0}^{|a|-1} \Delta a_i x_{i+j} + \sum_{i=0}^{|b|-1} \Delta b_i x_{i+j+f//2}$$

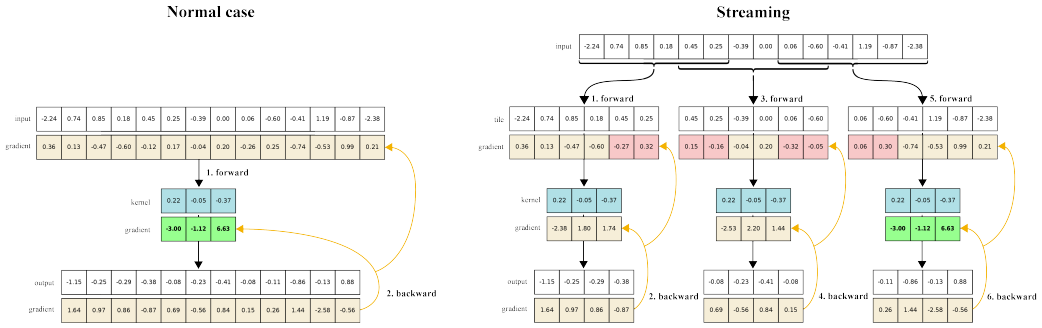
The gradient of the input can be calculated with a similar sum, but then shifted by the kernel size:

$$\Delta x_i = \sum_{j=0}^{n-1} \begin{cases} w_j \Delta p_{i-j}, & \text{if } i-j \geq 0 \text{ and } i-j < |p| \\ 0, & \text{otherwise} \end{cases}$$

This formula is equal to a convolution with a flipped kernel w on Δp padded with $n-1$ zeros (e.g., $flip(w) * [0, 0, \Delta p_1 ... \Delta p_n, 0, 0]$, when $n = 3$), often called a "full" convolution. Thus, analog to the forward pass, the backpropagation can also be streamed.

However, overlapping values of the output p are required when streaming the backpropagation, similar to the overlapping values of the input x required in the forward pass. To generate overlapping values for the output p, the overlap o for the input x needs to be increased to calculate the full Δx.[2]

[2]Zero-padding the tiles before the convolution does not help because these zeros do not exist in the original vector, hereby invalidating the gradients at the border as well.

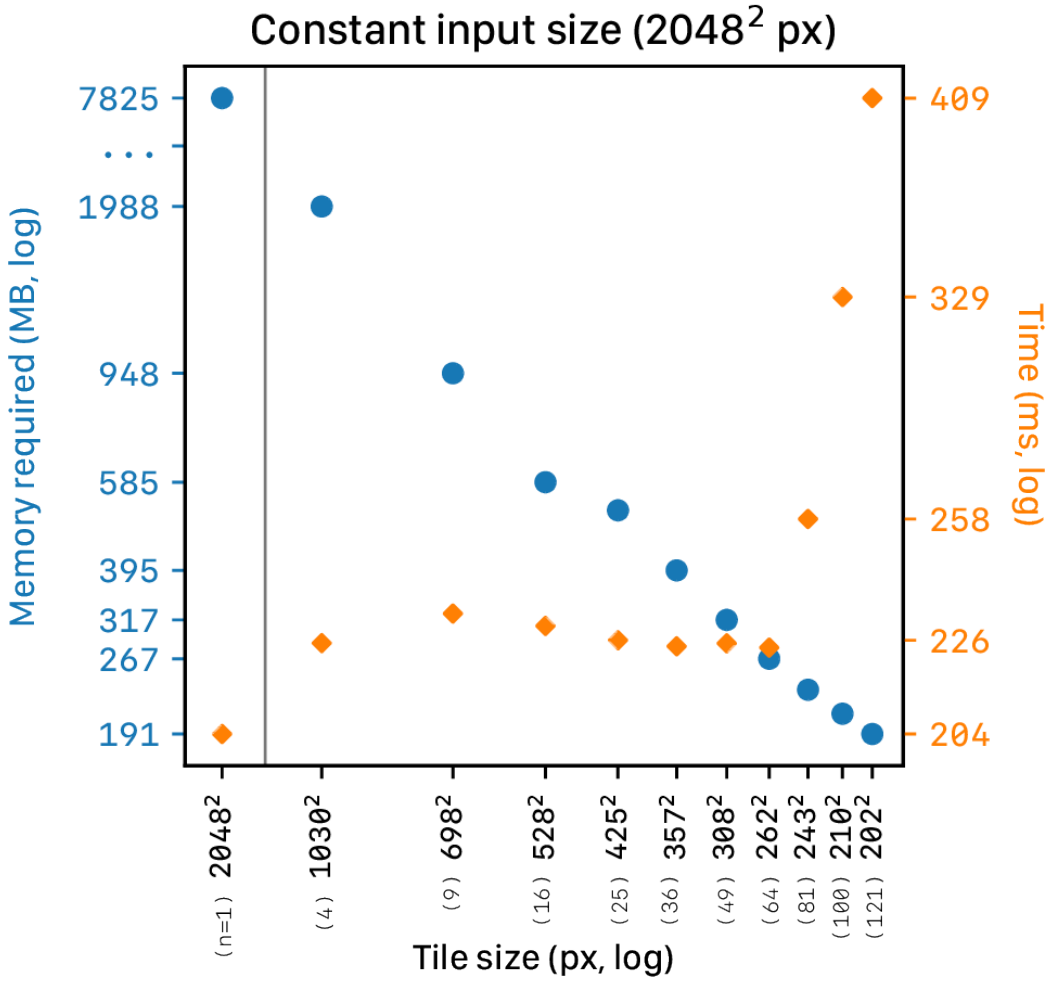


Efficiently calculating required tile overlap for complex architectures

Some recent state-of-the-art networks (e.g., ResNet and DenseNet) contain different paths through the network that sum or concatenate activations from different layers together. These paths make it difficult to manually calculate the required tile overlap for streaming.

To calculate the overlap for such networks, we temporarily replace all convolutional kernel parameters with $\frac{1}{n}$, where n was the length of the kernel. This causes each entry in the convolutional layer's output to be the average of the input image spanned by the convolutional kernel. We then pass an all-ones tile through the network. The required overlap will be the number of non-maximum values in the activation maps and gradients at the border of the tiles, see Algorithm [algorithm:crop].

$output\_stride \leftarrow 1$ $o \leftarrow \texttt{forward}(layers[i..n], o)$ $loss \leftarrow \texttt{criterion}(o)$ $g \leftarrow \texttt{backward}(layers[n..i], loss)$



Limitations

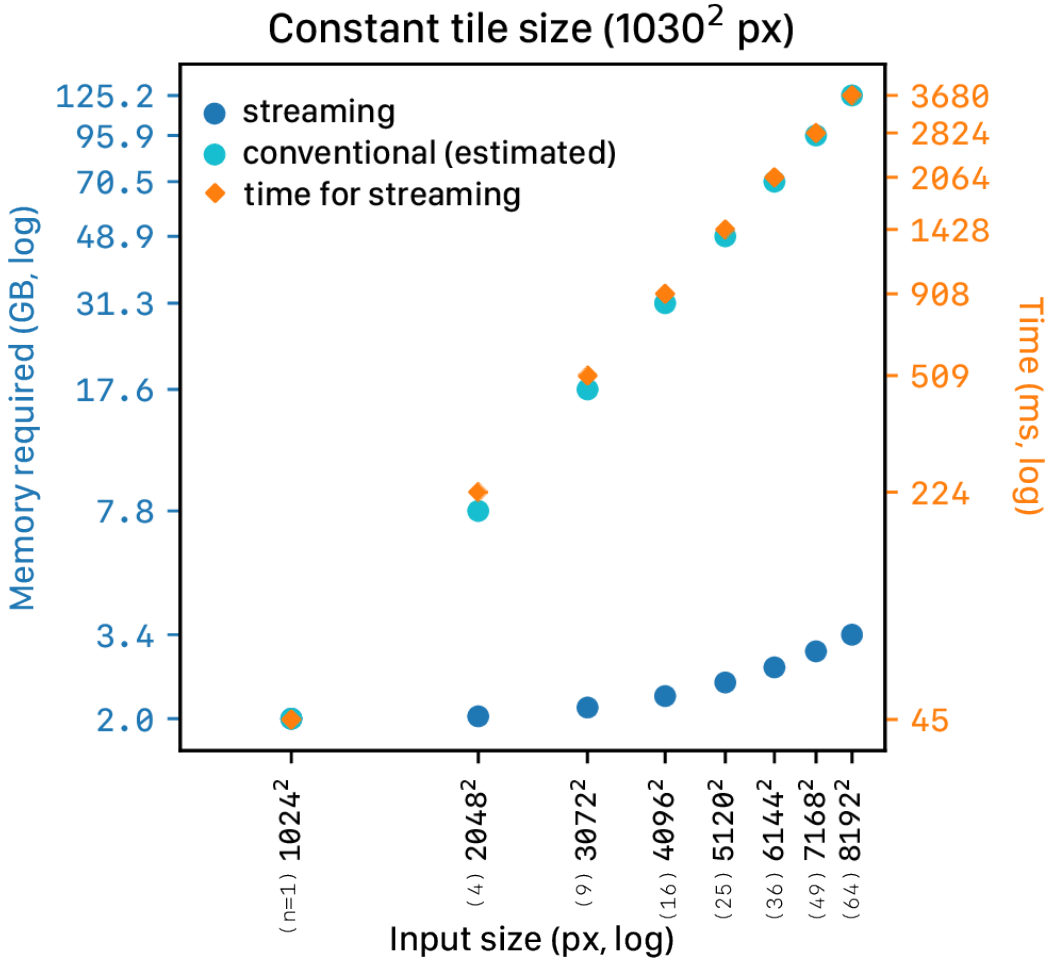
With small tiles, the overlap can be a significant part of the tile, counteracting the memory gains. Since we leverage the method for high-resolution images using large tiles, the memory gains outweigh this overhead.

Furthermore, due to the use of gradient checkpointing, the method will perform multiple forward and backward operations to calculate intermediate activations. This results in longer processing time than it would take

if the image could fit on the GPU (see Fig. [fig:constant_input] and [fig:constant_tile]). The processing time increases almost linearly with input size plus overlap.

For a network to be able to use this method, the intermediate feature maps and its gradients have to be able to fit on the GPU at a certain point. However, choosing a layer too deep into the network will require a lot of overlapping calculations, being less efficient. As such, choosing which layers to stream can be difficult. We suggest splitting the network and experimenting with random input to the final non-streaming layers to test if backpropagation fits on the GPU. Then, streaming the first layers with a tile size as large as possible.

Finally, since the method relies on the local properties of convolutions and pooling operations, trying to use other operations that break this locality will result in invalid results (e.g., operations that rely on all the feature map values such as BatchNormalization[31]). However, these operations can be used as soon as the whole feature map is reconstructed, after streaming, in the final part of the network.



Evaluation

We evaluated the streaming method with three different datasets and network architectures. First, in Section 5, we evaluated whether a CNN using streaming trains equivalently to the conventional training. Second, in Section 6, we evaluated the usage of streaming on a regression task in the public TUPAC16[9] dataset with high-resolution images (multiple gigapixels) and only image-level labels. We trained multiple networks using increasing image resolutions and network depth. Finally, in Section 7, we evaluated stream-

ing in a classification task using the image-level labels of the CAMELYON17 dataset[32].

An open-source implementation of the streaming algorithm and the ImageNette experiments can be found at https://github.com/DIAGNijmegen/StreamingCNN.

Experiments on ImageNette

We trained a CNN on small images using streaming and conventional training starting from the same initialization. We used a subset of the ImageNet dataset, ImageNette, using 10 ImageNet classes (tench, English springer, cassette player, chain saw, church, French horn, garbage truck, gas pump, golf ball, parachute), analog to[7].

Table 1: Network architecture for Imagenette experiment

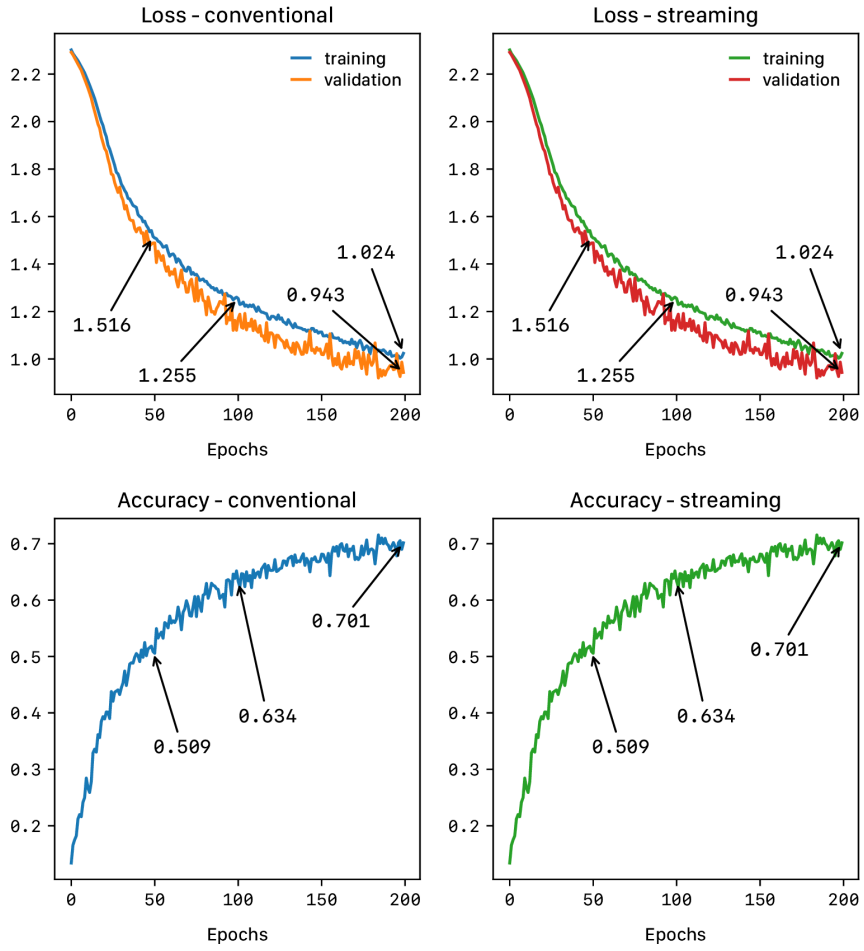
Layers	**Kernel size**	**Channels**
2D convolution	3x3	16
2D max-pool	2x2	16
2D convolution	3x3	32
2D max-pool	2x2	32
2D convolution	3x3	64
2D max-pool	2x2	64
2D convolution	3x3	128
2D max-pool	2x2	128
2D convolution	3x3	256
2D max-pool	10x10	256
Fully connected	10	

Data preparation

We used data augmentation for the training set following Szegedy et al.[33]. Patches of varying sizes were sampled from the image, distributed evenly between 8% and 100% of the image area with aspect ratio constrained to the interval $[\frac{3}{4}, \frac{4}{3}]$. For the tuning set, we sampled 320×320 patches from the center of the image.

Network architecture and training scheme

The CNN consisted of five blocks of a convolutional layer followed by a max-pool layer (see Table 1). The network was optimized for 200 epochs with stochastic gradient descent, using a learning rate of 1×10^{-3}, a mini-batch size of 32 images and weight decay 1×10^{-6}. For the streaming method, the first four layers were streamed with tiles of 32×32 pixels. The network was initialized according to He et al., 2015[34].



Results on Imagenette

The loss curves of both methods (Figure [figure:imagenette_exp]) were nearly identical, which empirically shows that training with streaming performed equivalently to conventional training. Small differences are likely due to losses of significance in floating point arithmetic; these differences accumulate during training and lead to small differences in loss values in later epochs.

Experiments on TUPAC16 dataset

To evaluate our method on a real-world task, we used the publicly available dataset of the TUPAC16 challenge[9]. This dataset consists of 500 hematoxylin and eosin (H&E) stained whole-slide images (WSI) from breast adenocarcinoma patients. The WSIs of these patients are available from The Cancer Genome Atlas[35] together with RNA expression profiles. The expression of 11 proliferation-associated genes was combined to create one objective measure for tumor growth, termed the PAM50 score[36]. This score has no known visual substrate in the images. Thus, manual labeling is considered impossible. We set aside 98 WSIs at random for tuning the algorithm and used the remaining slides for training. Additionally, an independent evaluation was performed by the challenge organizers on the test set of 321 WSIs, of which the public ground truth is not available. The submitted predictions were evaluated using Spearman's rank-order correlation between the prediction and the ground truth.

Table 2: Network architecture for TUPAC16 experiments.

Layers	**Kernel**	**Channels**	**Details**
2x 2D convolution	3x3	32	
2D max-pool	2x2	32	
2x 2D convolution	3x3	64	
2D max-pool	2x2	64	
2x 2D convolution	3x3	128	
2D max-pool	2x2	128	
2x 2D convolution	3x3	256	
2D max-pool	2x2	256	
2x 2D convolution	3x3	512	repeated for

2D max-pool	2x2	512	field of view experiment
2x 2D convolution	3x3	512	with BatchNormalization
2D max-pool	2x2	512	
2x 2D convolution	3x3	512	with BatchNormalization
2D max-pool	input size	512	
Dropout (p=0.5)		512	
Fully connected			continuous output, without non-linearity

To evaluate whether CNN models can leverage and use the higher resolution information that streaming makes possible, we performed two sets of experiments to end-to-end predict the PAM50 score. For one, we trained the same model with various image sizes (1024×1024, 2048×2048, and 4096×4096 pixels), thus increasing input image resolution. Different networks were trained in the second set, where the depth was increased with image size (22, 25, and 28 layers for respectively 2048×2048, 4096×2096, and 8192×8192). By also increasing the depth, the physical receptive field size before the last max-pool layer is kept constant (see Table 2). All networks were trained until convergence; the checkpoint with the highest Spearman's correlation coefficient on the tuning set was submitted for independent evaluation on the test set.

Data preparation

The images were extracted from the WSIs at image spacing $16.0\mu m$ for the 1024×1024 experiments, $8.0\mu m$ for 2048×2048, etc. (see Figure [fig:resolutions]). Background regions were cropped, and the resulting image was either randomly cropped or zero-padded to the predefined input size.

Since the challenge consists of a limited number of slides, we applied extensive data augmentations to increase the sample size (random rotations; random horizontal or vertical flipping; random brightness, contrast, saturation, and hue shifts; elastic transformations; and cutout[37]). For all experiments, the same hyperparameters and data preprocessing were used.

Network architecture and training scheme

The networks (see Table 2) were trained using the Adam optimizer[38] with a learning rate of 1×10^{-4}, with the default β parameters of $\beta_1 = 0.9$, $\beta_2 = 0.999$. We applied exponential decay to the learning rate of 0.99 per epoch. As an objective, we used the Huber loss with $\Delta = 1$, also called the smooth L1 loss[39]. The mini-batch size was 16 images. A dropout layer with $p = 0.5$ was inserted before the final classification layer. The networks were initialized following He et al., 2015[34]. The images were normalized using the mean and standard deviation values of the whole training set.

Streaming was applied until the final seven layers. Since BatchNormalization breaks the local properties of chained convolutional and pooling layers, it was only used in the last part of the network. Analysis of Santurkar et al.[40] suggests that adding only a few BatchNormalization layers towards the end of the network smooths the loss function significantly and helps optimization.

Results on TUPAC16

The task was evaluated using Spearman's correlation coefficient between the prediction and the ground truth PAM50 proliferation scores. In both experiments, an improvement of the metric was seen with increasing input sizes.

Table 3: TUPAC16: performance of the models on the independent test test, Spearman's rho correlation coefficient

Experiment	**Input size**	**Test set performa**
Equal number of parameters	1024x1024	0.485 (0.441-0.527)
	2048x2048	0.491 (0.448-0.533)
	4096x4096	0.536 (0.495-0.575)
Equal field of view before global max-pool (increasing depth)	2048x2048	0.491 (0.448-0.533)
	4096x4096	0.570 (0.531-0.606)
	8192x8192	0.560 (0.520-0.597)

The result of the network with the input image resolution of 4096×4096 approached state-of-the-art for image-level regression with a score of 0.570. Note that the first entry of the leaderboard used an additional set of manual

annotations of mitotic figures and is therefore not directly comparable to our experiments.

Table 4: *method uses additional detailed annotations from another task in the challenge and does not train a single model to predict from slide to PAM50 score.

Experiment	**Corr. coefficient**
Lunit Inc., South Korea[9,41]	0.617*
Ours (4096x4096)	**0.570**
Ours (8192x8192)	**0.560**
Tellez et al., 2019[23]	0.557
Radboud UMC Nijmegen, The Netherlands[9]	0.516
Contextvision, Sweden[9]	0.503
Belarus National Academy of Sciences[9]	0.494
The Harker School, United States[9]	0.474

Experiments on CAMELYON17 dataset

CAMELYON17 was used to evaluate the streaming method on a classification task[32]. CAMELYON17 is a large public dataset and challenge to detect metastases of adenocarcinoma in breast tissue. The dataset consists of 500 labelled WSIs and 500 unlabeled WSIs, which were respectively used as the training and test sets. In the training set, for 450 slides image-level labels were provided, while for the remaining 50 slides dense annotations (precise delineation of the metastases) were supplied. The slides were collected from five different hospitals. The challenge differentiates three clinical relevant metastases types: macro-metastases ($>$ 2 mm), micro-metastases ($\leq$ 2.0 mm or $>$ 200 cells in a single cross-section), and isolated tumor cells ($\leq$ 0.2 mm or $<$ 200 cells in a single cross-section). We evaluate the slide level classification performance with multiple ROC analyses (metastasis-type vs. the negative class, and negative versus all positive classes).

Data preparation for this experiment was equal to the TUPAC16 challenge. We picked 90 WSIs of the challenge training set at random to be used as our tuning set.

Confidence intervals were obtained through bootstrapping of the test set, ensuring the same sampling across the different resolutions. Furthermore, we performed a permutation test to assess statistical significance.

Network architecture and training scheme

We used the same training schedule and underlying architecture as the TUPAC16 experiments. We altered the architecture by disabling dropout, and to reduce problems with exploding gradients in the beginning of the network, we replaced BatchNormalization with weight decay of 1×10^{-6} and layer-sequential unit-variance (LSUV) initialization[42]. We applied the LSUV scaling per kernel channel[43]. The mean and standard deviation per layer activation were calculated over ten mini-batches by keeping track of the sum and squares of the channels per tile during streaming; the reformulation of variance as $\mathbb{E}[X^2] - \mu^2$ was used to calculate the full standard deviation of ten mini-batches before applying LSUV.

Input	**Negative**	**Isolated tumor cells**	**Micro**
	n=260	n=35	n=83
2048^2	0.580 (0.529-0.628)	0.450 (0.363-0.539)	0.689 (
4096^2	0.648 (0.601-0.696, p_1=0.03)	**0.533** (0.422-0.642, p_1=0.13)	0.669 (
8192^2	**0.706** (0.660-0.751, p_1<0.001, p_2=0.06)	0.463 (0.359-0.569, p_1=0.43, p_2=0.83)	**0.709**

Results on CAMELYON17

The network trained with 8192×8192 images is significantly better than the models trained with 4096×4096 images in the discriminating macrometastases from negative cases, and significantly better than the 2048×2048 model in discriminating negative cases from cases with any metastasis.

Saliency maps

Saliency maps were created for the networks trained with the largest resolution (8192×8192 pixels) according to Simonyan et al.[44]. For better visualization on lower resolution, a Gaussian blur was applied with $\sigma = 50$ for 8192×8192 network, $\sigma = 25$ for the 4096×4096 models, and $\sigma = 12, 5$ for the 2048×2048 models. Since a few gradient values can be significantly higher

than others, we capped the upper gradient values at the 99$^{\text{th}}$ percentile[45]. The upper 50$^{\text{th}}$ percentile was overlayed on top of the original image (See Figure 2).

Discussion and conclusion

We presented a novel streaming method to train CNNs with tiled inputs, allowing inputs of arbitrary size. We showed that the reconstructed gradients of the neural network weights using tiles were equivalent to those obtained with non-tiled inputs.

In the first experiment on ImageNette, we empirically showed that the training behavior of our proposed streaming method was similar to the behavior in the non-streaming case. Small differences occur later in training due to loss of significance in floating-point arithmetic. These differences accumulated during training and lead to the small difference in loss values in later epochs. However, they do not seem to harm performance. Most modern frameworks have similar problems due to their use of non-deterministic operations.

The second and third experiments showed that our streaming method can train CNNs with multi-megapixel images that, due to memory requirements in the non-streaming case, would not be able to fit on current hardware. When trained using the conventional method, without streaming, the experiment with the highest-resolution images (8192×8192 pixels) would require ~50 gigabytes per image, summing up to ~825 gigabytes of memory per mini-batch.

Results on the TUPAC16 dataset (Table 3) showed an increasing correlation between the prediction and the proliferation score with increasing input sizes. Our 4096×4096 pixel network performed best. A jump in performance from 0.491 to 0.570 was seen from 2048×2048 to 4096×4096 pixels, respectively. We hypothesize that this is because tumor tissue can be discriminated from other types of tissue at these higher resolutions. However, an 8192×8192 pixel input size did not further improve the performance on the test set, although the difference is minor, and the confidence interval is quite wide and overlapping. The nuclear details of cells at this resolution remain vague, which suggests that most of the information is still obtained from the morphology like in 4096×4096 images. Higher resolutions may be necessary to

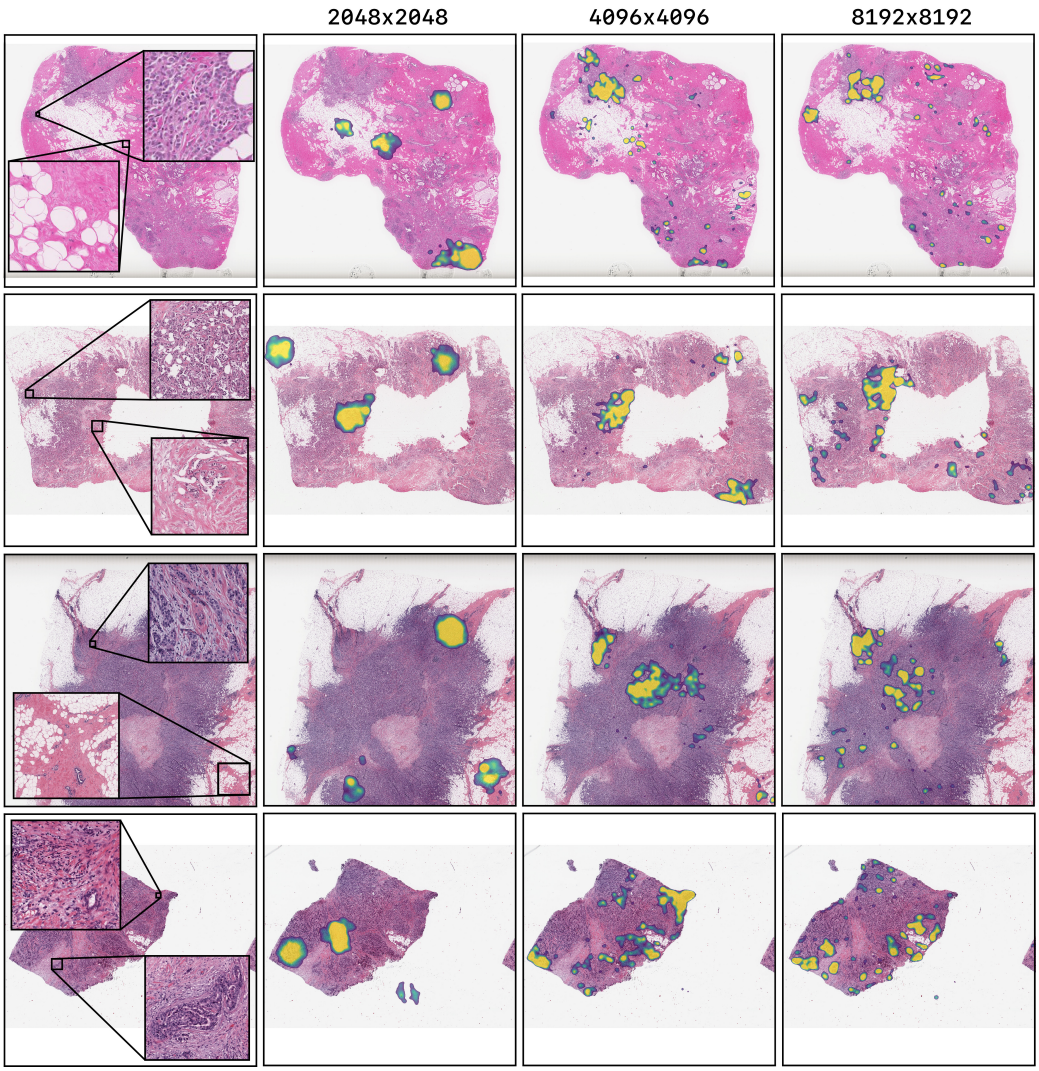


Figure 1: Saliency maps for test set images of the TUPAC16 experiment using the best performing models. The TUPAC16 network shows highlights in cell-dense and cancerous regions. There is a trend in which the higher the input solution of the model, the less it focuses on healthy tissue. Also, higher resolution models focus on more locations of the tissue.

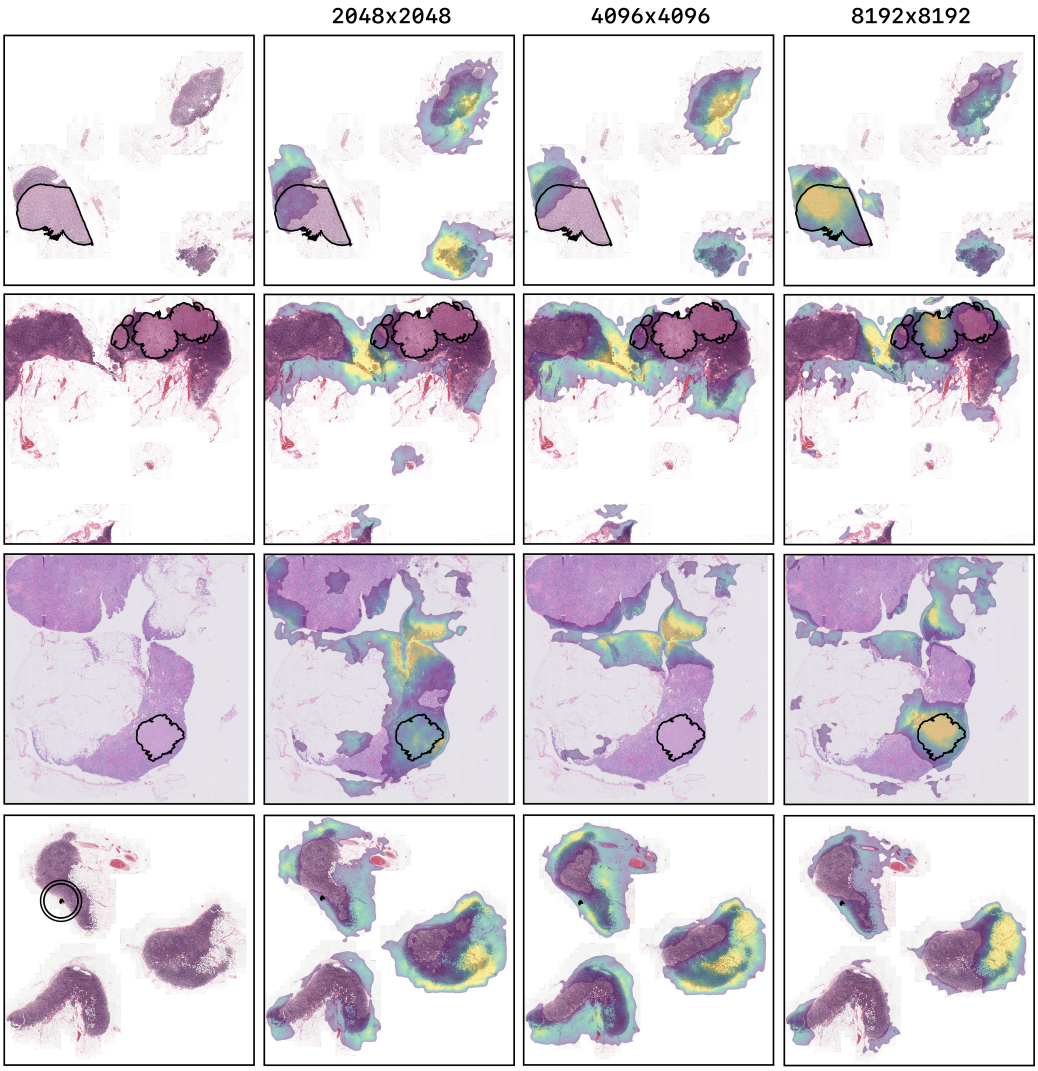
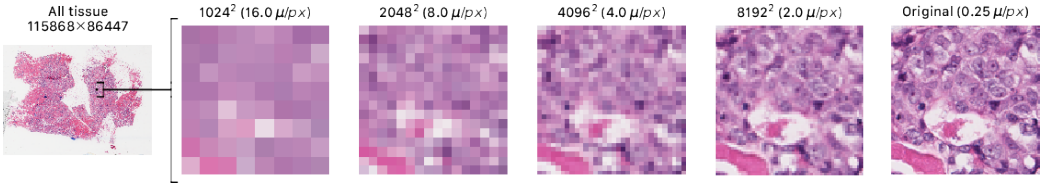


Figure 2: Saliency maps for images of the tuning set of the CAMELYON17 experiment. The highest resolution model, trained on image-level labels shows highlights corresponding to the ground truth pixel-level annotation of a breast cancer metastasis. The lower resolution models have lower probability for the ground truth class and show little correspondence to the location of the metastases. The last row shows a micro metastasis for which models failed to recognize.

further improve performance, although we may also have hit the ceiling for the performance of this network architecture, training setup, and data. Another explanation for the lack of improvement is the increasing difficulty for the network to find the sparse information in just 400 slides using a single label or a misrepresented tuning set due to the small provided training set. As such, it is likely that for some tasks and datasets, higher-resolutions are not beneficial. Our best result on TUPAC16 approached that of the challenge winner, who used task-specific information (a network trained on mitosis detection) instead of a pure regression of one label per WSI. Our method outperformed all other methods in the challenge.

Results on the CAMELYON17 dataset show improvement with increasing resolution. An exception occurs for the isolated tumor cells class; even at the highest resolution applied, the CNN was unable to differentiate isolated tumor cells. To accurately identify lesions of that size, the resolution would probably need to be increased by at least a factor of four. Furthermore, this class is also underrepresented (n=31) in the provided training set. The 8192×8192 network was significantly better than 4096×4096 and 2048×2048 in the discriminating macro-metastases from negative cases and significantly better than 2048×2048 in discriminating negative cases from cases with any metastasis.



Using saliency maps, we visualized what the models would change on the input to make it more closely resemble the assigned class. These maps show us which parts of the image the model takes into consideration[44]. Saliency maps of our CNNs trained on higher resolutions suggest that the networks learn the relevant features of the high-resolution images (see Figure 2). The image-level trained CAMELYON17 network shows highlights corresponding to the ground truth pixel-level annotation of a breast cancer metastasis. The TUPAC16 network shows highlights in cell-dense regions.

The streaming method has advantages over prior work on this topic. For streaming, we do not need to alter the dataset by resizing or creating additional pixel-level labels (which is sometimes not possible). Also, we do not

need to change the usage of the dataset like in the MIL paradigm or use compression techniques. Finally, we are not limited to specific architectural choices for our network, such as in RevNet; streaming can be applied to any state-of-the-art network, such as Inception or DenseNet.

While increasing input sizes and resolutions are beneficial in various tasks, there are some drawbacks. A limitation is the increase in computation time with increasing input sizes (Fig. [fig:constant_tile]). This can be partially counteracted by dividing the batch over multiple GPUs. Due to this limitation, we did not increase resolution further in our experiments. Future research could attempt to speed up computation on the tiles, e.g., by training with mixed precision[29] or depthwise separable convolutions[25]. One could also try to start with a pre-trained network (e.g., on ImageNet) and fine-tune for a shorter period.

Another limitation is the inability to use feature map-wide operations in the streaming part of the network, e.g., BatchNormalization. In this work, we replaced some benefits of BatchNormalization, namely the robustness against bad initialization and the regularization, with LSUV initialization and weight decay. Future work could focus on normalization techniques that retain the local properties of the relation between the output and input of the streaming part of the network, e.g., weight normalization[46].

Although this approach can, in theory, be used for segmentation, streaming a segmentation network, such as U-Net[47], will require some engineering. We would have to stream the encoder, "checkpointing" and reconstructing feature maps at the final convolution of every level, for the skip connections. Then, we would have to stream the decoding separately and carefully supply the spatially correct regions of the checkpointed skip connections to each tile. Equally for the backpropagation, reconstructing the gradients at the beginning of each decode level to make the skip connections work. This would be the case for a regular U-Net, if we would add more levels to U-Net, you will have to train middle layers without streaming, as the field of view of these layers could be too high (requiring a big overlap, making streaming less memory efficient).

Improving the performance of the high-resolution-trained networks could be a research topic of interest. In the TUPAC16 and CAMELYON17 experiments, we increased depth as we increased the input size. However, a recent work[26] – though on a maximum 480×480 image size – suggests a "compound" scaling

rule in which the input resolution is scaled together with depth and width of the network.

This paper focused on streaming two-dimensional images, but since convolutions over higher-dimensional data have the same local properties, one could leverage the same technique for, for example, 3D volumetric radiological images.

Acknowledgements

The authors would like to thank Erdi Çallı for his help in proofreading the equations.

1. Russakovsky O, Deng J, Su H, et al. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision*. 2015;115(3):211-252. doi:10.1007/s11263-015-0816-y
2. Krizhevsky A, Sutskever I, Hinton GE. ImageNet Classification with Deep Convolutional Neural Networks. In: *Proceedings of the 25th Advances in Neural Information Processing Systems.*; 2012. http://code.google.com/p/cuda-convnet/ http://papers.nips.cc/paper/4824-imagenet-classification-w%5Cnpapers3://publication/uuid/1ECF396A-CEDA-45CD-9A9F-03344449DA2A
3. He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. In: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. Vol 2016. IEEE Computer Society; 2016:770-778. doi:10.1109/CVPR.2016.90
4. Zagoruyko S, Komodakis N. Wide Residual Networks. Published online May 2016. http://arxiv.org/abs/1605.07146
5. Krizhevsky A. *Learning Multiple Layers of Features from Tiny Images*. University of Toronto; 2009.
6. Chen T, Xu B, Zhang C, Guestrin C. Training Deep Nets with Sublinear Memory Cost. *arXiv preprint*. 2016;1604.06174. http://arxiv.org/abs/1604.06174
7. Jeremy Howard. Imagenette: a smaller subset of 10 easily classified classes from Imagenet, and a little more French. https://github.com/fastai/imagenette

8. Litjens G, Bandi P, Bejnordi BE, et al. 1399 H&E-stained sentinel lymph node sections of breast cancer patients: The CAMELYON dataset. 2018;7. doi:10.1093/gigascience/giy065

9. Veta M, Heng YJ, Stathonikos N, et al. Predicting breast tumor proliferation from whole-slide images: The TUPAC16 challenge. *Medical Image Analysis*. 2019;54:111-121. doi:10.1016/j.media.2019.02.012

10. Ma L, Liu Y, Zhang X, Ye Y, Yin G, Johnson BA. Deep learning in remote sensing applications: A meta-analysis and review. 2019;152:166-177. doi:10.1016/j.isprsjprs.2019.04.015

11. Litjens G, Kooi T, Bejnordi BE, et al. A survey on deep learning in medical image analysis. 2017;42:60-88. doi:10.1016/j.media.2017.07.005

12. Campanella G, Hanna MG, Geneslaw L, et al. Clinical-grade computational pathology using weakly supervised deep learning on whole slide images. *Nature Medicine*. 2019;25(8):1301-1309. doi:10.1038/s41591-019-0508-1

13. Courtiol P, Tramel EW, Sanselme M, Wainrib G. Classification and Disease Localization in Histopathology Using Only Global Labels: A Weakly-Supervised Approach. *arXiv preprint*. 2018;1802.02212. http://arxiv.org/abs/1802.02212

14. Quellec G, Cazuguel G, Cochener B, Lamard M. Multiple-Instance Learning for Medical Image and Video Analysis. *IEEE Reviews in Biomedical Engineering*. 2017;10:213-234. doi:10.1109/RBME.2017.2651164

15. Ilse M, Tomczak JM, Welling M. Attention-based Deep Multiple Instance Learning. *arXiv preprint*. 2018;1802.04712. http://arxiv.org/abs/1802.04712

16. Couture HD, Marron JS, Perou CM, Troester MA, Niethammer M. Multiple Instance Learning for Heterogeneous Images: Training a CNN for Histopathology. In: Frangi AF, Schnabel JA, Davatzikos C, Alberola-López C, Fichtinger G, eds. *Medical Image Computing and Computer Assisted Intervention – MICCAI 2018*. Springer International Publishing; 2018:254-262.

17. Ianni JD, Soans RE, Sankarapandian S, et al. Tailored for Real-World: A Whole Slide Image Classification System Validated on Uncurated Multi-Site Data Emulating the Prospective Pathology Workload. *Scientific Reports*. 2020;10(1):3217. doi:10.1038/s41598-020-59985-2

18. Hou L, Samaras D, Kurc TM, Gao Y, Davis JE, Saltz JH. Patch-Based Convolutional Neural Network for Whole Slide Tissue Image Classification. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.; 2016:2424-2433. doi:10.1109/CVPR.2016.266

19. Dong N, Kampffmeyer M, Liang X, Wang Z, Dai W, Xing EP. Reinforced Auto-Zoom Net: Towards Accurate and Fast Breast Cancer Segmentation in Whole-slide Images. *CoRR*. 2018;abs/1807.1. http://arxiv.org/abs/1807.11113

20. Mnih V, Heess N, Graves A, Kavukcuoglu K. Recurrent Models of Visual Attention. In: *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*. MIT Press; 2014:2204-2212. http://dl.acm.org/citation.cfm?id=2969033.2969073

21. Katharopoulos A, Fleuret F. Processing Megapixel Images with Deep Attention-Sampling Models. In: *Proceedings of the 36th International Conference on Machine Learning*.; 2019. http://arxiv.org/abs/1905.03711

22. Recasens A, Kellnhofer P, Stent S, Matusik W, Torralba A. Learning to Zoom: a Saliency-Based Sampling Layer for Neural Networks. In: *European Conference on Computer Vision*.; 2018. http://arxiv.org/abs/1809.03355

23. Tellez D, Litjens G, Laak J van der, Ciompi F. Neural Image Compression for Gigapixel Histopathology Image Analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. in press. doi:10.1109/TPAMI.2019.2936841

24. Gomez AN, Ren M, Urtasun R, Grosse RB. The Reversible Residual Network: Backpropagation Without Storing Activations. In: *Proceedings of the 31st International Conference on Neural Information Processing Systems*.; 2017:2211-2221. http://arxiv.org/abs/1707.04585

25. Chollet F. Xception: Deep Learning with Depthwise Separable Convolutions. In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.; 2017:1800-1807. doi:10.1109/CVPR.2017.195

26. Tan M, Le QV. EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. In: *Proceedings of the 36th International Conference on Machine Learning.*; 2019. http://arxiv.org/abs/1905.11946

27. Huang G, Liu Z, Van Der Maaten L, Weinberger KQ. Densely connected convolutional networks. In: *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition.* Institute of Electrical; Electronics Engineers Inc.; 2017:2261-2269. doi:10.1109/CVPR.2017.243

28. Bulo SR, Porzi L, Kontschieder P. In-place Activated BatchNorm for Memory-Optimized Training of DNNs. In: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition.*; 2018. doi:10.1109/CVPR.2018.00591

29. Vanhoucke V, Senior A, Mao MZ. Improving the speed of neural networks on CPUs. In: *Deep Learning and Unsupervised Feature Learning Workshop, NIPS 2011.*; 2011.

30. Zhang J, Yeung SH, Shu Y, He B, Wang W. Efficient Memory Management for GPU-based Deep Learning Systems. Published online February 2019. http://arxiv.org/abs/1903.06631

31. Ioffe S, Szegedy C. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In: *Proceedings of the 32Nd International Conference on International Conference on Machine Learning - Volume 37.* ICML'15. JMLR.org; 2015:448-456. http://dl.acm.org/citation.cfm?id=3045118.3045167

32. Bándi P, Geessink O, Manson Q, et al. From Detection of Individual Metastases to Classification of Lymph Node Status at the Patient Level: The CAMELYON17 Challenge. *IEEE Transactions on Medical Imaging.* 2019;38(2):550-560. doi:10.1109/TMI.2018.2867350

33. Szegedy C, Wei Liu, Yangqing Jia, et al. Going deeper with convolutions. In: *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR).*; 2015:1-9. doi:10.1109/CVPR.2015.7298594

34. He K, Zhang X, Ren S, Sun J. Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification. In: *Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV).* IEEE Computer Society; 2015:1026-1034. doi:10.1109/ICCV.2015.123

35. Weinstein JN, Collisson EA, Mills GB, et al. The Cancer Genome Atlas Pan-Cancer analysis project. *Nature Genetics*. 2013;45(10):1113-1120. doi:10.1038/ng.2764

36. Nielsen TO, Parker JS, Leung S, et al. A Comparison of PAM50 Intrinsic Subtyping with Immunohistochemistry and Clinical Prognostic Factors in Tamoxifen-Treated Estrogen Receptor–Positive Breast Cancer. *Clinical Cancer Research*. 2010;16(21):5222-5232. doi:10.1158/1078-0432.CCR-10-1282

37. DeVries T, Taylor GW. Improved Regularization of Convolutional Neural Networks with Cutout. *CoRR*. 2017;abs/1708.0. http://arxiv.org/abs/1708.04552

38. Kingma DP, Ba J. Adam: A method for stochastic optimization. In: *International Conference on Learning Representations (ICLR)*.; 2015.

39. Girshick R. Fast R-CNN. In: *Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV)*. ICCV '15. IEEE Computer Society; 2015:1440-1448. doi:10.1109/ICCV.2015.169

40. Santurkar S, Tsipras D, Ilyas A, Madry A. How does batch normalization help optimization? In: *Advances in Neural Information Processing Systems*. Vol 2018-Decem.; 2018:2483-2493. https://papers.nips.cc/paper/7515-how-does-batch-normalization-help-optimization.pdf

41. Paeng K, Hwang S, Park S, Kim M. A Unified Framework for Tumor Proliferation Score Prediction in Breast Histopathology. *CoRR*. 2016;abs/1612.0. http://arxiv.org/abs/1612.07180

42. Mishkin D, Matas J. All you need is a good init. In: *International Conference on Learning Representations (ICLR)*.; 2016. http://arxiv.org/abs/1511.06422

43. Krähenbühl P, Doersch C, Donahue J, Darrell T. Data-dependent Initializations of Convolutional Neural Networks. In: *International Conference on Learning Representations (ICLR)*.; 2016. http://arxiv.org/abs/1511.06856

44. Simonyan K, Vedaldi A, Zisserman A. Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps. *CoRR*. Published online December 2013. http://arxiv.org/abs/1312.6034

45. Smilkov D, Thorat N, Kim B, Viégas FB, Wattenberg M. SmoothGrad: removing noise by adding noise. In: *ICML Workshop on Visualization for Deep Learning.*; 2017. http://arxiv.org/abs/1706.03825

46. Salimans T, Kingma DP. Weight normalization: A simple reparameterization to accelerate training of deep neural networks. In: *Advances in Neural Information Processing Systems.*; 2016.

47. Ronneberger O, Fischer P, Brox T. U-net: Convolutional networks for biomedical image segmentation. In: *International Conference on Medical Image Computing and Computer-Assisted Intervention.* Springer; 2015:234-241.

Detection of prostate cancer in whole-slide images through end-to-end training with image-level labels

Hans Pinckaers, Wouter Bulten, Jeroen van der Laak, Geert Litjens *†

Abstract

Prostate cancer is the most prevalent cancer among men in Western countries, with 1.1 million new diagnoses every year. The gold standard for the diagnosis of prostate cancer is a pathologists' evaluation of prostate tissue.

To potentially assist pathologists deep-learning-based cancer detection systems have been developed. Many of the state-of-the-art models are patch-based convolutional neural networks, as the use of entire scanned slides is hampered by memory limitations on accelerator cards. Patch-based systems typically require detailed, pixel-level annotations for effective training. However, such annotations are seldom readily available, in contrast to the clinical reports of pathologists, which contain slide-level labels. As such, developing algorithms which do not require manual pixel-wise annotations, but can learn using only the clinical report would be a significant advancement for the field.

In this paper, we propose to use a streaming implementation of convolutional layers, to train a modern CNN (ResNet-34) with 21 million

*Manuscript submitted on June 6, 2020. This work was supported by the Dutch Cancer Society under Grant KUN 2015-7970.

†Hans Pinckaers, Wouter Bulten, Jeroen van der Laak and Geert Litjens are with the Computational Pathology Group, Department of Pathology, Radboud Institute for Health Sciences, Radboud University Medical Center, Nijmegen, The Netherlands; E-mail: {hans.pinckaers, wouter.bulten, jeroen.vanderlaak, geert.litjens}@radboudumc.nl. Additionally, Jeroen van der Laak is with the Center for Medical Image Science and Visualization, Linköping University, Linköping, Sweden.

parameters end-to-end on 4712 prostate biopsies. The method enables the use of entire biopsy images at high-resolution directly by reducing the GPU memory requirements by 2.4 TB. We show that modern CNNs, trained using our streaming approach, can extract meaningful features from high-resolution images without additional heuristics, reaching similar performance as state-of-the-art patch-based and multiple-instance learning methods. By circumventing the need for manual annotations, this approach can function as a blueprint for other tasks in histopathological diagnosis.

The source code to reproduce the streaming models is available at https://github.com/DIAGNijmegen/pathology-streaming-pipeline.

Introduction

The current state-of-the-art in computer vision for image classification tasks are convolutional neural networks (CNNs). Commonly, convolutional neural networks are developed with low-resolution labeled images, for example 0.001 megapixels for CIFAR-10[1], and 0.09-0.26 megapixels for ImageNet[2]. These images are evaluated by the network and the parameters are optimized with stochastic gradient descent by backpropagating the classification error. Neural networks learn to extract relevant features from their input. To effectively learn relevant features, optimizing these networks requires relatively large datasets[3].

In histopathology, due to the gigapixel size of scanned samples, generally referred to as whole-slide images (WSIs), the memory limitation of current accelerator cards prohibits training on the entire image, in contrast to most of the natural images used in general computer vision tasks. As such, most networks are trained on tiny patches from the whole-slide image. Acquiring labels for these patches can be expensive. They are generally based on detailed outlines of the classes (e.g., tumor regions) by an experienced pathologist. This outlining is not done in clinical practice, and is a tedious and time-consuming task. This limits the dataset size for training models. Also, we will need to create these annotations for every individual task.

Besides time constraints, the diagnosis also suffers from substantial inter-observer and intra-observer variability[4]. For prostate cancer, pathologists report the Gleason grading scheme[5]. Prognostically interesting growth patterns are categorized, resulting in three levels of aggressiveness. When cancer

is present, the reports will mention a Gleason score, a combination of the two most informative growth patterns. These are the most common patterns or the highest pattern. There is disagreement in the detection of prostate cancer, as in the grading using the Gleason scheme. Since pathologists can disagree between therapeutically relevant growth patterns and the presence of a tumor, there are clinically relevant consequences per individual case.

However, if we could circumvent labeling on a patch level, clinically evaluated biopsies could be cheaply labeled using their clinical reports. These reports contain all relevant information for clinical decisions, and are thus of large value for machine learning algorithms.

In this paper we will focus on prostate cancer detection, determining whether a biopsy contains cancerous glands or not. The diagnosis of prostate cancer—the most prevalent cancer for men in Western countries—is established by detection on histopathological slides by a pathologist. The microscopy slides containing cross-sections of biopsies can exhibit morphological changes to prostate glandular structures. In low-grade tumors, the epithelial cells still form glandular structures; however, in the case of high-grade tumors, the glandular structures are eventually lost[6].

In the presence of cancer, the percentage of cancerous tissue in a prostate biopsy can be as low as 1%, the evaluation of the biopsies can be tedious and error-prone, causing disagreement in the detection of prostate cancer, as in the grading using the Gleason scheme[4].

Besides substantial inter-observer and intra-observer variability, diagnosing prostate cancer is additionally challenging due to increasing numbers of biopsies as a result of the introduction of prostate-specific antigen (PSA) testing[7]. This number is likely to increase further due to the aging population. In the light of a shortage of pathologists[8], automated methods could alleviate workload.

To reduce potential errors and workload, recent work[9–15], has shown the potential to automatically detect prostate cancer in biopsies. These studies either use expensive, pixel-level annotations or train CNNs with slide-level labels only, using a patch-based approach.

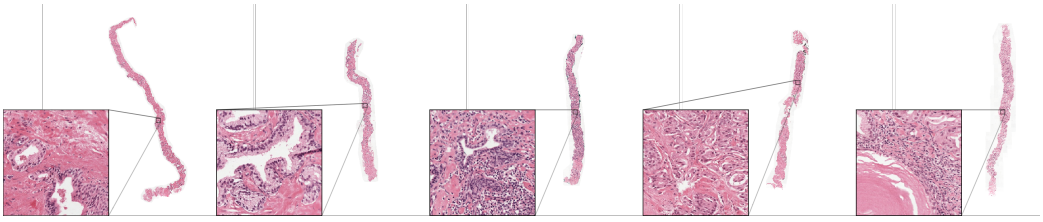
One popular strategy is based on multiple-instance-learning (MIL)[16–18]. In this approach, the whole-slide image (WSI) is subdivided into a grid of patches. The MIL assumption states that in a cancerous slide ('positive bag'),

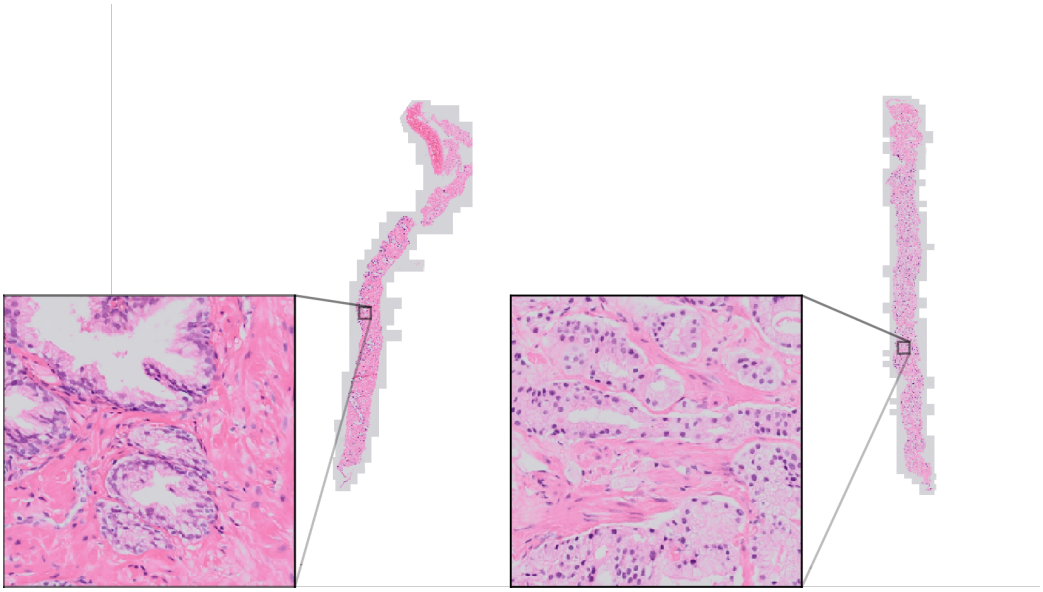
at least one patch will contain tumorous tissue, whereas negative slides have no patches containing tumour. Under this assumption, a CNN is trained on a patch-level to find the most tumorous patch.

However, this approach has several disadvantages[19]. First, this method only works for tasks where the label can be predicted from one individual patch and a single adversarial patch can result in a false positive detection. Second, it is essentially a patch-based approach, therefore, the size of the patch constrains the field-of-view of the network.

In this paper, we propose a novel method, using streaming[20], to train a modern CNN (ResNet-34) with 21 million parameters end-to-end to detect prostate cancer in whole-slide images of biopsies. We also investigate the use of transfer learning with this approach. This method does not suffer from the same disadvantages as the aforementioned approaches based on MIL: it can use the entire content of the whole-slide image for its prediction and the field-of-view is not limited to an arbitrary patch-size. We compare our approach against the methods by Campanella *et al.*[10] and Bulten *et al.*[9]. Since deep learning algorithm in computational pathology can suffer from bad generalization towards other scanners[21], we evaluated the generalization of the MIL- and streaming-trained ResNet-34 on additional biopsies acquired with a different scanner, previously used by Litjens *et al.*[12].

The streaming implementation allows us to train a convolutional neural network directly on entire biopsy images at high-resolution (268 megapixels) using only slide-level labels. We show that a state-of-the-art CNN can extract meaningful features from high-resolution images using labels from pathology reports without additional heuristics or post-processing. Subsequently, we show that transfer learning from ImageNet performs well for images that are 5000x bigger than the original images used for training (224x224), improving accuracy en decreasing train time.





Related works

For prostate cancer detection, previous works have used more traditional machine learning (i.e., feature-engineering) approaches[22–24]. Recently, researchers transitioned to using deep-learning-based methods for the detection of cancer[10,12]. Besides detection, research on prostate cancer grading has also been published[9,13,14].

In this work, we train on labels for individual biopsies. Since in other work, the memory of the accelerator restricts the input size of the image, published methods are based on searching relevant patches of the original slide[10,25–28], or compressing the slide into a smaller latent space[29].

We explicitly compare against the state-of-the-art method from Campanella *et al.*[10]. As mentioned before, their multiple-instance-learning approach is based on the single most-informative patch, and thus leads to a small field-of-view for the network, and potential false positives because of a few adversarial patches. To circumvent some of these problems, Campanelle *et al.*[10], tried to increase the field-of-view to multiple patches using a recurrent neural networks with some improvement. Their system achieved an area-under-the-receiver-operating curve (AUC) of 0.986. the aggregation method increased the AUC to 0.991. To make the comparison fair, we trained a ResNet-34

network architecture for both methods. However, when training end-to-end, the context of the whole image is automatically taken into account.

Campanella *et al.* showed that performance decreases when using smaller datasets, concluding that at least 10,000 biopsies are necessary for a good performance. Since they did not use data augmentation (probably because of the big dataset at hand), we investigated if we could reach similar performances with smaller dataset sizes using data augmentation.

Since the mentioned implementation of multiple-instance-learning only considers one patch, which may be less efficient, others[26,27] improved the method by using multiple resolution patches and attention mechanisms. Li *et al.* trained two models on low and high resolution patches, only patches that were predicted as suspicious by the lower resolution model were used to train the higher resolution model. Additionally, to calculate the attention mechanisms, all patches need to be kept in memory, limiting the size of the patches. Lu *et al.*[26] showed that, additionally to attention mechanisms, a frozen model pretrained on ImageNet decreases training time and improves data efficiency. We also use ImageNet weights, but by using the streaming-implementation of convolutions, can unfreeze the model and train the whole network end-to-end. However, in both papers, no comparison to the original method of Campanella *et al.* was performed.

Materials

We used the same dataset as Bulten *et al.*[9], we will briefly reiterate the collection of the dataset here. We built our dataset by retrospectively collecting biopsies and associated pathology reports of patients. Subsequently, we divided the patients between training, validation, and test set. As standard practice, we optimized the model using the training set and assessed generalization using the validation set during development. After development, we evaluated the model on the test set. The dataset, except for the test set, is publicly available as a Kaggle challenge at https://www.kaggle.com/c/prostate-cancer-grade-assessment. An additional set, termed Olympus set, was used for evaluation with a different scanner, originally extracted by Litjens et al[12].

Data collection

We retrieved pathologists reports of prostate biopsies for patients with a suspicion of prostate cancer, dated between Jan 1, 2012, and Dec 31, 2017, from digital patient records at the Radboud University Medical Center, excluding patients who underwent neoadjuvant or adjuvant therapy. The local ethics review board waived the need for informed consent (IRB approval 2016–2275).

After anonymization, we performed a text search on the anonymized pathology reports to divide the biopsies into positive and negative cases. Afterward, we divided the patient reports randomly into training, validation, and test set. By stratifying the biopsies on the primary Gleason score, we retrieved a comparable grade distribution in all sets. From the multiple cross-sections which were available per patient, we selected the standard hematoxylin-and-eosin-stained glass slide containing the most aggressive or prevalent part of malignant tissue for scanning.

We digitized the selected glass slides using a 3DHistech Pannoramic Flash II 250 (3DHistech, Hungary) scanner at a pixel resolution of $0.24\mu m$. Since each slide could contain one to six unique biopsies, commonly with two consecutive sections of the biopsies per slide, trained non-experts coarsely outlined each biopsy, assigning each with either the reported Gleason score, or labeling negative, based on the individual biopsy descriptions in the pathology report.

We collected 1243 glass slides, containing 5759 biopsies sections. After division, the training set consisted of 4712 biopsies, the validation set of 497 biopsies, and the test set of 550 biopsies (Table 1, Fig. [fig:example]). We extracted the individual biopsies from the scanned slides at a pixel resolution of $0.96\mu m$, visually approximately equivalent to 100x total magnification (i.e., 10x microscope objective with a standard 10x ocular lens). Subsequently, we trimmed the whitespace around the tissue using a tissue-segmentation neural network[30].

Reference standard test set

To determine a strong reference standard, three specialized pathologists reviewed the slides in three rounds. In the first round, each pathologist graded the biopsies independently. In the second round, each biopsy for which no consensus was reached in the first round, consensus was regraded by the

pathologist whose score differed from the other two, with the help of the pathologist's first score and the two anonymous Gleason scores of the other pathologists. In the third round, the pathologists discussed the biopsies without consensus after round two. In total 15 biopsies were discarded by the panel as they could not be reliably graded, resulting in a total test set size of 535 biopsies. See[9] for a complete overview of the grading protocol.

Smaller subsampled training set

To test our method with smaller datasets, we sampled 250 (5%) and 500 (10%) biopsies from the training set. Half of the cases in the new sets were negatives. For the positive biopsies, we stratified on primary Gleason grade and sampled equal amounts of each. Thus, we kept the distribution of the positive biopsies equal over all the datasets. We used the 5% (250 biopsies) and 10% (500 biopsies) datasets for training. The validation- and test-sets were equal to the ones used in the development of the model on the whole set.

Table 1: Distribution of datasets used in the experiments, stratisfied on primary Gleason pattern.

Dataset	**Total**	**Negative**	**3**	**4**	**5**
Training set	4712	16%	32%	45%	7%
Validation set	497	39%	23%	29%	9%
10% set	500	50%	17%	17%	17%
5% set	250	51%	16%	16%	16%
Test set	535	47%	25%	19%	9%
Olympus set	205	58%	25%	11%	4%

Olympus set

For the Olympus set, we used the slides of Litjens *et al.*, 2016[12]. That set contained 255 glass slides, scanned using an Olympus VS120-S5 system (Olympus, Japan). In comparison to the original paper, we used all biopsies on a negative slide, instead of only one, resulting in 291 biopsies (Fig. [fig:olympusexample]). Since patients in this set were biopsied in 2012, there

was a small overlap with the primary dataset used in this paper. We excluded 86 biopsies from 53 duplicate patients, resulting in a set of 205 biopsies.

Methods

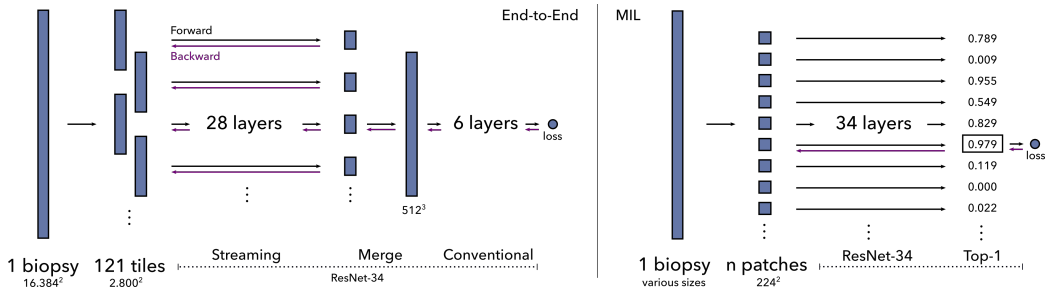
End-to-end streaming model

We trained a ResNet-34[31] convolutional neural network. Since the individual biopsy images differ in size, we padded or center/cropped them to 16384×16384 input. 99% of our dataset biopsies fitted within this input size. Since padding small biopsies results in a lot of whitespace, we changed the final pooling layer of ResNet-34 to a global max-pool layer.

For regularization, we used extensive data augmentation. To make augmentation of these images feasible with reasonable memory usage and speed, we used the open-source library VIPS[32]. Elastic random transformation, color augmentation (hue, saturation, and brightness), random horizontal and vertical flipping, and rotations were applied. We normalized the images based on training dataset statistics.

We initialized the networks using ImageNet-trained weights. As an optimizer, we used standard SGD (learning rate of $2e-4$) with momentum (0.9) and a mini-batch size of 16 images. Because when using streaming, we do not have a full image on the GPU, we cannot use batch normalization, thus we froze the batch normalization mean and variance, using the transfer-learned ImageNet running mean and variance. We randomly oversampled negative cases to counter the imbalance in the dataset[33].

For the experiments with random weights, we initialized the networks using He *et al.*[34]. We also used mixed precision training[35] to speed up training since these networks needed more epochs to convergence.



Streaming CNN

Most convolutional neural network architectures trained for a classification task require more memory in the first layers than in the latter because of the large feature maps. Our previously published method termed 'streaming'[20] circumvents these high memory requirements in the first layers by performing the operations on a tile-by-tile basis. This method is possible because CNNs use small kernels; hence the result at any given location is only defined by a small area of the input. This area is called the field-of-view. Since the field-of-view at the beginning of a network is vastly smaller than the full input image, we can use tiles (which have to be bigger than the field-of-view) to perform the convolutions serially. Thereby only requiring the amount of memory for the calculation on a single tile instead of the whole input image. After streaming, we concatenate the tile outputs to retrieve the complete intermediate feature map of the last streamed layer. This complete feature map is equal to the feature map we would get when training on a infinite-memory GPU.

During the forward pass of these memory-heavy first layers, we keep the final layer output and remove the output of the other intermediate layers, to save memory. We stream as many layers as needed until the last streamed layer's output can fit into GPU memory. This feature map can subsequently be fed through the rest of the neural network at once, resulting in the final output.

For the backward pass, we can use a similar implementation. The last layers, until the last streamed layer, can be backpropagated as usual. Then, we correctly tile the gradient of the last streamed layer's output. We use these gradient tiles for tile-by-tile backpropagation of the streamed layers. Leveraging the input tile, we recalculate the first layers' intermediate feature maps with a forward pass (this is commonly called gradient checkpointing[36]. With the recalculated features and the gradient tile, we can finish the backpropagation for the respective tile. We perform this for every tile. This way, we can recover the gradients of all parameters, as would be the case if training with the original input image. See Figure [figure:streamingSGD] for a graphical representation of the methods.

To train the ResNet-34, we streamed with a tile size of 2800×2800 (Fig. [fig:memrelation]) over the first 28 layers of the network. After these layers, the whole feature map (with dimensions 512×512×512) could fit into GPU

memory. It is possible to use the streaming implementation for more layers of the network, however, to improve speed it is better to stream until the feature map is just small enough. Finally, we fed the map through the remaining six layers to calculate the final output.

For the experiments with random weights in mixed precision, due to the decrease in memory usage, we could use a tile size of 3136×3136 to increase speed, and decrease the number of streamed layers to the first 27.

Training schedule

In transfer learning, often the first layers are treated as a feature extraction algorithm. After the feature extraction part, the second part is trained for the specific task[37]. Since the domain of histopathology differs significantly from the natural images in ImageNet, we froze the first three (of the four) residual blocks of the network (the first 27 layers) as feature extractor, only training the last block for our task. This also has the benefit of training faster, since we do not need to calculate gradients for the first layers. After 25 epochs, all the networks were stabilized and stopped improving the validation loss, showing slightly lower train losses.

From these epochs, we picked a checkpoint with a low validation loss to resume fine-tuning the whole network, unfreezing the weights of the first three residual blocks. Due to the relatively small validation set, the loss curve was less smooth than the training loss curve. To account for a sporadic checkpoint with a low loss, we calculated a moving average over five epochs. From these averages, we picked the window with the lowest loss, taking the middle checkpoint of the averaging window.

Starting from this checkpoint, we fine-tuned the whole network with a learning rate of $6e-5$. After approximately 50 epochs, all the networks stopped improving. For inference, we choose the checkpoints based on a moving average of five epochs with the lowest validation set loss. We averaged the weights of these checkpoints to improve generalization[38].

For the streaming experiments with random weights, we used the exact same training schedule except for the learning rate. The loss would go to infinity in the first few batches. When training from scratch, we could not use the first layers as feature extractor. We fine-tuned the whole network with a learning rate of $1e-5$ requiring 100 epochs until the validation loss did stabilized. We

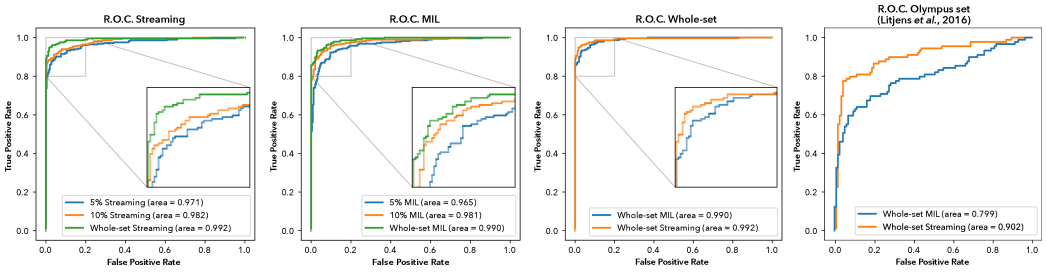
subsequently lowered the learning rate to $3e-6$ for 200 epochs after which the validation loss stopped improving.

The optimization and training procedure was fully conducted using the validation set, the test set, and the Olympus set were untouched during the development of the model.

Gradient accumulation and parallelization

Gradient accumulation is a technique to do a forward and backward pass on multiple images in series on the accelerator card, and averaging the parameter gradients over those images. Only after averaging, we perform a gradient descent step. Averaging the gradients over multiple images in series results in effectively training a mini-batch of these multiple images, while only requiring the memory for one image at a time. We used gradient accumulation over multiple biopsies to achieve an effective mini-batch size of 16 images.

We trained over multiple GPUs by splitting the mini-batch. For the streaming experiments, we used four GPUs (either NVIDIA RTX 2080ti or GTX 1080ti).



Multiple-instance-learning model

As a baseline, we implemented the multiple-instance-learning method as described in[10].

This method divides the images into a grid of smaller patches with the assumption that an individual patch could determine the image-level label. The task is to find the most informative patch. In our binary detection task, the most informative patch is determined by the patch with the highest probability of tumor. If there is a patch with a high probability of tumorous tissue, the whole biopsy is labeled tumorous.

We train such a model, per epoch, in two phases. The first phase is the inference phase, where we process all the patches of a biopsy, thereby finding the patch with the highest probability. This patch gets assigned the image-level label. Then, in the training phase, using only patches with the highest probability (the top-1 patch), the model parameters are optimized with a loss calculated on the patch probability and the label.

We followed the implementation from Campanella *et al.*[10], but tweaked it for our dataset sizes. We used standard SGD (learning rate of $1e-5$) with momentum (0.9) with a mini-batch size of 16 images. We froze the Batch-Normalization mean and variance, due to the smaller mini-batch size and to keep the features equal between the inference phase and the training phase. Equally, we oversampled negative cases to counter the imbalance in the dataset, instead of weighting[33].

We updated the whole model for 100 epochs when transfer learning, and 200 epochs when training from random weights. From these epochs, we picked the checkpoint with the lowest loss using the same scheme as the streaming model. Afterward, we trained for another 100 epochs with a learning rate of $3e-6$. The networks trained from random initialization on the 10% and 5% required 300 epochs. We again choose the checkpoint based on the lowest validation set loss, using a moving average of 5 epochs. We also used weight averaging for these checkpoints.

For regularization, we used the same data augmentation as the streaming model. We made sure that the same augmented patch was used in the inferencing and training phase. We used ImageNet statistics to normalize the patches.

Quantitative evaluation

The quantitative evaluation of both methods is performed using receiver-operating characteristic (ROC) analysis. Specifically, we look at the area under the ROC curve. To calculate a confidence interval, we used bootstrapping. We sampled the number of the biopsies in the set, with replacement, and calculated the area under the receiver-operating-curve based on the new sample. Repeating this procedure 10.000 times resulted in a distribution from which we calculated the 95% confidence interval (2.5 and 97.5 percentile)

Qualitative evaluation

To assess the correlation of certain regions to the cancerous label, we created heatmaps for both techniques. For MIL, we used the patch probabilities. For streaming, we used sensitivity maps using SmoothGrad[39]. As implementation of SmoothGrad, we averaged 25 sensitivity maps on Gaussian-noise-augmented versions of a biopsy. We used a standard deviation of 5% of the image-wide standard deviation for the Gaussian noise. As a comparison, we show pixel-level segmentations from the model published in Bulten *et al.*[9] as well.

In addition, we did a thorough analysis of the false positives and negatives of both the MIL and the streaming methods.

Experiments

We performed three experiments for both methods using three datasets. One experiment on all the data, and two on subsampled training sets, the 10% (500 biopsies) and 5% (250 biopsies) datasets.

Table 2: Area under the receiver-operating-curve comparison between the methods on the test set, *trained using transfer learning*.

Dataset	**Method**	**AUC**
Whole set	Streaming	0.992 (0.985–0.997)
	MIL	0.990 (0.984–0.995)
	Bulten *et al.*[9]	0.990 (0.982–0.996)
10% set	Streaming	0.982 (0.972–0.990)
	MIL	0.981 (0.970–0.990)
5% set	Streaming	0.971 (0.960–0.982)
	MIL	0.965 (0.949–0.978)
Olympus set	Streaming	0.909 (0.863–0.949)
	MIL	0.799 (0.732–0.861)

Table 3: Area under the receiver-operating-curve comparison between the methods on the test set, *trained from random initialization*.

Dataset	**Method**	**AUC**
Whole set	Streaming	0.967 (0.952–0.980)
	MIL	0.918 (0.894–0.941)
10% set	Streaming	0.924 (0.900–0.945)
	MIL	0.899 (0.871–0.924)
5% set	Streaming	0.915 (0.889–0.939)
	MIL	0.862 (0.831–0.892)

On the whole dataset, the streaming model achieved an AUC of 0.992 (0.985–0.997) and the MIL model an AUC of 0.990 (0.984–0.995). Interestingly, our models trained on the whole dataset reached similar performance to previous work on this dataset[9], which utilized a segmentation network trained using dense annotations obtained in a semi-supervised fashion.

For streaming, the performance on the smaller dataset sizes are similar between the two. 5% dataset has an AUC of 0.971 (0.960–0.982) for 5% and 0.982 (0.972–0.990) for 10% (Table 2). The models trained with more data generalize better (Fig. [figure:ROCcomparison]).

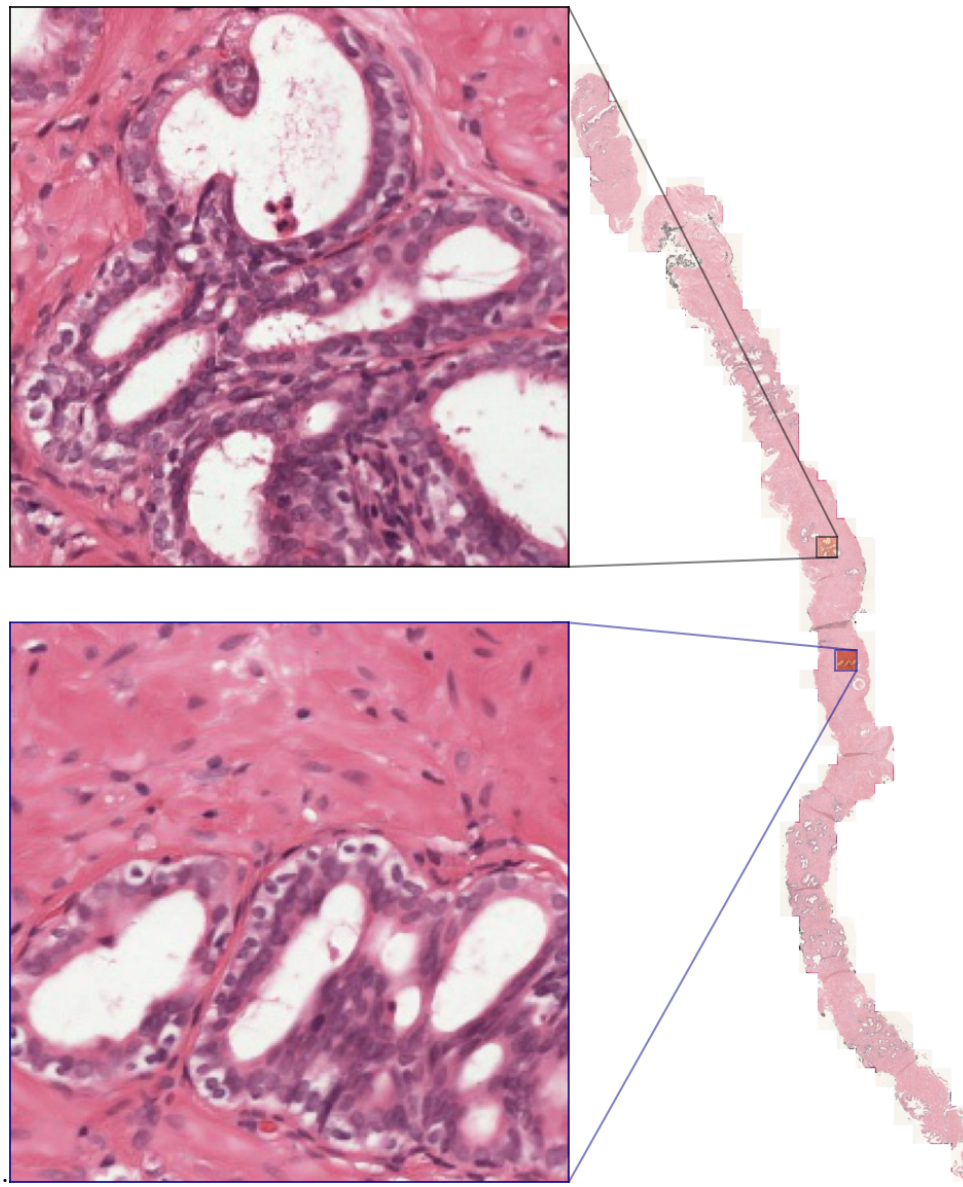
Also for multiple-instance learning there is a clear improvement going from a model trained on the smallest dataset size, with an AUC of 0.965 (0.949–0.978), increasing to 0.981 (0.970–0.990) on the 10% dataset.

There seems to be a trend that the MIL model performs slightly worse (Fig. [figure:ROCcomparison]), however, this difference falls within the confidence intervals.

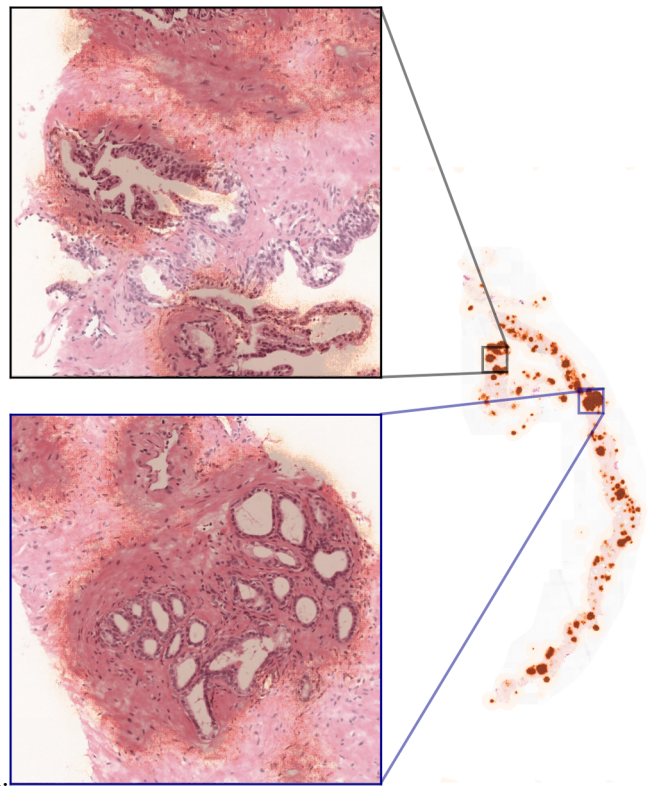
In the experiments trained from random weights, there is a larger separation between the methods, without overlap of the confidence intervals. Streaming achieves an AUC of 0.967 (0.952–0.980) when using the whole set (Table 3) in comparison to MIL with 0.918 (0.894–0.941). For the 10% set using streaming also results in higher metrics 0.924 (0.900–0.945) versus 0.899 (0.871–0.924). Finally, the 5% set gets an AUC of 0.915 (0.889–0.939) for streaming and 0.862 (0.831–0.892) for MIL.

In general, the areas identified by MIL and streaming in the heatmaps correspond well to the pixel-level segmentations from Bulten *et al.*, showing that both methods pick up the relevant regions for cancer identification (Figure [fig:heatmaps]). Most errors of the models seem to be due to normal epithelium mimicking tumorous glands in the case for false positives, and the small size of some tumorous regions as a possible reason for the false negatives. (Table 4)

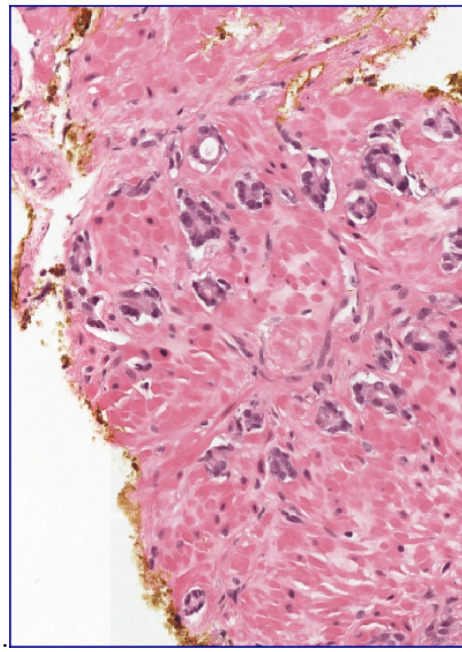
For the Olympus set, existing of biopsies scanned by the Olympus VS-system, there is a larger separation between the methods. Streaming reaches an AUC of 0.909 (0.863–0.949), with MIL scoring 0.799 (0.732–0.861). For this dataset, MIL has 36 false negatives versus 20 for streaming, and 8 false positive versus 5 from streaming.



Identified by the MIL model.

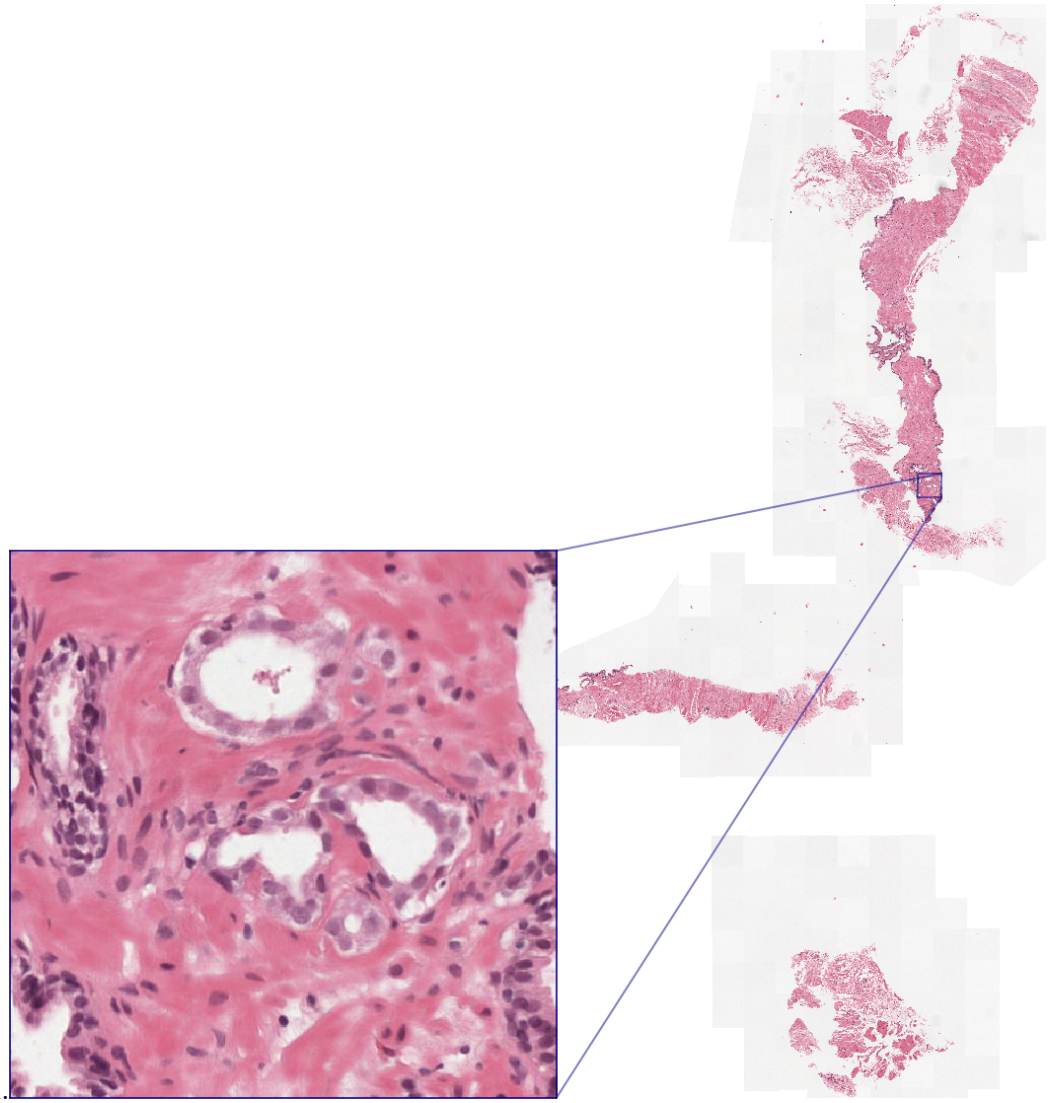


Identified by the streaming model.



Small tumorous glands mimicking vessels. Missed by both models.

Very limited amount of tumor (four glands), missed by the streaming



network.

Table 4: The predictions were manually judged and divided in the following categories. False positives and negatives were selected at the point of maximum accuracy in the ROC curve.

False positives	**Streaming** (5)	**MIL** (13)
Normal mimicking tumor	2	7
Inflammation	1	4

False positives	**Streaming** (5)	**MIL** (13)
Tissue artefacts	1	1
Bladder epithelium	1	0
Colon epithelium	0	1
False negatives	**Streaming** (13)	**MIL** (12)
Little amount of tumor	7	4
Tissue artefacts	3	1
Low-grade tumor	1	2
Inflammation-like	1	2
Unclear reason	1	2

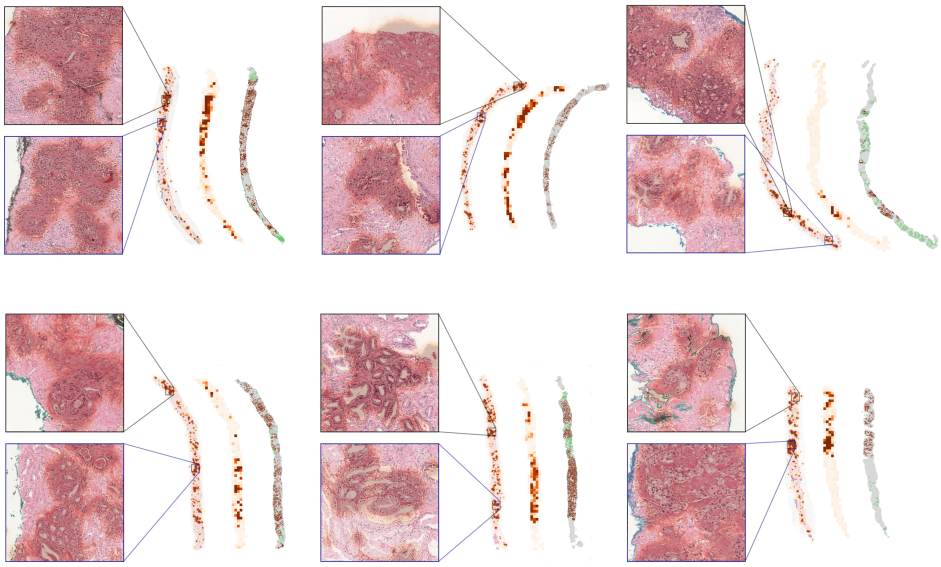


Table 5: When fine-tuning only the last six of the ResNet-34 are trained, all other layers are frozen. All metrics are in seconds. Preprocessing times not shown, in our experiments they accounted for 8 seconds for new biopsies. Mixed precision streaming is relatively fast due to bigger tile-size and one less layer to stream.

	Training	**Fine-tuning**	**Inferencing**
Full precision streaming	32.3 s	12.5 s	8.5 s
Mixed precision streaming	17.2 s	3.6 s	3.5 s
MIL	0.5 s	n.a.	0.25 s

Discussion and conclusions

In this paper, we proposed using streaming[20] convolution neural networks to directly train a state-of-the-art ResNet-34 architecture on whole prostate biopsies with slide-level labels from pathology reports. We are the first to train such high-resolution (268 megapixels) images end-to-end, without further heuristics. Accomplishing this without the streaming implementation would require a accelerator card with 2.4 terabyte of memory.

We showed it is possible to train a residual neural network with biopsy level labels and reach similar performance to a popular multiple-instance-learning (MIL) based method. Our models trained on the whole dataset reached an AUC of 0.992 for streaming training, and 0.990 for MIL. In addition, we achieved equal performance to a method trained on patch-based labels, with an AUC of 0.990[9] on the same dataset. Although, it should be noted that Bulten *et al.* used weakly-supervised labels, they used a cascade of models to go from epithelium antibody-staining to semi-automatic pixel-level annotations, to generate a model trained at the patch level.

Looking at the failure cases (Table 4), multiple-instance-learning suffers from interpreting normal glands as tumorous (Fig. [fig:errors_pos] and [fig:errors_neg]). We hypothesize this is due to the lack of context, in all but three cases the misclassification was due to one patch. For false negatives, both models fail when there is a small amount of tumor, however the streaming model seems to suffer more from this. A possible solution

would be to incorporate attention mechanisms into the network, allowing it to focus to smaller parts of the biopsy.

To study the benefits of transfer learning, we trained the networks from randomly initialized weights according to He *et al.*[34]. These networks took longer to converge (approximately 3-4x more iterations needed) and reached lower performances. In this case, MIL is less capable of extracting relevant information from the patches and scores worse than networks trained with streaming, scoring an AUC of 0.918 versus 0.959, respectively. We think training from random weights introduced additional noise in the MIL-training process. Since some biopsies contain cancerous tissue that only falls within a few patches, ImageNet weights can provide a better starting point to find these relevant patches during training. However, when training from random initialization, the noise of the benign patches in a cancerous biopsy may make it harder to learn. When possible, we advise the usage of pretrained network to increase convergence speed and final performance.

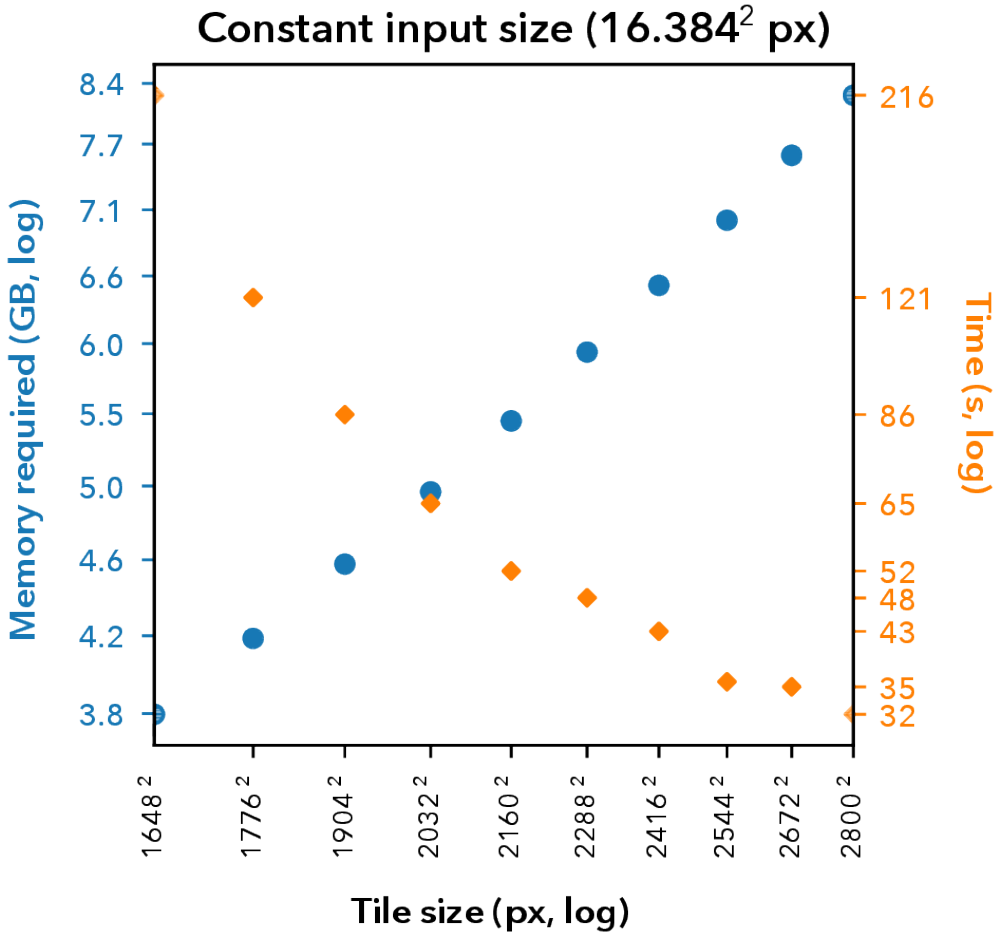
MIL performs weaker than the streaming network on the Olympus set, with the main error being misclassifying 36 biopsies with tumor as negative. The external dataset has other color characteristics due to the different scanner used. Since both network have been trained with the same data augmentation, MIL seems to benefit less from this augmentation thus generalizing worse. The improvement seen in generalization on the Olympus set and the trend of higher performance overall suggest that streaming extracts more meaningful features from the same dataset.

In this paper, we compared against a MIL implementation of Campanella *et al.* In their MIL implementation, only the top-1 patch is used for training per epoch. The method's data efficiency is reliant on how often different patches are selected in the first phase. Our results on the smallest dataset sample (5%, 250 slides) hint towards reduced data efficiency for MIL. However, the performance on the smaller datasets was already close to optimal, suggesting effective use of the transferred ImageNet-weights. Even though it is not the same test set as in their original paper, this seems to suggest a better performance for smaller datasets than Campanella *et al.* reported. Hypothetically, this could be due to data augmentation, which they did not use, and increased randomness with smaller mini-batch size in our study.

For MIL, selecting different patches per image, every epoch, is important to circumvent overfitting. We used lower minibatch-sizes, 16 vs 512, and

learning rates, $1e-5$ vs $1e-4$ as the original implementation[10]. We saw increased stability in training using smaller mini-batch sizes and learning rates, especially for the smaller datasets, where the whole dataset would otherwise fit in one mini-batch. Lower mini-batch sizes increased some noise, thereby picking different patches per epoch.

The streaming implementation of convolutional neural networks is computationally slower than our baseline. Mainly due to the number (121) and overlap (~650 pixels) of the tiles during backpropagation. For inference new slides, taking into account the preprocessing that needs to happen (roughly 8 seconds for extracting patches or extracting the whole biopsy), MIL takes half the time (8.25 seconds) compared to streaming (16.5 seconds) (Table IV). The most significant difference lies in the train speed, where for full precision, streaming is ~65 times slower than MIL (Table 5). Streaming did require half the number of epochs needed to converge, but the gap is still large. However, an algorithm only needs to be trained once, and the inference speed for both streaming and MIL is fast enough for use in clinical practice.



Improving the speed of the streaming implementation of convolutional operations is of interest. In this work, we improved training speed by first freezing the first layers of the neural network, not having to calculate gradients. Using this training scheme in the multiple-instance-learning baseline resulted in unstable training and worse performance. Further research could focus on decreasing the number of calculations needed by using variable input sizes[1] or lower-level implementations that ignore whitespace, such as sparse implementations of convolutions[40].

Streaming training with high-resolution images opens up the possibility to quickly gather large datasets with labels from pathology reports to train

[1]An example implementation of this can be found in the open source repository.

convolutional neural networks. Although linking individual biopsies to the pathology report is still a manual task, it is more efficient than annotating the individual slides. However, some pathology labs will manufacture one slide per biopsy and report systematically on these individual biopsies. Training from a whole slide, with multiple biopsies, is left for future research.

Since multiple-instance-learning, in the end, predicts the final label on a single patch, tasks that require information from different sites of the biopsy could be hard to engineer in this framework. For example, in the Gleason grading scheme, the two most informative growth patterns are reported. These patterns could lie on different parts of the biopsy, outside of the field-of-view of a single patch. Also, additional growth patterns could be present. The first reported growth pattern of Gleason grading is the most prevalent. Since multiple-instance-learning works patch-based, volumes that span larger than one patch are not used for the prediction. Streaming allows for training complex tasks, such as cancer grading, even with slide-level labels.

Our heatmaps show that indeed the streaming model uses information from multiple regions in the biopsy (Fig. [fig:heatmaps]). Even though our model is not trained on a patch-level, the sensitivity maps highlight similar regions as the MIL method and the segmentation algorithm from Bulten *et al.* Thus, interestingly, a modern convolutional neural network, originally developed for tiny input sizes, can extract useful information from 268 megapixel images.

Besides allowing the entire slide to inform predictions, streaming training also has the advantage of being able to learn with hard or impossible to annotate global information. For example, in the medical domain, survival prediction can be of great interest. Future work could be to predict survival from histopathology tissue directly. Reliably annotating for this task can be difficult. Since streaming can find patterns and features from the whole image using just the retrospective patient prognosis, this method can be beneficial in automatically finding new relevant biomarkers.

We provide source code of the streaming pipeline at GitHub[2]. We tried to make it easy to use with other datasets. Additionally to methods used in this paper, we added mixed precision support for even more memory efficient and faster training.

[2]https://github.com/DIAGNijmegen/pathology-streaming-pipeline

1. Krizhevsky A. *Learning Multiple Layers of Features from Tiny Images*. University of Toronto; 2009.

2. Russakovsky O, Deng J, Su H, et al. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision*. 2015;115(3):211-252. doi:10.1007/s11263-015-0816-y

3. Sun C, Shrivastava A, Singh S, Gupta A. Revisiting Unreasonable Effectiveness of Data in Deep Learning Era. *2017 IEEE International Conference on Computer Vision (ICCV)*. Published online 2017:843-852.

4. Ozkan TA, Eruyar AT, Cebeci OO, Memik O, Ozcan L, Kuskonmaz I. Interobserver variability in Gleason histological grading of prostate cancer. *Scandinavian Journal of Urology*. 2016;50(6):420-424. doi:10.1080/21681805.2016.1206619

5. Epstein JI. An Update of the Gleason Grading System. *Journal of Urology*. 2010;183(2):433-440. doi:10.1016/j.juro.2009.10.046

6. Fine SW, Amin MB, Berney DM, et al. A contemporary update on pathology reporting for prostate cancer: Biopsy and radical prostatectomy specimens. *European Urology*. 2012;62(1):20-39. doi:10.1016/j.eururo.2012.02.055

7. Welch HG, Albertsen PC. Prostate Cancer Diagnosis and Treatment After the Introduction of Prostate-Specific Antigen Screening: 1986–2005. *JNCI: Journal of the National Cancer Institute*. 2009;101(19):1325-1329. doi:10.1093/jnci/djp278

8. Wilson ML, Fleming KA, Kuti MA, Looi LM, Lago N, Ru K. Access to pathology and laboratory medicine services: a crucial gap. *Lancet (London, England)*. 2018;391(10133):1927-1938. doi:10.1016/S0140-6736(18)30458-6

9. Bulten W, Pinckaers H, Boven H van, et al. Automated deep-learning system for Gleason grading of prostate cancer using biopsies: a diagnostic study. *The Lancet Oncology*. 2020;21(2):233-241. doi:10.1016/S1470-2045(19)30739-9

10. Campanella G, Hanna MG, Geneslaw L, et al. Clinical-grade computational pathology using weakly supervised deep learning on whole slide images. *Nature Medicine*. 2019;25(8):1301-1309. doi:10.1038/s41591-019-0508-1

11. Nagpal K, Foote D, Liu Y, et al. Development and validation of a deep learning algorithm for improving Gleason scoring of prostate cancer. *npj Digital Medicine*. 2019;2(1):48. doi:10.1038/s41746-019-0112-2

12. Litjens G, Sánchez CI, Timofeeva N, et al. Deep learning as a tool for increased accuracy and efficiency of histopathological diagnosis. *Scientific Reports*. 2016;6(1):26286. doi:10.1038/srep26286

13. Arvaniti E, Fricker KS, Moret M, et al. Automated Gleason grading of prostate cancer tissue microarrays via deep learning. *Scientific Reports*. 2018;8(1):12054. doi:10.1038/s41598-018-30535-1

14. Lucas M, Jansen I, Savci-Heijink CD, et al. Deep learning for automatic Gleason pattern classification for grade group determination of prostate biopsies. *Virchows Archiv*. 2019;475(1):77-83. doi:10.1007/s00428-019-02577-x

15. Ström P, Kartasalo K, Olsson H, et al. Artificial intelligence for diagnosis and grading of prostate cancer in biopsies: a population-based, diagnostic study. *The Lancet Oncology*. 2020;21(2):222-232. doi:10.1016/S1470-2045(19)30738-7

16. Courtiol P, Tramel EW, Sanselme M, Wainrib G. Classification and Disease Localization in Histopathology Using Only Global Labels: A Weakly-Supervised Approach. *arXiv preprint*. 2018;1802.02212. http://arxiv.org/abs/1802.02212

17. Ilse M, Tomczak JM, Welling M. Attention-based Deep Multiple Instance Learning. *arXiv preprint*. 2018;1802.04712. http://arxiv.org/abs/1802.04712

18. Amores J. Multiple instance classification: Review, taxonomy and comparative study. *Artificial Intelligence*. 2013;201:81-105. doi:10.1016/j.artint.2013.06.003

19. Laak J van der, Ciompi F, Litjens G. No pixel-level annotations needed. *Nature Biomedical Engineering*. 2019;3(11):855-856. doi:10.1038/s41551-019-0472-6

20. Pinckaers H, Ginneken B van, Litjens G. Streaming convolutional neural networks for end-to-end learning with multi-megapixel images. *arXiv preprint*. 2019;1911.04432. https://arxiv.org/abs/1911.04432

21. Swiderska-Chadaj Z, Bel T de, Blanchet L, et al. Impact of rescanning and normalization on convolutional neural network performance in multi-center, whole-slide classification of prostate cancer. *Scientific Reports*. 2020;10(1):14398. doi:10.1038/s41598-020-71420-0

22. Gertych A, Ing N, Ma Z, et al. Machine learning approaches to analyze histological images of tissues from radical prostatectomies. *Computerized Medical Imaging and Graphics*. 2015;46:197-208. doi:10.1016/j.compmedimag.2015.08.002

23. Nguyen TH, Sridharan S, Macias V, et al. Automatic Gleason grading of prostate cancer using quantitative phase imaging and machine learning. *Journal of biomedical optics*. 2017;22(3):36015.

24. Naik S, Doyle S, Feldman M, Tomaszewski J, Madabhushi A. Gland Segmentation and Computerized Gleason Grading of Prostate Histology by Integrating Low- , High-level and Domain Specific Information. In: *Proceedings of 2nd Workshop on Microsopic Image Analysis with Applications in Biology.*; 2007:1-8.

25. Ianni JD, Soans RE, Sankarapandian S, et al. Tailored for Real-World: A Whole Slide Image Classification System Validated on Uncurated Multi-Site Data Emulating the Prospective Pathology Workload. *Scientific Reports*. 2020;10(1):3217. doi:10.1038/s41598-020-59985-2

26. Lu MY, Williamson DFK, Chen TY, Chen RJ, Barbieri M, Mahmood F. Data Efficient and Weakly Supervised Computational Pathology on Whole Slide Images. *arXiv preprint*. 2020;2004.09666. https://arxiv.org/abs/2004.09666

27. Li J, Li W, Gertych A, Knudsen BS, Speier W, Arnold CW. An attention-based multi-resolution model for prostate whole slide imageclassification and localization. *arXiv preprint*. 2019;1905.13208. http://arxiv.org/abs/1905.13208

28. Mercan C, Aksoy S, Mercan E, Shapiro LG, Weaver DL, Elmore JG. Multi-Instance Multi-Label Learning for Multi-Class Classification of Whole Slide Breast Histopathology Images. *IEEE Transactions on Medical Imaging*. 2018;37(1):316-325. doi:10.1109/TMI.2017.2758580

29. Tellez D, Litjens G, Laak J van der, Ciompi F. Neural Image Compression for Gigapixel Histopathology Image Analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. in press. doi:10.1109/TPAMI.2019.2936841

30. Bándi P, Balkenhol M, Ginneken B van, Laak J van der, Litjens G. Resolution-agnostic tissue segmentation in whole-slide histopathology images with convolutional neural networks. *PeerJ*. 2019;7:e8242. doi:10.7717/peerj.8242

31. He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. In: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. Vol 2016. IEEE Computer Society; 2016:770-778. doi:10.1109/CVPR.2016.90

32. Cupitt J, Martinez K. VIPS: An imaging processing system for large images. *Proceedings of SPIE - The International Society for Optical Engineering*. 1996;1663:19-28. doi:10.1117/12.233043

33. Buda M, Maki A, Mazurowski MA. A systematic study of the class imbalance problem in convolutional neural networks. *Neural networks : the official journal of the International Neural Network Society*. 2018;106:249-259. doi:10.1016/j.neunet.2018.07.011

34. He K, Zhang X, Ren S, Sun J. Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification. In: *Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV)*. IEEE Computer Society; 2015:1026-1034. doi:10.1109/ICCV.2015.123

35. Micikevicius P, Narang S, Alben J, et al. Mixed Precision Training. *arXiv preprint*. 2017;1710.03740. http://arxiv.org/abs/1710.03740

36. Chen T, Xu B, Zhang C, Guestrin C. Training Deep Nets with Sublinear Memory Cost. *arXiv preprint*. 2016;1604.06174. http://arxiv.org/abs/1604.06174

37. Tan C, Sun F, Kong T, Zhang W, Yang C, Liu C. A Survey on Deep Transfer Learning. In: *Artificial Neural Networks and Machine Learning - {ICANN} 2018 - 27th International Conference on Artificial Neural Networks, Rhodes, Greece, October 4-7, 2018, Proceedings, Part {III}*. Vol 11141. Lecture notes in computer science. Springer; 2018:270-279. doi:10.1007/978-3-030-01424-7_27

38. Izmailov P, Podoprikhin D, Garipov T, Vetrov D, Wilson AG. Averaging Weights Leads to Wider Optima and Better Generalization. *arXiv preprint*. 2018;1803.05407. http://arxiv.org/abs/1803.05407

39. Smilkov D, Thorat N, Kim B, Viégas FB, Wattenberg M. SmoothGrad: removing noise by adding noise. In: *ICML Workshop on Visualization for Deep Learning.*; 2017. http://arxiv.org/abs/1706.03825

40. Park J, Li S, Wen W, et al. Faster CNNs with Direct Sparse Convolutions and Guided Pruning. Published online November 2016. https://openreview.net/forum?id=rJPcZ3txx

Predicting biochemical recurrence of prostate cancer with artificial intelligence

Hans Pinckaers[a*], Jolique van Ipenburg[a], Jonathan Melamed[b], Angelo De Marzo

Abstract

Background: The first sign of metastatic prostate cancer after radical prostatectomy is rising PSA levels in the blood, termed biochemical recurrence. The prediction of recurrence relies mainly on the morphological assessment of prostate cancer using the Gleason grading system. However, in this system, within-grade morphological patterns and subtle histopathological features are currently omitted, leaving a significant amount of prognostic potential unexplored.

Methods: To discover additional prognostic information using artificial intelligence, we trained a deep learning system to predict biochemical recurrence from tissue in H&E-stained microarray cores directly. We developed a morphological biomarker using convolutional neural networks leveraging a nested case-control study of 685 patients and validated on an independent cohort of 204 patients. We use concept-based explainability methods to interpret the learned tissue patterns.

Results: The biomarker provides a strong correlation with biochemical recurrence in two sets (n=182 and n=204) from separate institutions. Concept-based explanations provided tissue patterns interpretable by pathologists.

Conclusions: These results show that the model finds predictive power in the tissue beyond the morphological ISUP grading.

Corresponding author: Hans Pinckaers, Radboud University Medical Center, Postbus 9101, 6500 HB Nijmegen, The Netherlands (tel +31 634 856 950, hans.pinckaers@radboudumc.nl)

[a] Department of Pathology, Radboud Institute for Health Sciences, Radboud University Medical Center, Nijmegen, The Netherlands

[b] Department of Pathology, New York University Langone Medical Center, New York, USA

[c] Departments of Pathology, Urology and Oncology, The Brady Urological Research Institute and the Sidney Kimmel Comprehensive Cancer Center at Johns Hopkins, Baltimore, Maryland, USA

[d] Department of Epidemiology, Johns Hopkins Bloomberg School of Public Health, Baltimore, Maryland, USA

[e]Center for Medical Image Science and Visualization, Linköping University, Linköping, Sweden

Introduction

Prostate cancer is a common malignancy among men, affecting 1.4 million per year.[1] A significant proportion of these men will receive the primary curative treatment of a prostatectomy. This surgery's success can partly be judged by the concentration of prostate-specific antigen (PSA) in the blood. While it has a dubious role in prostate cancer screening[2(ppHeijnsdijk2018-yn)], this protein is a valuable biomarker in PCa patients' follow-up post-prostatectomy. In a successful surgery, the concentration will mostly be undetectable (<0.1 ng/mL) after four to six weeks[3].

However, in approximately 30% of the patients[6], PSA will rise again after surgery, called biochemical recurrence, pointing to regrowth of prostate cancer cells. Biochemical recurrence is a prognostic indicator for subsequent progression to clinical metastases and prostate cancer death.[7] Estimating chances of biochemical recurrence could help to better stratify patients for specific adjuvant treatments.

The risk of biochemical recurrence of prostate cancer is currently assessed in clinical practice through a combination of the ISUP grade[8], the PSA value at diagnosis and the TNM staging criteria. In a recent European consensus guideline, these factors were proposed to separate the patients into a low-risk, intermediate-risk and high-risk group.[9] A high ISUP grade independently can,

independently of other factors, assign a patient to the intermediate (grade 2/3) or high-risk group (grade 4/5).

Based on the distribution of the Gleason growth patterns[10], which are prognostically predictive morphological patterns of prostate cancer, pathologists assign cancerous tissue obtained via biopsy or prostatectomy into one of five groups. They are commonly referred to as International Society of Urological Pathology (ISUP) grade groups, the ISUP grade, Gleason grade groups, or just grade groups.[13]. Throughout this paper we will use the term *ISUP grade*. The ISUP grade suffers from several well-known limitations. For example, there is substantial disagreement in the grading using the Gleason scheme.[13]. Furthermore, although the Gleason growth patterns have seen significant updates and additions since their inception in the 1960s, they remain relatively coarse descriptors of tissue morphology. As such, the prognostic potential of more fine-grained morphological features has been underexplored. We hypothesize that artificial intelligence, and more specifically deep learning, has the potential to discover such information and unlock the true prognostic value of morphological assessment of cancer. Specifically, we developed a deep learning system (DLS), trained on H&E-stained histopathological tissue sections, yielding a score for the likelihood of early biochemical recurrence.

Deep learning is a recent new class of machine learning algorithms that encompasses models called neural networks. These networks are optimized using training data; images with labels, such as recurrence information. From the training data, relevant features to predict the labels are automatically inferred. During development, the generalization of these features is tested on separated training data, which is not used for learning. Afterwards, a third independent set of data, the test set, is used to ensure generalization. Since features are inferred, handcrafted feature engineering is not needed anymore to develop machine learning models. Neural networks are the current state-of-the-art in image classification[15].

Deep learning has previously been shown to find visual patterns to predict genetic mutations from morphology, for example, in lymphoma[16] and lung cancer[17]. Additionally, deep learning has been used for feature discovery in colorectal cancer[18] and intrahepatic cholangiocarcinoma[19] using survival data. Although deep learning has been used with biochemical recurrence data on prostate cancer, Leo *et al.*[20] assumed manual feature selection beforehand, strongly limiting the extent of new features to be discovered. Yamamoto

et al.[21] used whole slide images and a deep-learning-based encoding of the slides to tackle the slides' high resolution. They leverage classical regression techniques and support-vector machine models on these encodings. The deep learning model was not directly trained on the outcome, limiting the feature discovery in this work as well.

A common critique of deep learning is its black-box nature of the inferred features.[22] Especially in the medical field, decisions based on these algorithms should be extensively validated and be explainable. Besides making the algorithms' prediction trustworthy and transparent, from a research perspective, it would be beneficial to visualize the data patterns which the model learned, allowing insight into the inferred features. We can visualize the patterns learned by the network leveraging a new technique called Automatic Concept Explanations (ACE)[23]. ACE clusters patches of the input image using their intermediate inferred features showing common patterns inferred by the network. We were interested in finding these common concepts over a range of images to unravel patterns that the model has identified.

This study aimed to use deep learning to develop a new prognostic biomarker based on tissue morphology for recurrence in patients with prostate cancer treated by radical prostatectomy. As training data, we used a nested case-control study[24]. This study design ensured we could evaluate whether the network learned differentiating patterns independent of Gleason patterns.

Methods

Cohorts

Two independent cohorts of patients who underwent prostatectomy for clinically localized prostate cancer were used in this study. Patients were treated at either the Johns Hopkins Hospital in Baltimore or New York Langone Medical Center. Both cohorts were accessed via the Prostate Cancer Biorepository Network[25].

For the development of the novel deep-learning-based biomarker (further referred to as DLS biomarker), we used a nested case-control study of patients from Johns Hopkins. This study consists of 524 matched pairs (724 unique patients) containing four tissue spots per patient. They were sampled from 4,860 prostate cancer patients with clinically localized prostate cancer

who received radical retropubic prostatectomy between 1993 and 2001. Men were routinely checked after prostatectomy at 3 months and at least yearly thereafter. Surveillance for recurrence was conducted using digital rectal examination and measurement of serum PSA concentration. Patients were followed for outcome until 2005, with a median follow-up of 4.0 years. The outcome was defined as recurrence, based on biochemical recurrence (serum PSA >0.2 ng/mL on 2 or more occasions after a previously undetectable level after prostatectomy), or events indicating biochemical recurrence before this was measured; local recurrence, systemic metastases, or death from prostate cancer. Controls were paired to cases with recurrence using incidence density sampling[26]. For each case, a control was selected who had not experienced recurrence by the date of the case's recurrence and was additionally matched based on age at surgery, race, pathologic stage, and Gleason sum in the prostatectomy specimen based on the pathology reports. Given the incidence density sampling of controls, some men were used as controls for multiple cases, and some controls developed recurrence later and became cases for that time period.

Table 1: Baseline characteristics of test set and development set from the John Hopkins Hospital, prostate cancer recurrence cases and controls, men who underwent radical prostatectomy for clinically localized disease between 1993 to 2001.

	Development set			**Test set**		
	Recurrence cases	**No events cases**	**P**	**Recurrence cases**	**Controls***	**P**
N	368	135		91	91	
Age, mean (SD)	58.9 (6.2)	59.3 (6.3)	p=0.540	58.4 (6.1)	58.3 (6.3)	*Matched*

Table 1: Baseline characteristics of test set and development set from the John Hopkins Hospital, prostate cancer recurrence cases and controls, men who underwent radical prostatectomy for clinically localized disease between 1993 to 2001.

preop. PSA (ng/mL), mean (SD)	12.3 (10.0)	10.1 (7.5)	p=0.010	12.3 (10.8)	10.5 (7.7)	p=0.195
Race, n (%)			p=0.599			*Matched*
White	327 (88.9)	120 (88.9)		72 (79.1)	75 (82.4)	
Black or African American	32 (8.7)	14 (10.4)		12 (13.2)	10 (11.0)	

Table 1: Baseline characteristics of test set and development set from the John Hopkins Hospital, prostate cancer recurrence cases and controls, men who underwent radical prostatectomy for clinically localized disease between 1993 to 2001.

Other	9 (2.4)	1 (0.7)		7 (7.7)	6 (6.6)	
Pathological stage			p=0.107			*Matched*
pT2	43 (11.7)	25 (18.5)		20 (22.0)	19 (20.9)	
pT3a	199 (54.1)	63 (46.7)		50 (54.9)	51 (56.0)	
pT3b or N1	126 (34.2)	47 (34.8)		21 (23.1)	21 (23.1)	

Table 1: Baseline characteristics of test set and development set from the John Hopkins Hospital, prostate cancer recurrence cases and controls, men who underwent radical prostatectomy for clinically localized disease between 1993 to 2001.

Gleason sum prostatectomy (%)			p=0.179		*Matched*
6	38 (10.3)	25 (18.5)		20 (22.0)	23 (25.3)
7	233 (63.3)	76 (56.3)		51 (56.0)	50 (54.9)
8+	97 (26.4)	34 (25.2)		20 (22.0)	18 (19.8)

Table 1: Baseline characteristics of test set and development set from the John Hopkins Hospital, prostate cancer recurrence cases and controls, men who underwent radical prostatectomy for clinically localized disease between 1993 to 2001.

ISUP grade, n (%)			p=0.002		p=0.851
1	38 (10.3)	25 (18.5)		20 (22.0)	23 (25.3)
2	140 (38.0)	61 (45.2)		35 (38.5)	38 (41.8)
3	93 (25.3)	15 (11.1)		16 (17.6)	12 (13.2)
4	49 (13.3)	21 (15.6)		13 (14.3)	10 (11.0)

Table 1: Baseline characteristics of test set and development set from the John Hopkins Hospital, prostate cancer recurrence cases and controls, men who underwent radical prostatectomy for clinically localized disease between 1993 to 2001.

5	48 (13.0)	13 (9.6)		7 (7.7)	8 (8.8)	
Positive surgical margins	140 (38.1)	24 (17.8)	p<0.001	36 (39.6)	20 (22.0)	p=0.016
Mean year of surgery	1997.0 (2.3)	1995.5 (2.3)	p<0.001	1997 (2.3)	1995 (2.1)	p<0.001

Table 1: Baseline characteristics of test set and development set from the John Hopkins Hospital, prostate cancer recurrence cases and controls, men who underwent radical prostatectomy for clinically localized disease between 1993 to 2001.

* due to the nested case-control nature, some controls could have a biochemical recurrence, but always later than their matched case.

Table 2: Baseline characteristics of the cohort from New York Langone hospital, prostate cancer recurrence cases and controls, men who underwent radical prostatectomy between 2001 to 2003

	Recurrence cases	**Controls**	**P**
N	38	166	
preop. PSA (ng/mL), mean (SD)	11.6 (11.5)	6.7 (3.9)	p=0.014
Age, mean (SD)	61.7 (8.9)	60.3 (6.6)	p=0.359
Race, n (%)			p=0.401
African-American	2 (5.3)	4 (2.4)	
Asian	2 (5.3)	3 (1.8)	
Caucasian	33 (86.8)	144 (86.7)	
Not reported	0 (0)	2 (1.2)	
Other	1 (2.6)	13 (7.8)	
Pathological stage, n (%)			p<0.001
pT2a	0 (0)	12 (7.2)	
pT2b	3 (7.9)	5 (3.0)	
pT2c	16 (42.1)	114 (68.7)	
pT3a	10 (26.3)	27 (16.3)	
pT3b	9 (23.7)	8 (4.8)	
ISUP grade, n (%)			p<0.001
1	3 (7.9)	67 (40.4)	
2	13 (34.2)	76 (45.8)	
3	6 (15.8)	13 (7.8)	
4	5 (13.2)	3 (1.8)	
5	11 (28.9)	7 (4.2)	
Surgical Margins, n (%)			p=0.060
Focal	10 (26.3)	20 (12.0)	
Free of tumour	27 (71.1)	144 (86.7)	
Widespread	1 (2.6)	2 (1.2)	

The TMA spots were cores (0.6 mm in diameter) from the highest-grade tumour nodule. Random subsamples were taken in quadruplicate for each case. The whole slides were scanned using a Hamamatsu NanoZoomer-XR slide scanner at 0.23 /px. TMA core images were extracted using QuPath (v0.2.3,[27]). We discarded analysis of cores with less than 25% tissue. The cores were manually checked (HP) for prostate cancer, excluding 535 cores without clear cancer cells present in the TMA cross-section, resulting in a total of 2343 TMA spots. The nested case-control set was split based on the matched pairs into a development set (268 unique pairs), and a test set (91 pairs); the latter was used for evaluation only. We leveraged cross-validation by subdividing the development into three folds to tune the models on different parts of the development set. We divided paired patient, randomly, keeping into account the distribution of the matched variables. The random assignment was done using the scikit-multilearn package[28], specifically the 'IterativeStratification' method in 'skmultilearn.model_selection'. After splitting the dataset into training and test, we split the training dataset into three folds using the same method for the cross-validation.

To validate the DLS biomarker on a fully independent external set, we used the cohort from New York Langone Medical Center. This external validation cohort consists of 204 patients with localized prostate cancer treated with radical prostatectomy between 2001 and 2003. Patients were followed for outcome until 2019, with a median follow-up of 5 years. Biochemical recurrence was defined as either a single PSA measurement of 0.4 ng/m or PSA level of 0.2 ng/ml followed by increasing PSA values in subsequent follow-up. Cores were sampled from the largest tumour focus or any higher-grade focus ($>$ 3mm). Subsamples were taken in quadruplicate for each case. Images were scanned using a Leica Aperio AT2 slide scanner at 0.25 /px.

Model details

For developing the convolutional neural networks (CNNs) we used PyTorch[29]. As an architecture, we used ResNet50-D[30] pretrained on ImageNet from PyTorch Image Models[31]. We used the Lookahead optimizer[32] with RAdam[33], with a learning rate of 2e-4 and mini-batch size of 16 images. We used weight decay (7e-3), and a drop-out layer (p=0.15) before the final fully-connected layer. We used EfficientNet-style[34] dropping of residual connections (p=0.3) as implemented in PyTorch Image Models. We used Bayesian Optimization to find the optimal values.

We resized the TMAs to 1.0 mu/pixel spacing and cropped to 768x768 pixels. Extensive data augmentations were used to promote generalization. The transformations were: flipping, rotations, warping, random crop, HSV color augmentations, jpeg compression, elastic transformations, Gaussian blurring, contrast alterations, gamma alterations, brightness alterations, embossing, sharpening, Gaussian noise and cutout[35]. Augmentations were implemented by albumentations[36] and fast.ai[37].

TMA spots from cases experiencing recurrence were assigned a value of 0-4, depending on the year on which the first event, either biochemical recurrence, metastases, or prostate cancer-related death, was recorded, with 0 meaning recurrence within a year, 4 meaning after 4+ years. TMA spots from cases without an event were also assigned the label 4.

We validated the model on the development validation fold each epoch with a moving average of the weights from 5 subsequent epochs. We used the concordance index as a metric to decide which model performed the best.

As the final prediction at the patient level, the TMA spot with the highest score was used. The final DLS consists of an ensemble of 15 convolutional neural networks. Using cross-validation as described above, 15 networks were trained for each fold, of which the five best performing were used for the DLS.

Figure 1. Overview of the methods summarizing the biomarker development and the Automatic Concept Explanations (ACE) process. Cores were extracted from TMA slides and used to train a neural network to predict the years to biochemical recurrence. On the nested case-control test set, a matched analysis was performed. For ACE, patches were generated from the cores, inferenced through the network and clustered based on their intermediate features.

Statistical analysis

For primary analysis of the nested case-control study, odds ratios (OR) and 95% confidence intervals (CI) were calculated using conditional logistic regression, following Dluzniewski et al.[38]. Due to the study design, calculating hazard ratios using a Cox proportional hazard regression is not appropriate. For the primary analysis, the continuous DLS marker was given as the only variable. For a secondary analysis, we added the non-matched variables PSA, positive surgical margins, and a binned indicator variable for year of surgery. Since matching was done on Gleason sum, and our goal was to identify pat-

terns beyond currently used Gleason patterns, we corrected for the residual differences of the ISUP grade between cases and control (see Table 1). A correction was performed by adding a continuous covariate since, due to the small differences, an indicator covariate did not converge. Analysis was done using the lifelines Python package (v. 0.25.10)[39] with Python (v. 3.7.8). Since the DLS predicts the time to recurrence, high values indicate a low probability of recurrence. We multiplied the DLS output by -1 to make the analysis more interpretable. For three patients (1 from the Johns Hopkins cohort and 2 from the New York Langone cohort), PSA values were missing and were therefore replaced by the median.

For primary analysis of the New York Langone cohort, we calculated hazard ratios (HR) using a Cox proportional hazards regression. We report a secondary multivariable analysis including indicator variables for relevant clinical covariates, Gleason sum, pathological stage, and surgical margin status. We tested the proportional hazards assumption as satisfactory (every p-value above 0.01) using the Pearson correlation between the residuals and the rank of follow-up time. Kaplan Meier plots were generated for the New York Langone cohort. Due to the nested case-control design for the Johns Hopkins set, this set could not be visualized in a Kaplan Meier plot.

Automatic Concept Explanations

To generate concepts, we picked the best performing single CNN from the DLS based on its validation set fold. We used a combination of the methods of Yeh *et al.*, 2020[40] and Ghorbani *et al.*, 2019[23].

We tiled the TMA images into 256x256 patches within the tissue, discarding patches with more than 50% whitespace. These patches were padded to the original input shape of the CNN (768x768 pixels). The latent space of layer 42 of 50 was saved for each tile. Afterwards, we used PCA (50 components) to lower the dimensionality and then performed k-means (k=15) to cluster the latent spaces.

In contrast to Yeh *et al.* and Ghorbani *et al.*, we did not sort the concepts on completeness of the explanations or importance for prediction of individual samples. We sorted the concepts to find interesting new patterns related to recurrence across images by ranking the concepts based on the DLS score of the TMA spot from which they originated.

For each concept, 25 examples were randomly picked and visually inspected

by a pathologist (JvI), with a special interest in uropathology, blinded to the case characteristics and prediction of the network.

Results

The DLS system was developed on the Johns Hopkins cohort with 2343 TMA spots of 685 included unique patients (39 patients were excluded due to insufficient tumour amount in the cores). 492 patients were recurrence cases (72%). The 685 included patients were split into a development set of 503 unique patients and a test set of 91 matched pairs of cases and controls (182 unique patients).

In the external validation cohort, 38 out of the 204 patients (19%) had biochemical recurrence after complete remission, PSA nadir after 3 months post-prostatectomy. From the 204 patients, 620 TMA spots were included. Clinical characteristics of the cohorts can be found in Table 1 and Table 2.

The DLS marker showed a strong association in the primary analyses on the test set of the Johns Hopkins cohort with an OR of recurrence of 3.28 (95% CI 1.73-6.23; $p<0.005$) per unit increase, with DLS system continuous output ranging from 0-3, with two cases below 0 (-0.27 and -0.24) (Table 3).

In addition, for the John Hopkins cohort, we checked for confounding by ISUP grade, PSA level at diagnosis, positive surgical margins, and year of prostatectomy. Neither covariate was found to bias the estimates of effect substantially. The biomarker maintained a strong correlation of OR 3.32 (CI 1.63 - 6.77; $p=0.001$) per unit increase, adjusting for these factors and the continuous term for the residual difference between cases and controls in the ISUP grade.

In the univariable analysis, the DLS marker was strongly associated with recurrence in the New York Langone external validation cohort with an HR of 5.78 (95% CI 2.44-13.72; $p<0.005$) per unit increase. In the multivariate model, including ISUP grade and the other prognostic indicators in addition to the DLS biomarker, the DLS biomarker was still strongly associated with recurrence with an HR of 3.02 (CI 1.10 - 8.29; $p=0.03$) per unit increase. Kaplan Meier curves based on a median cut-off, and four-group categorization, show a clear separation of the low-risk and high-risk groups (Figure 3).

Table 3: Conditional logistic regression analyses of the Johns Hopkins test set.

Covariate	**Matched analysis Johns Hopkins (OR)[1]**	**Multivariate analysis Johns Hopkins (OR)**
Biomarker	3.28 (CI 1.73 - 6.23; p<0.005)	3.32 (CI 1.63 - 6.77; p=0.001)
preop. PSA (ng/mL)		1.04 (CI 0.99 - 1.10; p=0.10)
Surgical margins (pos)		1.69 (CI 0.69 - 4.18; p=0.25)
ISUP grade (cont.)*		1.34 (CI 0.64 - 2.82; p=0.44)
Mean year of surgery		
1992 - 1994 (n=75)		*1.0*
1994 - 1997 (n=55)		3.35 (CI 1.13 - 9.91; p=0.03)
1997 - 2001 (n=52)		8.22 (CI 2.38 - 28.37; p=0.0009)

[1] Cases and controls were matched on age at surgery, race, pathologic stage, and Gleason sum in the prostatectomy specimen.
[2] The ISUP grade covariate was added to correct for the residual differences left after matching cases with controls on prostatectomy Gleason sum.

Table 4: Cox proportional hazard analyses of New York Langone external validation cohort.

Covariate	**Univariate analysis NYU (HR)**	**Multivariate analysis NYU (HR)**
Biomarker	4.79 (CI 2.09 - 10.96; p=0.0002)	3.02 (CI 1.10 - 8.29; p=0.03)
preop. PSA (ng/mL)		1.07 (CI 1.02 - 1.12; p=0.004)
ISUP grade		
1		1.0
2		2.64 (CI 0.73 - 9.58; p=0.14)
3		8.74 (CI 2.16 - 35.30; p=0.00)
4		12.78 (CI 2.82 - 57.91; p=0.00)
5		9.60 (CI 2.32 - 39.69; p=0.00)
Pathological stage		
pT2a + b		1.0
pT2c		1.02 (CI 0.27 - 3.80; p=0.98)
pT3a		1.26 (CI 0.28 - 5.67; p=0.77)
pT3b		2.77 (CI 0.66 - 11.62; p=0.16)
Surgical margins		
Free		1.0
Focal		2.13 (CI 0.76 - 5.96; p=0.15)

Table 4: Cox proportional hazard analyses of New York Langone external validation cohort.	
Widespread	0.20 (CI 0.01 - 3.39; p=0.27)

Automatic Concept Explanations provided semantically meaningful concepts (Figure 1). Concepts were identified that correlated with either a relatively rapid or slow biochemical recurrence. Visual inspection by JvI reveals that generally, the concepts with adverse behaviour show mainly Gleason pattern 4 and some Gleason pattern 5, with cribriform configuration in TMAs within the concepts with most adverse behaviour. The two intermediate concepts show mainly stroma and less aggressive growth patterns. The two concepts predicted to be part of late recurrence cases show mainly Gleason 3 patterns, with readily recognizable well-formed glands. See the supplementary materials for a detailed analysis.

Figure 2. Automatic Concepts Explanations. Sorted by their average score for the cores in which the pattern occurs. Showing the two most benign concepts, two intermediate and two aggressive concepts. Green, yellow and red shaded areas indicate 33%, 66% percentiles.
See the supplementary materials for all concepts.

Figure 3. Kaplan Meier plot for New York Langone external validation cohort, Groups were separated using the median DLS biomarker score in this cohort (left) and using four thresholds (right).

Discussion

We have developed a deep-learning-based morphological biomarker for the prediction of prostate cancer biochemical recurrence based on prostatectomy

tissue microarrays. Using a nested case-control study, we trained convolutional neural networks end-to-end with biochemical recurrence data. The DLS marker provides a continuous score based on the speed of biochemical recurrence it perceived. The DLS marker had an OR of 3.32 (CI 1.63 - 6.77; p=0.001) per unit increase for the test set, and an HR of 3.02 (CI 1.10 - 8.29; p=0.03) per unit increase for the external validation set. These findings support our hypothesis that there is more morphological information in the tissue besides the ISUP grade.

In the Kaplan Meier plot (Figure 3) the biomarker especially seems able to separate men with relatively rapid recurrence from men without (<5 years). However, we hypothesize that the decreased long-term separation in those survival curves is less due to the training cohort containing a median follow-up for four years. Furthermore, we choose to group patients together with more than four years of no biochemical recurrence, This limits the model's capabilities to differentiate patients with very late recurrence. Additionally, due to the limitations of the morphology of the present tumour to inform about long-term outcomes (e.g., cells that escaped the primary tumour may subsequently acquire genomic changes that influence recurrence). Furthermore, it should be noted that the number of at-risk patients was small at these long-term time points.

The nested case-control study contained follow-up information in timespans of years, this limited the use of survival based loss functions[41]. When more granular follow-up information is at hand, future work could investigate usage of Cox regression based loss functions to better leverage the information of the clinical cohort.

The DLS marker showed strong and similar association in both cohorts prepared at different pathology laboratories, which supports the robustness to differences in tissue preparation, staining protocols and scanners.

We showed that Automatic Concept Explanation may be helpful to find concepts correlated with good and poor prognosis. The most discriminatory concepts followed the morphological patterns of Gleason grading. Well-defined prostate cancer glands were predicted to undergo biochemical recurrence later than disorganized sheets of prostate cancer cells. These concepts support the DLS system capturing the expected morphological patterns in support of the validity of the DLS approach.

This study focused on the use of deep learning to automatically discover features relevant for biochemical recurrence prediction. Compared to before-mentioned studies on prostate cancer prognostics models[21], we are the first paper to directly optimize a neural network from prostatectomy tissue towards biochemical recurrence. Additionally, we report that training towards the biochemical recurrence endpoint results in patterns in the networks' features aligning with the ISUP grading.

In the increasing digitalisation of pathology labs, our DLS marker may be applied on digitally chosen regions of interest. Our marker is trained on tissue microarray spots that were selected at the highest grade cancer focus. Furthermore, it has to be noted that a TMA core allows for only limited assessment of the overall prostate cancer growth patterns. Since these tissue cores represent only limited samples from what is usually a much larger tumour lesion, the potential more aggressive patterns may still be present outside of the chosen regions, including regions of potential extraprostatic extension and perineural invasion. Validation will need to be done on entire prostatectomy sections and across cancer foci.

There have been improvements to prostate cancer grading[10], and recently the cribriform pattern is suggested to be important for prognostics[13]. However, the evaluation of this pattern can show a range of inter-observer variability[43], although a recent consensus approach could help decrease this variability[44]. Although we certainly have to keep in mind all the before-mentioned limitations, our findings are in line with outcomes concerning adverse behaviour in earlier work. The DLS system identified a concept that consisted of fields with cribriform-like growth patterns. This cribriform-like growth pattern was found to be part of the concept that was most associated with early recurrent cases.

The results in this study are limited to newer insights of prostate cancer growth, information on cribriform-growth and intraductal carcinoma were not readily available for use in the multivariate analysis, although the external validation cohort was graded using the 2005 ISUP consensus[45] partly encoding the presence of cribriform growth inside the ISUP grade.

Although biochemical recurrence is a common end-point to study prostate cancer progression, a clinical utility would be mostly found in assessing time-to-metastases or death. However, time-wise, they are typically significantly further separated from the surgical event, making it harder to identify rela-

tionships between tissue morphology and these end-points. Nevertheless, we would like to investigate them in the future.

Conclusions

In summary, we have developed a deep-learning-based visual biomarker for prostate cancer recurrence based on tissue microarray hotspots of prostatectomies. The DLS marker provides a continuous score predicting the speed of biochemical recurrence. We obtained an odds ratio of 3.32 (CI 1.63 - 6.77; p=0.001) for a nested case-control study from Johns Hopkins Hospital, matched on Gleason sum on other factors. Additionally, we obtained an HR of 3.02 (CI 1.10 - 8.29; p=0.03) for an external validation cohort from the New York Langone hospital, adjusted for ISUP grade, pathological stage, preoperative PSA concentration, and surgical margins status. Thus, this visual biomarker may provide prognostic information in addition to the current morphological ISUP grade.

Acknowledgments

This work was supported by the Dutch Cancer Society under Grant KUN 2015-7970.

This work was additionally supported by the Department of Defense Prostate Cancer Research Program, DOD Award No W81XWH-18-2-0013, W81XWH-18-2-0015, W81XWH-18-2-0016, W81XWH-18-2-0017, W81XWH-18-2-0018, W81XWH-18-2-0019 PCRP Prostate Cancer Biorepository Network (PCBN), DAMD17-03-1-0273, and supported by Prostate Cancer NCI-NIH grant (P50 CA58236).

References

Appendix can be found here:

1. Sung H, Ferlay J, Siegel RL, et al. Global cancer statistics 2020: GLOBOCAN estimates of incidence and mortality worldwide for 36 cancers in 185 countries. *CA Cancer J Clin.* Published online February 2021.
2. US Preventive Services Task Force, Grossman DC, Curry SJ, et al. Screening for prostate cancer: US preventive services task force recommendation statement. *JAMA*. 2018;319(18):1901-1913.

3. Goonewardene SS, Phull JS, Bahl A, Persad RA. Interpretation of PSA levels after radical therapy for prostate cancer. *Trends Urol Men s Health*. 2014;5(4):30-34.

4. Amling CL, Blute ML, Bergstralh EJ, Seay TM, Slezak J, Zincke H. Long-term hazard of progression after radical prostatectomy for clinically localized prostate cancer: Continued risk of biochemical failure after 5 years. *J Urol*. 2000;164(1):101-105.

5. Freedland SJ, Humphreys EB, Mangold LA, et al. Risk of prostate Cancer–Specific mortality following biochemical recurrence after radical prostatectomy. *JAMA*. 2005;294(4):433-439.

6. Han M, Partin AW, Pound CR, Epstein JI, Walsh PC. Long-term biochemical disease-free and cancer-specific survival following anatomic radical retropubic prostatectomy. The 15-year johns hopkins experience. *Urol Clin North Am*. 2001;28(3):555-565.

7. Van den Broeck T, Bergh RCN van den, Arfi N, et al. Prognostic value of biochemical recurrence following treatment with curative intent for prostate cancer: A systematic review. *European Urology*. 2019;75:967-987.

8. Epstein JI, Egevad L, Amin MB, Delahunt B, Srigley JR, Humphrey PA. The 2014 international society of urological pathology (ISUP) consensus conference on gleason grading of prostatic carcinoma. *American Journal of Surgical Pathology*. 2016;40:244-252.

9. Mottet N, Bergh RCN van den, Briers E, et al. EAU-EANM-ESTRO-ESUR-SIOG guidelines on prostate cancer—2020 update. Part 1: Screening, diagnosis, and local treatment with curative intent. *Eur Urol*. 2021;79(2):243-262.

10. Epstein JI. An update of the gleason grading system. *J Urol*. 2010;183(2):433-440.

11. Pierorazio PM, Walsh PC, Partin AW, Epstein JI. Prognostic gleason grade grouping: Data based on the modified gleason scoring system. *BJU Int*. 2013;111(5):753-760.

12. Epstein JI, Zelefsky MJ, Sjoberg DD, et al. A contemporary prostate cancer grading system: A validated alternative to the gleason score. *Eur Urol*. 2016;69(3):428-435.

13. Leenders GJLH van, Kwast TH van der, Grignon DJ, et al. The 2019 international society of urological pathology (ISUP) consensus conference on grading of prostatic carcinoma. *Am J Surg Pathol*. 2020;44(8):e87-e99.

14. Ozkan TA, Eruyar AT, Cebeci OO, Memik O, Ozcan L, Kuskonmaz I. Interobserver variability in gleason histological grading of prostate cancer. *Scand J Urol*. 2016;50(6):420-424.

15. Krizhevsky A, Sutskever I, Hinton GE. Imagenet classification with deep convolutional neural networks. *Adv Neural Inf Process Syst*. 2012;25:1097-1105.

16. Swiderska-Chadaj Z, Hebeda KM, Brand M van den, Litjens G. Artificial intelligence to detect MYC translocation in slides of diffuse large b-cell lymphoma. *Virchows Arch*. Published online September 2020.

17. Coudray N, Ocampo PS, Sakellaropoulos T, et al. Classification and mutation prediction from non–small cell lung cancer histopathology images using deep learning. *Nat Med*. 2018;24(10):1559-1567.

18. Wulczyn E, Steiner DF, Moran M, et al. Interpretable survival prediction for colorectal cancer using deep learning. *NPJ Digit Med*. 2021;4(1):71.

19. Muhammad H, Xie C, Sigel CS, Doukas M, Alpert L, Fuchs TJ. EPIC-Survival: End-to-end part inferred clustering for survival analysis, featuring prognostic stratification boosting. *arXiv*. Published online 2021:2101.11085v2.

20. Leo P, Janowczyk A, Elliott R, et al. Computer extracted gland features from H&E predicts prostate cancer recurrence comparably to a genomic companion diagnostic test: A large multi-site study. *npj Precision Oncology*. 2021;5.

21. Yamamoto Y, Tsuzuki T, Akatsuka J, et al. Automated acquisition of explainable knowledge from unannotated histopathology images. *Nat Commun*. 2019;10:5642.

22. Rudin C. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence*. 2019;1(5):206-215.

23. Ghorbani A, Wexler J, Zou J, Kim B. Towards automatic concept-based explanations. *arXiv*. Published online 2019:1902.03129v3.

24. Toubaji A, Albadine R, Meeker AK, et al. Increased gene copy number of ERG on chromosome 21 but not TMPRSS2-ERG fusion predicts outcome in prostatic adenocarcinomas. *Mod Pathol.* 2011;24(11):1511-1520.

25. Prostate cancer biorepository network.

26. Wang MH, Shugart YY, Cole SR, Platz EA. A simulation study of control sampling methods for nested case-control studies of genetic and molecular biomarkers and prostate cancer progression. *Cancer Epidemiol Biomarkers Prev.* 2009;18(3):706-711.

27. Bankhead P, Loughrey MB, Fernández JA, et al. QuPath: Open source software for digital pathology image analysis. *Sci Rep.* 2017;7(1):16878.

28. Szymanski P, Kajdanowicz T. Scikit-multilearn: A scikit-based python environment for performing multi-label classification. *J Mach Learn Res.* 2019;20(1):209-230.

29. Paszke A, Gross S, Massa F, et al. PyTorch: An imperative style, High-Performance deep learning library. Published online December 2019. https://arxiv.org/abs/1912.01703

30. He T, Zhang Z, Zhang H, Zhang Z, Xie J, Li M. Bag of tricks for image classification with convolutional neural networks. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition.*; 2019:558-567.

31. Wightman R. PyTorch image models. Published online 2021.

32. Zhang MR, Lucas J, Hinton G, Ba J. Lookahead optimizer: K steps forward, 1 step back. Published online July 2019. https://arxiv.org/abs/1907.08610

33. Liu L, Jiang H, He P, et al. On the variance of the adaptive learning rate and beyond. Published online August 2019. https://arxiv.org/abs/1908.03265

34. Tan M, Le Q. Efficientnet: Rethinking model scaling for convolutional neural networks. In: *International Conference on Machine Learning.* PMLR; 2019:6105-6114.

35. DeVries T, Taylor GW. Improved regularization of convolutional neural networks with cutout. Published online August 2017. https://arxiv.org/abs/1708.04552

36. Buslaev A, Iglovikov VI, Khvedchenya E, Parinov A, Druzhinin M, Kalinin AA. Albumentations: Fast and flexible image augmentations. *Information*. 2020;11(2):125.

37. Howard J, Gugger S. Fastai: A layered API for deep learning. *Information*. 2020;11(2):108.

38. Dluzniewski PJ, Wang MH, Zheng SL, et al. Variation in IL10 and other genes involved in the immune response and in oxidation and prostate cancer recurrence. *Cancer Epidemiol Biomarkers Prev*. 2012;21(10):1774-1782.

39. Davidson-Pilon C, Kalderstam J, Jacobson N, et al. CamDavidson-Pilon/lifelines: 0.25.10. Published online 2021.

40. Yeh CK, Kim B, Arik S, Li CL, Pfister T, Ravikumar P. On completeness-aware Concept-Based explanations in deep neural networks. *Adv Neural Inf Process Syst*. 2020;33.

41. Kvamme H, Borgan Ø, Scheel I. Time-to-Event prediction with neural networks and cox regression. *J Mach Learn Res*. 2019;20(129):1-30.

42. Hollemans E, Verhoef EI, Bangma CH, et al. Cribriform architecture in radical prostatectomies predicts oncological outcome in gleason score 8 prostate cancer patients. *Mod Pathol*. 2021;34(1):184-193.

43. Slot MA van der, Hollemans E, Bakker MA den, et al. Inter-observer variability of cribriform architecture and percent gleason pattern 4 in prostate cancer: Relation to clinical outcome. *Virchows Arch*. 2021;478(2):249-256.

44. Kwast TH van der, Leenders GJ van, Berney DM, et al. ISUP consensus definition of cribriform pattern prostate cancer. *Am J Surg Pathol*. Published online May 2021.

45. Epstein JI, Allsbrook WC Jr, Amin MB, Egevad LL, ISUP Grading Committee. The 2005 international society of urological pathology (ISUP) consensus conference on gleason grading of prostatic carcinoma. *Am J Surg Pathol*. 2005;29(9):1228-1242.

Discussion

In this PhD thesis, we addressed the main problem of the dimensionality of whole slide images in computational pathology and prostate cancer prognosis. These images are so large that previous works typically focused on the patch level or employed pre-trained mechanisms to predict clinically relevant endpoints based on the whole slides. In the first chapter of this thesis, we proposed a new method called "streaming," which aims to solve the dimensionality problem with a memory-efficient implementation of convolutions. We demonstrated that this approach is numerically equivalent to a convolutional neural network, without batch normalization, on a subset of ImageNet. Additionally, we showed that increasing the resolution leads to higher performance due to the increase in detail. The second chapter of the thesis employed this new method to predict prostate cancer on whole slide images. When compared to a top multiple instance learning method, the streaming method outperformed it and demonstrated better generalization to an unseen scanner. We also showed that using gradient saliency, the method can provide some explainability by localizing where the network is focusing its attention. These gradient saliency maps correspond to the predictions of a multiple instance learning network, indicating their correlation to the classification endpoint. In the third chapter, we demonstrated that our method can predict prostate cancer recurrence in patients after prostatectomy. By using histology data, we could further improve our prognostication for these patients beyond typical clinical variables, such as Gleason scores. This chapter laid the foundation for extracting these features from whole slide images since, in this work, we used predefined extracted areas from the original slide based on the highest grade tumor. However, we also considered the possibility that there might be more information on the slide to predict prognostics. This hypothesis was explored in the final chapter, where we analyzed whole slide images of a population cohort of patients who had undergone prostatec-

tomy. We attempted to predict the time to biochemical recurrence for these patients. Although we could not achieve statistically significant results due to the size of the dataset, there was a discernible signal in the slides. Using explainability methods, we demonstrated that the network focuses on the tumor and other relevant areas. This lays the groundwork for fully learning end-to-end, clinically interesting endpoints from histology images while harnessing the full potential of neural networks to find relevant features without manual feature engineering. One could argue that working on patches adds assumptions to the task and lacks context due to cropping the slide. However, deep learning has shown that neural networks can learn these assumptions and signals themselves, given enough data and appropriate labels. Interestingly, chapter 2 revealed that the streaming method is data-efficient, even compared to multiple instance learning, which utilizes patches. We hypothesize that this is because streaming employs very large feature maps that are harder to overfit. Furthermore, we use an aggregation layer later in the network while the feature map is still relatively large, making it more difficult to overfit to specific noise.

Generalization

One of the challenges in this field is the problem of model generalization. Many publications focus on creating a variation of a model for a specific dataset, which may not generalize well to other datasets, containing slides of different labs or different scanners. This may be the result of differences in color distribution, compression, or image properties in various datasets. To bring these algorithms into the clinic, it is essential to address the bias in the networks.

Academic and industry overlap

Unfortunately, the current academic incentives means that a lot of effort in the research community is not focused on these core problems in the field. Several factors contribute to this, such as funding concerns, the need for PhD students to publish, and the competitive nature of the field. This leads to an unfair competition where larger AI firms with more resources drive the field forward, while academia often lags behind. [TODO fix dictation]So another class that I see of the problem of tracking smallest plural fast pathologist was using deep learning as a alright, let's switch and put them in it's still fairly

easy to get a tornado set for certain Mainly in the event of a task with the baton of this and trading them off the shelf for small variation of existing model on them.This is almost guaranteed to work and it might be almost the wonders of finally creating a publication, given the incentives of a PhD, in this country, everyone needs to publish a certain number of papers.It is very appealing to go this route.Since it is still happening in radiology of fields, which has been digitized way earlier than pathology. I'm afraid that these publications will keep getting written and worked on in the near future. It is telling that when asking around, people rarely read each other's research in this field, attempting to read the papers of the bigger AI discussing methods. In my opinion, this leads to research that reinvents the wheel all of the time and doesn't build upon each other. And given the hype of deep learning, given a big enough dataset, it is still possible to publish in a very high impact journal.We published a Gleason grading algorithm in a very high impact journal. And, cynically, we could say that we already knew it was going to work if we did not do that. AI researchers in this field will not learn that much from this paper, physicians may learn that it's possible to do these tasks at an expert level. But I hope that this is soon not a surprise anymore and everybody knows that given enough data, these algorithms can find the right patterns. Another issue is the career ladder in academia, where personal publications are very important. This may lead to researchers focusing on short-term projects that are not as impactful in the long run. In my opinion, academia, funded by public money and without profit incentives, should focus on long-term goals and moon-shot ideas, leaving short-term projects to companies that can develop and implement them more quickly. On short term research, as mentioned above, like smaller papers, smaller tasks. In my opinion, when you can create an algorithm that is commercially interesting, meaning it can actually be implemented and sold in the clinic and you can develop this algorithm within a year, it may not make sense to develop those in academia, but let companies develop them. Since academia is funded by public money and doesn't have any profit incentive they should focus on the long-term work, on moonshot ideas. With the risk of not getting funding and the fact that PhD students have to publish, make this unlikely to happen. Meaning that right now, academia is doing a lot of similar work as companies do, often with way less funding. This leads to an unfair advantage or unfair competition. The companies often have way more hardware. Especially in the field of image analysis in general. After language processing you see that bigger AI firms with large sums of money drive the field forward. And

academia often hobbles after those companies. Nonetheless, fundamental research remains crucial, and the importance of computational pathology and prostate cancer prognosis should not be underestimated. It is essential to continue exploring new methods and techniques to improve patient outcomes and advance the field.

Bigger picture

The computational pathology field has started working on predicting human-defined features, such as mitotic count, tumor grading, and regular disease classification. What is happening now is that we are deriving more features using deep learning in cohorts of patients where clinical endpoints are available. Since deep learning works on input images, it has the potential to identify interesting features from histology that could predict treatment response. These solutions can assist oncologists in helping patients, providing broader benefits than just automating small tasks for the pathologist. There are still plenty of papers published where one can quickly compile a dataset for a very specific, narrow task, perhaps using just a couple of hundred patients. The approach is to take this task, develop a model, and then attempt to predict outcomes for this task. What we have seen recently in natural language processing (NLP) and other fields is that given enough data and correlations, large-scale models can be successful even when tackling tasks that are not specifically designed for them. We should learn from these NLP foundational models and aim for bigger models, rather than focusing on publishing papers for small, narrow tasks. By expanding our scope and ambitions, we can make greater strides in computational pathology and related fields. One of the really exciting projects in this field is the EU's Big Picture project, where a total of one million slides will be collected. Given enough data and the right techniques, such as self-supervised learning, models can predict or estimate prognosis better than tumor grading. Self-supervised learning networks can discover patterns on their own when provided with sufficient data. These networks have the potential to automatically generate grading schemes and extract valuable information from the data. It is not surprising that self-supervised learning networks can provide more prognostic information than traditional methods used by oncologists. Instead of relying on discrete grades (e.g., 3-5 grades), these models can operate in a continuous manner, enabling a more nuanced understanding of the data. As a result, statistical models can predict or estimate diagnoses more effectively

than the human-based approaches currently in use. In our study, a couple of hundred patients were analyzed using a specific model. We found that by clustering the latent space we could find the cribiform growth pattern, even if they are not specifically targeted. However, we must also be careful with interpreting patterns or, in this case, clustering of data. It is crucial not to overstate the importance of certain patterns, especially since we may not know their relevance in the larger context. The data we can derive from biological tissue, as well as the knowledge encoded in medical literature, can be used to build self-supervised foundational models that can potentially discover relationships we are not yet aware of. Extend.. (all the omics etc) (Although interobserver variability is a significant problem in computational pathology, the model's predictions do not stand alone in determining a patient's treatment. Instead, they contribute to predicting whether a patient will benefit from a specific treatment.)