Journal of Machine Learning for Biomedical Imaging 2023:004 vol. 2, pp. 96–150 Submitted 07/2022
Published 03/2023

Deep Weakly-Supervised Learning Methods for Classification and Localization in Histology Images: A Survey

Jérôme Rony jerome.rony.1@etsmtl.net
LIVIA, Dept. of Systems Engineering, École de technologie supérieure, Montreal, Canada
Soufiane Belharbi soufiane.belharbi.1@ens.etsmtl.ca
LIVIA, Dept. of Systems Engineering, École de technologie supérieure, Montreal, Canada
Jose Dolz jose.dolz@etsmtl.ca
LIVIA, Dept. of Software and IT Engineering, École de technologie supérieure, Montreal, Canada
Ismail Ben Ayed ismail.benayed@etsmtl.ca
LIVIA, Dept. of Systems Engineering, École de technologie supérieure, Montreal, Canada
Luke McCaffrey luke.mccaffrey@mcgill.ca
Goodman Cancer Research Centre, Dept. of Oncology, McGill University, Montreal, Canada
Eric Granger eric.granger@etsmtl.ca
LIVIA, Dept. of Systems Engineering, École de technologie supérieure, Montreal, Canada

Abstract

Using state-of-the-art deep learning (DL) models to diagnose cancer from histology data presents several challenges related to the nature and availability of labeled histology images, including image size, stain variations, and label ambiguity. In addition, cancer grading and the localization of regions of interest (ROIs) in such images normally rely on both image- and pixel-level labels, with the latter requiring a costly annotation process. Deep weakly-supervised object localization (WSOL) methods provide different strategies for low-cost training of DL models. Given only image-class annotations, these methods can be trained to simultaneously classify an image, and yield class activation maps (CAMs) for ROI localization. This paper provides a review of deep WSOL methods to identify and locate diseases in histology images, without the need for pixel-level annotations. We propose a taxonomy in which these methods are divided into bottom-up and top-down methods according to the information flow in models. Although the latter have seen only limited progress, recent bottom-up methods are currently driving a lot of progress with the use of deep WSOL methods. Early works focused on designing different spatial pooling functions. However, those methods quickly peaked in term of localization accuracy and revealed a major limitation, namely, – the under-activation of CAMs, which leads to high false negative localization. Subsequent works aimed to alleviate this shortcoming and recover the complete object from the background, using different techniques such as perturbation, self-attention, shallow features, pseudo-annotation, and task decoupling. In the present paper, representative deep WSOL methods from our taxonomy are also evaluated and compared in terms of classification and localization accuracy using two challenging public histology datasets – one for colon cancer (`GlaS`), and a second, for breast cancer (`CAMELYON16`). Overall, the results indicate poor localization performance, particularly for generic methods that were initially designed to process natural images. Methods designed to address the challenges posed by histology data often use priors such as ROI size, or additional pixel-wise supervision estimated from a pre-trained classifier, allowing them to achieve better results. However, all the methods suffer from high false positive/negative localization. Classification performance is mainly affected by the model

https://melba-journal.org/2023:004

selection process, which uses either the classification or the localization metric. Finally, four key challenges are identified in the application of deep WSOL methods in histology, namely, – under-/over-activation of CAMs, sensitivity to thresholding, and model selection – and research avenues are provided to mitigate them. Our code is publicly available at `https://github.com/jeromerony/survey_wsl_histology`.

Keywords: Medical/Histology Image Analysis, Computer-Aided Diagnosis, Deep Learning, Weakly Supervised Object Localization, Weakly Supervised Learning, Image Classification.

1. Introduction

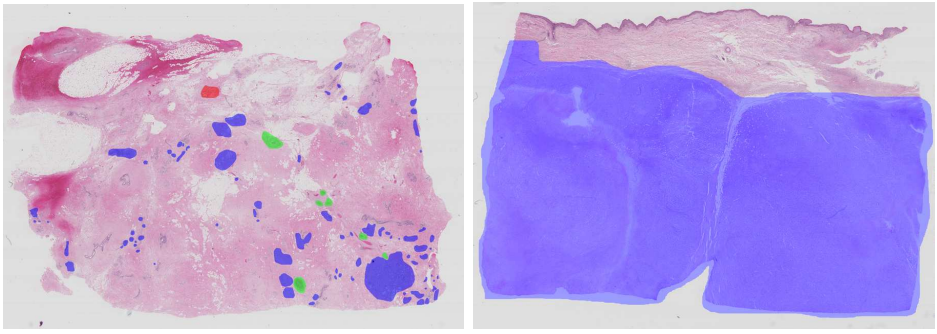
The advent of Whole Slide Imaging (WSI) scanners opened new possibilities in pathology image analysis (He et al., 2012; Madabhushi, 2009). Histology slides provide more comprehensive views of diseases and of their effect on tissue (Hipp et al., 2011), since their preparation preserves the underlying tissue structure (He et al., 2012). For instance, some disease characteristics (e.g., lymphatic infiltration of cancer) may be predicted using only histology images (Gurcan et al., 2009). Histology images analysis remains the gold standard in diagnosing several diseases, including most types of cancer (Gurcan et al., 2009; He et al., 2012; Veta et al., 2014). Breast cancer, which is the most prevalent cancer in women worldwide, relies on medical imaging systems as a primary diagnostic tool for its early detection (Daisuke and Shumpei, 2018; Veta et al., 2014; Xie et al., 2019).

Cancer is mainly diagnosed by pathologists who analyze WSIs to identify and assess epithelial cells organized into ducts, lobules, or malignant clusters, and embedded within a heterogeneous stroma. Manual analysis of histology tissues depends heavily on the expertise and experience of histopathologists. Such manual interpretation is time-consuming and difficult to grade in a reproducible manner. Analyzing WSIs from digitized histology slides enables facilitated, and potentially automated, Computer-Aided Diagnosis in pathology, where the main goal is to confirm the presence or absence of disease and to grade or measure disease progression.

Given the large number of digitized exams in use, automated systems have become a part of the clinical routines for breast cancer detection (Tang et al., 2009). Automated analysis of the spatial structures in histology images can be traced back to early works (Bartels et al., 1992; Hamilton et al., 1994; Weind et al., 1998). Various image processing and machine learning (ML) techniques have been investigated in a bid to identify discriminative structures and classify histology images (He et al., 2012); these include thresholding (Gurcan et al., 2006; Petushi et al., 2006), active contours (Bamford and Lovell, 2001), Bayesian classifiers (Naik et al., 2007), graphs used to model spatial structures (Bilgin et al., 2007; Tabesh et al., 2007), and ensemble methods based on Support Vector Machines and Adaboost (Doyle et al., 2006; Qureshi et al., 2008). An overview of these techniques and their applications is provided in (Gurcan et al., 2009; He et al., 2012; Veta et al., 2014). Recently, deep learning (DL) models have attracted a lot of attention in histology image analysis (Belharbi et al., 2021, 2022a, 2019, 2022b; Courtiol et al., 2018; Dimitriou et al., 2019; Iizuka et al., 2020; Janowczyk and Madabhushi, 2016; Li and Ping, 2018; Srinidhi et al., 2019). In the present paper, we continue in the same vein and focus on the application of DL models in histology image analysis.

DL models (Goodfellow et al., 2016), and convolutional neural networks (CNNs) in particular, provide state-of-the-art performance in many visual recognition applications such as image classification (Krizhevsky et al., 2012), object detection (Redmon et al., 2016), and segmentation (Dolz et al., 2018). These supervised learning architectures are trained end-to-end with large amounts of annotated data. More recently, the potential of DL models has begun to be explored in assisted pathology diagnosis (Daisuke and Shumpei, 2018; Janowczyk and Madabhushi, 2016; Li and Ping, 2018). Given the growing availability of histology slides, DL models have not only been proposed for disease prediction (Hou et al., 2016; Li and Ping, 2018; Sheikhzadeh et al., 2016; Spanhol et al., 2016a; Xu et al., 2016), but also for related tasks such as the detection and segmentation of tumor regions within WSI (Kieffer et al., 2017; Mungle et al., 2017), scoring of immunostaining (Sheikhzadeh et al., 2016; Wang et al., 2015), cancer staging (Shah et al., 2017; Spanhol et al., 2016a), mitosis detection (Chen et al., 2016; Cireşan et al., 2013; Roux et al., 2013), gland segmentation (Caie et al., 2014; Gertych et al., 2015; Sirinukunwattana et al., 2017), and detection and quantification of vascular invasion (Caicedo et al., 2011).

Histology images present additional challenges for ML/DL models because of their (1) high resolution where, for instance, a single core of prostate biopsy tissue digitized at $40\times$ magnification is approximately (15000×15000) elements ($\sim$ 225 million pixels); (2) heterogeneous nature resulting mainly from the variation of the WSI production process; and (3) noisy/ambiguous labels (Daisuke and Shumpei, 2018) caused by the annotation process that is conducted by assigning the worst stage of cancer to the image. Therefore, a WSI that is annotated with a specific grade is also more likely to contain regions with lower grades. This leads to imbalanced datasets having fewer images with high grades. Noisy/ambiguous labels are an issue for models trained through multi-instance learning (Carbonneau et al., 2018; Cheplygina et al., 2019; Wang et al., 2018; Zhou, 2004), where the WSI label is transferred to sampled image patches and can introduce annotation errors. Such label inconsistencies can degrade model performance, and hinder learning (Frenay and Verleysen, 2014; Sukhbaatar et al., 2014; Zhang et al., 2017) (see Figs. 1 and 2).



(a) (b)

Figure 1: Segmentation of two WSIs from the ICIAR 2018 BACH Challenge. Colors represent different types of cancerous regions: red for `Benign`, green for `In Situ Carcinoma` and blue for `Invasive Carcinoma`. These examples highlight the diversity in size and regions (Aresta et al., 2018).

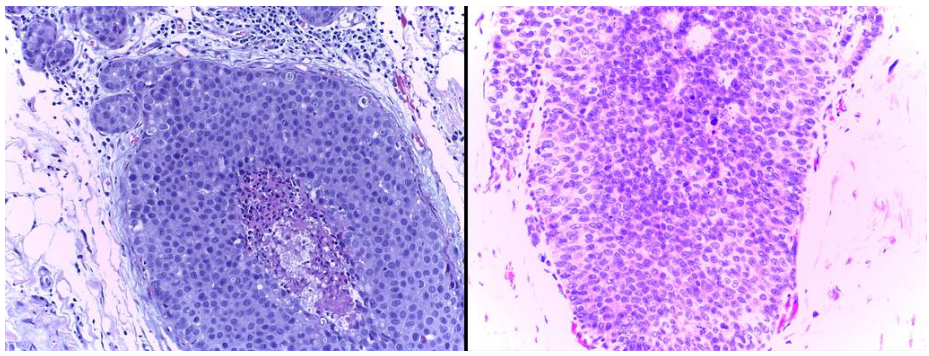


Figure 2: Difference in staining for two images both labeled as *In Situ Carcinoma* extracted from different WSIs (Aresta et al., 2018).

Training accurate DL models to analyze histology images often requires full supervision to address key tasks, such as classification, localization, and segmentation (Daisuke and Shumpei, 2018; Janowczyk and Madabhushi, 2016). Learning to accurately localize cancerous regions typically requires a large number of images with pixel-wise annotations. Considering the size and complexity of such images, dense annotations come at a considerable cost, and require highly trained experts. Outsourcing this task to standard workers such as Mechanical Turk Workers is not an option. As a result, histology datasets are often composed of large images that are coarsely annotated according to the diagnosis. It is therefore clear that training powerful DL models to simultaneously predict the image class and, localize important image regions linked to a prediction *without* dense annotations; is highly desirable for histology image analysis.

Despite their intrinsic challenges (Choe et al., 2020), techniques for weakly-supervised learning (WSL) (Zhou, 2017) have recently emerged to alleviate the need for dense annotation, particularly for computer vision applications. These techniques are adapted to different forms of weak supervision, including image-tags (image-levels label) (Kim et al., 2017; Pathak et al., 2015; Teh et al., 2016; Wei et al., 2017), scribbles (Lin et al., 2016; Tang et al., 2018), points (Bearman et al., 2016), bounding boxes (Dai et al., 2015; Khoreva et al., 2017), global image statistics, such as the target size (Bateson et al., 2019; Jia et al., 2017; Kervadec et al., 2019a,b). The reduced weak supervision requirement provides an appealing learning framework. In this paper, we focus on WSL methods that allow training a DL model using only image-level annotations for the classification of histology images and for the localization of image ROIs linked to class predictions. These methods perform the weakly-supervised object localization (WSOL) task, which can produce localization under the form of activation maps and bounding boxes (Choe et al., 2020).

Interpretability frameworks (Samek et al., 2019; Zhang et al., 2021) have attracted much attention in computer vision (Alber et al., 2019; Bau et al., 2017; Belharbi et al., 2021; Dabkowski and Gal, 2017; Fong et al., 2019; Fong and Vedaldi, 2017; Goh et al., 2020; Murdoch et al., 2019; Petsiuk et al., 2018, 2020; Ribeiro et al., 2016; Samek et al., 2020; Zhang et al., 2020b), and medical image analysis (Cruz-Roa et al., 2013; De La Torre et al., 2020; Fan et al., 2020; Ghosal and Shah, 2020; Hägele et al., 2020; Hao et al., 2019; Korbar et al., 2017; Saleem et al., 2021; Tavolara et al., 2020). They are related to WSOL in the

sense that it also allows providing a spatial map associated with a class prediction decision. However, interpretability methods are often evaluated differently, using for instance the pointing game (Zhang et al., 2018b), which allows localizing an object via a point. Therefore, we limit the focus of this paper to DL models in the literature that were designed and evaluated mainly for the localization task.

Currently, Class Activation Mapping (CAM) methods are practically the only technique for WSOL (Belharbi et al., 2022c). CAMs are built on top of convolution responses over an image, leading to a natural emergence of ROIs. Strong spatial activations in CAMs correspond to discriminative ROIs (Zhou et al., 2016) which allow object localization. Note that localization maps in CAM-based methods are part of the model itself. Such methods have been widely studied in the literature for the weakly-supervised object localization task. In parallel, other methods have emerged for the interpretability, explainability, and visualization of machine learning models (Samek et al., 2019). These methods often provide visualization tools, such as saliency maps to characterize the response of a pre-trained network for the input image. These methods include approaches such as attribution methods (Dabkowski and Gal, 2017; Fong and Vedaldi, 2017; Fong et al., 2019; Petsiuk et al., 2018; Zeiler and Fergus, 2014). Different from the CAM, they produce saliency maps that are external to the network architecture, and that are often estimated by solving an optimization problem. In addition to not being commonly used for object localization, these methods have their own evaluation metrics, such as the pointing game (Zhang et al., 2018b). Apart from CAM methods, (Meethal et al., 2020) presents the only work that aims to directly produce a bounding box, without using any CAMs, in order to localize objects. In (Zhang et al., 2020a), the authors aim to train a regressor to produce bounding boxes, where the target boxes are estimated from CAMs. Our review has shown that there very few works weakly localize objects without using CAMs, because of the difficulty in producing a bounding box using only global labels. CAMs have emerged as a natural response of convolution over visible pixels. However, a bounding box is an abstract and invisible shape, making it difficult to produce without explicit supervision, i.e., bounding box target. This difference between the two approaches explains the current state of the literature on WSOL. Generally, the goal of CAM methods is to build a model that is able to correctly classify an image where only image-class labels are needed. The methods also yield a per-class activation map, i.e., a CAM, under the form of a soft-segmentation map allowing the pixel-wise localization of objects. This map can also be post-processed to estimate a bounding box. Therefore, the scope of this paper is limited to CAM methods.

In this work, we provide a review of state-of-the-art deep WSOL methods proposed from 2013 to early 2022. Most of these reviewed methods have been proposed and evaluated on natural image datasets, with only few having been developed and evaluated with histology images in mind. The performance of representative methods is compared using two public histology datasets for breast and colon cancer, allowing to assess their classification and localization performance. While there have been different reviews of ML/DL models for medical image analysis, and particularly for the analysis of histology WSIs (Daisuke and Shumpei, 2018; Janowczyk and Madabhushi, 2016; Kandemir and Hamprecht, 2015; Litjens et al., 2017; Sudharshan et al., 2019) and medical video analysis (Quellec et al., 2017), these have focused on fully supervised tasks, semi-supervised tasks, or a mixture of different learning settings for classification and segmentation tasks (Litjens et al., 2017; Srinidhi et al.,

2019). To our knowledge, this paper represents the first review focused on deep WSOL models, trained on data with image-class labels for the classification of histology images and localization of ROIs.

Deep WSOL methods in the literature are divided into two main categories, based on the flow of information in models, namely, bottom-up and top-down methods. Our review shows that research in bottom-up methods is more active and dominant than is the case with to top-down methods, making the former state-of-the-art techniques. To address the shortcomings of CAMs, bottom-up methods have progressed from designing simple spatial pooling techniques to performing perturbations and self-attention, to using shallow features, and most recently, to exploiting pseudo-annotation and separating the training of classification from localization tasks. Recent successful WSOL techniques combine the use of shallow features, with pseudo-annotation, while decoupling classification and localization tasks. Top-down techniques for their part have seen less progress. The methods usually rely either on biologically-inspired processes, gradients, or confidence scores to build CAMs. Our comparative results study revealed that while deep WSOL methods proposed for histology data can yield good results, generic methods initially proposed for natural images nevertheless produced poor results. The former methods often rely on priors that aim to reduce false positives/negatives related to the ROI size, for example, or use explicit pixel-wise guidance collected from pre-trained classifiers. Overall, all WSOL methods suffer from high false positive/negative localization. We discuss several issues related to the application of such methods to histology data, including the under-/over-activation of CAMs, sensitivity to thresholding, and model selection. CAM over-activation is a new behavior that may be caused by the visual similarity between the foreground and the background.

In section 2, a taxonomy and a review of state-of-the-art deep WSOL methods are provided, followed by our experimental methodology (section 3) and results (section 4). We conclude this work with a discussion of the main findings, and key challenges facing the application of such WSOL methods in histology, and provide future directions to mitigate these challenges and potentially reduce the gap in performance between WSOL and fully supervised methods. More experimental details are provided in section A and section B. In addition, more visual results of localizations are presented in section C. Our code is publicly available.

2. A taxonomy of weakly-supervised object localization methods

In our taxonomy, we focus on deep WSL methods that allow classifying an image and *pixel-wise* localize its ROIs via a heat map (soft-segmentation map, CAM)[1] During training, only global image-class labels are required for supervision. These methods are referred to as weakly-supervised object localization methods (WSOL) (Choe et al., 2020).

Figure 3 illustrates the overall taxonomy. Among deep WSOL methods, we identify two main categories based on the information flow in the network to yield region localization (see Figure 4): (a) bottom-up methods, which are based on the forward pass information within a network, and (b) top-down methods, which exploit the backward information in addition to a forward pass. Each WSOL aims to stimulate the localization of ROI using a

1. In practice, these methods can also yield a bounding box after performing an image processing procedure over the CAM (Choe et al., 2020).

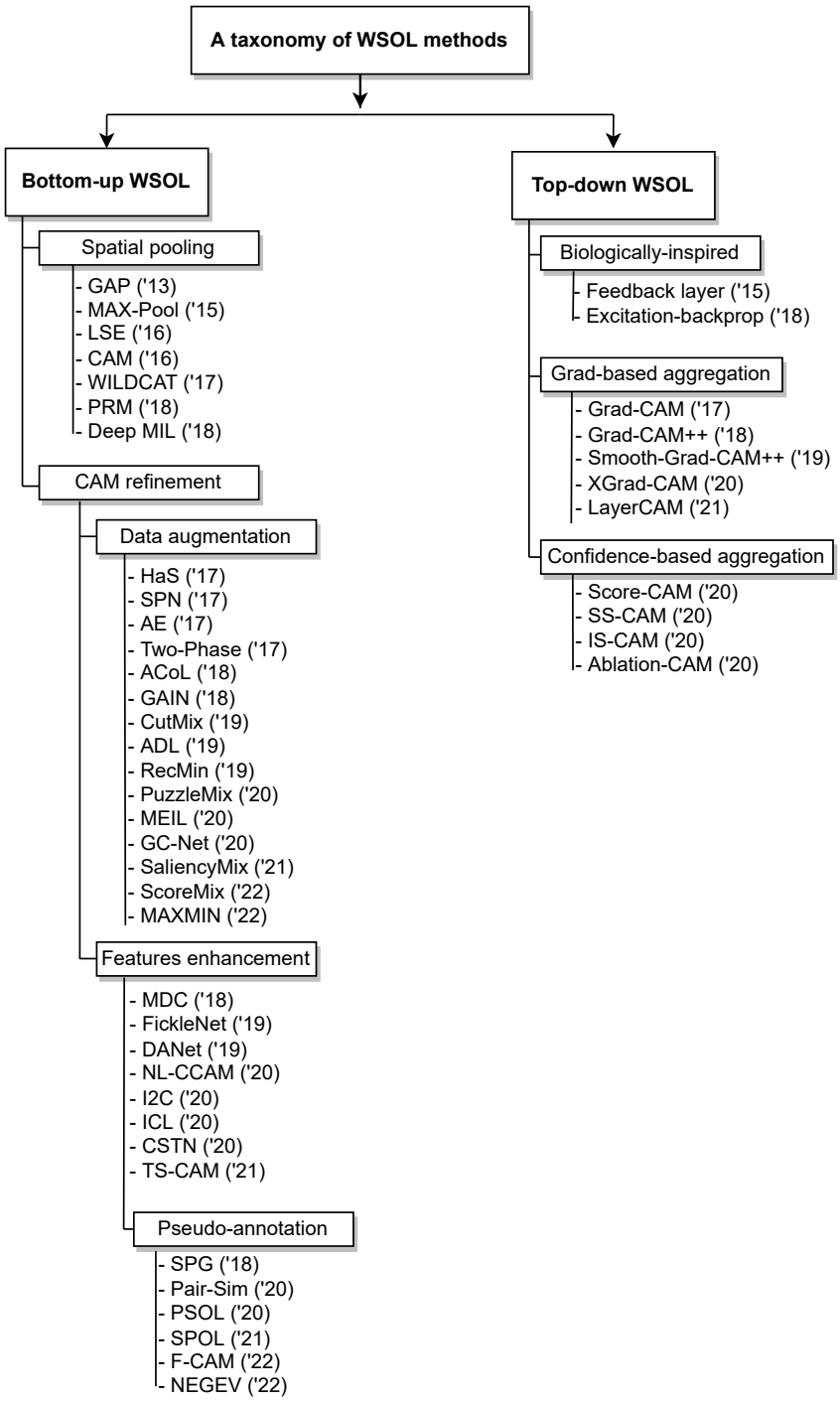


Figure 3: Overall taxonomy of deep WSOL methods for training on data with global image-class annotations, and classification and ROI localization. Methods in each category are ordered chronologically: **1. Bottom-up**: relies on forward pass information. **2. Top-down**: Exploits both forward and backward pass information.

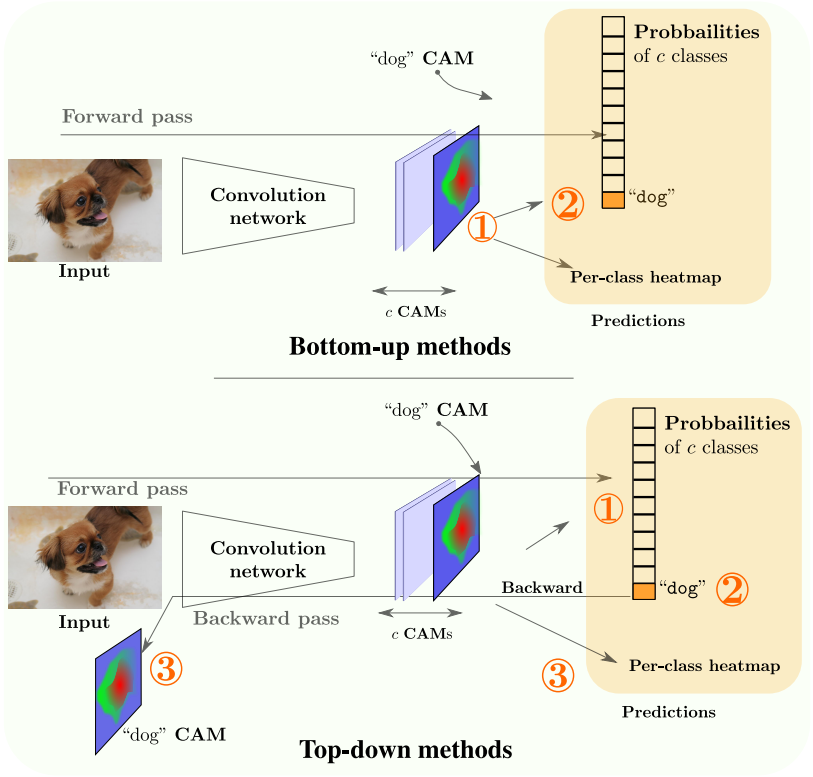


Figure 4: Illustration of the main differences between bottom-up (*top*) and top-down (*bottom*) methods for deep WSOL. Both approaches provide CAMs. However, bottom-up techniques produce them during the forward pass, while top-down techniques require a forward, then a backward pass to obtain them. The numbers 1, 2, and 3 in circles indicate the order of operations. The top-down methods can either produce CAMs at the top before the class pooling or the input of the network.

different mechanism. Both categories rely on building a spatial attention map that has a high magnitude response over ROI and low activations over the background. The rest of this section provides the notations used herein, details on the main categories and sub-categories, and highlights of the main emerging trends that contributed to the progress of the WSOL task.

Notation. To describe the mechanisms behind different methods, we introduce the following notation. Let us consider a set of training samples $\mathbb{D} = \{(\boldsymbol{x}^{(t)}, y^{(t)})\}$ of images $\boldsymbol{x}^{(t)} \in \mathbb{R}^{D \times H^{\text{in}} \times W^{\text{in}}}$ with H^{in}, W^{in}, D being the height, width and depth of the input image, respectively; its image-level label (i.e., class) is $y^{(t)} \in \mathcal{Y}$, with C possible classes. For simplicity, we refer to a training sample (an input and its label) as $(\boldsymbol{x}, y)$.

Let $f_{\boldsymbol{\theta}} : \mathbb{R}^{D \times H^{\text{in}} \times W^{\text{in}}} \rightarrow \mathcal{Y}$ be a function that models a neural network, where the input $\boldsymbol{x}$ has an arbitrary height and width and $\boldsymbol{\theta}$ is the set of model parameters. The training procedure aims to optimize parameters $\boldsymbol{\theta}$ to achieve a specific task. In a multi-class scenario, the network typically outputs a vector of scores $\boldsymbol{s} \in \mathbb{R}^C$ in response to an input image. This

vector is then normalized to obtain a posterior probability using a softmax function,

$$\Pr(y = i|\boldsymbol{x}) = \text{softmax}(\boldsymbol{s})_i = \frac{\exp(\boldsymbol{s}_i)}{\sum_{j=1}^{C} \exp(\boldsymbol{s}_j)} \ . \tag{1}$$

The model predicts the class label corresponding to the maximum probability: $\text{argmax}\{\Pr(y = i|\boldsymbol{x}) : i = 1, 2, ..., C\} = \text{argmax}\{\boldsymbol{s}_i : i = 1, 2, ..., C\}$.

Besides the classification of the input image, we are also interested in the pixel-wise localization of ROIs within the image. Typically, a WSOL method can predict a set of C activation maps of height H and width W to indicate the location of the regions of each class. We note this set as a tensor of shape $\boldsymbol{M} \in \mathbb{R}^{C \times H \times W}$, where $\boldsymbol{M}_c$ indicates the c^{th} map. $\boldsymbol{M}$ is commonly referred to as *Class Activation Maps* (CAMs). Due to convolutional and downsampling operations, typical CAMs have a low resolution as compared to the input image. We note the downscale factor as S, such that $H = H^{\text{in}}/S$ and $W = W^{\text{in}}/S$. Interpolation is often required to yield a CAM of the same size as the image.

2.1 Bottom-up WSOL techniques

In bottom-up methods, the pixel-wise localization is based on the activation of the feature maps resulting from the standard flow of information within a network from the input layer into the output layer (forward pass, Figure 4 (*top*)). Within this category, we identify two different subcategories of techniques to address weakly supervised localization. The first category contains techniques that rely mainly on spatial pooling. Different ways were proposed to pool class scores while simultaneously stimulating a spatial response in CAM to localize ROIs. These methods had limited success. Therefore, another type of method emerged and aimed to refine CAMs directly while using simple spatial pooling techniques. In the next subsections, we present these methods and their variants.

2.2 Spatial pooling methods

This family of techniques aims to design different spatial pooling methods to compute per-class scores, which are then used to train the whole network for classification using standard cross-entropy. In some cases, the pooling is performed to build an image representation, i.e., bag features. Such spatial pooling allows building maps (CAMs) to localize ROIs. Each method promotes the emergence of ROIs localization differently. This strategy undergirds WSOL, and is considered a pioneering mechanism that introduced weakly supervised localization in deep models (Lin et al., 2013). Learning preserving spatial information in CAMs allows ROIs *localization* while requiring only global class annotation. Different methods have been proposed to compute the class scores from spatial maps, whith each pooling strategy having a direct impact on the emerging localization. The challenge is to stimulate the emergence of just ROI in the CAM. All techniques usually start off the same way: a CNN extracts K feature maps $\boldsymbol{F} \in \mathbb{R}^{K \times H \times W}$, where K is the number of feature maps, which is architecture-dependent. The feature maps $\boldsymbol{F}$ are then used to compute a per-class score using a spatial pooling technique.

The first method is Global Average Pooling (GAP) (Lin et al., 2013). It simply averages each feature map in $\boldsymbol{F} \in \mathbb{R}^{K \times H \times W}$ to yield the per-map score in order to build the global

representation $\boldsymbol{f} \in \mathbb{R}^K$ of the input image,

$$\boldsymbol{f}_k = \frac{1}{H\,W} \sum_{i=1,j=1}^{H,W} \boldsymbol{F}_{k,i,j} \ , \tag{2}$$

where $\boldsymbol{f}_k$ is the k^{th} feature of the output. The class-specific activations are then obtained by a linear combination of the features using the weights of the classification layer. Note that in practice, one can directly average CAMs, when available, to yield per-class scores instead of using an intermediate dense layer. In both cases, this pooling strategy ties the per-class score to *all* spatial locations on a map. This means that both ROIs and the background participate in the computation of the per-class score. The CAM literature shows that this pooling strategy can be used to allow a CNN to perform localization using only global labels (Zhou et al., 2016). Typically, in a CNN, the last layer which classifies the representation $\boldsymbol{f}$ is a fully connected layer parameterized by $\boldsymbol{W} \in \mathbb{R}^{C\times K}$ such that $\boldsymbol{s} = \boldsymbol{W}\boldsymbol{f}$ (bias is omitted for simplicity). The CAMs, denoted as $\boldsymbol{M} \in \mathbb{R}^{C\times H\times W}$, are then obtained using a weighted sum of the spatial feature $\boldsymbol{F}$,

$$\boldsymbol{M}_c = \sum_{k=1}^{K} \boldsymbol{W}_{c,k}\,\boldsymbol{F}_k \ . \tag{3}$$

This strategy has been widely used for natural scene images as well as for medical images (Feng et al., 2017; Gondal et al., 2017; Izadyyazdanabadi et al., 2018; Sedai et al., 2018).

An early work on CAM methods (Zhou et al., 2016) revealed a fundamental issue, namely, under-activation. CAMs tend to activate only on small discriminative regions, and therefore, localizing only a small part of the object while missing a large part of it. This leads to high false negatives. Subsequent works in WSOL aimed mainly to tackle this issue by pushing activations in CAM to cover the entire object. This is done either through a different pooling strategy or by explicitly designing a method aiming to recover the full object (subsubsection 2.2.1).

As an alternative to averaging spatial responses, authors in (Oquab et al., 2015) consider using the *maximum* response value on the CAM as a per-class score (MAX-Pool). The method therefore avoids including potential background regions in the class score, thus reducing false positives. However, this pooling technique tends also to focus on small discriminative parts of objects since the per-class score is tied only to one pixel of the response map[2]. To alleviate this problem (Pinheiro and Collobert, 2015; Sun et al., 2016) consider using a smoothed approximation of the maximum function to discover larger parts of objects of interest using the *log-sum-exp* function (LSE),

$$\boldsymbol{s}_c = \frac{1}{q} \log\big[\frac{1}{H\,W} \sum_{i=1,j=1}^{H,W} \exp(q\,\boldsymbol{M}_{c,i,j})\big] \ , \tag{4}$$

where $q \in \mathbb{R}_+^*$ controls the smoothness of the approximation. A small q value makes the approximation closer to the average function, while a large q makes it close to the maximum

2. Note that one pixel in the CAM corresponds to a large surface in the input image depending on the size of the receptive field of the network at the CAM layer.

function. Thus, with small q values, make the network consider large regions, while large values consider only small regions.

Instead of considering the maximum of the map (Oquab et al., 2015), i.e., a single high response point, authors in (Zhou et al., 2018) (PRM) propose to use *local maxima*. This amounts to using local peak responses which are more likely to cover a larger part of the object than occurs when using only the single maximum response,

$$\boldsymbol{s}_c = \boldsymbol{M}^c * G^c = \frac{1}{N^c} \sum_{k=1}^{N^c} \boldsymbol{M}^c_{i_k,j_k} \ , \tag{5}$$

where $G^c \in \mathbb{R}^{H \times W}$ is a sampling kernel, $*$ is the convolution operation, and N^c is the number of local maxima. Depending on the size of the kernel, this pooling allows stimulating different distant locations, which can help recover adjacent regions with an object. Similarly, it is likely that background regions are stimulated. This highlights the challenge faced with transferring global labels into local pixels. Note that such a transfer of supervision is known to be an ill-posed problem in the field of WSOL (Wan et al., 2018).

All the pooling methods discussed thus far rely on high responses to yield per-class scores. The assumption with CAMs is that strong responses indicate potential ROIs, while low responses are more likely to represent backgrounds. This assumption is incorporated in the computation of per-class scores, and therefore, has a direct impact on the localization in CAMs. Authors in (Durand et al., 2017, 2016) (WILDCAT) pursue a different strategy by including low activation, i.e., negative evidence, in the computation of the per-class score. They argue that such pooling plays a regularization role and prevents overfitting, allowing better classification performance. However, it remains unclear how tying negative regions to class scores improves localization, since the aim is to maximize the per-class score of the true label. Nevertheless, the authors provide an architectural change of the pooling layer where several *modality maps* per class are considered. Hence, these modalities allow to capture several parts of the object, leading to better localization. Formally, the pooling is written as,

$$\boldsymbol{s}_c = \frac{Z_c^+}{n^+} + \alpha \frac{Z_c^-}{n^-} \ , \tag{6}$$

where Z_c^+ and Z_c^- correspond to the sum of the n^+ highest and n^- lowest activations of $\boldsymbol{M}_c$ respectively, and α is a hyper-parameter that controls the importance of the minimum scoring regions. Such an operation consists in selecting for each class the n^+ highest activation and the n^- lowest activation within the corresponding map. This method has also been used in the medical field for weakly supervised region localization and image classification in histology images (Belharbi et al., 2019, 2022b). In (Courtiol et al., 2018), instead of operating on pixels, the authors consider adapting (Durand et al., 2017, 2016) for WSIs to operate on instances (tiles).

The aforementioned methods build a bag (image) representation, and then compute CAMs that hold local localization responses, and finally, pull the per-class scores. Authors in (Ilse et al., 2018) (Deep MIL) rely explicitly on a multi-instance learning (MIL) framework (Carbonneau et al., 2018; Cheplygina et al., 2019; Wang et al., 2018; Zhou, 2004). Here, instance representations are firstly built. Then, using the attention mechanism (Bahdanau et al., 2015), a bag representation is computed using a weighted average of the instances

representations. In this case, it is the attention weights that represent the CAM. Strong weights indicate instances with ROIs, while small weights indicate background instances. This method requires changes to standard CNN models. In addition, it is tied to binary classification only. Adjusting to a multi-class context requires further changes to the architecture. Formally, given a set of features $\boldsymbol{F} \in \mathbb{R}^{K \times H \times W}$ extracted for an image, the representation $\boldsymbol{f}$ of the image is computed as,

$$\boldsymbol{f} = \sum_{i=1,j=1}^{H,W} \boldsymbol{A}_{i,j} \boldsymbol{F}_{i,j} \ , \tag{7}$$

$$\text{and} \quad \boldsymbol{A}_{i,j} = \frac{\exp(\psi(\boldsymbol{F}_{i,j}))}{\sum_{i=1,j=1}^{H,W} \exp(\psi(\boldsymbol{F}_{i,j}))} \ ,$$

where $\boldsymbol{F}_{i,j}$ is the feature vector of the location (i.e., instance) indexed by i and j. $\psi : \mathbb{R}^K \to \mathbb{R}$ is a scoring function. The resulting representation $\boldsymbol{f}$ is then classified by a fully connected layer. Two scoring functions are considered (Ilse et al., 2018),

$$\psi_1(\boldsymbol{f}) = \boldsymbol{w} \tanh(\boldsymbol{V}\boldsymbol{f}) \ , \tag{8}$$

$$\psi_2(\boldsymbol{f}) = \boldsymbol{w}\big[\tanh(\boldsymbol{V}\boldsymbol{f}) \odot \sigma(\boldsymbol{U}\boldsymbol{f})\big] \ , \tag{9}$$

where $\boldsymbol{w} \in \mathbb{R}^L$, $(\boldsymbol{V}, \boldsymbol{U}) \in \mathbb{R}^{L \times K}$ are learnable weights, and $\odot$ is an element-wise multiplication. This approach is designed specifically for binary classification and produces a matrix of attention weights $\boldsymbol{A} \in [0,1]^{H \times W}$ with $\sum \boldsymbol{A} = 1$. In the next section, we present a second bottom-up category that aims to refine the CAMs directly.

2.2.1 CAM refinement methods

While spatial pooling methods have helped the emergence of some discriminative regions in CAMs, they have limited success when it comes to covering the full foreground object. Under-activation of CAMs is still a major ongoing issue in the WSOL field, reflecting the difficulty face in to transferring global labels to pixel level. Ever since this became clear, research has shifted from improving the pooling function to explicitly overcoming the under-activation issue and recovering the entire object. Often, this is achieved while using simple pooling functions such as the GAP method (Lin et al., 2013), and to this end, different strategies have been proposed. We divide these into two main categories: methods that use data augmentation to *mine* more discriminative regions, and methods that aim to enhance and learn better internal features of a CNN.

Data augmentation methods. Data augmentation is a strategy often used in machine learning to prevent overfitting and improve performance (Goodfellow et al., 2016). It has been similarly been used in the WSOL field to prevent models from overfitting one single discriminative region, i.e., from under-activating. This is often achieved by pushing the model to seek, i.e., *mine*, other discriminative regions, thereby promoting a large coverage of objects in CAM. Data augmentation most commonly takes the form as information suppression, i.e., *erasing*, where part of an input signal is deleted. This can be performed over input images or intermediate features. Conceptually, this can be seen as a *perturbation* process to stimulate the emergence of more ROIs. For instance, authors in (Singh and Lee, 2017)

propose a 'Hide-And-Seek' (HaS) training strategy, where the input image is divided into multiple patches. During the training phase, only some of these patches are randomly set to be visible while the rest are hidden. Such data augmentation has already been shown to regularize CNN models and improve their classification performance (Devries and Taylor, 2017) (cutout). This is similar to applying a dropout (Srivastava et al., 2014) over the input image, where the target regions consist of a whole patch instead of a single pixel. As a result, the network will not overly rely on the most discriminative patches, and will seek other discriminative regions. While this is an advantage, it can be counter-productive as the network may inadvertently be pushed to consider the background as discriminative, especially for small objects that can be easily deleted.

Other data augmentations have been exploited to improve localization. For instance, the MixUp method (Zhang et al., 2018a) was designed to regularize neural networks by making them less sensitive to adversarial examples and reducing their memorization. This is done by blending two training images to certain degree, in which case the label of the augmented image is assigned by the linear combination of the labels of the two images. Despite the augmented images looking unnatural and locally ambiguous, the method improves classification accuracy. Authors in (Yun et al., 2019) (CutMix) adapt this method to improve localization. Instead of fully blending images, they propose to randomly cut a patch from an image and mix it with a second image. The label is mixed proportionally to the size of the patch. In essence, this is similar to cutout (Devries and Taylor, 2017) and HaS (Singh and Lee, 2017), but instead patches being filled with black or random noise, they are filled with pixels from another image. In practice, this has been shown to improve localization performance. However, due to the randomness in the source patch selection process, this method may select background regions leading to wrong mixed labels, which then leads to the classifier learning unexpected feature representations. Similarly, (Kim et al., 2020) proposed PuzzleMix, which jointly optimizes two objectives: selecting an optimal mask and selecting an optimal mixing plan. Here, the mixing of the input images is no longer random, but uses image saliency, which emerges from image statistics. The mask tries to reveal the most salient data of the two images. Meanwhile, the optimal transport plan aims to maximize the saliency of the revealed portion of the data. In the same vein, SaliencyMix (Uddin et al., 2021) exploits image saliency, but uses a bounding box to capture a region remix instead of a mask. Note that relying on image saliency is a major drawback for less salient images such as those bearing histology data since the foreground and background look similar. Authors in (Stegmüller et al., 2022) (ScoreMix) applied this type of approach to histology data by using proposed regions via attention. Mixing region approach is based on classifier attention instead of image statistics. Discriminative regions from the sources are cut and mixed over non-discriminative regions of the target. Conceptually, this gives a better regional mixing. However, since the learned attention can easily hold false positives/negatives, the mixing can still be vulnerable. In addition, the obtained results seem relatively close to those of the CutMix method (Yun et al., 2019).

In (Wei et al., 2017) (AE), the authors propose an iterative strategy to mine discriminative regions for semantic segmentation. Similarly to the HaS method (Singh and Lee, 2017), they erase regions with the highest response values through learning epochs of a classifier. This allows the emergence of large parts of the model. The emerging segmentation proposals are used to train the model for semantic segmentation. Sequential erasing yields a computationally

expensive process since multiple rounds are required. To improve this, ACoL (Zhang et al., 2018c) designed two branch classifiers to predict the discriminative region and corresponding complementary area simultaneously. The MEIL method (Mai et al., 2020) proceeds in a similar fashion by adding multiple output branches that exploit the erasing process within the learning.

Guided Attention Inference Network (GAIN) (Li et al., 2018) method uses two sequential networks with a shared backbone to mine ROIs. The first network yields an attention map of ROIs, which is used to erase discriminative regions in the image. The erased image is then fed into the next network, where its class response with respect to the target label is used to ensure that no discriminative regions are left in the image after the erasing process. The ROI suppression process is expected to push the first model to seek more discriminative regions, hence large ROIs are covered by the CAM. Similarly, authors in (Kim et al., 2017) (Two-Phase) consider two-phase training of two networks. The first network is trained until convergence. Then, it is used, with frozen weights, in front of a second network to produce a CAM of the target label. The CAM is thresholded to localize the most discriminative regions. Instead of masking the input image as done in the GAIN method (Li et al., 2018), the authors consider masking intermediate feature maps. Once again, results show that this type of information hiding at the feature level allows exploring more ROIs to uncover complete objects.

The GC-Net method (Lu et al., 2020) considers incorporating Geometry Constraints (GC) to train a network to localize objects. Specifically, the authors use 3 models: a detector that yields object localization under the form of a box or an ellipse; a mask generator, which generates a mask based on the generated localization, and a classifier that is evaluated over the ROIs covered by the mask and its complement, i.e., background. The detector is trained to produce small ROIs in which the classifier has a high score while a low score is achieved over the background.

Authors in (Belharbi et al., 2019) (RecMin) consider a recursive mining algorithm integrated directly into back-propagation, allowing to mine ROIs on the fly. All these methods perform mining-erasing of information over the input image. The ADL (Choe and Shim, 2019) method builds a self-attention map per layer to spot potential ROIs. Then, it stochastically erases locations over multiple intermediate feature maps at once during forward propagation through simple element-wise multiplication. The erasing is performed by simple dropout over the attention mask. Such a procedure allows the enhancement of both classification and localization performance. Note that self-attention was already used prior to ADL in (Zhu et al., 2017) (SPN) as a layer to yield proposal regions that are coupled with feature maps allowing only potential ROIs to pass to the next layer, filtering out background/noise. Authors in (Belharbi et al., 2022b) (MAXMIN) use two models: a localizer, followed by a classifier. The localizer aims to build a CAM to localize ROIs at the pixel level. The input image is masked by the produced CAM, and then fed to the classifier. The authors explicitly include the background prior to learning the CAM by constraining it to holding both foreground and background regions. This prevents under-/over-activations, which in turn reduces false positives/negatives. Using entropy, the target classifier scores are constrained to be low over the background and high over the foreground, thus ensuring that no ROI is left in the background. A significant downside of these erasing/mining-based methods is their inherent risk of over-mining since there are no clear criteria to stop mining.

While they are efficient at expanding and gathering object regions, it is very easy to expand to non-discriminative regions, which directly increases false positives.

Features enhancement methods. Other methods aim to improve localization by learning better features. This is often achieved through architectural changes of standard models or by exploiting different levels of features for localization, such as shallow features. Additionally, using pseudo-labels to explicitly guide learning has emerged as an alternative approach for tackling WSOL tasks.

Authors in (Wei et al., 2018) analyze the impact of the object scale on predictions and propose to exploit dilated-convolution (Chen et al., 2018, 2015). They equip a classifier with a varying dilation rate: multi-dilated convolutional (MDC) blocks. This has been shown to effectively enlarge the receptive fields of convolutional kernels, and more importantly, to transfer the surrounding discriminative information to non-discriminative object regions, promoting the emergence of ROI while suppressing the background. Unlike most works that pull CAMs from the top layer (high level), authors in (Yang et al., 2020) (NL-CCAM) consider non-local features by combining low- and high-level features to promote better localization. In addition, rather than using a per-class map as the final CAM, they combine all CAMs using a weighted sum after ordering them using their posterior class probabilities. This allows to gather several parts of the objects and to suppress background regions in the final localization map. The FickleNet method (Lee et al., 2019) randomly selects hidden locations over feature maps. During training, for each random selection, a new CAM is generated. Therefore, for each input image, multiple CAMs can be generated to predict the most discriminative parts. This allows building CAMs that better cover the object. This method is related to ADL (Choe and Shim, 2019), which uses attention, followed by dropout, to mask features. FickleNet does not rely on attention, and simply drops random locations.

DANet (Xue et al., 2019) uses multi-branch outputs at different layers to yield a CAM with different resolutions. This allows to obtain a hierarchical localization. To spread activation over the entire object *without* deteriorating the classification performance, the authors consider a joint optimization of two different terms. A discrepant divergent activation loss constrains CAMs of the same class to cover *different* regions. The authors note that classes with similar visual features are typically suppressed in standard CNNs, since the latter are not discriminative. To recover these regions, they propose a hierarchical divergent activation loss. Meta-classes are created hierarchically to gather previous meta-classes, in which the bottom of the hierarchy contains the original classes. At a specific level, the classifier is trained to assign the same meta-class for all samples assigned to it. This pushes shared similar features to activate within that meta-class, hence recovering similar features in original classes.

In the I^2C method (Zhang et al., 2020c), the authors propose to leverage pixel-wise similarities at the spatial feature level via Inter-Image Communication (I^2C) for better localization. Local and global discriminative features are pushed to be consistent. A local constraint aims to learn the stochastic feature consistency among discriminative pixels, which are randomly sampled from a pair of images within a batch. A global constraint is employed, where a global center feature per-class is maintained and updated in memory after each mini-batch. Average local random features are constrained to be close to the center class features. The ICL method (Ki et al., 2020) aims to deal with over-activation by preventing CAMs from spiking over the background. An attention contrastive loss is proposed. Similar

to ADL (Choe and Shim, 2019), an attention map is estimated from feature maps. Very high and very low activations are used to estimate potential foreground and background regions. The middle activations could be either foreground or background. To expand the activation from foreground into uncertain region, the contrastive loss aims to push activation with *spatial features* similar to foreground features to be foreground while activations with similar spatial features to background features are pushed to be background. This allows a careful expansion of foreground activation toward background regions. In addition, attention at the top layer, which is semantically rich, is used in a self-learning setup to align and guide low layer attention, which is often noisy.

The WSOL task has also benefited from recent advances in architectural design in deep learning. Transformers (Dosovitskiy et al., 2021) in particular have seen their first use in such a task in the TS-CAM (Gao et al., 2021) method. A visual transformer (Dosovitskiy et al., 2021) constructs a sequence of tokens by splitting an input image into patches with positional embedding and applying cascaded transformer blocks to extract a visual representation. Visual transformers can learn complex spatial transforms and reflect long-range semantic correlations adaptively via self-attention mechanism and multilayer perceptrons. This occurs to be crucial for localizing full object. TS-CAM (Gao et al., 2021) improves patch embeddings by exploiting self-attention. In addition, a class-agnostic map is built at each layer. The authors equipped the transformers' output with a CAM module allowing to obtain semantic maps. The Final CAM is an aggregation between the CAM yielded by the CAM module and the average class-agnostic maps across all layers. This shows to help improve localization. In the CSTN method (Meethal et al., 2020), the authors replace standard convolution filters with Spatial Transformer Networks (STNs) (Jaderberg et al., 2015). In addition to using multi-scale localization, this STN model learns affine transformations, which can cover different variations including translation, scale, and rotation, allowing to better attend different object variations.

Recently, a new trend has emerged in WSOL, in which *pseudo-annotations* are exploited. An external model or a WSOL classifier is initially trained using weak labels. Then, it is used to collect pseudo-labels which represent a substitute for the missing full supervision. They are then used to fine-tune a final model. This provides explicit localization guidance for training. However, such methods inherit a major drawback in the form of learning with inaccurate/noisy labels, which must be dealt with. For instance, in (Rahimi et al., 2020) (Pair-Sim), the authors use a fully supervised source dataset to train a proposal generator (Faster-RCNN (Ren et al., 2015)). Then, they apply the generator over a target weakly supervised dataset for the WSOL task to yield proposals for each sample, i.e., bag. The classical MIL framework (Carbonneau et al., 2018; Cheplygina et al., 2019; Wang et al., 2018; Zhou, 2004) is applied by splitting the target dataset into 2 subsets conditioned on the class; one with positive samples with that class label, and the other holding negative samples. The MIL framework is solved such as to yield exactly one proposal per positive sample. A unary score regarding the proposal abjectness that is learned from the source is used, in addition to a pairwise score that measures the compatibility of two proposals conditioned on the bag class.

Authors in (Zhang et al., 2020a) show that localization and classification interfere with each other in WSOL, and that these should be divided into two separate tasks. They first train a classifier, which is used to generate Pseudo Supervised Object Localization (PSOL).

This pseudo-supervision is then used to train a separate class-agnostic localizer. In the same vein, the work in (Wei et al., 2021) demonstrates the benefits of shallow features for localization. The authors exploit low level (Shallow) features to yield Pseudo supervised Object Localization (SPOL), which is used to guide the training of another network. The F-CAM method (Belharbi et al., 2022c) also exploits shallow features by equipping standard classifiers with a segmentation decoder to form a U-Net architecture (Ronneberger et al., 2015). Such a model builds the final CAM through top and low features using skip-connections. The authors show the impact of the CAM size on localization performance, with a lower localization performance seen with a smaller CAM size. CAMs are often interpolated to have the same size as the input image. Since the interpolation algorithm does not take into consideration the image statistics, the authors propose to gradually increase the resolution of the CAMs via a parametric decoder. A low resolution CAM, image statistics and generic size priors are used to train the decoder. The authors propose a *stochastic* sampling of local evidence as opposed to common practice in the literature, where pseudo-labels are selected and fixed before training. The F-CAM method was further adapted for transformer-based methods (Murtaza et al., 2023, 2022) for WASOL in drone-surveillance, and subsequently, for WSOL in videos (Belharbi et al., 2023). Following F-CAM architecture, NEGEV (Belharbi et al., 2022a) was proposed for histology data to improve localization and classifier interpretability. However, the authors focus mainly on using negative evidence collected from a pre-trained classifier, as well as evidence occurring naturally in datasets, i.e., fully negative samples. This allows the method to achieve state-of-the-art performance in localization. Additional experiments also show that the stochastic sampling proposed in (Belharbi et al., 2022c) outperforms the fixed selection of local evidence by a large margin. The Self-Produced Guidance method (SPG) (Zhang et al., 2018d) extracts several attention maps from the top- and low-level layers to benefit from global and detailed localization. Discrete locations of potential ROIs are collected from the maps using thresholding, and are then used to train different layers in a self-supervised way. Note that this approach is related to SPN (Zhu et al., 2017) and ADL (Choe and Shim, 2019), which exploit attention as a self-guidance mechanism to steer the focus toward potential ROIs by masking feature maps. However, the SPG method explicitly learns a segmentation mask using discrete pixel-wise information collected from attention as supervision.

2.3 Top-down WSOL techniques

This second main category is based essentially on the backward pass information within a network to build an attention map that localizes ROIs with respect to a selected target class (backward pass, Figure 4 (*bottom*)). We distinguish three main sub-categories which differ in the way the top signal is back-traced. The first category exploits a secondary conductive feedback network; the second relies on gradient information to aggregate backbone spatial feature maps, while the last exploits posterior class scores for aggregation.

Biologically-inspired methods. These methods are often inspired by cognitive science. For instance, authors in (Cao et al., 2015) argue that visual attention in humans is typically dominated by a target, i.e., a 'goal', in a top-down fashion. Biased competition theory (Beck and Kastner, 2009; Desimone, 1998; Desimone and Duncan, 1995) explains that the human visual cortex is enhanced by top-down stimuli and irrelevant neurons are suppressed in

feedback loops when searching for objects. The work in (Cao et al., 2015) mimics this top-down loop with a *feedback network* that is attached to standard feedforward networks and holds binary variables in addition to ReLU activations (Nair and Hinton, 2010). These binary variables are activated by a top-down message. Given a target class, a standard forward step is performed within the feedforward network to maximize the posterior score of the target. Then, a backward pass is achieved within the feedback network. To promote *selectivity* in the feedback loop (Desimone and Duncan, 1995), the L_1 norm is used as a sparsity term over the binary variables of the feedback network. The forward/backward process is performed several times in order to optimize a loss function composed of the posterior score of a target class, and the sparsity term over the binary variables. For a localization task, the backward loop can reach the network input layer to yield an attention map that indicates ROIs associated with the target class selected at the top of the network. Despite the benefits of this method for CNN visualization and ROI localization, its iterative optimization process to obtain localization makes it less practical for WSOL tasks. The excitation-backprop (Zhang et al., 2018b) method follows a similar top-down scheme. In particular, the authors consider a top-down Winner-Take-ALL (WTA) process (Tsotsos et al., 1995) which selects one winning path. To avoid selecting one deterministic path, which is less representative and leads to binary maps, the authors propose a *probabilistic* WTA downstream process that models all paths. This process integrates both bottom-up and top-down information to compute the winning probability for each neuron. To improve the localization accuracy of the attention map, particularly for images with multi-objects, the authors propose a contrastive top-down attention which captures the differential effect between a pair of top-down attention signals. This allows the attention map to hold activations only for one target class. While both methods (Cao et al., 2015; Zhang et al., 2018b) yield good results, they require substantial changes to standard CNN architectures. In addition, the methods are often used for interpretability and explainability of deep models (Samek et al., 2019).

In the next two categories, we describe how intermediate spatial feature maps are used to pull discriminative maps to localize ROIs associated with a fixed target class. Usually, an aggregation scheme via a weight-per-feature map is performed. The key element in these methods is how the weighting coefficients are estimated. All these weights are back-streamed from the per-class posterior score at the top of the network.

Grad-based aggregation. This family of methods relies on the gradient of posterior class scores of the true label with respect to the feature maps to determine aggregation weights. Such approaches are also used as *visual tools* to explain a network's decision(Samek et al., 2019). In (Selvaraju et al., 2017), the authors propose the Grad-CAM method. In order to compute the CAMs, they propose to *aggregate* spatial feature maps using gradient coefficients. The coefficient of each feature map is computed using the gradient of the score of the selected target class with respect to that map. This gradient indicates how much a pixel location

contributes to the target output. The CAM for the class c is computed as,

$$\boldsymbol{M}_c = \text{ReLU}\left(\sum_{k=1}^{K} \boldsymbol{A}_{c,k}\, \boldsymbol{F}_k\right), \tag{10}$$

$$\text{where} \quad \boldsymbol{A}_{c,k} = \frac{1}{H\,W} \sum_{i=1,j=1}^{H,W} \frac{\partial \boldsymbol{s}_c}{\partial \boldsymbol{F}_{k,i,j}}, \tag{11}$$

where $\boldsymbol{s}_c$ is the score for the class c. This approach is a generalization of the CAM method (Zhou et al., 2016), where the derivative of the score with respect to the feature map is used instead of learned weights. This approach was improved in Grad-CAM++ (Chattopadhyay et al., 2018) and Smooth-Grad-CAM++ (Omeiza et al., 2019) to obtain better localization by covering a complete object, as well as explaining the occurrence of multiple instances in a single image. In (Fu et al., 2020), the authors propose theoretical grounds for CAM-based methods, in particular Grad-CAM, for a more accurate visualization. They include two important axioms: sensitivity and conservation, which determine how to better compute the importance weight of each feature map. Following these axioms, the authors propose a new gradient-based method, XGrad-CAM, which computes the coefficient differently. The coefficients are the solution to an optimization problem composed of the two axioms. To date, these methods have aggregated the feature maps at the last layer. LayerCAM method (Jiang et al., 2021) exploits top- and low-level layers to extract localization. Top layers are commonly known to hold coarse localization, while low-level layers hold detailed but noisy localization. This method extracts CAMs from each layer using a back-propagated gradient signal. Then, the final CAM is computed by fusing all CAMs estimated at each layer. Note that this family of methods is designed to *interrogate* a pre-trained model. While they are model-independent, and allow inspecting the decision of a trained model, the user cannot control their behavior during the training of the model. This ties the localization performance of these methods strongly to the localization information provided by the trained model.

Confidence-based aggregation. Several methods exploit raw scores of the target class, instead of the gradient, in order to *aggregate* spatial features of a backbone. In the Score-CAM method (Wang et al., 2020), each feature map is element-wise multiplied with the input image, and is then used to compute the target-class score. This allows one to obtain a posterior score for each spatial map, which is then compared to the score of the raw image. The difference between both scores yields Channel-wise Increase of Confidence (CIC), in which where high values indicate the presence of a strong ROI in the feature map. The final CAM is a linear weighted sum of all the feature maps, followed by ReLU non-linearity (Nair and Hinton, 2010) where CIC coefficients are used as weights,

$$\boldsymbol{M}^c_{Score-CAM} = ReLU\Big(\sum_k \alpha^c_k \boldsymbol{F}^l_k\Big), \tag{12}$$

where α^c_k is the CIC of class c for feature map k, and l is a layer. Smoothed-Score-CAM (SS-CAM) (Naidu and Michael, 2020) improves the Score-CAM (Wang et al., 2020). Instead of computing the CIC over a single masked image, SS-CAM averages many perturbed masked images. The authors propose either to perturbate the feature map or the input image. This

yields smoothed aggregation weights. The IS-CAM method (Naidu et al., 2020) performs a similar process to smooth weights. While these methods yield good localization, a recent empirical evaluation showed that they are computationally expensive (Belharbi et al., 2022c). For instance, computing a single CAM for an image of size 224×224 over the ResNet50 (He et al., 2016) model takes a few minutes on a decent GPU. This makes training such methods impractical. In the Ablation-CAM method (Desai and Ramaswamy, 2020), the authors consider a gradient-free method to avoid using gradients due to unreliability stemming from gradient saturation (Adebayo et al., 2018; Kindermans et al., 2019). The importance coefficient of a feature map is computed as a slope that measures the difference between the posterior class score and the score obtained when turning off that feature map. That difference is then normalized.

2.4 Critical Analysis

Our review of several works on WSOL carried out in recent years showed the emergence of different strategies for WSOL tasks. We cite two main families. Top-down methods, which aim to back-trace the posterior probability class to find ROIs. These methods rely either on biology processes, classifier confidence, or gradients. Gradient-based methods, which are the more dominant. They are model-independent and can be used for any trained network, in addition to being easy to implement. This family of methods are also used for CNN visualization, and interpretability (Samek et al., 2019).

While top-down methods have been successful, they have experienced slower progress than bottom-up methods, which seem to be the driving core of WSOL. Most successful WSOL methods derive from this family. They are easy to implement and follow the standard flow of information in feed-forward networks. Early works aimed to design different spatial pooling functions to obtain CAMs, but these methods quickly hit a fundamental snag in CAMs, in the form of under-activation. This also suggests that relying only on spatial pooling to transfer global labels to the pixel-level is not enough. Subsequent works have focused on this issue mainly by attempting to recover the localization of a complete object. To that end, several strategies have been proposed:

- **Perturbation** of input images or embeddings, i.e., intermediate features. It is often used to mine discriminative regions. The most common perturbation mechanism is suppression, in which a part of the signal is deleted stochastically either uniformly or via selective attention.
- **Self-attention and self-learning.** Training CNNs to localize objects using only global labels is an ill-posed problem (Wan et al., 2018). However, thanks to the convolutional filter properties, common patterns can emerge within intermediate spatial feature maps. Researchers exploit this property to collect self-attention maps, which often focus on objects. This self-attention has been successfully used to *guide* intermediate convolution layers to further focus on emerging ROI and filter out background and noisy features. In addition, most confident regions in a self-attention map have been used as self-supervisory signals at the pixel level.
- **Shallow features** have long been known to hold useful but noisy localization information in supervised learning, such as in segmentation (Ronneberger et al., 2015). It was not until recently, however, that shallow features began to be exploited as well in WSOL tasks, allowing further boost the localization performance.

- **Pseudo-annotation** provides a substitute for full supervision. Using only global labels for localization been somewhat very successful. They allow the identification of the most discriminative regions but are unable to recover full objects. Partial, noisy, and uncertain pseudo-supervision is currently deemed to be very useful to boost localization performance. It provides low-cost supervision, and yet, should be used with care since it could be very noisy, which could push the model in the wrong direction or trap it in local solutions that predict similar pseudo-annotations.
- **Decoupling classification and localization tasks.** Recent studies in WSOL have shown that these tasks are antagonistic: localization task converges during the very early epochs, and then later degrades, while the classification task converges toward the end of the training. Some works separate them by first training a classifier, then a localizer. The aim is to build a final framework that can yield the best performance over both tasks. Note that recent successful works on WSOL tasks combine shallow features with pseudo-annotations while separating classification and localization tasks.

3. Experimental methodology

In this section, we present the experimental procedure used to evaluate the performance of deep WSOL models. The aim of our experiments was to assess and compare their ability to accurately classify histology images and localize cancerous ROIs.

In order to compare the localization performance of WSOL techniques on histology data, we selected different representative methods from both categories (bottom-up and top-down). From the *bottom-up* category, we consider the following methods: GAP (Lin et al., 2013), MAX-Pool (Oquab et al., 2015), LSE (Sun et al., 2016), CAM (Zhou et al., 2016), HaS (Singh and Lee, 2017), WILDCAT (Durand et al., 2017), ACoL (Zhang et al., 2018c), SPG (Zhang et al., 2018d), Deep MIL (Ilse et al., 2018), PRM (Zhou et al., 2018), ADL (Choe and Shim, 2019), CutMix (Yun et al., 2019), TS-CAM (Gao et al., 2021), MAXMIN (Belharbi et al., 2022b), NEGEV (Belharbi et al., 2022a); while the following methods are considered from the *top-down* category: GradCAM (Selvaraju et al., 2017), GradCAM++ (Chattopadhyay et al., 2018), Smooth-GradCAM++ (Omeiza et al., 2019), XGradCAM (Fu et al., 2020), LayerCAM (Jiang et al., 2021).

Experiments are conducted on two public datasets of histology images, described in subsection 3.3. Most of the public datasets used are collected exclusively for classification or segmentation purposes (Daisuke and Shumpei, 2018), these include the BreaKHis (Spanhol et al., 2016b) and BACH (Aresta et al., 2018) datasets. The only dataset we found that contained both image-level and pixel-level annotations was `GlaS` (subsubsection 3.3.1). Using a single dataset for evaluation could be insufficient to draw meaningful conclusions. Therefore, we created an additional dataset with the required annotations by using a protocol to sample image patches from WSIs of the CAMELYON16 dataset (subsubsection 3.3.2).

3.1 Protocol

In all our experiments, we follow the same experimental protocol as found in (Choe et al., 2020) which defines a clear setup to evaluate ROI localization obtained by a weakly supervised classifier. The protocol includes two main elements, namely, *model selection*, and an *evaluation metric* at the pixel level.

In a weakly supervised setup, model selection is critical. The learning scenario considered in our experiments entails two main tasks: Classification and localization, which are shown to be antagonistic tasks (Belharbi et al., 2022c; Choe et al., 2020). While the localization task converges during the very early training epochs, the classification task converges at late epochs. Therefore, to yield a better localization model, an adequate model selection protocol is required. Following (Choe et al., 2020), and considering a full validation set labeled only at the global level, we randomly select a few samples to be labeled additionally at the pixel level. In particular, we select a few samples per class to yield a balanced set. These samples are used for model selection using localization measures. This selection is referred to as a `B-LOC` selection. Model selection using a full validation set employing only classification measures with global labels is referred to as `B-CL` selection. We provide results on both selection methods to assess their impact on performance. All results are reported using `B-LOC` selection unless specified otherwise. In the next section, we present the evaluation metrics.

3.2 Performance measures

For each task, i.e., classification and localization, we consider their respective metric.

3.2.1 Classification task

We use a standard classification accuracy `CL`,

$$\texttt{CL} = 100 \times \frac{\#\text{correctly classified samples}}{\#\text{samples}} \, (\%) \, , \tag{13}$$

where *#correctly classified samples* is the total number of correctly classified samples, and *#samples* is the total number of samples.

3.2.2 Localization task

The aim of WSOL is to produce a score map that is used to localize an object. In order to measure the quality of localization of ROIs, we consider the same protocol used in (Choe et al., 2020). Using the class activation map $\boldsymbol{S}$ of a target class, a binary map is obtained through thresholding. At pixel location (i, j), this map is compared to the ground true mask $\boldsymbol{T}$. Following (Choe et al., 2020), we threshold the score map at τ to generate the binary mask $\{(i, j) \mid s_{ij} \geq \tau\}$. We consider the following localization metrics:

`PxAP`: We use the `PxAP` metric, presented in (Choe et al., 2020), which measures the pixel-wise precision-recall. At a specific threshold τ, the pixel precision and recall are defined as,

$$\texttt{PxPrec}(\tau) = \frac{|\{s_{ij}^{(n)} \geq \tau\} \cap \{T_{ij}^{(n)} = 1\}|}{|\{s_{ij}^{(n)} \geq \tau\}|} \, , \tag{14}$$

$$\texttt{PxRec}(\tau) = \frac{|\{s_{ij}^{(n)} \geq \tau\} \cap \{T_{ij}^{(n)} = 1\}|}{|\{T_{ij}^{(n)} = 1\}|} \, . \tag{15}$$

The PxAP metric marginalizes the threshold τ over a predefined set of thresholds[3],

$$\texttt{PxAP} := \sum_{l} \texttt{PxPrec}(\tau_l) \times (\texttt{PxRec}(\tau_l) - \texttt{PxRec}(\tau_{l-1})) \,, \tag{16}$$

which is the area under the curve of the pixel precision-recall curve.

Confusion Matrix: Since we are dealing with a medical application, it is important to assess true positives/negatives and false positives/negatives performance at the pixel level in order to have real insights into localization accuracy. Such information is not explicitly provided via the PxAP metric. Therefore, we consider measuring the confusion matrix by marginalizing the threshold τ similarly to what is done in the PxAP metric. First, we compute each normalized component of the confusion matrix with respect to a fixed threshold as follows,

$$\texttt{TP}(\tau) = \frac{|\{s_{ij}^{(n)} \geq \tau\} \cap \{T_{ij}^{(n)} = 1\}|}{|\{T_{ij}^{(n)} = 1\}|} \,, \tag{17}$$

$$\texttt{FN}(\tau) = \frac{|\{s_{ij}^{(n)} < \tau\} \cap \{T_{ij}^{(n)} = 1\}|}{|\{T_{ij}^{(n)} = 1\}|} \,, \tag{18}$$

$$\texttt{FP}(\tau) = \frac{|\{s_{ij}^{(n)} \geq \tau\} \cap \{T_{ij}^{(n)} = 0\}|}{|\{T_{ij}^{(n)} = 0\}|} \,, \tag{19}$$

$$\texttt{TN}(\tau) = \frac{|\{s_{ij}^{(n)} < \tau\} \cap \{T_{ij}^{(n)} = 0\}|}{|\{T_{ij}^{(n)} = 0\}|} \,, \tag{20}$$

where $\texttt{TP}, \texttt{FN}, \texttt{FP}, \texttt{TN}$ are the true positives, false negatives, false positives, and true negatives, respectively. Each component can be represented as a graph with the x-axis as the threshold τ. Similarly to PxAP, we marginalize confusion matrix components over τ by measuring the area under each component, which is also the average since the step between thresholds is fixed. We report the percentage values of each component,

$$\texttt{TP*} := 100 \times \sum_{l} \texttt{TP}(\tau_l) \times (\tau_l - \tau_{l-1}) \,, \tag{21}$$

$$\texttt{FN*} := 100 \times \sum_{l} \texttt{FN}(\tau_l) \times (\tau_l - \tau_{l-1}) \,, \tag{22}$$

$$\texttt{FP*} := 100 \times \sum_{l} \texttt{FP}(\tau_l) \times (\tau_l - \tau_{l-1}) \,, \tag{23}$$

$$\texttt{TN*} := 100 \times \sum_{l} \texttt{TN}(\tau_l) \times (\tau_l - \tau_{l-1}) \,. \tag{24}$$

3.3 Datasets

In this section, we describe the two public datasets of histology images used in our experiments: GlaS for colon cancer, and CAMELYON16 for breast cancer.

3.3.1 GlaS dataset (GlaS)

This is a histology dataset for colon cancer diagnosis (Sirinukunwattana et al., 2017)[4]. It contains 165 images from 16 Hematoxylin and Eosin (H&E) histology sections and their

3. In all the experiments, we used $\tau \in [0, 1]$ with a step of 0.001 as in (Choe et al., 2020).
4. The Gland Segmentation in Colon Histology Contest: https://warwick.ac.uk/fac/sci/dcs/research/tia/glascontest

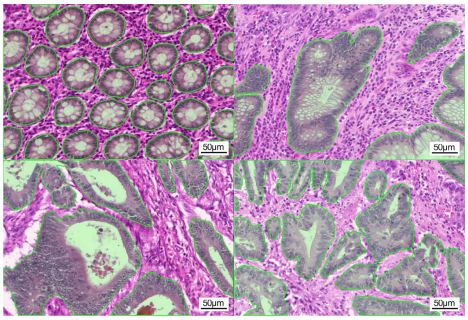


Figure 5: Example of images of different classes with their segmentations from `GlaS` dataset (Credit: (Sirinukunwattana et al., 2017)). *Row 1*: Benign. *Row 2*: Malignant.

corresponding labels. For each image, both pixel-level and image-level annotations for cancer grading (i.e., benign or malign) are provided. The whole dataset is split into training (67 samples), validation (18 samples), and testing (80 samples) subsets. Among the validation set, 3 samples per class are selected to be fully supervised, i.e., 6 samples in total for `B-LOC` selection.

3.3.2 Camelyon16 dataset (`CAMELYON16`)

This dataset[5] is composed of 399 WSI for detection of metastases in H&E stained tissue sections of sentinel auxiliary lymph nodes (SNLs) of women with breast cancer (Ehteshami Bejnordi et al., 2017). The WSIs are annotated globally as normal or metastases. The WSIs with metastases are further annotated at the pixel level to indicate regions of tumors. An example of a WSI is provided in Figure 6. Among the 399 WSIs provided, 270 are used for training, and 129 for testing[6]. The large size of the images makes their use in this survey inconvenient. Therefore, we designed a concise protocol to sample small sub-images for WSL with pixel-wise and image-level annotations. In summary, we sample sub-images of size 512×512 to form train, validation, and test sets, respectively (Fig.7). A detailed sampling protocol is provided in section B. This protocol generates a benchmark containing a total of 48,870 samples: 24,348 samples for training, 8,858 samples for validation, and 15,664 samples for testing. Each sub-set has balanced classes. For `B-LOC`, we randomly select 5 samples per class from the validation set to be fully supervised, i.e., 10 samples in total.

3.3.3 Implementation details

The training of all methods is performed using SGD with 32 batch size (Choe et al., 2020), 1000 epochs for `GlaS`, and 20 epochs for `CAMELYON16`. We use a weight decay of 10^{-4}. Images are resized to 256x256, and patches of size 224x224 are randomly sampled for training. Since almost all methods were evaluated on natural images, we cannot use the reported best hyper-parameters in their original papers. For each method, we perform a search

5. The Cancer Metastases in Lymph Nodes Challenge 2016 (CAMELLYON16): https://camelyon16.grand-challenge.org/Home
6. Sample `test_114` is discarded since the pixel level annotation was not provided. Therefore, the test set is composed of 128 samples with 48 samples with nodal metastases.

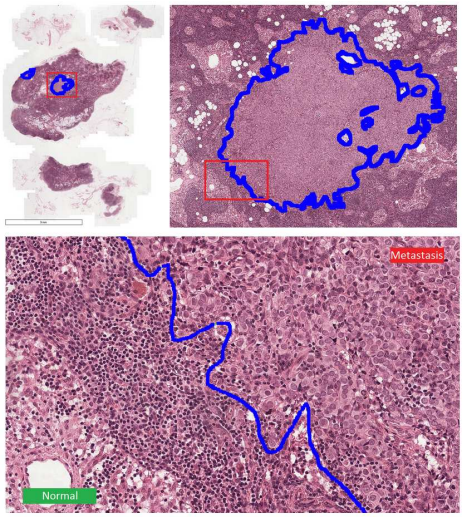


Figure 6: Example of metastatic regions in a WSI from CAMELYON16 dataset (Credit: (Sirinukunwattana et al., 2017)). *Top left*: WSI with tumor. *Top right*: Zoom-in of one of the metastatic regions. *Bottom*: Further zoom-in of the frontier between normal and metastatic regions.

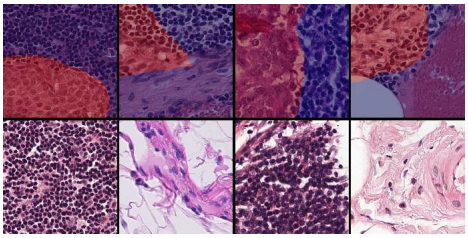


Figure 7: Examples of test images from metastatic (top) and normal (bottom) classes of `CAMELYON16` dataset of size 512×512. Metastatic regions are indicated with a red mask.

for the best hyper-parameter over the validation set, including the learning rate. For the methods (Belharbi et al., 2022a,b), we set part of the hyper-parameters as described in their papers, since they were evaluated on the same histology datasets. For each method, the number of hyper-parameters to tune ranges from one to six. We use three different common backbones (Choe et al., 2020): VGG16 (Simonyan and Zisserman, 2015), InceptionV3 (Szegedy et al., 2016), and ResNet50 (He et al., 2016). For the TS-CAM method (Gao et al., 2021), we use DeiT-based architectures (Touvron et al., 2021): DeiT-Ti (t), DeiT-S (s), DeiT-B (b). We use U-Net (Ronneberger et al., 2015) with full pixel annotation to yield an upper-bound segmentation performance. The weights of all architectures (backbones) are initialized using pre-trained models over Image-Net (Krizhevsky et al., 2012). Then, ann the weights are trained on the histology data. The U-Net decoder is initialized randomly. In section A, we provide full details on the hyper-parameters search.

	GlaS				CAMELYON16			
Methods / Metric	VGG	Inception	ResNet	Mean	VGG	Inception	ResNet	Mean
	PxAP (B-LOC)							
Bottom-up WSOL								
GAP (Lin et al., 2013) *(corr,2013)*	58.5	57.5	56.2	57.4	37.5	24.6	43.7	35.2
MAX-Pool (Oquab et al., 2015) *(cvpr,2015)*	58.5	57.1	46.2	53.9	42.1	40.9	20.2	34.4
LSE (Sun et al., 2016) *(cvpr,2016)*	63.9	62.8	59.1	61.9	63.1	29.0	42.1	44.7
CAM (Zhou et al., 2016) *(cvpr,2016)*	68.5	50.5	64.4	61.1	25.4	48.7	27.5	33.8
HaS (Singh and Lee, 2017) *(iccv,2017)*	65.5	65.4	63.5	64.8	25.4	47.1	29.7	34.0
WILDCAT (Durand et al., 2017) *(cvpr,2017)*	56.1	54.9	60.1	57.0	44.4	31.4	31.0	35.6
ACoL (Zhang et al., 2018c) *(cvpr,2018)*	63.7	58.2	54.2	58.7	31.3	39.3	31.3	33.9
SPG (Zhang et al., 2018d) *(eccv,2018)*	63.6	58.3	51.4	57.7	45.4	24.5	22.6	30.8
Deep MIL (Ilse et al., 2018) *(icml,2018)*	66.6	61.8	64.7	64.3	53.8	51.1	57.9	54.2
PRM (Zhou et al., 2018) *(cvpr,2018)*	59.8	53.1	62.3	58.4	46.0	41.7	23.2	36.9
ADL (Choe and Shim, 2019) *(cvpr,2019)*	65.0	60.6	54.1	59.9	19.0	46.0	46.0	37.0
CutMix (Yun et al., 2019) *(eccv,2019)*	59.9	50.4	56.7	55.6	56.4	44.9	20.7	40.6
TS-CAM (Gao et al., 2021) *(corr,2021)*	t:54.5	b:57.8	s:55.1	52.8	t:46.3	b:21.6	s:42.2	36.7
MAXMIN (Belharbi et al., 2022b) *(tmi,2022)*	75.0	49.1	81.2	68.4	50.4	**80.8**	**77.7**	**69.6**
NEGEV (Belharbi et al., 2022a) *(midl,2022)*	**81.3**	**70.1**	**82.0**	**77.8**	**70.3**	53.8	52.6	58.9
Top-down WSOL								
GradCAM (Selvaraju et al., 2017) *(iccv,2017)*	75.7	56.9	70.0	67.5	40.2	34.4	29.1	34.5
GradCAM++ (Chattopadhyay et al., 2018) *(wacv,2018)*	76.1	65.7	70.7	70.8	41.3	43.9	25.8	37.0
Smooth-GradCAM++ (Omeiza et al., 2019) *(corr,2019)*	71.3	67.6	75.5	71.4	35.1	31.6	25.1	30.6
XGradCAM (Fu et al., 2020) *(bmvc,2020)*	73.7	66.4	62.6	67.5	40.2	33.0	24.4	32.5
LayerCAM (Jiang et al., 2021) *(ieee,2021)*	67.8	66.1	70.9	68.2	34.1	25.0	29.1	29.4
Fully supervised								
U-Net (Ronneberger et al., 2015)*(miccai,2015)*	96.8	95.4	96.4	96.2	83.0	82.2	83.6	82.9

Table 1: PxAP performance over GlaS and CAMELYON16 test sets. Model selection: B-LOC.

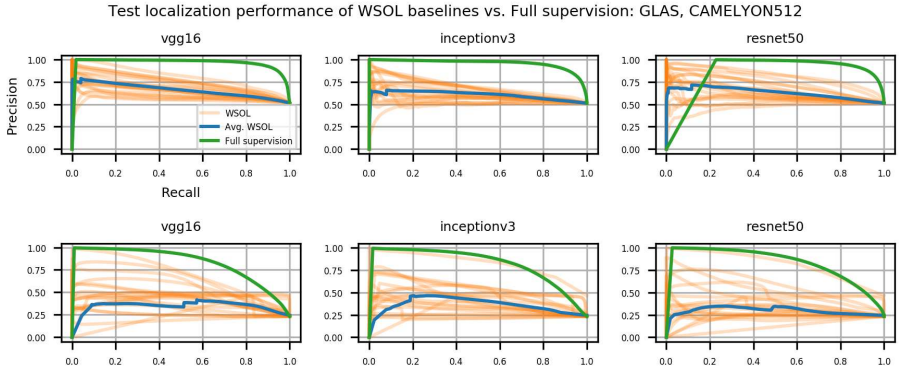


Figure 8: Localization sensitivity to thresholding: WSOL methods (orange), average WSOL methods (blue), fully supervised method (green). Top: GlaS. Bottom: CAMELYON16. Best visualized in color.

	VGG				Inception				ResNet			
Bottom-up WSOL	TP*	FN*	FP*	TN*	TP*	FN*	FP*	TN*	TP*	FN*	FP*	TN*
GAP (Lin et al., 2013) *(corr,2013)*	34.5	65.4	20.5	79.4	50.0	49.9	51.1	48.8	30.6	69.3	21.2	78.7
MAX-Pool (Oquab et al., 2015) *(cvpr,2015)*	38.7	61.2	31.5	68.4	41.2	58.7	36.3	63.6	50.1	49.8	55.7	44.2
LSE (Sun et al., 2016) *(cvpr,2016)*	52.0	47.9	41.4	58.5	**70.9**	**29.0**	63.7	36.2	51.0	48.9	44.9	55.0
CAM (Zhou et al., 2016) *(cvpr,2016)*	34.5	65.4	20.5	79.4	50.0	49.9	51.1	48.8	30.6	69.3	21.2	78.7
HaS (Singh and Lee, 2017) *(iccv,2017)*	42.3	57.6	31.4	68.5	26.4	73.5	16.1	83.8	36.0	63.9	27.5	72.4
WILDCAT (Durand et al., 2017) *(cvpr,2017)*	37.6	62.3	33.2	66.7	42.3	57.6	35.6	64.3	**74.3**	**25.6**	68.8	31.1
ACoL (Zhang et al., 2018c) *(cvpr,2018)*	28.3	71.6	11.1	88.8	6.6	93.3	4.6	95.3	16.1	83.8	6.4	93.5
SPG (Zhang et al., 2018d) *(eccv,2018)*	62.2	37.7	50.5	49.4	58.8	41.1	51.9	48.0	47.1	52.8	47.0	52.9
Deep MIL (Ilse et al., 2018) *(icml,2018)*	14.7	85.2	**7.3**	**92.6**	8.9	91.0	4.3	95.6	10.2	89.7	**4.0**	**95.9**
PRM (Zhou et al., 2018) *(cvpr,2018)*	41.8	58.1	34.7	65.2	61.1	38.8	59.7	40.2	37.3	62.6	29.9	70.0
ADL (Choe and Shim, 2019) *(cvpr,2019)*	41.3	58.6	30.2	69.7	47.3	52.6	38.5	61.4	34.4	65.5	31.4	68.5
CutMix (Yun et al., 2019) *(eccv,2019)*	41.3	58.6	33.6	66.3	38.9	61.0	39.4	60.5	31.3	68.6	28.1	71.8
TS-CAM (Gao et al., 2021) *(corr,2021)*	t:23.1	t:76.8	t:20.4	t:79.5	b:25.2	b:74.7	b:20.4	b:79.5	s:30.3	s:69.6	s:26.5	s:73.4
MAXMIN (Belharbi et al., 2022b) *(tmi,2022)*	**57.4**	**42.5**	41.8	58.1	43.0	56.9	44.3	55.6	56.0	43.9	38.6	61.3
NEGEV (Belharbi et al., 2022a) *(midl,2022)*	52.3	47.6	42.5	57.4	54.7	45.2	48.9	51.0	52.2	47.7	45.6	54.3
Top-down WSOL												
GradCAM (Selvaraju et al., 2017) *(iccv,2017)*	28.3	71.6	11.1	88.8	6.6	93.3	4.6	95.3	16.1	83.8	6.4	93.5
GradCAM++ (Chattopadhyay et al., 2018) *(wacv,2018)*	30.6	69.3	13.4	86.5	12.5	87.4	5.1	94.8	19.0	80.9	8.6	91.3
Smooth-GradCAM++ (Omeiza et al., 2019) *(corr,2019)*	31.3	68.6	17.0	82.9	15.6	84.3	5.7	94.2	24.8	75.1	10.0	89.9
XGradCAM (Fu et al., 2020) *(bmvc,2020)*	31.5	68.4	14.9	85.0	13.0	86.9	4.1	95.8	11.8	88.1	5.7	94.2
LayerCAM (Jiang et al., 2021) *(ieee,2021)*	35.4	64.5	21.6	78.3	11.0	88.9	**3.5**	**96.4**	18.2	81.7	8.1	91.8
Fully supervised												
U-Net (Ronneberger et al., 2015)*(miccai,2015)*	89.8	10.1	11.6	88.3	86.9	13.0	14.6	85.3	87.0	12.9	12.4	87.5

Table 2: Confusion matrix performance over GlaS test set. Model selection: B-LOC.

4. Results and discussion

4.1 Comparison of selected methods

Table 1 shows the localization performance (PxAP) of all methods over both the GlaS and CAMELYON16 datasets. Overall, we observed a discrepancy in performance between different backbones. Across all WSOL methods, we obtained an average PxAP localization performance of 66.86% for VGG, followed by 63.46% for ResNet50, and finally, 59.60% for Inception over the GlaS test set (Table 1). Over CAMELYON16, VGG still ranks first with 42.17%, followed by Inception with 40.61%, and then ResNet50 with 34.72%. This performance difference comes from the basic architectural design difference between these common backbones. In addition, the results of the WSOL methods show that the CAMELYON16 dataset, with an average localization performance of 39.01%, is more challenging than the GlaS dataset, which has an average localization performance of 62.75%. This reflects the inherited difficulty in the CAMELYON16 dataset. While both datasets are challenging, the task in the GlaS dataset boils down to localizing glands which often have a relatively distinct but variable shape or texture (Figure 5). This makes spotting them relatively easy even for a non-expert. However, ROIs in CAMELYON16 have no obvious/common shape or texture (Figure 7). They can seem completely random from a local perspective. Spotting these ROIs can be even extremely challenging for non-experts. This explains the difference in the localization performance of WSOL methods over both datasets. Note that methods that were designed for histology images such as MAXMIN (Belharbi et al., 2022b) and NEGEV (Belharbi et al., 2022a) yield the best localization performance compared to generic methods that were designed and

	VGG				Inception				ResNet			
Bottom-up WSOL	TP*	FN*	FP*	TN*	TP*	FN*	FP*	TN*	TP*	FN*	FP*	TN*
GAP (Lin et al., 2013) *(corr,2013)*	54.0	45.9	52.6	47.3	95.8	4.1	38.0	61.9	63.3	36.6	52.6	47.3
MAX-Pool (Oquab et al., 2015) *(cvpr,2015)*	**94.5**	**5.4**	95.8	4.1	70.8	29.1	56.5	43.4	75.7	24.2	85.0	14.9
LSE (Sun et al., 2016) *(cvpr,2016)*	80.9	19.0	53.5	46.4	52.2	47.7	48.8	51.1	87.4	12.5	76.2	23.7
CAM (Zhou et al., 2016) *(cvpr,2016)*	54.0	45.9	52.6	47.3	**95.8**	**4.1**	38.0	61.9	63.3	36.6	52.6	47.3
HaS (Singh and Lee, 2017) *(iccv,2017)*	54.0	45.9	52.6	47.3	90.5	9.4	36.1	63.8	53.8	46.1	48.6	51.3
WILDCAT (Durand et al., 2017) *(cvpr,2017)*	84.0	15.9	48.5	51.4	40.1	59.8	16.2	83.7	37.4	62.5	21.4	78.5
ACoL (Zhang et al., 2018c) *(cvpr,2018)*	13.4	86.5	4.6	95.3	14.1	85.8	7.2	92.7	7.0	92.9	3.8	96.1
SPG (Zhang et al., 2018d) *(eccv,2018)*	79.6	20.3	55.0	44.9	47.7	52.2	46.9	53.0	47.1	52.8	48.1	51.8
Deep MIL (Ilse et al., 2018) *(icml,2018)*	28.5	71.4	9.7	90.2	54.4	45.5	25.3	74.6	25.2	74.7	7.6	92.3
PRM (Zhou et al., 2018) *(cvpr,2018)*	94.4	5.5	36.5	63.4	63.5	36.4	40.6	59.3	00.0	100.0	00.0	100.0
ADL (Choe and Shim, 2019) *(cvpr,2019)*	51.8	48.1	56.5	43.4	82.5	17.4	41.7	58.2	**94.5**	**5.4**	35.9	64.0
CutMix (Yun et al., 2019) *(eccv,2019)*	79.3	20.6	52.5	47.4	73.7	26.2	49.4	50.5	1.5	98.4	22.0	77.9
TS-CAM (Gao et al., 2021) *(corr,2021)*	t:92.8	t:7.1	t:38.5	t:61.4	b:31.5	b:68.4	b:34.1	b:65.8	s:87.0	s:12.9	s:38.5	s:61.4
MAXMIN (Belharbi et al., 2022b) *(tmi,2022)*	47.1	52.8	47.1	52.8	78.9	21.0	29.6	70.3	62.1	37.8	14.5	85.4
NEGEV (Belharbi et al., 2022a) *(midl,2022)*	21.1	78.8	**4.2**	**95.7**	22.1	77.8	5.9	94.0	9.1	90.8	3.9	96.0
Top-down WSOL												
GradCAM (Selvaraju et al., 2017) *(iccv,2017)*	13.4	86.5	4.6	95.3	14.1	85.8	7.2	92.7	7.0	92.9	3.8	96.1
GradCAM++ (Chattopadhyay et al., 2018) *(wacv,2018)*	70.6	29.3	43.6	56.3	16.1	83.8	**5.3**	**94.6**	4.6	95.3	**2.9**	**97.0**
Smooth-GradCAM++ (Omeiza et al., 2019) *(corr,2019)*	33.7	66.2	22.5	77.4	14.7	85.2	8.6	91.3	34.0	65.9	31.8	68.1
XGradCAM (Fu et al., 2020) *(bmvc,2020)*	13.4	86.5	4.6	95.3	26.4	73.5	16.0	83.9	15.1	84.8	15.4	84.5
LayerCAM (Jiang et al., 2021) *(ieee,2021)*	13.2	86.7	6.2	93.7	25.7	74.2	23.8	76.1	5.9	94.0	3.7	96.2
Fully supervised												
U-Net (Ronneberger et al., 2015)*(miccai,2015)*	58.9	41.0	7.1	92.8	54.6	45.3	5.4	94.5	58.7	41.2	8.1	91.8

Table 3: Confusion matrix performance over CAMELYON16 test set. Model selection: B-LOC.

evaluated on natural images. Top-down methods, such as GradCAM++ (Chattopadhyay et al., 2018) and LayerCAM (Jiang et al., 2021), have been shown to be more efficient on GlaS, with an average of 69.08%, than bottom-up methods, with an average of 60.65%. However, bottom-up methods perform better on CAMELYON16, with an average of 41.09%, as compared to 32.80% for top-down methods. This also can be explained by the aforementioned difference between both datasets. Bottom-up methods rely on convolution-responses that allow spotting common patterns, which can be better detected in GlaS on CAMELYON16. However, top-down methods often rely on gradients that can spot arbitrary shapes giving these methods more advantage over CAMELYON16. The deep MIL method (Ilse et al., 2018) yielded interesting results on both datasets.

Overall, the results also show a large performance gap between learning with weak supervision (global label in this case) and the fully-supervised method. This highlights the difficulty with histology images as compared to natural images.

We also look into the confusion matrix to better assess the pixel-wise predictions on both the GlaS (Table 2) and CAMELYON16 (Table 3) datasets. The first observation is the high *false negative* rate, with a large part of the ROI considered as background. This metric goes up to $\sim 93\%$ over GlaS and 100% over CAMELYON16. This indicates that WSOL methods tend to under-activate by highlighting only a small part of the object and missing the rest. Under-activation is a common behavior in the WSOL method over natural images (Choe et al., 2020), which increases false negatives. We observe a new trend, namely, high *false positive*, which is less common in WSOL (Choe et al., 2020). This is caused by the over-

activation of the entire image, including the ROI and background. The visual similarity between foreground/background regions is the source of this issue, as the model is unable to discriminate between both regions. On average, false positives are much more frequent in `CAMELYON16` than in `GlaS`. However, in both datasets, the false negative rate is much higher than the false positive rate.

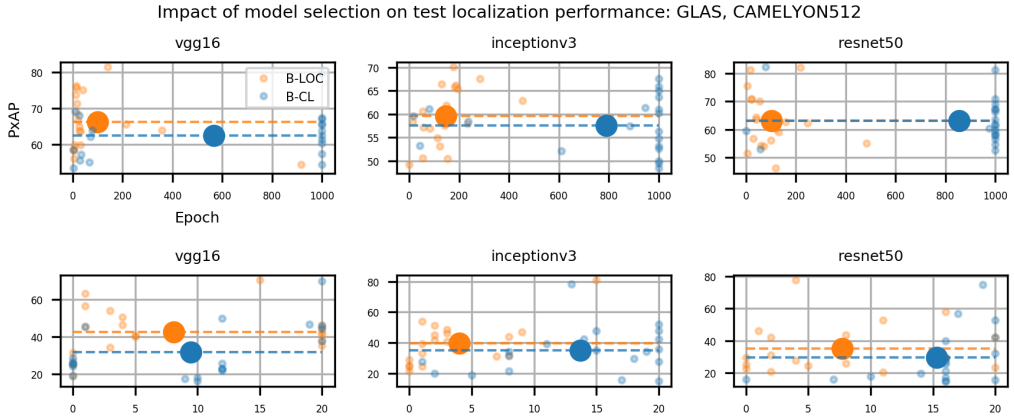
These results suggest that when dealing with histology images, the WSL method can exhibit two behaviors, either under-activate or over-activate, leading to high false negatives or positives. Both drawbacks should be considered when designing WSOL methods for this type of data. We provide visual results of both behaviors in section C. In the histology literature, two different ways are considered to alleviate these issues. In (Belharbi et al., 2022b), the authors consider explicitly adding a background prior to prevent over-activation, while simultaneously preventing under-activation by promoting large sizes for both the background and the foreground. The authors in (Belharbi et al., 2022a) have considered using pixel-level guidance from a pre-trained classifier. Empirically, this allowed consistent patterns to emerge while avoiding under-/over-activation. However, the main drawback of this method is its strong dependence on the quality of pixel-wise evidence collected from the pre-trained classifier.

4.2 Localization sensitivity to thresholds

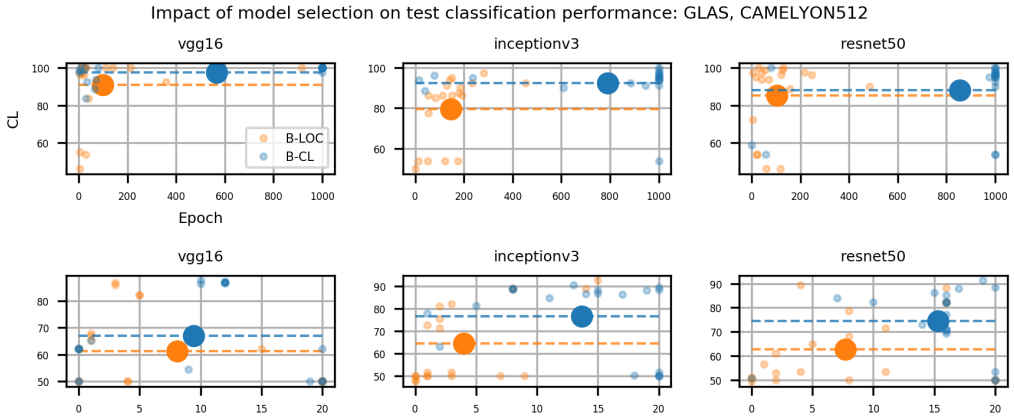
In WSOL, thresholding is required (Choe et al., 2020) to obtain a mask of a specific ROI. All the localization performances reported in this work are marginalized over a set of thresholds to allow a fair comparison (Choe et al., 2020). It has been shown that threshold values are critical for localization performance (Choe et al., 2020). In practice, for a test sample, one needs to threshold the CAM to yield a discrete localization of the ROI[7]. Ideally, the value of the threshold should not have a major impact on the ROI localization. However, that is not the case for WSOL. Evaluations on natural images (Belharbi et al., 2022c; Choe et al., 2020) have shown that localization performance from CAMs obtained in weakly-supervised setups is strongly tied to thresholds. We perform a similar analysis to the variation of localization performance with respect to the threshold in Figure 8. We observe a steep decline in performance when increasing the threshold, similar to the results obtained in (Belharbi et al., 2022c; Choe et al., 2020). This once again highlights the dependency of localization on the threshold, and also indicates that optimal thresholds are concentrated near zero, similarly to the observation of (Belharbi et al., 2022c; Choe et al., 2020). These results also suggest that CAM's activation distribution has a single mode located near zero as demonstrated in (Belharbi et al., 2022c; Choe et al., 2020). This makes finding an optimal threshold difficult, and therefore, makes CAMs sensitive to thresholding, which reflects the uncertainty in CAMs. Ideally, the activation distribution is expected to be bimodal: background mode near zero, and foreground mode near one. Consequently, separating the foreground from the background becomes easy and less sensitive to the threshold, such as in a fully supervised method. The vulnerability of CAMs to thresholding is still an open issue in WSOL (Belharbi et al., 2022c; Choe et al., 2020), and this should be considered in future designs of WSOL methods for general purposes, including histology data applications.

7. Activations of CAMs can also be exploited visually by the user to determine ROI and manually inspect them.

Results in Figure 8 also show that there is still a large performance gap between WSOL and fully supervised methods, with the gap being much larger in the `CAMELYON16` dataset.



(a) **Localization**: Impact of model selection (`B-LOC`: orange. vs. `B-CL`: blue) over test **localization** (`PxAP`) performance. Each point indicates the epoch (x-axis) at which the best model is selected and its corresponding localization performance (y-axis). Large circles indicate the average over all WSOL methods. Top: `GlaS`. Bottom: `CAMELYON16`.



(b) **Classification**: Impact of model selection (`B-LOC`: orange. vs. `B-CL`: blue) over test **classification** (`CL`) performance. Each point indicates the epoch (x-axis) at which the best model is selected and its corresponding classification performance (y-axis). Large circles indicate the average over all WSOL methods. Top: `GlaS`. Bottom: `CAMELYON16`.

Figure 9: Impact of model selection (`B-LOC`: orange. vs. `B-CL`: blue) on test **localization** (`PxAP`) and **classification** (`CL`) performance.

4.3 Importance of model selection

Typically, training a model for localization task with image class as weak supervision is done without having access to any localization information. This means that only classification information can be used to perform model selection using early stopping via a validation set, for instance. However, it has been shown that classification and localization tasks are

antagonistic (Belharbi et al., 2022c; Choe et al., 2020). This implies that model selection via localization (`B-LOC`) leads to good localization, and more likely than not, to poor classification. On the other hand, selection using classification (`B-CL`) yields good classification, and more likely than not, to poor localization, as observed empirically in (Belharbi et al., 2022c; Choe et al., 2020).

The authors in (Choe et al., 2020) suggested a standard protocol for WSOL, in which a few samples in the validation set are fully labeled, i.e., they bear localization annotations. Such subset is used for model selection using localization (`B-LOC`) performance. While unrealistic[8], this protocol allows a fair comparison between methods by removing user bias from the selection.

In 9a, we show the impact of model selection on localization performance. We observe two main trends. First, localization performance converges during the very early epochs, while classification converges toward the end. This indicates that localization performance reaches its peak early on, and then degrades with long training epochs. The opposite can be said about classification. This result is consistent with the findings in (Belharbi et al., 2022c; Choe et al., 2020) over natural images. A second important observation is that, on average, using localization information for model selection yields slightly better localization performance as compared to when using classification measures. Note that the total average gap varies on both datasets with $\sim 2\%$ for `GlaS`, and $\sim 8\%$ for `CAMELYON16`. This suggests that difficult datasets may benefit more from localization selection. Details of the differences in localization performance are reported in Table 6.

In parallel, we inspected the impact of model selection on classification performance 9b. In most cases, we found that selecting a model based on its classification performance over a validation set consistently yields a largely better model in terms of classification. In contrast, model selection using localization performance yields poor classification performance. From 7, 9b, in 70.39% of the cases, the classification selection outperforms the localization selection, with a performance gap of: [min: 0.10%, max: 54.3%, average: 15.08%]. In addition, the reverse is true for 11.18% of the cases with a performance gap of: [min: 0.29%, max: 11.3% avg: 4.45%]. In 18.42% of the cases, where both strategies yield the same classification performance. Therefore, based on these statistics, a better classifier is more likely to be selected using classification accuracy.

All these results mimic what is observed in (Choe et al., 2020), once again confirming that classification and localization under a weakly-supervised setup are more likely to be antagonistic. This is another challenge to consider when designing WSL methods.

The issue of model selection in WSOL was highlighted in (Choe et al., 2020), including its impact on localization and classification performance on natural images. A similar behavior is observed here on the histology dataset, with no clear solution. Most research works on WSOL aim primarily to improve the state-of-the-art localization perfromance. Works in (Belharbi et al., 2022a,c; Zhang et al., 2020a) propose to separate localization from the classification task. First, they train a classifier to get the best classification performance. Then, using information from the classifier, a localizer is trained. This strategy allows to obtain a final framework that yields the best classification and localization performance.

8. Since in a real weakly supervised application, we do not have any localization information

4.4 Computational complexity

Inference time is an important consideration for systems deployed in real-world applications. Table 4 shows that WSOL methods require run times that are suitable for daily clinical routines. However, these statistics show that bottom-up methods are relatively faster than top-down methods. Note that bottom-up methods simply require a forward pass to yield the classification and localization. However, top-down methods require an additional backward pass to perform localization which, adds to the processing time. Among top-down methods, confidence-based aggregation methods have proven to be very slow since they need a series of forward passes to compute multiple scores. In addition, we observe that methods behave differently depending on the backbone architecture, with ResNet50 appearing to yield a slower performance than other backbones. Table 5 presents more details about the memory resources required for backbones, including the number of parameters.

	CNN Backbones		
Methods	**VGG16**	**Inception**	**ResNet50**
CAM (Zhou et al., 2016) *(cvpr,2016)*	0.2ms	0.2ms	0.3ms
ACoL (Zhang et al., 2018c) *(cvpr,2018)*	12.0ms	19.2ms	24.9ms
SPG (Zhang et al., 2018d) *(eccv,2016)*	11.0ms	18.0ms	23.9ms
ADL (Choe and Shim, 2019) *(cvpr,2019)*	6.4ms	16.0ms	14.4ms
NEGEV (Belharbi et al., 2022a) *(midl,2022)*	6.2ms	25.5ms	18.5ms
GradCAM (Selvaraju et al., 2017) *(iccv,2017)*	7.7ms	21.1ms	27.8ms
GradCAM++ (Chattopadhyay et al., 2018) *(wacv,2018)*	23.5ms	23.7ms	28.0ms
Smooth-GradCAM (Omeiza et al., 2019) *(corr,2019)*	62.0ms	150.7ms	136.2ms
XGradCAM (Fu et al., 2020) *(bmvc,2020)*	2.9ms	19.2ms	14.2ms
LayerCAM (Jiang et al., 2021) *(ieee,2021)*	3.2ms	18.2ms	17.9ms
ScoreCAM (Wang et al., 2020) *(cvpr,2020)*	1.9s	3.4s	9.3s
SSCAM (Naidu and Michael, 2020) *(corr,2020)*	105s	136s	349s
IS-CAM (Naidu et al., 2020) *(corr,2020)*	30s	39s	99s

Table 4: Inference time required to produce CAMs using different WSOL methods with standard classifiers = encoder (VGG16, Inception, ResNet50) + global average pooling. The time needed to build a full-size CAM is estimated using an idle Tesla P100 GPU for one random RGB image of size 224×224 with 200 classes. SSCAM (Naidu and Michael, 2020) ($N = 35, \sigma = 2$), IS-CAM (Naidu et al., 2020) ($N = 10$), IS-CAM (Naidu et al., 2020) ($N = 10$) methods are evaluated with a batch size of 32 with their original hyper-parameters (N, and σ).

	CNN Backbones		
Measures	**VGG16**	**Inception**	**ResNet50**
# parameters	≈19.6M	≈25.6M	≈23.9M
# feature maps	1024	1024	2048

Table 5: The number of parameters per backbone, and the number of feature maps at the top layer. The size of feature maps at the top layer is 28×28.

	GlaS				CAMELYON16			
	VGG	Inception	ResNet	Mean	VGG	Inception	ResNet	Mean
Methods / Metric	PxAP: B-LOC/B-CL							
Bottom-up WSOL								
GAP (Lin et al., 2013) *(corr,2013)*	58.5/53.5	57.5/61.0	60.3/56.2	57.4/56.9	37.5/37.5	24.6/15.8	43.7/39.9	35.2/31.0
MAX-Pool (Oquab et al., 2015) *(cvpr,2015)*	58.5/58.5	57.1/56.2	46.2/59.6	53.9/58.1	42.1/28.3	40.9/35.1	20.2/19.4	34.4/27.6
LSE (Sun et al., 2016) *(cvpr,2016)*	63.9/62.2	62.8/61.3	59.1/58.1	61.9/60.5	63.1/25.7	29.0/27.9	42.1/32.0	44.7/28.5
CAM (Zhou et al., 2016) *(cvpr,2016)*	68.5/60.3	50.5/53.3	64.4/63.4	61.1/58.2	25.4/25.4	48.7/39.4	27.5/14.8	33.8/26.5
HaS (Singh and Lee, 2017) *(iccv,2017)*	65.5/61.4	65.4/63.6	63.5/59.9	64.8/61.6	25.4/16.0	47.1/47.8	29.7/17.7	34.0/27.1
WILDCAT (Durand et al., 2017) *(cvpr,2017)*	56.1/69.1	54.9/48.4	60.1/56.5	57.0/58.0	44.4/44.4	31.4/35.7	31.0/16.8	35.6/32.3
ACoL (Zhang et al., 2018c) *(cvpr,2018)*	63.7/57.2	58.2/54.8	54.2/53.0	58.7/55.0	31.3/18.1	39.3/42.2	31.3/32.0	33.9/30.7
SPG (Zhang et al., 2018d) *(eccv,2018)*	63.6/55.7	58.3/59.5	51.4/61.3	57.7/58.8	45.4/45.4	24.5/14.9	22.6/15.5	30.8/25.2
Deep MIL (Ilse et al., 2018) *(icml,2018)*	66.6/63.7	61.8/57.4	64.7/57.9	64.3/59.6	53.8/49.8	51.1/47.9	57.9/56.9	54.2/51.5
PRM (Zhou et al., 2018) *(cvpr,2018)*	59.8/57.4	53.1/52.1	62.3/58.8	58.4/56.1	46.0/46.0	41.7/21.6	23.2/16.0	36.9/27.8
ADL (Choe and Shim, 2019) *(cvpr,2019)*	65.0/62.5	60.6/49.5	54.1/61.8	59.9/57.9	19.0/19.0	46.0/29.8	46.0/16.0	37.0/21.6
CutMix (Yun et al., 2019) *(eccv,2019)*	59.9/55.2	50.4/49.9	56.7/52.5	55.6/52.5	56.4/25.4	44.9/27.6	20.7/14.7	40.6/22.5
TS-CAM (Gao et al., 2021) *(corr,2021)*	t:54.5/54.5	b:57.8/58.4	s:55.1/54.7	52.8/55.8	t:46.3/17.5	b:21.6/34.1	s:42.2/42.2	36.7/31.2
MAXMIN (Belharbi et al., 2022b) *(tmi,2022)*	75.0/63.8	49.1/61.0	81.2/82.3	68.4/69.0	50.4/46.6	80.8/78.1	77.7/74.9	69.6/66.5
Top-down WSOL								
GradCAM (Selvaraju et al., 2017) *(iccv,2017)*	75.7/65.8	56.9/52.6	70.0/67.2	67.5/61.8	40.2/22.7	34.4/34.2	29.1/29.1	34.5/28.6
GradCAM++ (Chattopadhyay et al., 2018) *(wacv,2018)*	76.1/67.3	65.7/53.1	70.7/69.0	70.8/63.1	41.3/26.6	43.9/42.5	25.8/26.8	37.0/31.9
Smooth-GradCAM++ (Omeiza et al., 2019) *(corr,2019)*	71.3/67.9	67.6/67.5	75.5/66.3	71.4/67.2	35.1/24.2	31.6/31.6	25.1/25.3	30.6/27.0
XGradCAM (Fu et al., 2020) *(bmvc,2020)*	73.7/65.3	66.4/60.2	62.6/67.1	67.5/64.2	40.2/22.7	33.0/18.8	24.4/21.1	32.5/20.8
LayerCAM (Jiang et al., 2021) *(ieee,2021)*	67.8/67.0	66.1/65.9	70.9/70.9	68.2/67.9	34.1/21.8	25.0/20.1	29.1/29.1	29.4/23.6
Fully supervised								
U-Net (Ronneberger et al., 2015) *(miccai,2015)*	96.8	95.4	96.4	96.2	83.0	82.2	83.6	82.9

Table 6: Comparison of localization performance (PxAP) with respect to model selection method: B-LOC/B-CL over GlaS and CAMELYON16 test sets. Colors: CL (B-LOC) $<$ CL (B-CL) means that localization performance PxAP obtained using B-LOC is worse than that obtained using B-CL. CL (B-LOC) $>$ CL (B-CL) means that localization performance obtained using B-LOC is better than that obtained using B-CL. Better visualized with color. The NEGEV method (Belharbi et al., 2022a) is not considered in this table because the classifier is pre-trained and frozen. Only B-LOC is used for model selection.

5. Conclusion and future directions

Training deep models for ROI localization in histology images requires costly dense annotation. In addition, such labeling is performed by medical experts. A weakly supervised object localization framework provides different techniques for low-cost training of deep models. Using only image-class annotation, WSOL methods can be trained to classify an image and yield a localization of ROIs via CAMs. Despite its success, the WSOL framework still faces a major challenge, namely, correctly transferring image-class labels to the pixel-level. Moreover, histology images present additional challenges over natural images, including their size, stain variation, and ambiguity of labels. Most importantly, ROIs in histology data are less salient, which makes spotting them much more difficult. This easily opens WSOL models up to false positives/negatives.

In this work, we have presented a review of several deep WSOL methods covering the 2013 to early 2022 period. We divided them into two main categories based on the information flow in the model: bottom-up, and top-down methods. The latter have seen limited progress, while bottom-up methods are the current driving force behind WSOL task. They have

	GlaS				CAMELYON16			
	VGG	Inception	ResNet	Mean	VGG	Inception	ResNet	Mean
Methods / Metric	CL: B-LOC/B-CL							
WSOL								
GAP (Lin et al., 2013) *(corr,2013)*	46.2/98.7	93.7/96.2	87.5/95.0	75.8/96.6	50.0/50.0	50.0/86.3	68.1/69.2	56.0/68.5
MAX-Pool (Oquab et al., 2015) *(cvpr,2015)*	97.5/97.5	86.2/93.7	46.2/58.7	76.6/83.3	50.0/50.0	82.0/86.7	71.4/72.9	67.8/69.8
LSE (Sun et al., 2016) *(cvpr,2016)*	92.5/93.7	92.5/91.2	100/96.2	95.0/93.7	67.8/86.7	50.0/77.8	61.0/86.2	59.6/83.5
CAM (Zhou et al., 2016) *(cvpr,2016)*	100/100	53.7/88.7	97.5/98.7	83.7/95.8	62.2/62.2	51.3/84.7	53.5/71.0	55.6/72.6
HaS (Singh and Lee, 2017) *(iccv,2017)*	100/97.5	86.2/97.5	93.7/100	93.3/98.3	62.2/87.5	50.0/50.0	51.0/82.3	54.4/73.2
WILDCAT (Durand et al., 2017) *(cvpr,2017)*	55.0/100	86.2/95.0	96.2/100	79.1/98.3	50.0/50.0	50.0/50.0	50.0/77.0	50.0/59.0
ACoL (Zhang et al., 2018c) *(cvpr,2018)*	100/92.5	95.0/96.2	46.2/53.7	80.4/80.8	50.0/86.3	50.0/86.4	50.0/50.0	50.0/74.2
SPG (Zhang et al., 2018d) *(eccv,2018)*	53.7/83.7	53.7/93.7	72.5/97.5	59.9/91.6	65.1/65.1	50.0/50.0	49.4/88.5	54.8/67.8
Deep MIL (Ilse et al., 2018) *(icml,2018)*	96.2/100	81.2/92.5	98.7/95.0	92.0/95.8	86.6/87.0	71.3/88.0	88.1/87.8	82.0/87.6
PRM (Zhou et al., 2018) *(cvpr,2018)*	96.2/100	53.7/90.0	96.2/92.5	82.0/94.1	50.0/50.0	75.5/88.6	50.0/50.8	58.5/63.1
ADL (Choe and Shim, 2019) *(cvpr,2019)*	100/100	77.5/93.7	93.7/95.0	90.4/96.2	50.0/50.0	50.0/50.1	56.6/84.1	52.2/61.4
CutMix (Yun et al., 2019) *(eccv,2019)*	100/88.7	86.2/95.0	100/96.2	95.4/93.3	66.8/62.2	80.8/88.2	53.0/70.3	66.8/73.5
TS-CAM (Gao et al., 2021) *(corr,2021)*	t:100/100	b:92.5/95.0	s:90.0/90.0	94.1/95.0	t:50.0/54.4	b:48.3/88.1	s:50.0/50.0	49.4/64.1
MAXMIN (Belharbi et al., 2022b) *(tmi,2022)*	83.7/100	50.0/96.2	95.0/100	76.2/98.7	50.0/50.0	92.4/90.3	89.2/91.3	77.2/77.2
Top-down WSOL								
GradCAM (Selvaraju et al., 2017) *(iccv,2017)*	97.5/100	85.0/93.7	98.7/95.0	93.7/96.2	40.2/86.6	34.4/88.7	29.1/82.3	34.5/85.8
GradCAM++ (Chattopadhyay et al., 2018) *(wacv,2018)*	97.5/100	87.5/100	53.7/53.7	79.5/84.5	50.0/62.2	88.9/89.3	78.6/85.3	72.5/78.9
Smooth-GradCAM++ (Omeiza et al., 2019) *(corr,2019)*	100/100	97.5/98.7	97.5/97.5	98.3/98.7	50.0/62.2	88.5/88.5	51.0/82.3	63.1/77.6
XGradCAM (Fu et al., 2020) *(bmvc,2020)*	100/100	91.2/91.2	88.7/96.2	93.3/95.8	82.1/86.6	88.9/81.1	82.3/71.0	84.4/79.5
LayerCAM (Jiang et al., 2021) *(ieee,2022)*	100/100	90.0/97.5	53.7/53.7	81.2/83.7	85.8/86.6	47.4/62.9	82.1/82.1	71.7/77.2

Table 7: Comparison of classification accuracy (CL) with different model selection method: B-LOC/B-CL on GlaS and CAMELYON16 test sets. Colors: CL (B-LOC) < CL (B-CL) means that the classification accuracy obtained using B-LOC is worse than when using B-CL. CL (B-LOC) > CL (B-CL) means that the classification accuracy obtained using B-LOC is better than when using B-CL. Better visualized with color. The NEGEV method (Belharbi et al., 2022a) is not considered in this table because the classifier (CAM (Zhou et al., 2016)) is pre-trained and frozen. Only B-LOC is used for model selection.

undergone several major changes which have greatly improved the task. Early works focused on designing different spatial pooling functions. However, these methods quickly peaked in term of performance, revealing a major limitation to CAMs, namely, under-activation. Subsequent works aimed to alleviate this issue and recover the complete object using different techniques including: perturbation, self-attention, shallow features, pseudo-annotation, and tasks decoupling. Recent state-of-the-art methods combine several of these techniques.

To assess the localization and classification performance of WSOL techniques over histology data, we selected representative methods in our taxonomy for experimentation. We evaluated them over two public histology datasets: one for colon cancer (GlaS) and a second dataset for breast cancer (CAMELYON16), using the standard protocol for a WSOL task (Choe et al., 2020). Overall, the results indicate poor localization performance, particularly for generic methods that were designed and evaluated over natural images. Methods designed considering histology data challenges yielded good results. In general, all methods suffer from high false positive/negative localization. Furthermore, our analysis showed the following issues:

- **Under-activation,** where CAMs activate only over a small discriminative region, which increases false negatives. This is a documented behavior over natural images.

- Over-activation, where CAMs activate over the entire image, and increases false positives. This is a new behavior of WSOL which is mostly caused by the similarity between foreground and background regions. This makes discrimination between both regions difficult. The following are common strategies to reduce both issues: **1)** Use of priors over the region size (Belharbi et al., 2022b). The size of both the foreground and background in an image is constrained to be as large as possible. However, the analytical solution to their constraint peaks when each region covers half of the image. Results showed that such a trivial solution does not occur in practice due to the existence of other competing constraints. **2)** Use of pseudo-annotations (Belharbi et al., 2022a). The authors used pixel-wise evidence for the foreground and background from a pre-trained classifier. This explicitly provides pixel-wise supervision to the model. Since the pre-trained classifier could easily yield wrong pseudo-labels, this makes the model vulnerable to learning from wrong labels and ties its performance to the performance of the pre-trained classifier. A better solution consists in building more reliable pseudo-labels. Synthesizing better substitution to full supervision via pseudo-annotation is a potential path to explore. Using noisy pseudo-labels has already shown interesting results over medical and natural data (Song et al., 2020).

- Sensitivity to thresholding which is another documented issue for CAMs over natural images, which was observed over histology data as well. A typical solution is to push CAMs' activation to be more confident. However, altering the distribution of CAMs could deteriorate the classification performance. A possible solution is to separate the classification scoring function from CAMs such as in the architecture proposed in (Belharbi et al., 2022a,c, 2023).

- Model selection. Using only classification accuracy allowed to select models having high classification performance, but poor localization. On the other hand, using localization performance for model selection yielded the opposite. This is a common issue in the WSOL task. Recent works have suggested separating both tasks to obtain a framework that yields the best performance for both. The architecture proposed in (Belharbi et al., 2022a) represents a key solution to this issue. First, a classifier is trained, and then it is frozen. Next, the localizer is trained. When trained properly, the final model is expected to yield the best performance in both tasks.

Our final conclusion in this work is that the localization performance obtained with WSOL methods when applied to histology data still lags behind performance with full supervision. The methods are still unable to accurately localize ROI, mainly due to their non-saliency. We have cited several key issues to be considered when designing future WSOL techniques for histology data in order to close the performance gap between weakly and fully supervised methods.

Acknowledgments

This research was supported in part by the Canadian Institutes of Health Research, the Natural Sciences and Engineering Research Council of Canada, and the Digital Research Alliance of Canada (alliancecan.ca).

Ethical Standards

The work follows appropriate ethical standards in conducting research and writing the manuscript, following all applicable laws and regulations regarding treatment of animals or human subjects.

Conflicts of Interest

We declare we don't have any conflicts of interest.

References

J. Adebayo, J. Gilmer, M. Muelly, I. Goodfellow, M. Hardt, and B. Kim. Sanity checks for saliency maps. In *NeurIPS*, 2018.

M. Alber, S. Lapuschkin, P. Seegerer, M. Hägele, K.T. Schütt, G. Montavon, W. Samek, K.-R. Müller, S. Dähne, and P.-J. Kindermans. innvestigate neural networks! *JMLR*, 20: 93:1–93:8, 2019.

G. Aresta, T. Araújo, S. Kwok, et al. Bach: Grand challenge on breast cancer histology images. *CoRR*, abs/1808.04277, 2018.

D. Bahdanau, K. Cho, and Y. Bengio. Neural machine translation by jointly learning to align and translate. In *ICLR*, 2015.

P. Bamford and B. Lovell. Method for accurate unsupervised cell nucleus segmentation. In *Intern. Conf. of the IEEE Engineering in Medicine and Biology Society*, 2001.

P.H. Bartels, D. Thompson, M. Bibbo, et al. Bayesian belief networks in quantitative histopathology. *Analytical and Quantitative Cytology and Histology*, 14(6):459–473, 1992.

Mathilde Bateson, Hoel Kervadec, Jose Dolz, Hervé Lombaert, and Ismail Ben Ayed. Constrained domain adaptation for segmentation. In *MICCAI*, 2019.

D. Bau, B. Zhou, A. Khosla, A. Oliva, and A. Torralba. Network dissection: Quantifying interpretability of deep visual representations. In *CVPR*, 2017.

A. Bearman, O. Russakovsky, V. Ferrari, et al. What's the point: Semantic segmentation with point supervision. In *ECCV*, 2016.

D.M. Beck and S. Kastner. Top-down and bottom-up mechanisms in biasing competition in the human brain. *Vision research*, 49(10):1154–1165, Jun 2009.

S. Belharbi, J. Rony, J. Dolz, I. Ben Ayed, L. McCaffrey, and E. Granger. Min-max entropy for weakly supervised pointwise localization. *CoRR*, abs/1907.12934, 2019.

S. Belharbi, I. Ben Ayed, L. McCaffrey, and E. Granger. Deep active learning for joint classification & segmentation with weak annotator. In *WACV*, 2021.

S. Belharbi, M Pedersoli, I. Ben Ayed, L. McCaffrey, and E. Granger. Negative evidence matters in interpretable histology image classification. In *Medical Imaging with Deep Learning (MIDL)*, 2022a.

S. Belharbi, J. Rony, J. Dolz, I. Ben Ayed, L. McCaffrey, and E. Granger. Deep interpretable classification and weakly-supervised segmentation of histology images via max-min uncertainty. *IEEE Transactions on Medical Imaging*, 41:702–714, 2022b.

S. Belharbi, A. Sarraf, M. Pedersoli, I. Ben Ayed, L. McCaffrey, and E. Granger. F-cam: Full resolution class activation maps via guided parametric upscaling. In *WACV*, 2022c.

S. Belharbi, I. Ben Ayed, L. McCaffrey, and E. Granger. Ftcam: Temporal class activation maps for object localization in weakly-labeled unconstrained videos. In *WACV*, 2023.

C. Bilgin, C. Demir, C. Nagi, et al. Cell-graph mining for breast tissue modeling and classification. In *Intern. Conf. of the IEEE Engineering in Medicine and Biology Society*, 2007.

J. C. Caicedo, F. A. González, and E. Romero. Content-based histopathology image retrieval using a kernel-based semantic annotation framework. *Journal of Biomedical Informatics*, 44(4):519–528, 2011.

P. D. Caie, A. K. Turnbull, S. M. Farrington, et al. Quantification of tumour budding, lymphatic vessel density and invasion through image analysis in colorectal cancer. *Journal of Translational Medicine*, 12(1):156, 2014.

C. Cao, X. Liu, Y. Yang, et al. Look and think twice: Capturing top-down visual attention with feedback convolutional neural networks. In *ICCV*, 2015.

M.-A. Carbonneau, V. Cheplygina, E. Granger, and G. Gagnon. Multiple instance learning: A survey of problem characteristics and applications. *Pattern Recognition*, 77:329–353, 2018.

A. Chattopadhyay, A. Sarkar, P. Howlader, and V. N. Balasubramanian. Grad-cam++: Generalized gradient-based visual explanations for deep convolutional networks. In *WACV*, 2018.

H. Chen, X. Qi, L. Yu, et al. Dcan: deep contour-aware networks for accurate gland segmentation. In *CVPR*, 2016.

L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A.L. Yuille. Semantic image segmentation with deep convolutional nets and fully connected crfs. In *ICLR*, 2015.

L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A.L. Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *PAMI*, 40(4):834–848, 2018.

V. Cheplygina, M. de Bruijne, and J.P.W. Pluim. Not-so-supervised: A survey of semi-supervised, multi-instance, and transfer learning in medical image analysis. *Medical Image Analysis*, 54:280–296, 2019.

J. Choe and H. Shim. Attention-based dropout layer for weakly supervised object localization. In *CVPR*, 2019.

J. Choe, S. J. Oh, S. Lee, S. Chun, Z. Akata, and H. Shim. Evaluating weakly supervised object localization methods right. In *CVPR*, 2020.

D. C. Cireşan, A. Giusti, L. M. Gambardella, et al. Mitosis detection in breast cancer histology images with deep neural networks. In *MICCAI*, 2013.

P. Courtiol, E. W. Tramel, M. Sanselme, et al. Classification and disease localization in histopathology using only global labels: A weakly-supervised approach. *CoRR*, abs/1802.02212, 2018.

A. A. Cruz-Roa, J.E. A. Ovalle, A. Madabhushi, and F.A.G. Osorio. A deep learning architecture for image representation, visual interpretability and automated basal-cell carcinoma cancer detection. In *MICCAI*, 2013.

P. Dabkowski and Y. Gal. Real time image saliency for black box classifiers. In *NeurIPS*, 2017.

J. Dai, K. He, and J. Sun. Boxsup: Exploiting bounding boxes to supervise convolutional networks for semantic segmentation. In *ICCV*, 2015.

K. Daisuke and I. Shumpei. Machine learning methods for histopathological image analysis. *Computational and Structural Biotechnology Journal*, 16, 2018.

J. De La Torre, A. Valls, and D. Puig. A deep learning interpretable classifier for diabetic retinopathy disease grading. *Neurocomputing*, 396:465–476, 2020.

S. Desai and H.G. Ramaswamy. Ablation-cam: Visual explanations for deep convolutional network via gradient-free localization. In *WACV*, 2020.

R. Desimone. Visual attention mediated by biased competition in extrastriate visual cortex. *Philosophical transactions of the Royal Society of London. Series B, Biological sciences*, 353(1373):1245–1255, Aug 1998.

R. Desimone and J. Duncan. Neural mechanisms of selective visual attention. *Annual Review of Neuroscience*, 18(1):193–222, 1995.

T. Devries and G. W. Taylor. Improved regularization of convolutional neural networks with cutout. *CoRR*, abs/1708.04552, 2017.

N. Dimitriou, O. Arandjelović, and P.D. Caie. Deep learning for whole slide image analysis: An overview. *Frontiers in Medicine*, 6:264, 2019.

J. Dolz, C. Desrosiers, and I. Ben Ayed. 3D fully convolutional networks for subcortical segmentation in MRI: A large-scale study. *NeuroImage*, 2018.

A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021.

S. Doyle, C. Rodriguez, A. Madabhushi, et al. Detecting prostatic adenocarcinoma from digitized histology using a multi-scale hierarchical classification approach. In *Intern. Conf. of the IEEE Engineering in Medicine and Biology Society*, 2006.

T. Durand, N. Thome, and M. Cord. Weldon: Weakly supervised learning of deep convolutional neural networks. In *CVPR*, 2016.

T. Durand, T. Mordan, N. Thome, et al. Wildcat: Weakly supervised learning of deep convnets for image classification, pointwise localization and segmentation. In *CVPR*, 2017.

B. Ehteshami Bejnordi, M. Veta, P. Johannes van Diest, et al. Diagnostic Assessment of Deep Learning Algorithms for Detection of Lymph Node Metastases in Women With Breast Cancer. *Journal of the American Medical Association*, 318(22):2199–2210, 2017.

M. Fan, T. Chakraborti, E. I.-C. Chang, Y. Xu, and J. Rittscher. Microscopic fine-grained instance classification through deep attention. In *MICCAI*, Lecture Notes in Computer Science, 2020.

X. Feng, J. Yang, A. F. Laine, et al. Discriminative localization in cnns for weakly-supervised segmentation of pulmonary nodules. In *MICCAI*, 2017.

R. Fong, M. Patrick, and A. Vedaldi. Understanding deep networks via extremal perturbations and smooth masks. In *ICCV*, 2019.

Ruth C. Fong and Andrea Vedaldi. Interpretable explanations of black boxes by meaningful perturbation. In *ICCV*, 2017.

B. Frenay and M. Verleysen. Classification in the presence of label noise: A survey. *TNNLS*, 25, 2014.

R. Fu, Q. Hu, X. Dong, Y. Guo, Y. Gao, and B. Li. Axiom-based grad-cam: Towards accurate visualization and explanation of cnns. In *BMVC*, 2020.

W. Gao, F. Wan, X. Pan, Z. Peng, Q. Tian, Z. Han, B. Zhou, and Q. Ye. TS-CAM: token semantic coupled attention map for weakly supervised object localization. In *ICCV*, 2021.

A. Gertych, N. Ing, Z. Ma, et al. Machine learning approaches to analyze histological images of tissues from radical prostatectomies. *Computerized Medical Imaging and Graphics*, 46, 2015.

S. Ghosal and P. Shah. Interpretable and synergistic deep learning for visual explanation and statistical estimations of segmentation of disease features from medical images. *CoRR*, abs/2011.05791, 2020.

G. SW. Goh, S. Lapuschkin, L. Weber, W. Samek, and A. Binder. Understanding integrated gradients with smoothtaylor for deep neural network attribution. *CoRR*, abs/2004.10484, 2020.

W. M. Gondal, J. M. Köhler, R. Grzeszick, et al. Weakly-supervised localization of diabetic retinopathy lesions in retinal fundus images. In *Int. Conf. on Image Processing*, 2017.

I. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. MIT Press, 2016. `http://www.deeplearningbook.org`.

M. N. Gurcan, T. Pan, H. Shimada, et al. Image analysis for neuroblastoma classification: segmentation of cell nuclei. In *Intern. Conf. of the IEEE Engineering in Medicine and Biology Society*, 2006.

M. N. Gurcan, L. Boucheron, A. Can, et al. Histopathological image analysis: A review. *IEEE reviews in Biomedical Engineering*, 2:147, 2009.

M. Hägele, P. Seegerer, S. Lapuschkin, M. Bockmayr, W. Samek, F. Klauschen, K.-R. Müller, and A. Binder. Resolving challenges in deep learning-based analyses of histopathological images using explanation methods. *Scientific Reports*, 10(1):6423, 2020.

P.W. Hamilton, N. Anderson, P.H. Bartels, et al. Expert system support using bayesian belief networks in the diagnosis of fine needle aspiration biopsy specimens of the breast. *Journal of clinical pathology*, 47, 1994.

J. Hao, S.C. Kosaraju, N.Z. Tsaku, D.H. Song, and M. Kang. Page-net: Interpretable and integrative deep learning for survival analysis using histopathological images and genomic data. In *Pacific Symposium on Biocomputing*, 2019.

K. He, X. Zhang, S.g Ren, and J. Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

L. He, L. R. Long, S. Antani, et al. Histology image analysis for carcinoma detection and grading. *Computer Methods and Programs in Biomedicine*, 107(3):538–556, 2012.

J. D. Hipp, A. Fernandez, C. C. Compton, et al. Why a pathology image should not be considered as a radiology image. *Journal of Pathology Informatics*, 2, 2011.

L. Hou, D. Samaras, T. M. Kurc, et al. Patch-based convolutional neural network for whole slide tissue image classification. In *CVPR*, 2016.

O. Iizuka, F. Kanavati, K. Kato, M. Rambeau, K. Arihiro, and M. Tsuneki. Deep learning models for histopathological classification of gastric and colonic epithelial tumours. *Scientific Reports*, 10(1):1–11, 2020.

M. Ilse, J. M. Tomczak, and M. Welling. Attention-based deep multiple instance learning. In *ICML*, 2018.

M. Izadyyazdanabadi, E. Belykh, C. Cavallo, et al. Weakly-supervised learning-based feature localization in confocal laser endomicroscopy glioma images. *CoRR*, abs/1804.09428, 2018.

M. Jaderberg, K. Simonyan, A. Zisserman, et al. Spatial transformer networks. In *NIPS*, 2015.

A. Janowczyk and A. Madabhushi. Deep learning for digital pathology image analysis: A comprehensive tutorial with selected use cases. *Journal of Pathology Informatics*, 7, 2016.

Z. Jia, X. Huang, I. Eric, C. Chang, and Y. Xu. Constrained deep weak supervision for histopathology image segmentation. *IEEE Transactions on Medical Imaging*, 36(11): 2376–2388, 2017.

P.-T. Jiang, C.-B. Zhang, Q. Hou, M.-M. Cheng, and Y. Wei. Layercam: Exploring hierarchical class activation maps for localization. *IEEE Trans. Image Process.*, 30: 5875–5888, 2021.

M. Kandemir and F.A. Hamprecht. Computer-aided diagnosis from weak supervision: A benchmarking study. *Computerized Medical Imaging and Graphics*, 42:44–50, 2015.

H. Kervadec, J. Dolz, E. Granger, and I. Ben Ayed. Curriculum semi-supervised segmentation. In *MICCAI*, 2019a.

H. Kervadec, J. Dolz, M. Tang, et al. Constrained-cnn losses for weakly supervised segmentation. *Medical Image Analysis*, 54:88–99, 2019b.

A. Khoreva, R. Benenson, J.H. Hosang, M. Hein, and B. Schiele. Simple does it: Weakly supervised instance and semantic segmentation. In *CVPR*, 2017.

M. Ki, Y. Uh, W. Lee, and H. Byun. In-sample contrastive learning and consistent attention for weakly supervised object localization. In *ACCV*, 2020.

B. Kieffer, M. Babaie, S. Kalra, et al. Convolutional neural networks for histopathology image classification: Training vs. using pre-trained networks. In *Int. Conf. on Image Processing Theory, Tools and Applications*, 2017.

D. Kim, D. Cho, D. Yoo, and I. So Kweon. Two-phase learning for weakly supervised object localization. In *ICCV*, 2017.

J.-H. Kim, W. Choo, and H.O. Song. Puzzle mix: Exploiting saliency and local statistics for optimal mixup. In *ICML*, 2020.

P.-J. Kindermans, S. Hooker, J. Adebayo, M. Alber, K. T. Schütt, S. Dähne, D. Erhan, and B. Kim. *The (Un)reliability of Saliency Methods*, pages 267–280. Springer International Publishing, 2019.

B. Korbar, A.M. Olofson, A.P. Miraflor, C.M. Nicka, M.A. Suriawinata, L. Torresani, A.A. Suriawinata, and S. Hassanpour. Looking under the hood: Deep neural network visualization to interpret whole-slide image analysis outcomes for colorectal polyps. In *CVPR workshops*, 2017.

A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *NeurIPS*. 2012.

J. Lee, E. Kim, S. Lee, J. Lee, and S. Yoon. Ficklenet: Weakly and semi-supervised semantic image segmentation using stochastic inference. In *CVPR*, 2019.

K. Li, Z. Wu, K.-C. Peng, et al. Tell me where to look: Guided attention inference network. In *CVPR*, 2018.

Y. Li and W. Ping. Cancer metastasis detection with neural conditional random field. In *Medical Imaging with Deep Learning*, 2018.

D. Lin, J. Dai, J. Jia, et al. Scribblesup: Scribble-supervised convolutional networks for semantic segmentation. In *CVPR*, 2016.

M. Lin, Q. Chen, and S. Yan. Network in network. *coRR*, abs/1312.4400, 2013.

G. Litjens, T. Kooi, B. E. Bejnordi, et al. A survey on deep learning in medical image analysis. *Medical Image Analysis*, 42:60 – 88, 2017.

W. Lu, X. Jia, W. Xie, L. Shen, Y. Zhou, and J. Duan. Geometry constrained weakly supervised object localization. In *ECCV*, 2020.

A. Madabhushi. Digital pathology image analysis: opportunities and challenges. *Imaging in Medicine*, 1(1):7, 2009.

J. Mai, M. Yang, and W. Luo. Erasing integrated learning: A simple yet effective approach for weakly supervised object localization. In *CVPR*, 2020.

A. Meethal, M. Pedersoli, S. Belharbi, and E. Granger. Convolutional stn for weakly supervised object localization and beyond. In *ICPR*, 2020.

T. Mungle, S. Tewary, D.K. Das, et al. Mrf-ann: a machine learning approach for automated er scoring of breast cancer immunohistochemical images. *Journal of Microscopy*, 267(2): 117–129, 2017.

W. J. Murdoch, C. Singh, K. Kumbier, R. Abbasi-Asl, and B. Yu. Definitions, methods, and applications in interpretable machine learning. *Proceedings of the National Academy of Sciences*, 116(44):22071–22080, 2019.

S. Murtaza, S. Belharbi, M. Pedersoli, A. Sarraf, and E. Granger. Constrained sampling for class-agnostic weakly supervised object localization. In *Montreal AI symposium*, 2022.

S. Murtaza, S. Belharbi, M. Pedersoli, A. Sarraf, and E. Granger. Discriminative sampling of proposals in self-supervised transformers for weakly supervised object localization. In *WACV workshop*, 2023.

R. Naidu and J. Michael. SS-CAM: smoothed score-cam for sharper visual feature localization. *CoRR*, abs/2006.14255, 2020.

R. Naidu, A. Ghosh, Y. Maurya, S. R. Nayak K, and S. S. Kundu. IS-CAM: integrated score-cam for axiomatic-based explanations. *CoRR*, abs/2010.03023, 2020.

S. Naik, S. Doyle, A. Madabhushi, et al. Automated gland segmentation and gleason grading of prostate histology by integrating low-, high-level and domain specific information. In *Workshop on Microscopic Image Analysis with Applications in Biology*, 2007.

V. Nair and G.E. Hinton. Rectified linear units improve restricted boltzmann machines. In *ICML*, 2010.

D. Omeiza, S. Speakman, C. Cintas, and K. Weldemariam. Smooth grad-cam++: An enhanced inference level visualization technique for deep convolutional neural network models. *CoRR*, abs/1908.01224, 2019.

M. Oquab, L. Bottou, I. Laptev, et al. Is object localization for free?-weakly-supervised learning with convolutional neural networks. In *CVPR*, 2015.

N. Otsu. A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man, and Cybernetics*, 9(1):62–66, 1979.

D. Pathak, P. Krahenbuhl, and T. Darrell. Constrained convolutional neural networks for weakly supervised segmentation. In *ICCV*, 2015.

V. Petsiuk, A. Das, and K. Saenko. RISE: randomized input sampling for explanation of black-box models. In *BMVC*, 2018.

V. Petsiuk et al. Black-box explanation of object detectors via saliency maps. *CoRR*, abs/2006.03204, 2020.

S. Petushi, F. U. Garcia, M. M. Haber, et al. Large-scale computations on histology images reveal grade-differentiating parameters for breast cancer. *BMC medical imaging*, 6(1):14, 2006.

P. O. Pinheiro and R. Collobert. From image-level to pixel-level labeling with convolutional networks. In *CVPR*, 2015.

G. Quellec, G. Cazuguel, B. Cochener, and M. Lamard. Multiple-instance learning for medical image and video analysis. *IEEE reviews in Biomedical Engineering*, 10:213–234, 2017.

H. Qureshi, O. Sertel, N. Rajpoot, R. Wilson, and M. Gurcan. Adaptive discriminant wavelet packet transform and local binary patterns for meningioma subtype classification. In *MICCAI*, 2008.

A. Rahimi, A. Shaban, T. Ajanthan, R.I. Hartley, and B. Boots. Pairwise similarity knowledge transfer for weakly supervised object localization. In *ECCV*, 2020.

J. Redmon, S. K. Divvala, R. B. Girshick, et al. You only look once: Unified, real-time object detection. In *CVPR*, 2016.

S. Ren, K. He, R.B. Girshick, and J. Sun. Faster R-CNN: towards real-time object detection with region proposal networks. In *NeurIPS*, 2015.

M. T. Ribeiro, S. Singh, and C. Guestrin. "why should i trust you?": Explaining the predictions of any classifier. In *ACM SIGKDD 2016*, 2016.

O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*, 2015.

L. Roux, D. Racoceanu, N. Loménie, et al. Mitosis detection in breast cancer histological images an icpr 2012 contest. *Journal of Pathology Informatics*, 4, 2013.

H. Saleem, A. Raza Shahid, and B. Raza. Visual interpretability in 3d brain tumor segmentation network. *Comput. Biol. Medicine*, 133:104410, 2021.

W. Samek, G. Montavon, A. Vedaldi, L. K. Hansen, and K.-R. Müller, editors. *Explainable AI: Interpreting, Explaining and Visualizing Deep Learning*, volume 11700 of *Lecture Notes in Computer Science*. Springer, 2019.

W. Samek, G. Montavon, S. Lapuschkin, C. J. Anders, and K.-R. Müller. Toward interpretable machine learning: Transparent deep neural networks and beyond. *CoRR*, abs/2003.07631, 2020.

S. Sedai, D. Mahapatra, Z. Ge, et al. Deep multiscale convolutional feature learning for weakly supervised localization of chest pathologies in x-ray images. In *Intern. Workshop on Machine Learning in Medical Imaging*, 2018.

R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *ICCV*, 2017.

M. Shah, D. Wang, C. Rubadue, et al. Deep learning assessment of tumor proliferation in breast cancer histological images. In *Int. Conf. on Bioinformatics and Biomedicine*, 2017.

F. Sheikhzadeh, M. Guillaud, and R. K. Ward. Automatic labeling of molecular biomarkers of whole slide immunohistochemistry images using fully convolutional networks. *CoRR*, abs/1612.09420, 2016.

K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In Yoshua Bengio and Yann LeCun, editors, *ICLR*, 2015.

K.K. Singh and Y.J. Lee. Hide-and-seek: Forcing a network to be meticulous for weakly-supervised object and action localization. In *ICCV*, 2017.

K. Sirinukunwattana, J. P.W. Pluim, H. Chen, et al. Gland segmentation in colon histology images: The glas challenge contest. *Medical Image Analysis*, 35:489–502, 2017.

H. Song, M. Kim, D. Park, and J.-G. Lee. Learning from noisy labels with deep neural networks: A survey. *CoRR*, abs/2007.08199, 2020.

F. A. Spanhol, L. S. Oliveira, C. Petitjean, et al. Breast cancer histopathological image classification using convolutional neural networks. In *International Joint Conference on Neural Network*, 2016a.

F. A. Spanhol, L. S. Oliveira, C. Petitjean, et al. A dataset for breast cancer histopathological image classification. *IEEE Transactions on Biomedical Engineering*, 63(7):1455–1462, 2016b.

C.L. Srinidhi, O. Ciga, and A.L. Martel. Deep neural network models for computational histopathology: A survey. *CoRR*, abs/1912.12378, 2019.

N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *JMLR*, 15(56):1929–1958, 2014.

T. Stegmüller, A. Spahr, B. Bozorgtabar, and J.-P. Thiran. Scorenet: Learning non-uniform attention and augmentation for transformer-based histopathological image classification. *CoRR*, abs/2202.07570, 2022.

P.J. Sudharshan, C. Petitjean, F. Spanhol, et al. Multiple instance learning for histopathological breast cancer image classification. *Expert Systems with Applications*, 117:103 – 111, 2019.

S. Sukhbaatar, J. Bruna, M. Paluri, et al. Training convolutional networks with noisy labels. *coRR*, abs/1406.2080, 2014.

C. Sun, M. Paluri, R. Collobert, et al. Pronet: Learning to propose object-specific boxes for cascaded neural networks. In *CVPR*, 2016.

C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna. Rethinking the inception architecture for computer vision. In *CVPR*, 2016.

A. Tabesh, M. Teverovskiy, H.-Y. Pang, et al. Multifeature prostate cancer diagnosis and gleason grading of histological images. *IEEE transactions on Medical Imaging*, 26(10): 1366–1378, 2007.

J. Tang, R. M. Rangayyan, J. Xu, et al. Computer-aided detection and diagnosis of breast cancer with mammography: recent advances. *IEEE Transactions on Information Technology in Biomedicine*, 13(2):236–251, 2009.

M. Tang, A. Djelouah, F. Perazzi, Y. Boykov, and C. Schroers. Normalized Cut Loss for Weakly-supervised CNN Segmentation. In *CVPR*, 2018.

T.E. Tavolara, M.K.K. Niazi, M. Ginese, C. Piedra-Mora, D. M. Gatti, G. Beamer, and M. N. Gurcan. Automatic discovery of clinically interpretable imaging biomarkers for mycobacterium tuberculosis supersusceptibility using deep learning. *EBioMedicine*, 62: 103094, 2020.

E.u. W. Teh, M. Rochan, and Y. Wang. Attention networks for weakly supervised object localization. In *BMVC*, 2016.

H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, and H. Jegou. Training data-efficient image transformers &; distillation through attention. In *ICML*, 2021.

J.K. Tsotsos, S. M. Culhane, W.Y.K. Wai, Y. Lai, N. Davis, and F. Nuflo. Modeling visual attention via selective tuning. *Artificial Intelligence*, 78(1):507–545, 1995. Special Volume on Computer Vision.

A.F.M. Shahab Uddin, M.S. Monira, W. Shin, T. Chung, and S.-H. Bae. Saliencymix: A saliency guided data augmentation strategy for better regularization. In *ICLR*, 2021.

M. Veta, J.P.W. Pluim, P. J. Van Diest, et al. Breast cancer histopathology image analysis: A review. *IEEE Transactions on Biomedical Engineering*, 61(5):1400–1411, 2014.

F. Wan, P. Wei, J. Jiao, et al. Min-entropy latent model for weakly supervised object detection. In *CVPR*, 2018.

D. Wang, D. J. Foran, J. Ren, et al. Exploring automatic prostate histopathology image gleason grading via local structure modeling. In *Int. Conf. the IEEE Engineering in Medicine and Biology Society*, 2015.

H. Wang, Z. Wang, M. Du, F. Yang, Z. Zhang, S. Ding, P. Mardziel, and X. Hu. Score-cam: Score-weighted visual explanations for convolutional neural networks. In *CVPR workshop*, 2020.

X. Wang, Y. Yan, P. Tang, X. Bai, and W. Liu. Revisiting multiple instance neural networks. *Pattern Recognition*, 74:15–24, 2018.

J. Wei, Q. Wang, Z. Li, S. Wang, S. K. Zhou, and S. Cui. Shallow feature matters for weakly supervised object localization. In *CVPR*, 2021.

Y. Wei, J. Feng, X. Liang, M.-M. Cheng, Y. Zhao, and S. Yan. Object region mining with adversarial erasing: A simple classification to semantic segmentation approach. In *CVPR*, 2017.

Y. Wei, H. Xiao, H. Shi, Z. Jie, J. Feng, and T.S. Huang. Revisiting dilated convolution: A simple approach for weakly- and semi-supervised semantic segmentation. In *CVPR*, 2018.

K. L. Weind, C. F. Maier, B. K. Rutt, et al. Invasive carcinomas and fibroadenomas of the breast: comparison of microvessel distributions–implications for imaging modalities. *Radiology*, 208(2):477–483, 1998.

J. Xie, R. Liu, I.V. Joseph Luttrell, et al. Deep learning based analysis of histopathological images of breast cancer. *Frontiers in Genetics*, 10, 2019.

J. Xu, X. Luo, G. Wang, et al. A deep convolutional neural network for segmenting and classifying epithelial and stromal regions in histopathological images. *Neurocomputing*, 191:214–223, 2016.

H. Xue, C. Liu, F. Wan, J. Jiao, X. Ji, and Q. Ye. Danet: Divergent activation for weakly supervised object localization. In *ICCV*, 2019.

S. Yang, Y. Kim, Y. Kim, and C. Kim. Combinational class activation maps for weakly supervised object localization. In *WACV*, 2020.

S. Yun, D. Han, S. Chun, S.J. Oh, Y. Yoo, and J. Choe. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *ICCV*, 2019.

M. D. Zeiler and R. Fergus. Visualizing and understanding convolutional networks. In *ECCV*, 2014.

C. Zhang, S. Bengio, M. Hardt, et al. Understanding deep learning requires rethinking generalization. *CoRR*, abs/1611.03530, 2017.

C.-L. Zhang, Y.-H. Cao, and J. Wu. Rethinking the route towards weakly supervised object localization. In *CVPR*, 2020a.

H. Zhang, M. Cissé, Y.N. Dauphin, and D. Lopez-Paz. mixup: Beyond empirical risk minimization. In *ICLR*, 2018a.

J. Zhang, S. A. Bargal, Z. Lin, et al. Top-down neural attention by excitation backprop. *International Journal of Computer Vision*, 126(10):1084–1102, 2018b.

X. Zhang, Y. Wei, J. Feng, Y. Yang, and T.S. Huang. Adversarial complementary learning for weakly supervised object localization. In *CVPR*, 2018c.

X. Zhang, Y. Wei, G. Kang, Y. Yang, and T.S. Huang. Self-produced guidance for weakly-supervised object localization. In *ECCV*, 2018d.

X. Zhang, N. Wang, H. Shen, S. Ji, X. Luo, and T. Wang. Interpretable deep learning under fire. In *USENIX Security Symposium*, 2020b.

X. Zhang, Y. Wei, and Y. Yang. Inter-image communication for weakly supervised localization. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *ECCV*, Lecture Notes in Computer Science, 2020c.

Y. Zhang, P. Tiňo, A. Leonardis, and Ke Tang. A survey on neural network interpretability. *IEEE Trans. Emerg. Top. Comput. Intell.*, 5(5):726–742, 2021.

B. Zhou, A. Khosla, A. Lapedriza, et al. Learning deep features for discriminative localization. In *CVPR*, 2016.

Y. Zhou, Y. Zhu, Q. Ye, Q. Qiu, and J. Jiao. Weakly supervised instance segmentation using class peak response. In *CVPR*, 2018.

Z.-H. Zhou. Multi-instance learning: A survey. *Department of Computer Science & Technology, Nanjing University, Tech. Rep*, 2004.

Z.-H. Zhou. A brief introduction to weakly supervised learning. *National Science Review*, 5 (1):44–53, 2017.

Y. Zhu, Y. Zhou, Q. Ye, et al. Soft proposal networks for weakly supervised object localization. In *ICCV*, 2017.

Appendix A. Hyper-parameter search

Table 8 presents the general hyper-parameters used for all methods. Table 9 holds hyper-parameters for specific methods.

Table 8: General hyper-parameters.

Hyper-parameter	Value
Fully sup. model f	U-Net
Backbones	VGG16, InceptionV3, ResNet50.
Optimizer	SGD
Nesterov acceleration	True
Momentum	0.9
Weight decay	0.0001
Learning rate	$\in \{0.01, 0.001, 0.1\}$
Learning rate decay	`GlaS`: 0.1 each 250 epochs. `CAMELYON16`: 0.1 each 5 epochs.
Mini-batch size	32
Random flip	Horizontal/vertical random flip
Random color jittering	Brightness, contrast, and saturation at 0.5 and hue at 0.05
Image size	Resize image to 225×225. Then, crop random patches of 224×224
Learning epochs	`GlaS`: 1000, `CAMELYON16`: 20

Appendix B. CAMELYON16 protocol for WSL

This appendix provides details on our protocol for creating a WSOL benchmark from the CAMELYON16 dataset (Ehteshami Bejnordi et al., 2017). Samples are patches from WSIs, and each patch has two levels of annotation:

- Image-level label y: the class of the patch, where $y \in \{\texttt{normal}, \texttt{metastatic}\}$.

- Pixel-level label $\boldsymbol{Y} = \{0,1\}^{H^{\text{in}} \times W^{\text{in}}}$: a binary mask where the value 1 indicates a `metastatic` pixel, and 0 a `normal` pixel. For `normal` patches, this mask will contain 0 only.

First, we split the CAMELYON16 dataset into training, validation, and test sets at the *WSI-level*. This prevents patches from the same WSI from ending up in different sets. All patches are sampled with the highest resolution from WSI –i.e., level = 0 in WSI terminology–. Next, we present our methodology of sampling metastatic and normal patches.

Table 9: Per-method hyper-parameters. Notation in this table follows the same notation in the original papers.

Hyper-parameter	Value
LSE (Sun et al., 2016)	$q \in \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$
HaS (Singh and Lee, 2017)	Grid size $\in \{8, 16, 32, 44, 56\}$, Drop rate $\in \{0.2, 0.3, 0.4, 0.5, 0.6\}$
WILDCAT (Durand et al., 2017)	$\alpha \in \{0.1, 0.6\}$ kmax $\in \{0.1, 0.3, 0.5, 0.6, 0.7\}$ kmin $\in \{0.1, 0.2, 0.3\}$ Modalities = 5
ACoL (Zhang et al., 2018c)	$\delta \in \{0.5, 0.6, 0.7, 0.8, 0.9\}$
SPG (Zhang et al., 2018d)	$\delta_{1h} \in \{0.5, 0.7\}$ $\delta_{1l} \in \{0.01, 0.05, 0.1\}$ $\delta_{2h} \in \{0.5, 0.6, 0.7\}$ $\delta_{2l} \in \{0.01, 0.05, 0.1\}$ $\delta_{3h} \in \{0.5, 0.6, 0.7\}$ $\delta_{3l} \in \{0.01, 0.05, 0.1\}$
Deep MIL (Ilse et al., 2018)	Mid-channels = 128. Gated attention: True/False.
PRM (Zhou et al., 2018)	$r \in \{3, 5, 7, 9, 11, 13\}$ Kernel stride $\in \{1, 3, 5, 7, 9, 11, 13\}$
ADL (Choe and Shim, 2019)	Drop rate $\in \{0., 0.25, 0.35, 0.45, 0.50, 0.75\}$ $\gamma \in \{0.75, 0.85, 0.90\}$
CutMix (Yun et al., 2019)	$\alpha = 1.0$ $\lambda \in \{0.5, 0.6, 0.7, 0.8, 0.9, 1.\}$
MAXMIN (Belharbi et al., 2022b)	Same as (Belharbi et al., 2022b)
NEGEV (Belharbi et al., 2022a)	Same as (Belharbi et al., 2022a)

Sampling metastatic patches. Metastatic patches are sampled only from metastatic WSIs around the cancerous regions. Sampled patches will have an image-level label, and a pixel-level label. The sampling follows these steps:

1. Consider a metastatic WSI.
2. Sample a patch $\boldsymbol{x}$ with size (H, W).
3. Binarize the patch into a mask $\boldsymbol{x}^{\text{b}}$ using the OTSU method (Otsu, 1979). Pixels with value 1 indicate tissue.
4. Let $p_t^{\boldsymbol{x}^{\text{b}}}$ be the tissue percentage within $\boldsymbol{x}^{\text{b}}$. If $p_t^{\boldsymbol{x}^{\text{b}}} < p_t$, discard the patch.
5. Compute the metastatic binary mask $\boldsymbol{Y}$ of the patch $\boldsymbol{x}$ using the pixel-level annotation of the WSI (values of 1 indicate a metastatic pixel).
6. Compute the percentage $p_m^{\boldsymbol{x}}$ of metastatic pixels within $\boldsymbol{Y}$.
7. If $p_m^{\boldsymbol{x}} < p_0$, discard the patch. Else, keep the patch $\boldsymbol{x}$ and set $y = \texttt{metastatic}$ and $\boldsymbol{Y}$ is its pixel-level annotation.

We note that we sample *all* possible metastatic patches from CAMELYON16 using the above approach. Sampling using such an approach will lead to a large number of metastatic patches with a high percentage of cancerous pixels (patches sampled from the center of the

cancerous regions). These patches will have their binary annotation mask $\boldsymbol{Y}$ full of 1s. Using these patches will shadow the performance measure of the localization of cancerous regions. To avoid this issue, we propose to perform a calibration of the sampled patches in order to remove most such patches. We define two categories of metastatic patches:

1. **Category 1**: Contains patches with $p_0 \leq p_m^{\boldsymbol{x}} \leq p_1$. Such patches are rare, and contain only a small region of cancerous pixels. They are often located at the edge of the cancerous regions within a WSI.
2. **Category 2**: Contains patches with $p_m^{\boldsymbol{x}} > p_1$. Such patches are extremely abundant, and contain a very large region of cancerous pixels (most often the entire patch is cancerous). Such patches are often located inside the cancerous regions within a WSI.

Our calibration method consists in keeping all patches within **Category 1**, and discarding most of the patches in **Category 2**. To this end, we apply the following sampling approach:

1. Assume we have n patches in **Category 1**. We will sample $n \times p_n$ patches from **Category 2**, where p_n is a predefined percentage.
2. Compute the histogram of the frequency of the percentage of cancerous pixels within all patches, assuming a histogram with b bins.
3. Among all the bins with $p_m^{\boldsymbol{x}} > p_1$, pick a bin uniformly.
4. Pick a patch within that bin uniformly.

This procedure is repeated until we sample $n \cdot p_n$ patches from **Category 2**.

In our experiments, patches are not overlapping. We use the following configuration: $p_0 = 20\%,\ p_1 = 50\%,\ p_t = 10\%,\ p_n = 1\%$. The number of bins in the histogram is obtained by dividing the interval $[0, 1]$ with a delta of 0.05. These hyper-parameters are not validated to optimize the performance of the models, but are set in a reasonable way to *automatically* sample consistent and unbalanced patches without the need to manually check the samples. Patch size is set to 512×512. Figure 10 illustrates an example of metastatic patches and their corresponding masks.

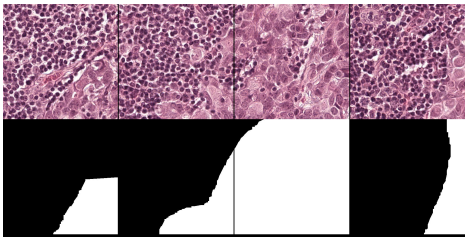


Figure 10: Example of metastatic patches with size 512×512 sampled from CAMELYON16 dataset (WSI: `tumor_001.tif`). *Top row*: Patches. *Bottom row*: Masks of metastatic regions (white color).

Sampling normal patches. Normal patches are sampled only from normal WSI. A normal patch is sampled randomly and uniformly from the WSI (without repetition or overlapping). If the patch has enough tissue ($p_t^{\boldsymbol{x}^{\mathrm{b}}} \geq p_t$), the patch is accepted. Tissue mass measurement

is performed at level = 6 where it is easy for the OTSU binarization method to split the tissue from the background. We double-check the tissue mass at level = 0.

Let us consider a set (train, validation, or test) at patch level. We first pick the corresponding metastatic patches from the metastatic WSI, assuming n_m is their total number. Assuming there are h normal WSIs in this set, we sample the same number of normal patches as the total number of metastatic ones. In order to mix the patches from all the normal WSI, we sample $\frac{n_m}{h}$ normal patches per normal WSI. In our experiment, we use the same setup as in the metastatic patches sampling case: $p_t = 10\%$. Figure 11 illustrates an example of normal patches.

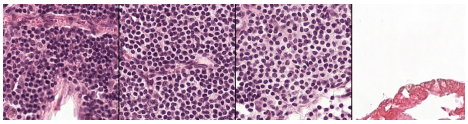


Figure 11: Example of normal patches with size 512×512 sampled from CAMELYON16 dataset (WSI: `normal_001.tif`).

Appendix C. Visual results

In this section, we provide visual results for the localization of different methods using the ResNet50 backbone: Figure 12, and Figure 13 for `GlaS` test set; and Figure 14 and Figure 15 for `CAMELYON16` test set.

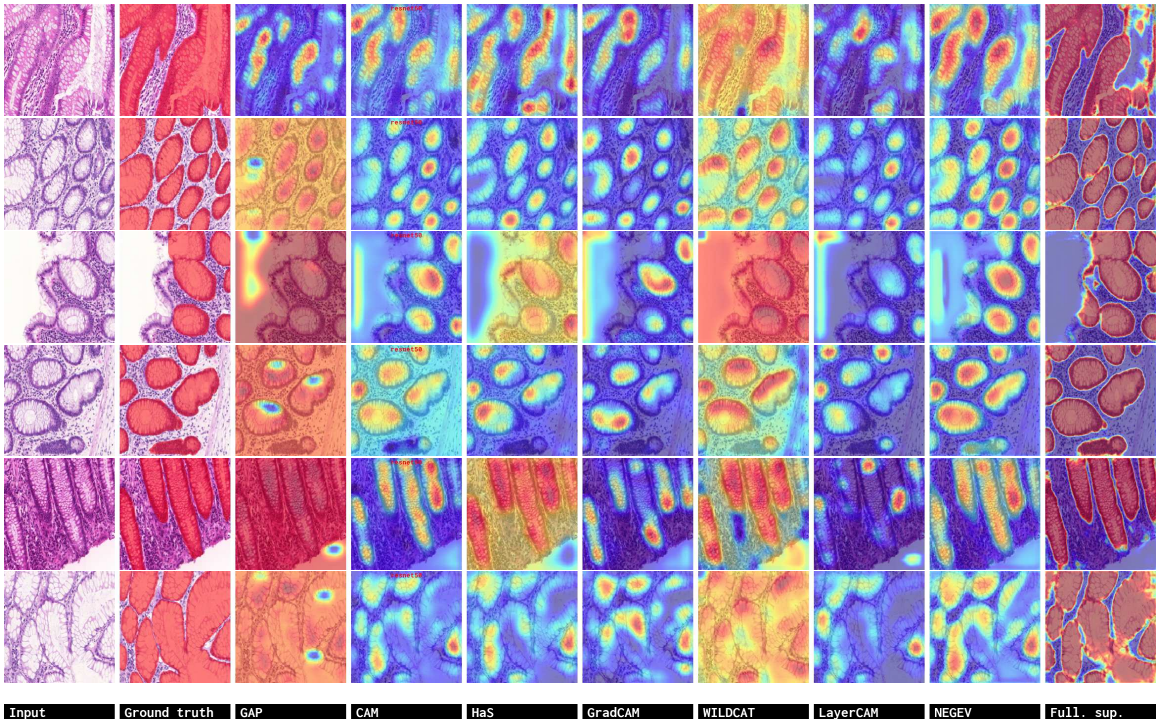


Figure 12: CAM predictions over **benign** test samples for `GlaS`. Ground truth ROIs are indicated with a red mask highlighting glands. In all predictions, strong CAM's activations indicate glands. Backbone: ResNet50.

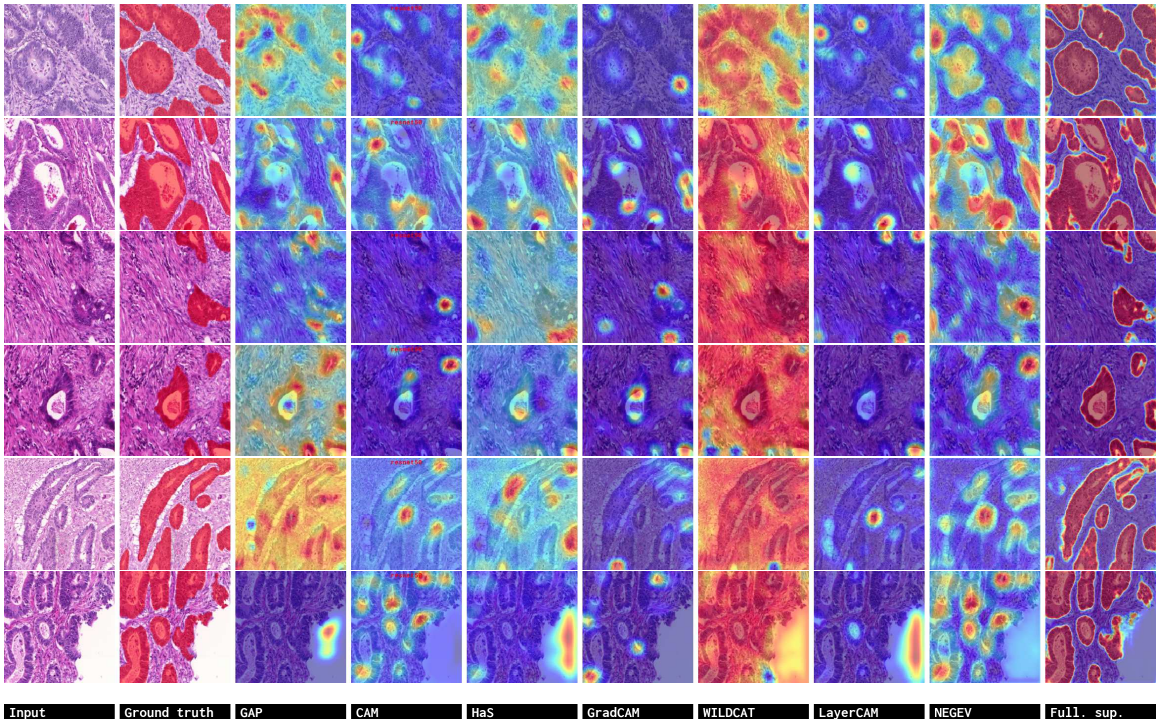


Figure 13: Predictions over **malignant** test samples for `GlaS`. Ground truth ROIs are indicated with a red mask highlighting glands. In all predictions, strong CAM's activations indicate glands. Backbone: ResNet50.

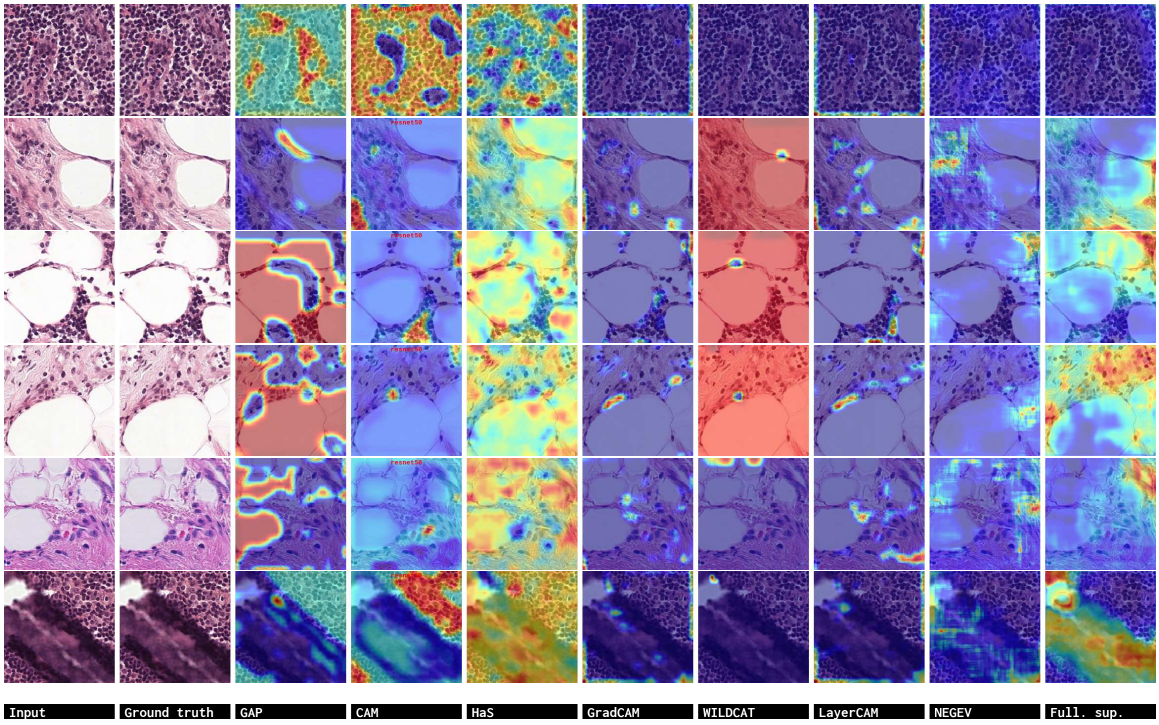


Figure 14: Predictions over **normal** test samples for `CAMELYON16`. Ground truth ROIs are indicated with a red mask highlighting metastatic regions. In all predictions, strong CAM's activations indicate metastatic regions. Backbone: ResNet50.

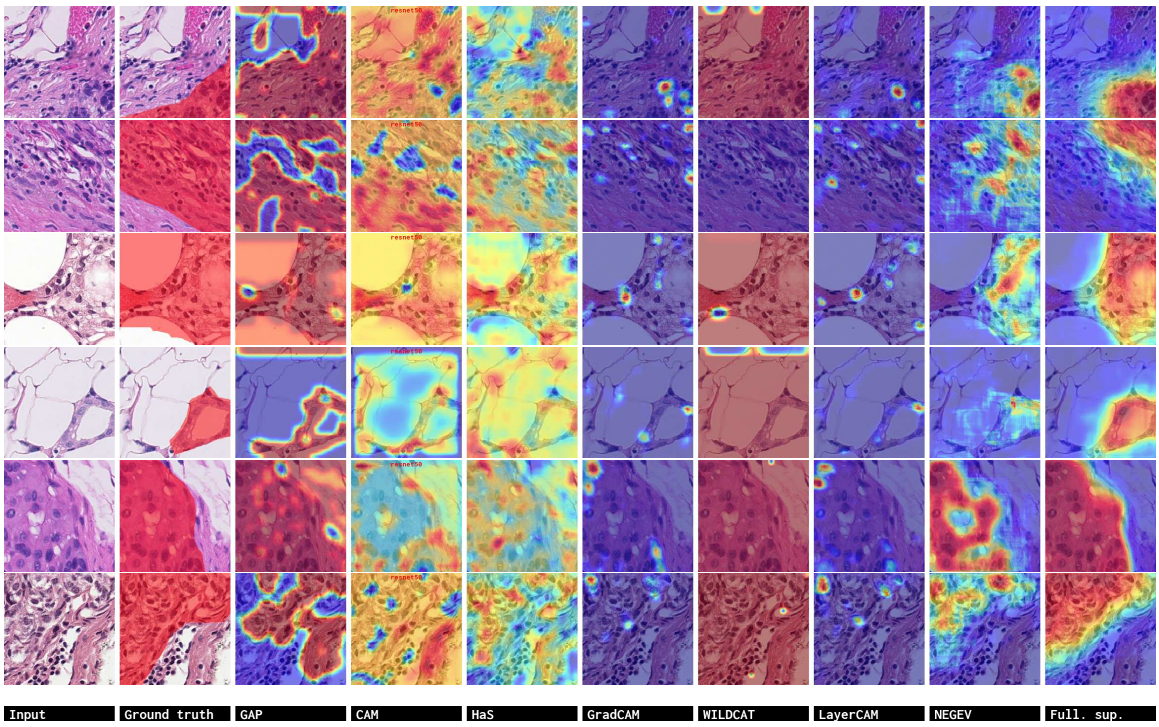


Figure 15: Predictions over **metastatic** test samples for `CAMELYON16`. Ground truth ROIs are indicated with a red mask highlighting metastatic regions. In all predictions, strong CAM's activations indicate metastatic regions. Backbone: ResNet50.