Processing Megapixel Images with Deep Attention-Sampling Models

Angelos Katharopoulos [1 2] **François Fleuret** [1 2]

Abstract

Existing deep architectures cannot operate on very large signals such as megapixel images due to computational and memory constraints. To tackle this limitation, we propose a fully differentiable end-to-end trainable model that samples and processes only a fraction of the full resolution input image. The locations to process are sampled from an attention distribution computed from a low resolution view of the input. We refer to our method as *attention sampling* and it can process images of several megapixels with a standard single GPU setup. We show that sampling from the attention distribution results in an unbiased estimator of the full model with minimal variance, and we derive an unbiased estimator of the gradient that we use to train our model end-to-end with a normal SGD procedure. This new method is evaluated on three classification tasks, where we show that it allows to reduce computation and memory footprint by an order of magnitude for the same accuracy as classical architectures. We also show the consistency of the sampling that indeed focuses on informative parts of the input images.

1. Introduction

For a variety of computer vision tasks, such as cancer detection, self driving vehicles, and satellite image processing, it is necessary to develop models that are able to handle high resolution images. The existing CNN architectures, that provide state-of-the-art performance in various computer vision fields such as image classification (He et al., 2016), object detection (Liu et al., 2016), semantic segmentation (Wu et al., 2019) etc., cannot operate directly on such images due to computational and memory requirements. To address this issue, a common practice is to downsample the original image before passing it to the network. However, this leads to loss of significant information possibly critical for certain tasks.

(a) (b) (c)

Figure 1: Common practice to process megapixel images with CNNs is to downsample them, however this results in significant loss of information (b, c).

Another research direction seeks to mitigate this problem by splitting the original high resolution image into patches and processing them separately (Hou et al., 2016; Golatkar et al., 2018; Nazeri et al., 2018). Naturally, these methods either waste computational resources on uninformative patches or require ground truth annotations for each patch. However, per patch labels are typically expensive to acquire and are not available for the majority of the available datasets.

The aforementioned limitations are addressed by two disjoint lines of work: the recurrent visual attention models (Mnih et al., 2014; Ba et al., 2014) and the attention based multiple instance learning (Ilse et al., 2018). The first seeks to limit the wasteful computations by only processing some parts of the full image. However, these models result in hard optimization problems that limit their applicability to high resolution images. The second line of work shows that regions of interest can be identified without explicit patch annotations by aggregating per patch features with an attention mechanism. Nevertheless, such methods do not address the computational and memory issues inherent in all patch based models.

This work aims at combining the benefits of both. Towards this goal, we propose an end-to-end trainable model able to handle multi-megapixel images using a single GPU or CPU. In particular, we sample locations of "informative patches"

[1]Idiap Research Institute, Martigny, Switzerland [2]EPFL, Lausanne, Switzerland. Correspondence to: Angelos Katharopoulos <firstname.lastname@idiap.ch>.

Proceedings of the 36^{th} International Conference on Machine Learning, Long Beach, California, PMLR 97, 2019.

from an "attention distribution" computed on a lower resolution version of the original image. This allows us to only process a fraction of the original image. Compared to previous works, due to our attention based weighted average feature aggregation, we are able to derive an unbiased estimator of the gradient of the virtual and intractable "full model" that would process the full-scale image in a standard feed-forward manner, and do not need to resort to reinforcement learning or variational methods to train. Furthermore, we prove that sampling patches from the attention distribution results in the minimum variance estimator of the "full model".

We evaluate our model on three classification tasks and we show that our proposed *attention sampling* achieves comparable test errors with Ilse et al. (2018), that considers all patches from the high resolution images, while being up to **25**$\times$ faster and requiring up to **30**$\times$ less memory.

2. Related Work

In this section, we discuss the most relevant body of work on attention-based models and techniques to process high resolution images using deep neural networks, which can be trained from a scene-level categorical label. Region proposal methods that require per-patch annotations, such as instance-level bounding boxes (Girshick et al., 2014; Redmon et al., 2016; Liu et al., 2016), do not fall in that category.

2.1. Recurrent visual attention models

This line of work includes models that learn to extract a sequence of regions from the original high resolution image and only process these at high resolution. The regions are processed in a sequential manner, namely the distribution to sample the n-th region depends on the previous $n-1$ regions. Mnih et al. (2014) were the first to employ a recurrent neural network to predict regions of interest on the high resolution image and process them sequentially. In order to train their model, which is not differentiable, they use reinforcement learning. In parallel, Ranzato (2014); Ba et al. (2014) proposed to additionally downsample the input image and use it to provide spatial context to the recurrent network. Ramapuram et al. (2018) improved upon the previous works by using variational inference and Spatial Transformer Networks (Jaderberg et al., 2015) to solve the same optimization problem.

All the aforementioned works seek to solve a complicated optimization problem that is non differentiable and is approximated with either reinforcement learning or variational methods. Instead of employing such a complicated model to aggregate the features of the patches and generate dependent attention distributions that result in a hard optimization problem, we propose to use an attention distribution to perform a weighted average of the features, which allows us to directly train our model with SGD.

2.2. Patch based models

Such models (Hou et al., 2016; Liu et al., 2017; Nazeri et al., 2018) divide the high resolution image into patches and process them separately. Due to the lack of per patch annotations, the above models need to introduce a separate method to provide labels for training the patch level network. Instead *attention sampling* does not require any patch annotations and through the attention mechanism learns to identify regions of interest in arbitrarily large images.

2.3. Attention models

Xu et al. (2015) were the first to use soft attention methods to generate image captions. More related to our work is the model of Ilse et al. (2018), where they use the attention distribution to aggregate a bag of features. To apply their method to images, they extract patches, compute features and aggregate them with an attention distribution that is computed from these features. This allows them to infer regions of interest without having access to per-patch labels. However, their model wastes computational resources by handling all patches, both informative and non-informative. Our method, instead, learns to focus only on informative regions of the image, thus resulting in orders of magnitude faster computation while retaining equally good performance.

2.4. Other methods

Jaderberg et al. (2015) propose Spatial Transformer Networks (STN) that learn to predict affine transformations of a feature map than includes cropping and rescaling. STNs employ several localization networks, that operate on the full image, to generate these transformations. As a result, they do not scale easily to megapixel images or larger. Recasens et al. (2018) use a low resolution view of the image to predict a saliency map that is used in conjunction with the differentiable STN sampler to focus on useful regions of the high resolution image by making them larger. In comparison, *attention sampling* focuses on regions by weighing the corresponding features with the attention weights.

3. Methodology

In this section, we formalize our proposed *attention-sampling* method. Initially, we introduce a generic formulation for attention and we show that sampling from the attention distribution generates an optimal approximation in terms of variance that significantly reduces the required computation. In § 3.2.2, we derive the gradient with respect to the parameters of the attention and the feature network through the sampling procedure. In § 3.3 and § 3.4, we

provide the methodology that allows us to speed up the processing of high resolution images using *attention sampling* and is used throughout our experiments.

3.1. Attention in neural networks

Let x, y denote an input-target pair from our dataset. We consider $\Psi(x;\Theta) = g(f(x;\Theta);\Theta)$ to be a neural network parameterized by Θ. $f(x;\Theta) \in \mathbb{R}^{K\times D}$ is an intermediate representation of the neural network that can be thought of as K features of dimension D, e.g. the last convolutional layer of a ResNet architecture or the previous hidden states and outputs of a recurrent neural network.

Employing an attention mechanism in the neural network $\Psi(\cdot)$ at the intermediate representation $f(\cdot)$ is equivalent to defining a function $a(x;\Theta) \in \mathbb{R}_+^K$ s.t. $\sum_{i=1}^K a(x;\Theta)_i = 1$ and changing the definition of the network to

$$\Psi(x;\Theta) = g\left(\sum_{i=1}^{K} a(x;\Theta)_i f(x;\Theta)_i\right), \tag{1}$$

given that the subscript i extracts the i-th row from a matrix or the i-th element from a vector.

3.2. Attention sampling

By definition, $a(\cdot)$ is a multinomial distribution over K discrete elements (e.g. locations in the images). Let I be a random variable sampled from $a(x;\Theta)$. We can rewrite the attention in the neural network $\Psi(\cdot)$ as the expectation of the intermediate features over the attention distribution $a(\cdot)$

$$\Psi(x;\Theta) = g\left(\sum_{i=1}^{K} a(x;\Theta)_i f(x;\Theta)_i\right) \tag{2}$$

$$= g\left(\mathop{\mathbb{E}}_{I\sim a(x;\Theta)}[f(x;\Theta)_I]\right). \tag{3}$$

Consequently, we can avoid computing all K features by approximating the expectation with a Monte Carlo estimate. We sample a set Q of N i.i.d. indices from the attention distribution, $Q = \{q_i \sim a(x;\Theta) \mid i \in \{1,2,\ldots,N\}\}$ and approximate the neural network with

$$\Psi(x;\Theta) \approx g\left(\frac{1}{N}\sum_{q\in Q} f(x;\Theta)_q\right). \tag{4}$$

3.2.1. Relation with importance-sampling

We are interested in deriving an approximation with minimum variance so that the output of the network does not change because of the sampling. In the following paragraphs, we show that sampling from $a(x;\Theta)$ is optimal in that respect.

Let P denote a discrete probability distribution on the K features with probabilities p_i. We want to sample from P such that the variance is minimized. Concretely, we seek P^* such that

$$P^* = \mathop{\text{argmin}}_P \mathop{\mathbb{V}}_{I\sim P}\left[\frac{a(x;\Theta)_I f(x;\Theta)_I}{p_I}\right]. \tag{5}$$

We divide by p_I to ensure that the expectation remains the same regardless of P. One can easily verify that $\mathbb{E}_{I\sim P}\left[\frac{a(x;\Theta)_I f(x;\Theta)_I}{p_I}\right] = \mathbb{E}_{I\sim a(x;\Theta)}[f(x;\Theta)_I]$. We continue our derivation as follows:

$$\mathop{\text{argmin}}_P \mathop{\mathbb{V}}_{I\sim P}\left[\frac{a(x;\Theta)_I f(x;\Theta)_I}{p_I}\right] \tag{6}$$

$$= \mathop{\text{argmin}}_P \mathop{\mathbb{E}}_{I\sim P}\left[\left(\frac{a(x;\Theta)_I}{p_I}\right)^2 \|f(x;\Theta)_I\|_2^2\right] \tag{7}$$

$$= \mathop{\text{argmin}}_P \sum_{i=1}^{K} \frac{a(x;\Theta)_i^2}{p_i} \|f(x;\Theta)_i\|_2^2. \tag{8}$$

The minimum of equation 8 is

$$p_i^* \propto a(x;\Theta)_i \|f(x;\Theta)_i\|_2, \tag{9}$$

which means that sampling according to the attention distribution is optimal when we do not have information about the norm of the features. This can be easily enforced by constraining the features to have the same L_2 norm.

3.2.2. Gradient derivation

In order to use a neural network as our attention distribution we need to derive the gradient of the loss with respect to the parameters of the attention function $a(\cdot;\Theta)$ through the sampling of the set of indices Q. Namely, we need to compute

$$\frac{\partial \frac{1}{N}\sum_{q\in Q} f(x;\Theta)_q}{\partial \theta} \tag{10}$$

for all $\theta \in \Theta$ including the ones that affect $a(\cdot)$.

By exploiting the Monte Carlo approximation and the multiply by one trick, we show that

$$\frac{\partial}{\partial\theta}\frac{1}{N}\sum_{q\in Q} f(x;\Theta)_q \approx \mathop{\mathbb{E}}_{I\sim a(x;\Theta)}\left[\frac{\frac{\partial}{\partial\theta}\left[a(x;\Theta)_I f(x;\Theta)_I\right]}{a(x;\Theta)_I}\right]. \tag{11}$$

In equation 11, the gradient of each feature is weighed inversely proportionally to the probability of sampling that feature. This result is expected, because the "effect" of rare samples should be increased to account for the low observation frequency (Kahn & Harris, 1951). This allows

us to derive the gradients of our *attention sampling* method as follows:

$$\frac{\partial}{\partial\theta}\frac{1}{N}\sum_{q\in Q} f(x;\Theta)_q = \frac{1}{N}\sum_{q\in Q}\frac{\frac{\partial}{\partial\theta}\left[a(x;\Theta)_q f(x;\Theta)_q\right]}{a(x;\Theta)_q}, \tag{12}$$

which requires computing only the rows of $f(\cdot)$ for the sampled indices in Q. Due to lack of space, a detailed derivation of equation 11, can be found in our supplementary material.

3.2.3. Sampling without replacement

In our initial analysis, we assume that Q is sampled i.i.d. from $a(x;\Theta)$. However, this means that it is probable to sample the same element multiple times, especially as the entropy of the distribution decreases during the training of the attention network. To avoid computing a feature multiple times and to make the best use of the available computational budget we propose sampling without replacement.

We model sampling without replacement as follows: Initially, we sample a position i_1 with probability $p_1(i) \propto a(x;\Theta)_i \,\forall i$. Subsequently, we sample the second position i_2, given the first, with probability $p_2(i \mid i_1) \propto a(x;\Theta)_i \,\forall i \neq i_1$. Following this reasoning, we can define sampling the n-th position with probability

$$\forall i \notin \{i_1, i_2, \ldots, i_{n-1}\},\quad p_n(i \mid i_1, i_2, \ldots, i_{n-1}) \propto a(x;\Theta)_i \tag{13}$$

Simply averaging the features, as in equation 4, would result in a biased estimator. Instead, we use

$$\mathop{\mathbb{E}}_{I_1,I_2,\ldots,I_n}\left[\sum_{k=1}^{n-1} a(x;\Theta)_{I_k} f(x;\Theta)_{I_k} + \right. \tag{14}$$

$$\left. f(x;\Theta)_{I_n}\sum_{t\notin\{I_1,I_2,\ldots,I_{n-1}\}} a(x;\Theta)_t\right] = \tag{15}$$

$$\mathop{\mathbb{E}}_{I_1,I_2,\ldots,I_n}\left[\sum_{i=1}^{K} a(x;\Theta)_i f(x;\Theta)_i\right] = \tag{16}$$

$$\mathop{\mathbb{E}}_{I\sim a(x;\Theta)}\left[f(x;\Theta)_I\right]. \tag{17}$$

We assume that I_1 to I_n are sampled from $p_1(i)$ to $p_n(i)$ accordingly. Following the reasoning of § 3.2.2, we compute the gradient through the sampling in an efficient and numerically stable way. The complete analysis is given in the supplementary material.

3.3. Multi-resolution data

For most implementations of attention in neural networks, $a(\cdot)$ is a function of the features $f(\cdot)$ (Ilse et al., 2018). This means that in order to compute the attention distribution we need to compute all the features. However, in order to take advantage of our Monte Carlo Estimation of equation 4 and avoid computing all K features, we use a lower resolution view of the data. This allows us to gain significant speedup from *attention sampling*.

Given an image $x \in \mathbb{R}^{H\times W\times C}$ where H, W, C denote the height, width and channels respectively, and its corresponding view $V(x,s) \in \mathbb{R}^{h\times w\times C}$ at scale s we compute the attention as

$$a(V(x,s);\Theta) : \mathbb{R}^{h\times w\times C} \to \mathbb{R}^{hw}, \tag{18}$$

where $h < H$ and $w < W$. We also define a function $P(x,i)$ that extracts a patch from the full resolution image x centered around the corresponding i-th pixel in $V(x,s)$.

Based on the above, we derive a model capable of only considering few patches from the full size image x, as follows:

$$\Psi(x) = g\left(\sum_{i=1}^{hw} a(V(x,s))_i f(P(x,i))\right) \tag{19}$$

$$\approx g\left(\frac{1}{N}\sum_{q\in Q} f(P(x,q))\right). \tag{20}$$

Note that both the attention and feature functions have trainable parameters Θ which we omit for clarity. In the formulation of equation 20, we do not consider the location of the sampled patches $P(x,q)$. This is not an inherent limitation of the model since we can simply pass the location as a parameter in our feature function $f(\cdot)$.

3.4. Implementation details

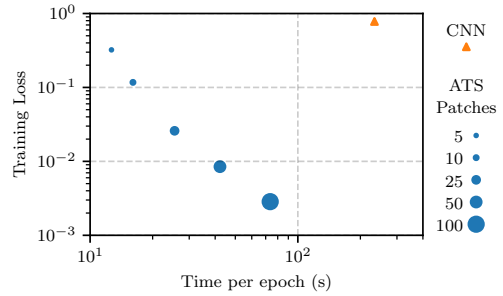
In this section, we discuss the specifics of our proposed *attention sampling*. Equation $f(\cdot)$ is implemented by a neural network which we refer to as *feature network*. Similarly $a(\cdot)$ is another neural network, typically, significantly smaller, referred to as *attention network*. Finally, function $g(\cdot)$ is a linear classification layer.

In order to control the exploration-exploitation dilemma we introduce an entropy regularizer for the attention distribution. Namely given a loss function $\mathcal{L}(x,y;\Theta)$ we use

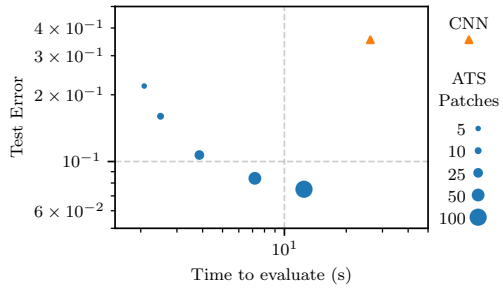
$$\mathcal{L}'(x,y;\Theta) = \mathcal{L}(x,y;\Theta) - \lambda\mathcal{H}(a(x;\Theta)), \tag{21}$$

where $\mathcal{H}(x)$ denotes the entropy of the distribution x. This regularizer prevents the attention network from quickly deciding which patches are informative. This results in an exploration of the available patch space during the initial stage of training. Due to lack of space we evaluate the impact of this regularizer qualitatively in our supplementary.

As already mentioned in § 3.2.1, normalizing the features in terms of the L_2 norm guarantees that the attention distribution produces the minimum variance estimator of the



(a) Performance on training set



(b) Performance on test set

Figure 2: Comparison of *attention sampling* (ATS) with a CNN on Megapixel MNIST. We observe that we can trade optimization accuracy for time by sampling fewer patches.

"full model"; thus in all the feature networks we add L_2 normalization as the final layer.

4. Experimental evaluation

In this section, we analyse experimentally the performance of our *attention sampling* approach on three classification tasks. We showcase the ability of our model to focus on informative parts of the input image which results in significantly reduced computational requirements. We refer to our approach as *ATS* or *ATS-XX* where XX denotes the number of sampled patches. Note that we do not consider per-patch annotations for any of the used datasets. The code used for the experiments can be found in https://github.com/idiap/attention-sampling.

4.1. Introduction

4.1.1. Baselines

Most related to our method is the patch based approach of Ilse et al. (2018) that implements the attention as a function of the features of each patch. For the rest of the experiments, we refer to this method as *Deep MIL*. For *Deep MIL*, we specify the patches to be extracted from each high resolution image by a regular grid of varying size depending on the dimensions of the input image and the patch size. Note that the architecture of the feature network and the patch size used for *Deep MIL* is always the same as the ones used for our *attention sampling* method.

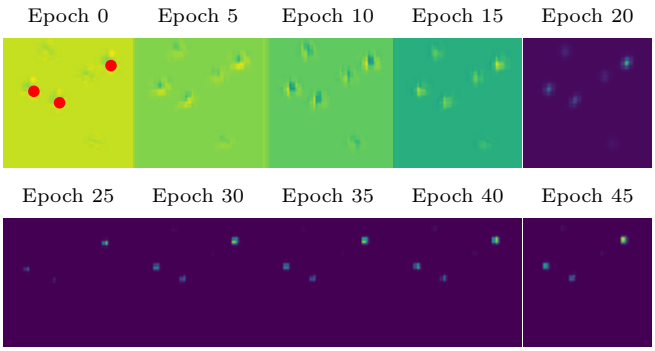


Figure 3: The evolution of the attention distribution on Megapixel MNIST. Yellow means higher attention. At the first image (epoch 0) we mark the position of the digits with the red dots. The attention finds the three digits and focuses on them instead of the noise which can be clearly seen in epochs 10 and 15.

To showcase that existing CNN architectures are unable to operate on megapixel images, we also compare our method to traditional CNN models. Typically, the approach for handling high resolution images with deep neural networks is to downsample the input images. Thus; for a fair comparison, we train the CNN baselines using images at various scales. The specifics of each network architecture are described in the corresponding experiment. For more details, we refer the reader to our supplementary material.

Finally, to show that the learned attention distribution is non-trivial, we replace the attention network of our model with a fixed network that predicts the uniform distribution and compare the results. We refer to this baseline as *U-XX* where XX denotes the number of sampled patches.

4.1.2. Metrics

Our proposed model allows us to trade off computation with increased performance. Therefore, besides reporting just the achieved test error, we also measure the computational and memory requirements. To this end, we report the per sample wall-clock time for a forward/backward pass and the peak GPU memory allocated for training with a batch size of 1, as reported by the TensorFlow (Abadi et al., 2016) profiler. Note that for *attention sampling*, extracting a patch, reading it from main memory and moving it to the GPU memory is always included in the reported time. Regarding the memory requirements of our baselines, it is important to mention that the maximum used memory depends on the size of the high resolution image, whereas for *attention sampling* it only depends on the number sampled patches and the patch size. For a fair comparison in terms of both memory and computational requirements, with *Deep MIL*,

we make sure that the patches are extracted from a grid with a stride at least half the size of the patch. Finally, due to lack of space, we provide extensive qualitative results of the learned attention distribution in the supplementary material.

4.2. Megapixel MNIST

We evaluate *attention sampling* on an artificial dataset based on the MNIST digit classification task (LeCun et al., 2010). We generate 6000 empty images of size 1500×1500 and we place patches of random noise at 50 random locations. The size of each patch is equal to an MNIST digit. In addition, we randomly position 5 digits sampled from the MNIST dataset, 3 belonging to the same class and 2 to a random class. The task is to identify the digit with the most occurrences. We use 5000 images for training and 1000 for testing.

For ATS, the attention network is a three layer convolutional network and the feature network is inspired from LeNet-1 (LeCun et al., 1995). To compute the attention, we downsample the image to 180×180 which results in $32,400$ patches to sample from. The sampled patches from the high resolution image have size 50×50 pixels. For the CNN baseline, we train it on the full size images. Regarding uniform sampling, we note that it does not perform better than random guessing, due to the very large sampling space ($32,400$ possible patches); thus we omit it from this experiment. Furthermore, we also omit *Deep MIL* because the required memory for a batch size of 1 exceeds the available GPU memory.

4.2.1. Performance

Initially, we examine the effect of the number of sampled patches on the performance of our method. We sample $\{5, 10, 25, 50, 100\}$ patches for each image which corresponds to 0.01% to 0.3% of the available sampling space. We train our models 5 independent runs for 500 epochs and the averaged results are depicted in figures 2a and 2b. The figures show the training loss and test error, respectively, with respect to wall clock time both for ATS and the CNN baseline. Even though the CNN has comparably increased capacity, we observe that ATS is order of magnitudes faster and performs better.

As expected, we observe that *attention sampling* directly trades performance for speed, namely sampling fewer patches results in both higher training loss and test error. Although the CNN baseline performs better than random guessing, achieving roughly 40% error, it is still more than an order of magnitude higher than ATS.

4.2.2. Evolution of the attention distribution

The quantitative results of the previous section demonstrate that *attention sampling* processes high resolution images both faster and more accurately than the CNN baseline. However, another important benefit of using attention is the increased interpretability of the decisions of the network. This can be noticed from Figure 3, where we visualize the evolution of the attention distribution as the training progresses. In particular, we select a patch from a random image from the dataset that contains 6 distinct items, 3 pieces of noise and 3 digits, and draw the attention distribution for that patch. We observe that the attention distribution starts as uniform. However, during training, we note that the attention network first learns to distinguish empty space from noise and digits and subsequently even noise from digits. This explains why by only sampling 5 patches we achieve approximately 20% error, even though it is the minimum required to be able to confidently classify an image.

4.3. Histopathology images

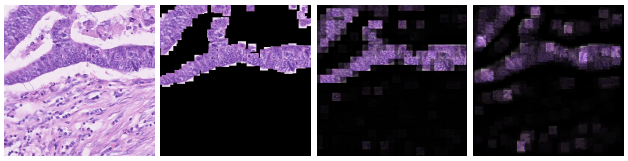
In this experiment, we evaluate *attention sampling* on the *colon cancer* dataset introduced by Sirinukunwattana et al. (2016) to detect whether epithelial cells exist in a hematoxylin and eosin (H&E) stained image.

This dataset contains 100 images of dimensions 500×500. The images originate both from malignant and normal tissue and contain approximately 22,000 annotated cells. Following the experimental setup of Ilse et al. (2018), we treat the problem as binary classification where the positive images are the ones that contain at least one cell belonging in the epithelial class. While the size of the images in this dataset is less than one megapixel, our method can easily scale to datasets with much larger images, as the computational and memory requirements depend only on the size and the number of the patches. However, this does not apply to our baselines, where both the memory and the computational requirements scale linearly with the size of the input image. As a result, this experiment is a best case scenario for our baselines.

For our model, we downsample the images by a factor of 5 and we use the attention network described in § 4.2. The feature network of our model is the same as the one proposed by Ilse et al. (2018) with input patches of size 27×27. For *Deep MIL*, we extract 2,500 patches per image at a regular grid. Regarding the CNN baseline, we use a ResNet (He et al., 2016) architecture. Furthermore, we perform data augmentation by small random adjustments to the brightness and contrast of each image. Following Ilse et al. (2018), we perform 5 independent runs and report the mean and the standard error of the mean.

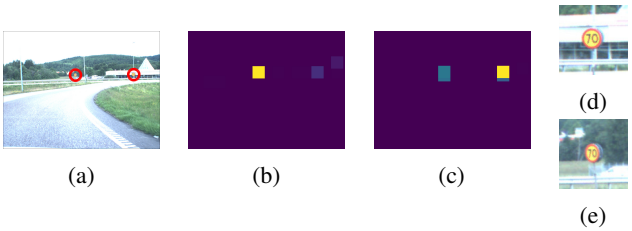
Method	Scale	Train Loss	Test Error	Time/sample	Memory/sample
U-10	0.2/1	0.210 ± 0.031	0.156 ± 0.006	1.8 ms	19 MB
U-50	0.2/1	0.075 ± 0.000	0.124 ± 0.010	4.6 ms	24 MB
CNN	0.5	0.002 ± 0.000	0.104 ± 0.009	4.8 ms	65 MB
CNN	1	0.002 ± 0.000	0.092 ± 0.012	18.7 ms	250 MB
Deep MIL (Ilse et al., 2018)	1	0.007 ± 0.000	0.093 ± 0.004	48.5 ms	644 MB
ATS-10	0.2/1	0.083 ± 0.019	0.093 ± 0.014	1.8 ms	21 MB
ATS-50	0.2/1	0.028 ± 0.002	0.093 ± 0.019	4.5 ms	26 MB

Table 1: Performance comparison of *attention sampling* (ATS) with a CNN and *Deep MIL* on the *colon cancer* dataset comprised of H&E stained images. The experiments were run 5 times and the average ($\pm$ a standard error of the mean) is reported. ATS performs equally well to *Deep MIL* and CNN in terms of test error, while being at least **10x** faster.



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

Figure 4: Visualization of the learned attention distributions for *Deep MIL* (c) and our *attention sampling* (d) on an H&E stained image from the *colon cancer* dataset. (a) depicts the raw image and (b) depicts the cells that belong to the epithelial class. Images (c) and (d) are created by multiplying every patch in (a) by the corresponding normalized attention weight. Both methods localize the attention distribution effectively on the informative parts of the image.



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

Figure 5: Visualization of the learned attention distributions for *Deep MIL* (b) and our *attention sampling* (c) on an image from the speed limits dataset (a). (d) and (e) depict the marked regions from (a) which are also selected by *attention sampling*. We observe that both of them contain speed limit signs unrecognizable in the low resolution image.

4.3.1. Performance

The results of this experiment are summarized in Table 1. We observe that sampling from the uniform distribution 10 and 50 patches is clearly better than random guessing by achieving 15.6% and 12.4% error respectively. This stems from the fact that each positive sample contains hundreds of regions of interest, namely epithelial cells, and we only need one to classify the image. As expected, *attention sampling* learns to focus only on informative parts of the image thus resulting in approximately 35% lower test error and 3 times lower training loss. Furthermore, compared to *Deep MIL* and CNN, ATS-10 performs equally well while being **25x** and **10x** faster respectively. Moreover, the most memory efficient baseline (CNN) needs at least **3x** more memory compared to *attention sampling*, while *Deep MIL* needs **30x** more.

4.3.2. Attention Distribution

To show that our proposed model indeed learns to focus on informative parts of the image, we visualize the learned attention distribution at the end of training. In particular, we select an image from the test set and we compute the attention distribution both for *Deep MIL* and *attention sampling*. Subsequently, we weigh each corresponding patch with a normalized attention value that is computed as $w_i = \frac{a_i - \min(a)}{\max(a) - \min(a)}$. For reference, in Figure 4, apart from the two attention distributions, we also visualize the patches that contain an epithelial cell. Both models identify epithelial cells without having access to per-patch annotations. In order to properly classify an image as positive or not, we just need to find a single patch that contains an epithelial cell. Therefore, despite the fact that the learned attention using *attention sampling* matches less well the distribution of the epithelial cells (Figure 4b), compared to *Deep MIL*, it is not necessarily worse for the classification task that we are interested in. However, it is less helpful for detecting regions of interest. In addition, we also observe that both attentions have significant overlap even on mistakenly selected patches such as the bottom center of the images.

4.4. Speed limit sign detection

In this experiment, we seek to classify images based on whether they contain no speed limit or a limit sign of 50, 70 or 80 kilometers per hour. We use a subset of the Swedish traffic signs dataset (Larsson & Felsberg, 2011), for which we do not use explicit annotations of the signs, just one label for each image. The dataset contains $3,777$ images annotated with 20 different traffic sign classes. Each image is 1.3 megapixels, namely 960×1280 pixels. As some classes contain less than 20 samples, we limit the classification task to the one described above. The resulting dataset consists of 747 training images and 684 test images, distributed approx-

Method	Scale	Train Loss	Test Error	Time/sample	Memory/sample
U-5	0.3/1	1.468 ± 0.317	0.531 ± 0.004	7.8 ms	39 MB
U-10	0.3/1	0.851 ± 0.408	0.472 ± 0.008	10.8 ms	78 MB
CNN	0.3	0.003 ± 0.001	0.311 ± 0.049	6.6 ms	86 MB
CNN	0.5	0.002 ± 0.001	0.295 ± 0.039	15.6 ms	239 MB
CNN	1	0.002 ± 0.000	0.247 ± 0.001	64.2 ms	958 MB
Deep MIL (Ilse et al., 2018)	1	0.077 ± 0.089	0.083 ± 0.006	97.2 ms	1,497 MB
ATS-5	0.3/1	0.162 ± 0.124	0.089 ± 0.002	8.5 ms	86 MB
ATS-10	0.3/1	0.082 ± 0.032	0.095 ± 0.008	10.3 ms	118 MB

Table 2: Performance comparison of *attention sampling* (ATS) with a CNN and *Deep MIL* on the *speed limits* dataset. The experiments were run 3 times and the average ($\pm$ a standard error of the mean) is reported. ATS performs equally well as *Deep MIL* while being at least **10x** faster. Regarding CNN, we note that both *Deep MIL* and *attention sampling* perform significantly better.

imately as 100 images for each speed limit sign and 400 for the background class, namely no limit sign.

An interesting fact about this dataset is that in order to properly classify all images it is mandatory to process them in high resolution. This is illustrated in Figure 1, where from the downsampled image one can deduce the existence of a speed limit sign, without being able to identify the number of kilometers written on it. Objects that are physically far from the moving camera become unrecognizable when downsampling the input image. This property might be critical, for early detection of pedestrians or collision avoidance in a self-driving car scenario.

For *attention sampling*, we downsample the original image by approximately a factor of 3 to 288×384. The attention network is a four layer convolutional network and the feature network of both our model and *Deep MIL* is a simple ResNet. For *Deep MIL*, we extract 192 patches on a grid 12×16 of patch size 100×100. For a fair comparison, we evaluate the CNN baseline using images at various resolutions, namely scales 0.3, 0.5 and 1.0.

Again also for this dataset, we perform data augmentation, namely random translations and contrast brightness adjustments. In addition, due to class imbalance, for all evaluated methods, we use a crossentropy loss weighted with the inverse of the prior of each class. We perform 3 independent runs and report the mean and the standard error of the mean.

4.4.1. Performance

Table 2 compares the proposed model to our baselines on the speed limits dataset. We observe that although the CNN learns the training set perfectly, it fails to generalise. For the downsampled images, this is expected as the limits on the traffic signs are indistinguishable. Similarly, due to the small number of informative patches, uniform sampling fails to correctly classify both the training set and the test set. We observe that *attention sampling* achieves comparable test error to *Deep MIL* by using just 5 patches, instead of 192. This results in significant speedups of more than an order of magnitude. Regarding the required memory, *attention sampling* needs **17x** less memory compared to *Deep MIL*.

4.4.2. Attention Distribution

In this section, we compare qualitatively the learned attention distribution of *Deep MIL* and *attention sampling* on an image from the test set of the speed limits dataset. In Figure 5a, we mark the positions of speed limit signs with red circles and visualize the corresponding patches in figures 5d and 5e. We observe that the attention distribution from our proposed model has high probability for both patches whereas *Deep MIL* locates both but selects only one. Also in this dataset, both models identify regions of interest in the images without being given any explicit per-patch label.

5. Conclusions

We have presented a novel algorithm to efficiently process megapixel images in a single CPU or GPU. Our algorithm only processes fractions of the input image, relying on an attention distribution to discover informative regions of the input. We show that we can derive the gradients through the sampling and train our model end-to-end with SGD. Furthermore, we show that sampling with the attention distribution is the optimal approximation, in terms of variance, of the model that processes the whole image.

Our experiments show that our algorithm effectively identifies the important regions in two real world tasks and an artificial dataset without any patch specific annotation. In addition, our model executes an order of magnitude faster and requires an order of magnitude less memory than state of the art patch based methods and traditional CNNs.

The presented line of research opens several directions for future work. We believe that a nested model of *attention sampling* can be used to efficiently learn to discover informative regions and classify up to gigapixel images using a single GPU. In addition, *attention sampling* can be used in resource constrained scenarios to finely control the trade-off between accuracy and spent computation.

Acknowledgement

This work is supported by the Swiss National Science Foundation under grant number FNS-30209 "ISUL".

References

Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., et al. Tensorflow: a system for large-scale machine learning. In *OSDI*, volume 16, pp. 265–283, 2016.

Ba, J., Mnih, V., and Kavukcuoglu, K. Multiple object recognition with visual attention. *arXiv preprint arXiv:1412.7755*, 2014.

Girshick, R., Donahue, J., Darrell, T., and Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 580–587, 2014.

Golatkar, A., Anand, D., and Sethi, A. Classification of breast cancer histology using deep learning. In *International Conference Image Analysis and Recognition*, pp. 837–844. Springer, 2018.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Hou, L., Samaras, D., Kurc, T. M., Gao, Y., Davis, J. E., and Saltz, J. H. Patch-based convolutional neural network for whole slide tissue image classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2424–2433, 2016.

Ilse, M., Tomczak, J., and Welling, M. Attention-based deep multiple instance learning. In Dy, J. and Krause, A. (eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 2127–2136, Stockholmsmssan, Stockholm Sweden, 10–15 Jul 2018. PMLR. URL http://proceedings.mlr.press/v80/ilse18a.html.

Jaderberg, M., Simonyan, K., Zisserman, A., et al. Spatial transformer networks. In *Advances in neural information processing systems*, pp. 2017–2025, 2015.

Kahn, H. and Harris, T. E. Estimation of particle transmission by random sampling. *National Bureau of Standards applied mathematics series*, 12:27–30, 1951.

Larsson, F. and Felsberg, M. Using fourier descriptors and spatial models for traffic sign recognition. In *Scandinavian Conference on Image Analysis*, pp. 238–249. Springer, 2011.

LeCun, Y., Jackel, L., Bottou, L., Brunot, A., Cortes, C., Denker, J., Drucker, H., Guyon, I., Muller, U., Sackinger, E., et al. Comparison of learning algorithms for handwritten digit recognition. In *International conference on artificial neural networks*, volume 60, pp. 53–60. Perth, Australia, 1995.

LeCun, Y., Cortes, C., and Burges, C. Mnist handwritten digit database. *AT&T Labs [Online]. Available: http://yann. lecun. com/exdb/mnist*, 2, 2010.

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., and Berg, A. C. Ssd: Single shot multibox detector. In *European conference on computer vision*, pp. 21–37. Springer, 2016.

Liu, Y., Gadepalli, K., Norouzi, M., Dahl, G. E., Kohlberger, T., Boyko, A., Venugopalan, S., Timofeev, A., Nelson, P. Q., Corrado, G. S., et al. Detecting cancer metastases on gigapixel pathology images. *arXiv preprint arXiv:1703.02442*, 2017.

Mnih, V., Heess, N., Graves, A., et al. Recurrent models of visual attention. In *Advances in neural information processing systems*, pp. 2204–2212, 2014.

Nazeri, K., Aminpour, A., and Ebrahimi, M. Two-stage convolutional neural network for breast cancer histology image classification. In *International Conference Image Analysis and Recognition*, pp. 717–726. Springer, 2018.

Ramapuram, J., Diephuis, M., Webb, R., and Kalousis, A. Variational saccading: Efficient inference for large resolution images. *arXiv preprint arXiv:1812.03170*, 2018.

Ranzato, M. On learning where to look. *arXiv preprint arXiv:1405.5488*, 2014.

Recasens, A., Kellnhofer, P., Stent, S., Matusik, W., and Torralba, A. Learning to zoom: a saliency-based sampling layer for neural networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 51–66, 2018.

Redmon, J., Divvala, S., Girshick, R., and Farhadi, A. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 779–788, 2016.

Sirinukunwattana, K., Raza, S. E. A., Tsang, Y.-W., Snead, D. R., Cree, I. A., and Rajpoot, N. M. Locality sensitive deep learning for detection and classification of nuclei in routine colon cancer histology images. *IEEE transactions on medical imaging*, 35(5):1196–1206, 2016.

Wu, Z., Shen, C., and Van Den Hengel, A. Wider or deeper: Revisiting the resnet model for visual recognition. *Pattern Recognition*, 2019.

Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A., Salakhudinov, R., Zemel, R., and Bengio, Y. Show, attend and tell: Neural image caption generation with visual attention. In *International conference on machine learning*, pp. 2048–2057, 2015.