Medical Image Analysis 97 (2024) 103259

Contents lists available at ScienceDirect

Medical Image Analysis

journal homepage: www.elsevier.com/locate/media

Uncertainty-aware multiple-instance learning for reliable classification: Application to optical coherence tomography

Coen de Vente [a,b,c,*], Bram van Ginneken [c], Carel B. Hoyng [d], Caroline C.W. Klaver [d,e], Clara I. Sánchez [a,b]

[a] Quantitative Healthcare Analysis (QurAI) Group, Informatics Institute, University of Amsterdam, Amsterdam, Noord-Holland, Netherlands
[b] Department of Biomedical Engineering and Physics, Amsterdam University Medical Center, Amsterdam, Noord-Holland, Netherlands
[c] Diagnostic Image Analysis Group (DIAG), Department of Radiology and Nuclear Medicine, Radboudumc, Nijmegen, Gelderland, Netherlands
[d] Department of Ophthalmology, Radboudumc, Nijmegen, Gelderland, Netherlands
[e] Ophthalmology & Epidemiology, Erasmus MC, Rotterdam, Zuid-Holland, Netherlands

ARTICLE INFO

Keywords:
Out-of-distribution detection
Generalizability
Interpretability
Optical coherence tomography

ABSTRACT

Deep learning classification models for medical image analysis often perform well on data from scanners that were used to acquire the training data. However, when these models are applied to data from different vendors, their performance tends to drop substantially. Artifacts that only occur within scans from specific scanners are major causes of this poor generalizability. We aimed to enhance the reliability of deep learning classification models using a novel method called Uncertainty-Based Instance eXclusion (UBIX). UBIX is an *inference-time* module that can be employed in multiple-instance learning (MIL) settings. MIL is a paradigm in which instances (generally crops or slices) of a bag (generally an image) contribute towards a bag-level output. Instead of assuming equal contribution of all instances to the bag-level output, UBIX detects instances corrupted due to local artifacts on-the-fly using uncertainty estimation, reducing or fully ignoring their contributions before MIL pooling. In our experiments, instances are 2D slices and bags are volumetric images, but alternative definitions are also possible. Although UBIX is generally applicable to diverse classification tasks, we focused on the staging of age-related macular degeneration in optical coherence tomography. Our models were trained on data from a single scanner and tested on external datasets from different vendors, which included vendor-specific artifacts. UBIX showed reliable behavior, with a slight decrease in performance (a decrease of the quadratic weighted kappa (κ_w) from 0.861 to 0.708), when applied to images from different vendors containing artifacts; while a state-of-the-art 3D neural network without UBIX suffered from a significant detriment of performance (κ_w from 0.852 to 0.084) on the same test set. We showed that instances with unseen artifacts can be identified with OOD detection. UBIX can reduce their contribution to the bag-level predictions, improving reliability without retraining on new data. This potentially increases the applicability of artificial intelligence models to data from other scanners than the ones for which they were developed. The source code for UBIX, including trained model weights, is publicly available through https://github.com/qurAI-amsterdam/ubix-for-reliable-classification.

1. Introduction

Deep learning models for medical image analysis applications are often trained on data that is acquired with one or a selected number of scanner types and/or acquisition protocols. When applying these trained models on data from different scanners or protocols, the performance tends to plummet (Yanagihara et al., 2020; De Fauw et al., 2018). This negatively affects the reliability of these systems, which is a main aspect of trustworthy AI (González-Gonzalo et al., 2021; European Commission, 2019), and its wide integration and adoption in clinical practice. In general, convolutional neural networks (CNNs) are known to fail when they are applied under dataset shift or to out-of-distribution (OOD) datasets; and approaches to address this effect are being investigated (Ovadia et al., 2019). This OOD nature of the data occasionally only stems from local areas in images, such as local artifacts. These local artifacts occur frequently in data from specific vendors or particular scanning protocols, and can be found in multiple medical imaging fields, such as contrast-enhanced mammography data (Neppalli et al., 2021) and optical coherence tomography (Bazvand and Ghassemi, 2020). In images with these types of artifacts, there generally are sufficient parts in a sample which are in-distribution (ID)

* Corresponding author at: Quantitative Healthcare Analysis (QurAI) Group, Informatics Institute, University of Amsterdam, Amsterdam, Noord-Holland, Netherlands.
E-mail address: research@coendevente.com (C. de Vente).

https://doi.org/10.1016/j.media.2024.103259
Received 21 January 2023; Received in revised form 17 June 2024; Accepted 24 June 2024
Available online 27 June 2024

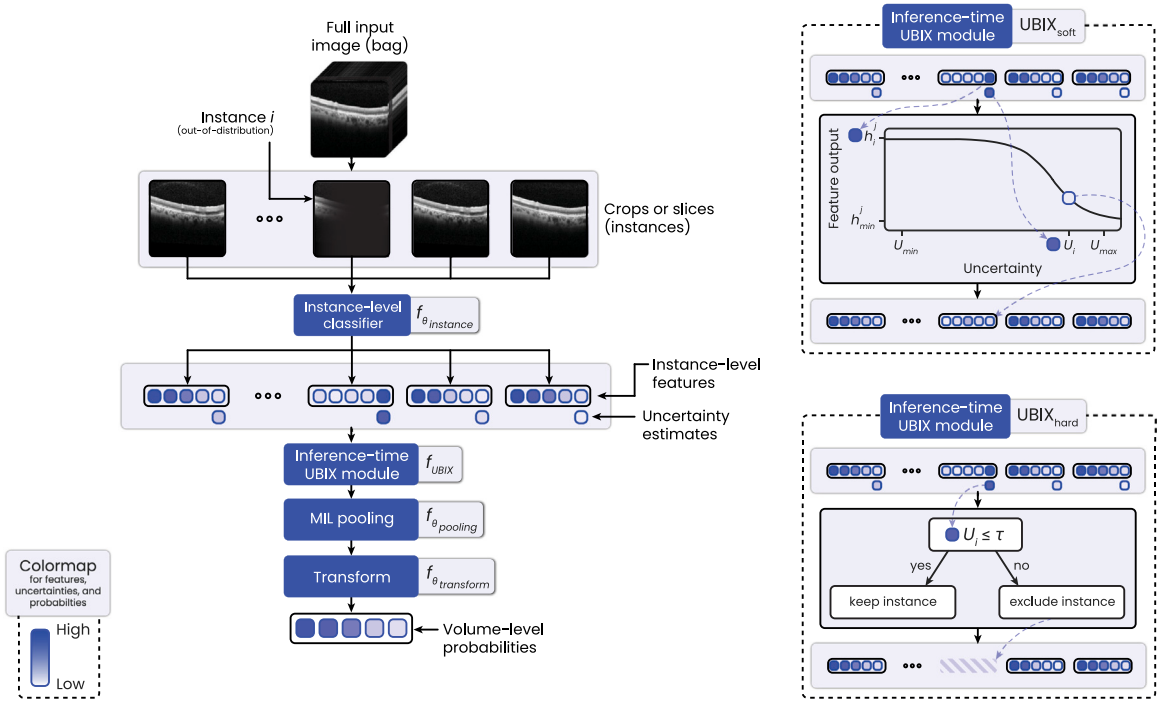
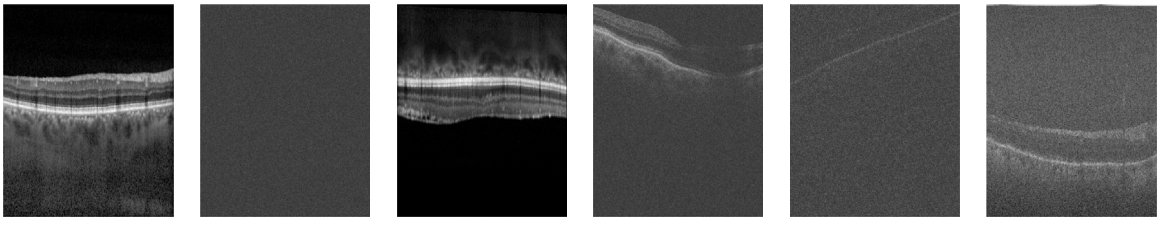


Fig. 1. Overview of the general MIL pipeline (left) and how the two UBIX variants are integrated into this pipeline (right). Each MIL instance is fed into the same classifier. During inference time, a UBIX function converts pre-UBIX instance-level features based on their respective uncertainties to post-UBIX instance-level features. The instance-level post-UBIX features are then converted to bag-level outputs using MIL pooling. During training, the pre-UBIX features are fed directly into the MIL pooling function. The thin dashed blue arrows indicate the flow of one example feature and its associated uncertainty. h_i^j is the instance-level feature value for instance i and feature j. h_{min}^j is the minimum instance-level feature value in the validation set for feature j. U_i is the uncertainty associated with instance i. U_{min} and U_{max} are the minimum and maximum uncertainty in the validation set, respectively.



(a) No imaging artifacts. (b) Blinking artifact. (c) Vertically flipped. (d) Shadow. (e) Noise. (f) Noise.

Fig. 2. Examples of real imaging artifacts in OCT. Subfigure (a) shows a normal B-scan without imaging artifacts for reference. The other subfigures show various common imaging artifacts. The images (a) and (c) originate from the H_{val} dataset, while images (b), (d), and (e) originate from the T_{test} dataset, and image (f) originates from the B_{test} dataset.

to form a correct prediction if the model would in some way only focus on those parts of the data and neglect the OOD areas.

To achieve this increased robustness to local OOD areas in images, we propose Uncertainty-based Instance eXclusion (UBIX). This approach builds upon multiple-instance learning (MIL), a form of weakly supervised learning popular in medical image analysis (Cheplygina et al., 2019; Ilse et al., 2018). In MIL, a labeled bag (usually the whole input image) consists of multiple unlabeled instances (usually image patches, regions or slices). During deep MIL, instances are considered individually by a neural network, and the instance-level outputs, each contributing equally, are combined to obtain a bag-level prediction using a MIL pooling function (Wang et al., 2018). Instead of assuming equal contribution, the UBIX approach assumes that some of the instances might be corrupted due to local artifacts, identifies these instances on-the-fly using uncertainty estimation, and reduces or ignores its contribution to the bag-level prediction using the so-called UBIX function before the MIL pooling function (see Fig. 1). To the best of our knowledge, this is the first method that uses OOD detection in such a manner to increase reliability.

Although our method is applicable to any instance definition (such as 2D or 3D patches) or bag definition, we focus on MIL problems in which slices are instances and full 3D volumes are bags. Specifically, we focus on classifying age-related macular degeneration (AMD) in optical coherence tomography (OCT), as there is a plurality of manufacturers and scanner versions in the field of OCT. Swanson and Fujimoto (2017) list fourteen companies that produce OCT scanners for ophthalmic applications and the type of imaging artifacts that occur can differ substantially across scanners (Bazvand and Ghassemi, 2020). These artifacts include slices (B-scans) that are fully black due to blinking, vertically flipped B-scans, shadows, and noise. For example, blinking artifacts and very noisy B-scans are much less common in certain scanners, specifically ones with higher speed and eye tracking software (Bazvand and Ghassemi, 2020), due to internal B-scan averaging and rescanning strategies (Puzyeyeva et al., 2011). A number of examples of imaging artifacts occurring in OCT are shown in Fig. 2.

While our primary investigation centers on UBIX's assessment in ophthalmic applications, its potential utility spans diverse medical image tasks. UBIX aims to increase the robustness to local imaging artifacts, which may be under-represented during training but at least sporadically manifest in clinical settings. A relevant example pertains to the observation from Linmans et al. (2023) of small colon tissue regions in a histopathology prostate training set, leading to locally

OOD instances. Furthermore, various artifacts in histopathology, such as foreign objects, scratches, elastic deformations, and fingerprints, have been outlined by Schömig-Markiefka et al. (2021). These artifacts can result from natural origins, preprocessing, and digitization steps. Wellenberg et al. (2018) discussed metal artifacts in CT scans. Commercially available CT scanners employ a range of techniques to address metal artifacts, leading to varied appearances of these artifacts in scans (Wellenberg et al., 2018). In these CT and histopathology examples, instances could be defined as crops (volumetric crops in CT, and 2D crops in histopathology), while the whole slide image and CT volume could be the bags. We expect that classification models developed in these contexts may potentially benefit from UBIX as well.

We evaluate the generalizability of our proposed models by training on data acquired with a scanner from one vendor, while evaluating with data from scanners of other vendors. We show that UBIX increases this generalizability using a baseline comparison. Moreover, we systematically analyze the ability of UBIX to detect OOD instances by gradually introducing artificial image artifacts that occur naturally as well. The trained algorithm is publicly available for inference on the online platform of Grand Challenge.[1]

2. Related work

2.1. Multiple-instance learning in medical imaging

One of the most common medical application in which MIL is applied is histopathology (Xu et al., 2019; Patil et al., 2019; Chikontwe et al., 2020; Tomczak et al., 2018), mainly because it is very labor-intensive and time-consuming to manually annotate entire whole slide images on instance-level. Ilse et al. (2018) used an attention-based MIL pooling layer and evaluated it on an MNIST-based dataset and histopathology datasets. Other medical modalities to which MIL with deep learning has been applied include ultrasound (Yin et al., 2019; Shin et al., 2018), computed tomography (Han et al., 2020; Xu et al., 2020) and magnetic resonance imaging (Zhu et al., 2021; Qiu et al., 2021).

Several methods have been proposed to integrate confidence or uncertainty estimation at instance-level in MIL approaches. For example, Integrated Instance-Level and Bag-Level MIL (IIB-MIL) (Ren et al., 2023) is a MIL approach that integrates instance-level and bag-level supervision, using a frozen instance-level encoder pre-trained with weak supervision. After pre-training, it optimizes the instance-level features with a label-disambiguation module, which incorporates a confidence bank, aiming to mitigate the effect of training with noisy labels. This confidence bank is a form of uncertainty estimation but serves a different purpose in their pipeline than the uncertainty estimation in UBIX does.

Weakly supervised knowledge distillation (WENO) (Qu et al., 2022) is a student–teacher framework that consists of an instance-level classifier (the student) and a bag-level classifier (the teacher). Attention-based scores from the teacher are used to train the student. It also uses hard positive instance mining, which relies on a form of confidence estimation to find hard examples. In WENO, this ensures that the model not only learns to define a positive bag from easy positive instances but also predicts harder positive instances as positive. Even though Qu et al. (2022) show this is an effective strategy to increase model performance, they do not aim to increase robustness to OOD instances. This is in contrast with the aim of UBIX.

Unlike our proposed method, these approaches may suffer from limited robustness when transferred to other distributions, as they do not explicitly employ methods that aim to improve robustness when transferring to OOD data.

[1] https://grand-challenge.org/algorithms/amd-classification-in-oct-with-ubix/

2.2. Out-of-distribution detection

OOD detection is the identification of samples that originate from a different distribution than the training distribution. Such samples generally have high model predictive uncertainties, given a good uncertainty estimation method. Hendrycks and Gimpel (2016) proposed a simple baseline for OOD detection using the maximum class probability as confidence scores. Another early work was Monte Carlo dropout (MC-DO), in which they leveraged dropout to estimate uncertainty (Gal and Ghahramani, 2016). Ovadia et al. (2019) compared a number of methods for OOD detection and uncertainty estimation including MC-DO. They found that deep ensembling (Lakshminarayanan et al., 2017) was one of the top-performing methods for OOD detection. Since then, other popular methods for uncertainty estimation and OOD detection have been published (Hsu et al., 2020; Liu et al., 2020; Tack et al., 2020).

Uncertainty estimation has been investigated for medical image analysis as well, such as Mehrtash et al. (2020), who used deep ensembles to calibrate probabilities in segmentation maps. Calli et al. (2019) detected incorrect orientation or anatomy in X-rays using an OOD detection metric called FRODO, defined as the Mahalanobis distance of test samples to samples in the train set. Furthermore, Linmans et al. (2020, 2023) used multi-head CNNs, an approach similar to deep ensembles, to detect images with lymphoma in histopathology as OOD samples.

In general, uncertainty estimation in medical images are used as an additional output to assess the behavior of the developed models or to identify abnormalities as OOD samples. In contrast, our proposed approach takes into account OOD detection during inference to increase classification robustness against data shift.

2.3. Robustness against data shift in OCT

Related works have successfully applied machine learning methods for AMD classification from OCT, but did not specifically focus on robustness to OOD data (Apostolopoulos et al., 2017; Venhuizen et al., 2017; Rasti et al., 2017; Kurmann et al., 2019; Lee et al., 2017; Wang et al., 2020).

The following works studied robustness against data shift in OCT. De Fauw et al. (2018) used an OCT segmentation network of which the output was fed into a classification network, which in turn outputted a referral suggestion, diagnosis probabilities for multiple retinal disease features, such as choroidal neovascularization (CNV) and geographic atrophy (GA), and volume estimations of drusen and epiretinal membranes. The error rate on their internal test set with OCTs from the same scanner, as the development set, *i.e.*, Topcon, was 5.5%, but the error rate increased to 46.6% when transferred to an external set with OCTs from a different scanner, *i.e.*, Heidelberg Spectralis. When retraining their segmentation network with data from this scanner, the error rate improved to 3.4%. Seeböck et al. (2019) and Romo-Bucheli et al. (2020) used a CycleGAN to transform OCT scans acquired on a device that was not used during training to have a similar appearance as the training data. For retinal fluid (Seeböck et al., 2019; Romo-Bucheli et al., 2020) and layer (Romo-Bucheli et al., 2020) segmentation, they observed a generalizability improvement when applying this domain adaptation technique, compared to traditional transformation strategies.

The main downside of these methods is their requirement for – albeit annotated or not – data from the new setting. Acquiring and annotating this new data, as well as any potential retraining, is a time-consuming and expensive process. Moreover, if these models are unknowingly applied in settings that are highly different from the development setting, models can fail silently, potentially causing misdiagnoses. We propose a method that reduces the performance drop when a model is transferred to a setting unlike its development setting, without the requirement for acquiring or labeling data originating from this new setting.

3. Methods

Our paper introduces UBIX, a method that enhances the robustness of classification models to locally OOD data. Specifically, our approach is designed for scenarios where a portion of a given sample, such as a crop or slice of an image, differs from the training data distribution. UBIX is implemented in the MIL paradigm and is *solely applied during inference*. It can theoretically be applied seamlessly to any existing MIL model that has already been trained. In Section 3.1, we provide an introduction to MIL and related concepts. UBIX relies on the identification of local OOD areas in the input sample, by performing uncertainty estimation on MIL instances. In Section 3.2, we present various state-of-the-art uncertainty estimation methods and propose variants of uncertainty measures that are optimized for ordinal labels. Finally, Section 3.3 elucidates how UBIX works and how it integrates instance-level uncertainty estimation into MIL models.

3.1. Multiple instance learning (MIL)

3.1.1. The MIL paradigm

MIL is a form of supervised learning designed to address scenarios where our dataset is organized into bags of instances. Each bag, denoted as $X = \{\mathbf{x}_1, \dots, \mathbf{x}_I\}$, can contain a varying number of instances, and each bag is associated with a bag-level label, $Y \in \{1, \dots, C\}$, where C represents the number of possible classes. Individual instances within a bag do not have explicit labels. During training, only bag-level labels are available.

MIL employs an instance-level classifier, which transforms each instance $\mathbf{x}_i$ into a feature vector $\mathbf{h}_i \in \mathbb{R}^J$ using a function parameterized by $\theta_{instance}$, represented as $\mathbf{h}_i = f_{\theta_{instance}}(\mathbf{x}_i)$. There are two primary types of MIL approaches: instance-based and embedding-based. In the instance-based approach, J is equal to C, which means that the instance-level classifier produces instance-level logits for each class. In contrast, the embedding-based approach allows J to represent the number of features, potentially using a higher-dimensional space.

Once the instance-level classifier has been applied, MIL pooling is employed to aggregate these vectors into a bag-level feature representation, denoted as $\mathbf{h}_X$, using a function parameterized by $f_{\theta_{pooling}}$. More formally, $\mathbf{h}_X = f_{\theta_{pooling}}(\mathbf{h}_1, \dots, \mathbf{h}_I)$. In most MIL pooling functions, $\mathbf{h}_X \in \mathbb{R}^J$. However, when distribution pooling (Oner et al., 2023) is used, a marginal feature distribution is estimated instead of a single scalar value for each feature. This case is elaborated upon further in Section 3.1.2. Following MIL pooling, a bag-level representation transformation function, parameterized by $\theta_{transform}$, is applied to the bag-level feature vector $\mathbf{h}_X$. This transformation maps $\mathbf{h}_X$ to the predicted bag-level label, denoted as $\hat{Y}$, and is expressed as $\hat{Y} = f_{\theta_{transform}}(\mathbf{h}_X)$.

In deep learning-based MIL, the instance-level classifier $f_{\theta_{instance}}$ is typically implemented as a neural network. In the instance-based approach, the bag-level representation transformation $f_{\theta_{transform}}$ is the identity function, enabling a direct link between each class in the bag-level outputs and the outputs on the instance-level. In the embedding-based approach, $f_{\theta_{transform}}$ is parameterized using a neural network. Depending on the chosen pooling function, even the MIL pooling function $f_{\theta_{pooling}}$ may be parameterized using a neural network (see Section 3.1.2).

3.1.2. MIL pooling functions

The MIL pooling function $\theta_{pooling}$ aggregates instance-level feature vectors $\mathbf{h}_1, \dots, \mathbf{h}_I$, where $\mathbf{h}_i = [h_i^1, \dots, h_i^J]$ and $h_i^j \in \mathbb{R}$, to a bag-level feature vector $\mathbf{h}_X = [h_X^1, \dots, h_X^J]$, where $h_X^j \in \mathbb{R}$. A requirement for the MIL pooling function is that it can be applied to any number of instances J. The following common MIL pooling functions all follow this requirement and are also described by Oner et al. (2020):

- **Max pooling:** Max pooling selects the maximum value for each feature across all instances in the bag:

$$h_X^j = \max_{i=1}^{I} h_i^j, \quad \text{for } j = 1, 2, \dots, J.$$

- **Mean pooling:** Mean pooling computes the mean value of each feature across all instances in the bag:

$$h_X^j = \frac{1}{I} \sum_{i=1}^{I} h_i^j, \quad \text{for } j = 1, 2, \dots, J.$$

- **Attention pooling** (Ilse et al., 2018): Attention pooling uses a weighted average of instances, where the weights are determined by a neural network. This approach allows for generating bag-level features based on the importance of each instance. The Attention pooling mechanism is mathematically represented as follows:

$$h_X^j = \sum_{i=1}^{I} a_i h_i^j, \quad \text{for } j = 1, 2, \dots, J,$$

where a_i are the instance-specific attention weights, calculated as:

$$a_i = \frac{\exp(\mathbf{w}^T \tanh(\mathbf{V}\mathbf{h}_i^T))}{\sum_{k=1}^{I} \exp(\mathbf{w}^T \tanh(\mathbf{V}\mathbf{h}_k^T))}.$$

Here, $\mathbf{w} \in \mathbb{R}^{L \times 1}$ and $\mathbf{V} \in \mathbb{R}^{L \times J}$ are learnable parameters.

- **Distribution pooling** (Oner et al., 2020): Distribution pooling estimates marginal feature distributions for each feature dimension. This provides a richer representation of the data, as individual features are defined as distributions instead of single scalar values. The formula for distribution pooling involves estimating these marginal distributions. Given instance-level feature vectors $\mathbf{h}_1, \dots, \mathbf{h}_I$, the goal is to find a bag-level representation $\mathbf{h}_X = [p_X^1, p_X^2, \dots, p_X^J]$, and p_X^j is the estimated marginal distribution of the feature j. The estimated marginal distribution p_X^j is calculated using kernel density estimation with a Gaussian kernel having a standard deviation σ:

$$p_X^j(v) = \frac{1}{I} \sum_{i=1}^{I} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2\sigma^2}(v - h_i^j)^2} \quad \text{for } j = 1, 2, \dots, J.$$

Additionally, in order to use these distributions as vectors in a neural network, the distributions are binned. These binned representations h_X^j, are obtained by sampling the values v from p_X^j at M equally spaced bins within the range of possible values, $v_b = \frac{b}{M-1}$ for $b = 0, 1, \dots, M-1$. This results in $h_j^X \in \mathbb{R}^M$ for $j = 1, 2, \dots, J$, and the binning formula can be expressed as:

$$h_X^j = \left[p_X^j(v = v_b) \;\middle|\; v_b = \frac{b}{M-1},\ b = 0, 1, \dots, M-1 \right] \quad \text{for } j = 1, 2, \dots, J.$$

Distribution pooling is flexible and can capture rich information about the distribution of features within a bag. It avoids the loss of information associated with point estimate-based pooling methods.

- **Distribution with attention pooling** (Oner et al., 2020): Distribution with attention pooling combines Attention pooling with Distribution pooling. Feature distributions are calculated in the same way for each feature as with distribution pooling, except an attention weight a_i is incorporated:

$$p_X^j(v) = \frac{1}{I} \sum_{i=1}^{I} a_i \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2\sigma^2}(v - h_i^j)^2} \quad \text{for } j = 1, 2, \dots, J.$$

The instance-level weight a_i is calculated in the same way as in Attention pooling and, similarly to distribution pooling, binning is performed to transform these distributions into vectors.

- **TransMIL** (Shao et al., 2021): TransMIL is a MIL pooling method that is based on the Transformer architecture (Vaswani et al., 2017). The TransMIL pooling function can be expressed as follows:

  $$\mathbf{h}_X = \text{TPT}(\mathbf{h}_1, \dots, \mathbf{h}_I),$$

  where the TPT is a module that consists of two Transformer layers and a position encoding layer. Further details about the architecture can be found in the paper from Shao et al. (2021).

3.2. Uncertainty estimation

Uncertainty estimation provides a measure of the uncertainty related to the model's prediction. In this work, we define an uncertainty estimation technique as a function $f_{uncertainty}$ that maps input data to a single scalar representing an uncertainty estimation value. This is typically executed on individual samples in a dataset, but can also be applied to instances in MIL. Given an instance $\mathbf{x}_i$, the uncertainty can be estimated as $U_i = f_{uncertainty}(\mathbf{x}_i)$.

Several uncertainty estimation techniques also need the selection of an uncertainty measure ($f_{measure}$) to aggregate multiple probability vectors. In the last part of this section, we also define several of these uncertainty measure functions. Some popular uncertainty estimation techniques are:

- **Deep Ensemble** (Lakshminarayanan et al., 2017): This method utilizes multiple deep networks trained with various random model weight initializations. To estimate uncertainty, each model's output is considered:

  $$f_{uncertainty}(\mathbf{x}_i) = f_{measure}(f_{\theta_{instance,1}}(\mathbf{x}_i), \dots, f_{\theta_{instance,M}}(\mathbf{x}_i)),$$

  where each $f_{\theta_{instance,j}}$ is the j-th instance-level classification model in the ensemble and M is the total number of models in the ensemble.
- **Monte-Carlo dropout (MC-DO)** (Gal and Ghahramani, 2016): This method estimates uncertainty by employing dropout at test time and performing multiple forward passes:

  $$f_{uncertainty}(\mathbf{x}_i) = f_{measure}(f_{\theta_{instance,MCDO}}(\mathbf{x}_i; t_1), \dots, f_{\theta_{instance,MCDO}}(\mathbf{x}_i; t_T)),$$

  where $f_{\theta_{instance,MCDO}}$ is an instance-level classification model with dropout enabled during test-time. The term t_k represents the k-th stochastic forward pass out of a total of T passes.
- **Test-time augmentation (TTA)** (Ayhan et al., 2020): This method augments the input sample at test time and observes the variation in the classifier's output. The uncertainty is calculated as:

  $$f_{uncertainty}(\mathbf{x}_i) = f_{measure}(f_{\theta_{instance}}(Aug_1(\mathbf{x}_i)), \dots, f_{\theta_{instance}}(Aug_T(\mathbf{x}_i))),$$

  where $f_{\theta_{instance}}$ is an instance-level classification model and $Aug_k(\mathbf{x}_i)$ represents the application of the kth augmentation to the instance $\mathbf{x}_i$ out of a total of T augmentations.
- **MaxLogit** (Hendrycks et al., 2019): This technique considers the maximum logit value (before softmax) as a confidence measure. To obtain an uncertainty value, rather than a confidence value, we take the negative of the maximum logit value. The uncertainty is directly given by:

  $$U_i = -\max(\mathbf{l}_i),$$

  where $\mathbf{l}_i$ is the logit vector for the instance $\mathbf{x}_i$. Please note this is the only uncertainty estimation technique described in this work that does not utilize $f_{measure}$.

The list below describes a number of uncertainty measures. Each measure utilizes a function $f_{measure}$ that operates on a series of model outputs for an instance, producing an uncertainty value. Given a series of model outputs $p_{i,1}, p_{i,2}, \dots, p_{i,T}$ for instance $\mathbf{x}_i$, where T can for example represent multiple models in an ensemble, stochastic forward passes in MC-DO, or multiple augmentations in TTA, the uncertainty measure function $f_{measure}$ produces:

$$U_i = f_{measure}(p_{i,1}, p_{i,2}, \dots, p_{i,T}).$$

- **Maximum class probability:**

  $$f_{measure}(p_{i,1}, p_{i,2}, \dots, p_{i,T}) = -\max_{c=1}^{C} \frac{1}{T} \sum_{t=1}^{T} p_{i,t,c},$$

  where U_i is the uncertainty associated with instance $\mathbf{x}_i$, C is the number of classes. $p_{i,t,c}$ is the probability assigned by the t-th model, stochastic forward pass or augmentation for instance $\mathbf{x}_i$ and class c. Since we are interested in obtaining uncertainty values, rather than confidence values, we employ a minus-sign at the start of the equation.
- **Mean class variance:**

  $$f_{measure}(p_{i,1}, p_{i,2}, \dots, p_{i,T}) = \frac{1}{C} \sum_{c=1}^{C} \frac{1}{T} \sum_{t=1}^{T} (p_{i,t,c} - \mu_{i,c})^2,$$

  where $\mu_{i,c} = \frac{1}{T} \sum_{t=1}^{T} p_{i,t,c}$ is the mean probability of the c-th class for instance $\mathbf{x}_i$.
- **Entropy:**

  $$f_{measure}(p_{i,1}, p_{i,2}, \dots, p_{i,T}) = -\sum_{c=1}^{C} \mu_{i,c} \log \mu_{i,c},$$

  where $\mu_{i,c} = \frac{1}{T} \sum_{t=1}^{T} p_{i,t,c}$ is as previously defined.

For ordinal classification tasks, such as those with a staging scale, it is critical to note that conventional uncertainty measures might not accurately capture the differences between stages. Specifically, they might treat uncertainties between neighboring stages the same as those between distant stages. Consider a scenario where probabilities for classes 1 and 2 are both 50%, rendering other classes at 0%. The uncertainty should be lower than a scenario where class 1 and class 5 have probabilities of 50%. To address this, we propose ordinal variants of mean class variance and entropy, known as ordinal variance and ordinal entropy. These measures specifically consider the ordinal relationship between classes, ensuring that larger uncertainties between distant classes are weighted more heavily than those between closer classes:

- **Ordinal variance:**

  $$f_{measure}(p_{i,1}, p_{i,2}, \dots, p_{i,T}) = \frac{1}{T} \sum_{t=1}^{T} (q_{i,t} - \mu_i)^2,$$

  where $q_{i,t} = \sum_{c=1}^{C} (c-1) \cdot p_{i,t,c}$ represents the weighted sum of class probabilities by their ordinal rank for instance $\mathbf{x}_i$. This essentially converts the class probabilities to a single scalar value, similar to a value that a regular regression model would output. The term μ_i denotes the average of these weighted sums across all models or stochastic passes, calculated as $\mu_i = \frac{1}{T} \sum_{t=1}^{T} q_{i,t}$.
- **Ordinal entropy:**

  $$f_{measure}(p_{i,1}, p_{i,2}, \dots, p_{i,T}) = -\sum_{c=1}^{C-1} \left(\underbrace{\sum_{d=1}^{c} \mu_{i,d} \log \sum_{d=1}^{c} \mu_{i,d}}_{\text{Entropy of classes up to } c} + \underbrace{\sum_{d=c+1}^{C} \mu_{i,d} \log \sum_{d=c+1}^{C} \mu_{i,d}}_{\text{Entropy of classes beyond } c} \right),$$

where $\mu_{i,c} = \frac{1}{T}\sum_{t=1}^{T} p_{i,t,c}$ represents the mean probability of instance $\mathbf{x}_i$ being of class c across all models or stochastic forward passes. This measure calculates multiple entropies based on binary partitions of the classes. For every class c except the last class, C, it breaks down the classes into two groups: one with classes up to c and the other with classes greater than c. By doing this, it constructs binary uncertainties for each partition and computes their entropies. These entropies are then summed up, producing a measure that captures ordinal relationships between classes.

3.3. Uncertainty-based Instance eXclusion (UBIX)

UBIX is neither a standalone MIL method, like Max pooling or Distribution pooling, nor an alternative uncertainty estimation approach, such as deep ensembles or MaxLogit. Instead, it synergistically leverages existing MIL methods and uncertainty estimation techniques. Moreover, UBIX can seamlessly integrate with a broad spectrum of MIL methods and uncertainty estimation techniques.

In UBIX, there is a deliberate inference-time manipulation of instance-level features based on the associated uncertainties. As opposed to assuming equal contribution, UBIX identifies instances that might be corrupted due to local artifacts, and reduces or ignores the contributions of these instances on-the-fly using these associated instance-level uncertainties. We can categorize two distinct UBIX variants (see Fig. 1), which both perform this manipulation in a different manner. Specifically, $\text{UBIX}_{\text{soft}}$ reduces the contributions of uncertain instances and $\text{UBIX}_{\text{hard}}$ fully removes their contributions.

- **$\text{UBIX}_{\text{soft}}$**: For clarity, we first present UBIX for a straightforward scenario, without the intricacies introduced by techniques such as deep ensembles or TTA. $\text{UBIX}_{\text{soft}}$ is specifically designed for scenarios where $f_{\theta_{pooling}}$ employs Max pooling in conjunction with the instance-based MIL approach. We formulate this restriction because this UBIX variant actively reduces instance-level features as uncertainty increases, a behavior that is appropriate solely when Max pooling is applied.
  f_{UBIX} can be conceptualized as a modified sigmoid function. During inference, it maps the instance-level features, $\mathbf{h}_i$, based on the associated uncertainty, U_i, to updated instance-level features. These updated instance-level features are then used in the downstream MIL pipeline. The intention is for $f_{UBIX}(\mathbf{h}_i, U_i)$ to remain close to $\mathbf{h}_i$ when uncertainty U_i is minimal. Therefore, the sigmoid function approaches $\mathbf{h}_i$ as uncertainty decreases, *i.e.,* the upper asymptote is set to $\mathbf{h}_i$. Conversely, when U_i is high, we desire the logits to be diminished. Therefore, the sigmoid function converges towards the lowest feature value observed in the validation set as the uncertainty increases.
  More formally, the instance-level features are transformed by the UBIX function f_{UBIX} as follows:

$$f_{UBIX}(\mathbf{h}_i, U_i) = \frac{\mathbf{h}_i - \mathbf{h}_{min}}{1 + e^{\delta(U_i - \hat{\gamma})}} + \mathbf{h}_{min},$$

  where δ is a hyperparameter dictating the smoothness of f_{UBIX}. The term $\mathbf{h}_{min}$ is the vector of length J (which is equal to C in the instance-based approach), containing the minimum instance-level outputs within the validation set for each class, defined as:

$$\mathbf{h}_{min} = [\min_{\mathbf{x}_i \in \mathcal{D}_{val}} \{h_i^1\}, \ldots, \min_{\mathbf{x}_i \in \mathcal{D}_{val}} \{h_i^J\}],$$

  with $\mathcal{D}_{val}$ representing the validation dataset. The steepest gradient of f_{UBIX} with respect to $\mathbf{h}_i$ occurs when U_i equals $\hat{\gamma}$, which is defined as:

$$\hat{\gamma}(\gamma) = \gamma(U_{max} - U_{min}) + U_{min},$$

  with γ being a tunable hyperparameter. The extremities of uncertainties are described by:

$$U_{min} = \min_{\mathbf{x}_i \in \mathcal{D}_{val}} \{U_i\},$$

  and:

$$U_{max} = \max_{\mathbf{x}_i \in \mathcal{D}_{val}} \{U_i\}.$$

  In conclusion, the sole change that UBIX introduces is during the inference phase where the MIL pooling function $f_{\theta_{pooling}}$ does not directly utilize $\mathbf{h}_i$. Instead, $f_{\theta_{pooling}}$ takes the transformed instance-level features $f_{UBIX}(\mathbf{h}_i, U_i)$ as input.
  When multiple forward passes are employed during inference due to the use of techniques like deep ensembles, TTA, or MC-DO, the UBIX function f_{UBIX} operates in the same way. There will be multiple values for $\mathbf{h}_i$, one for each forward pass. $f_{\theta_{transform}}(f_{\theta_{pooling}}(f_{UBIX}(\mathbf{h}_i)))$ is performed separately for these different values for $\mathbf{h}_i$. Only the bag-level probabilities are subsequently aggregated.
- **$\text{UBIX}_{\text{hard}}$**: This variant of UBIX operates by fully excluding certain instances based on their uncertainty values. Specifically, the UBIX function is defined differently in this approach:

$$f_{UBIX}(\mathbf{h}_i, U_i) = \begin{cases} \mathbf{h}_i & \text{if } U_i \leq \tau \\ \text{exclude} & \text{otherwise} \end{cases}$$

  where τ is a hyperparameter that determines the uncertainty threshold for instance exclusion. In this approach, any instance with an uncertainty value exceeding τ is completely excluded from further processing.
  When implementing $\text{UBIX}_{\text{hard}}$, the MIL pooling function, $f_{\theta_{pooling}}$, is fed with a subset of instance-level features after exclusion.

Table 1
CIRCL grading system.

AMD Stage	Criteria
1. No AMD	No drusen or small, hard drusen only.
2. Early AMD	>10 small (<63 μm), drusen and pigmentary changes or 1–15 intermediate (63–124 μm) drusen.
3. Intermediate AMD	>15 intermediate (63–124 μm) drusen or any large (≥ 125 μm) drusen or GA not in the central circle of the ETDRS grid.
4. Advanced AMD: GA	Presence of central GA.
5. Advanced AMD: CNV	Evidence of active or previous CNV lesion.
6. CNV without signs forAMD	Chosen if CNV is present but no drusen of any size are present within the Field 2.
7. Cannot grade	Image is regarded as not gradable.

4. Data

4.1. Data

Three different data sets from three different vendors were used to develop and evaluate the proposed solution: a dataset with Heidelberg OCTs served as a training set, referred to as H_{train}, a validation set, referred to as H_{val}, and internal test set, referred to as H_{test} (Section 4.1.1); and two external test sets were used to evaluate the generalizability of our models, one with Topcon scans, referred to as T_{test}, (Section 4.1.2) and one with Bioptigen scans, referred to as B_{test} (Section 4.1.3).

Table 2
AMD stage distribution in each dataset. The table shows the number of OCT scans.

	H_{train}	H_{val}	H_{test}	T_{test}	B_{test}
No AMD	597	216	212	942	115
Early AMD	202	74	70		
Intermediate AMD	329	106	114	149	269
Advanced AMD: GA	72	29	28	37	
Advanced AMD: CNV	737	236	256	56	

4.1.1. Heidelberg dataset (H_{train}, H_{val} and H_{test})

For development and internal testing, we used the European Genetic Database (EUGENDA), a large multi-center database for clinical and molecular analysis of AMD (van de Ven et al., 2012; Fauser et al., 2011), containing 3,278 OCT from 1,013 patients in total. The training set H_{train}, validation set H_{val}, and test set H_{test} contained 1937, 661 and 680 OCTs from 607 (60%), 202 (20%) and 204 (20%) patients, respectively.

Manual grading of the scans was performed by the Cologne Image Reading Center and Laboratory (CIRCL). They categorized the OCTs using the criteria described in Table 1. Samples with grade 6 and 7 were excluded from this study. The number of OCTs for each of the remaining five stages is shown in Table 2. The OCTs were acquired with a Spectralis HRA+OCT (Heidelberg Engineering, Heidelberg, Germany) scanner. We resampled all B-scans to the same pixel spacing of 13.9 μm × 3.9 μm. The number of B-scans in each OCT scan was left unchanged, which varied from 14 to 73.

4.1.2. Topcon dataset (T_{test})

One of the external test sets was derived from the Rotterdam Study (Ikram et al., 2017). This is a prospective cohort study in the city of Rotterdam, the Netherlands, that started in 1990 to investigate age-related diseases. There were in total 1184 OCT scans available from this dataset, originating from 713 patients. All OCTs were graded using the Wisconsin Age-related maculopathy grading system (WARMGS) (Klein et al., 1991) and manually harmonized to the CIRCL grading system. The number of OCTs for the resulting four classes is shown in Table 2. The OCTs from this dataset were taken with an OCT scanner from Topcon Corp., Tokyo, Japan. Each OCT volume contained 128 B-scans. Similarly to the Heidelberg set, all B-scans were resampled to have a pixel spacing of 13.9 μm × 3.9 μm.

4.1.3. Bioptigen dataset (B_{test})

The other external test set was described by Farsiu et al. (2014), containing normal patients and patients with intermediate AMD. For each of these subjects one OCT volume, acquired with an SD-OCT scanner from Bioptigen, Inc (Research Triangle Parc, NC), was available. The AREDS2 system (Chew et al., 2012) was used for grading and was harmonized to CIRCL grading system. The number of OCTs for these two classes is given in Table 2. All OCT volumes contained 100 B-scans and, all B-scans were again resampled to have a pixel spacing of 13.9 μm × 3.9 μm.

5. Experimental design

5.1. Vendor generalizability and interpretability

To assess vendor generalizability of UBIX, we first calculated the performance on the internal test set, H_{test}, which is from the same distribution as the one used for training, H_{train}. Subsequently, we evaluated the performance when transferring to the two external datasets T_{test} and B_{test}. We evaluated the performance on these datasets for $UBIX_{soft}$ and $UBIX_{hard}$. Additionally, we compared the performance of the proposed model with three different approaches, namely a 3D CNN approach, a traditional MIL approach (without UBIX) and an ensemble of multiple MIL approaches. The 3D CNN was a ResNet-18 (He et al., 2016) with 3D convolutions and the instance-level classifiers in the MIL approaches were ResNet-18's with 2D convolutions. All models in our experiments were trained end-to-end.

To better show the effect of the proposed methodologies on scans with vendor-specific artifacts, we also separately evaluated the performance of the five aforementioned UBIX variants and baselines on a subset of OCT volumes in T_{test} with blinking artifacts, referred to as T_{blink} (n = 33). These volumes generally have multiple B-scans in which the retina is not visible.

The interpretability of UBIX is illustrated qualitatively by showing the instance-level predictions and uncertainties for several OCT scans.

5.2. Effect of artificial artifacts

To demonstrate the effect of UBIX more clearly, we performed experiments where we artificially corrupted the dataset T_{test} with artifacts that also occur naturally in OCT scans. The different artifact types were blinking artifacts, vertically flipped B-scans, shadows and noise. Fig. 3 shows a number of examples. We gradually introduced more OCT volumes with artificial artifacts and compared the performance for $UBIX_{soft}$, $UBIX_{hard}$ and MIL.

When one of these artifacts was applied to an OCT volume, a portion of the B-scans were affected, as happens in clinical scenarios. Artificial artifacts were then added to either one or two groups of adjacent B-scans. Both scenarios had an equal probability. The sizes of these groups had sizes of between 2% and 15% of B-scans, which we experimentally found to be representative of real artifacts.

Vertically flipped B-scans are caused by a Fourier-domain detection artifact, as described by Ho et al. (2010). Shadows and noise are usually caused by media opacities, such as corneal scarring and cataract. The artificial artifacts were implemented as follows:

- To generate B-scans with artificial blinking artifacts, we started with an image where all pixel values were set to 0. Next, we added random Gaussian noise to this image. The mean of this noise was set to the median pixel value found in the full OCT scan. The standard deviation of this noise was matched to the standard deviation of the OCT scan.
- The vertically flipped B-scans were generated by flipping the B-scans along the horizontal axis.
- To generate the shadow artifact for a particular B-scan, we adapted each A-scan (column in an OCT B-scan) $a_1, \ldots, a_A$ separately, where A is the number of A-scans in the B-scan. All A-scans a_i were transformed using the shadow function $S(a_i)$:

$$S(a_i) = a_i(1 - s(i)), \tag{1}$$

where $s(i)$ is sampled from a normal probability density function:

$$s(i) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}(\frac{i-\mu}{\sigma})^2}, \tag{2}$$

where μ is randomly selected between 0 and A, and σ is randomly defined between $A/4$ and $3A/4$. μ and σ are kept the same within one OCT volume.
- The noise artifact is Gaussian noise added to the original image with a mean of 0 and a standard deviation of 4 times the standard deviation within the original OCT volume.

5.3. Applicability to several MIL settings and uncertainty estimation approaches

We performed several experiments to assess the applicability of UBIX to different uncertainty estimation approaches and MIL settings. We evaluated this on the external test sets T_{test}, T_{blink}, and B_{test}. The uncertainty measures and techniques that we evaluated were the ones presented in Section 3.2. The MIL pooling functions we used in these experiments were described in Section 3.1.2.

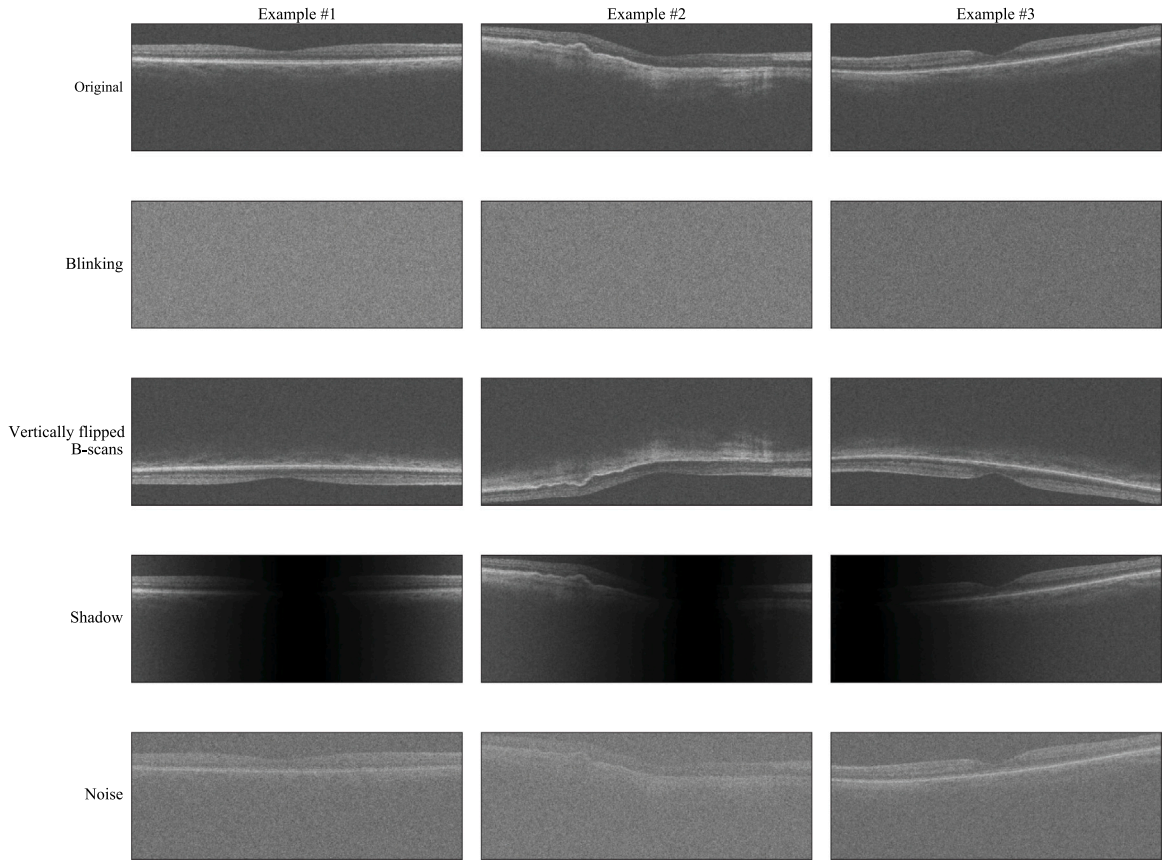


Fig. 3. Examples of artificial artifacts. Each row shows the middle B-scans of a random OCT volume from T_{test}. The image in the first row is the original, unaltered B-scan. The other rows each depict a different artificial artifact applied to that B-scan.

Some of the MIL pooling functions are only compatible with the embedding-based approach. Therefore, in the experiment where we compared these MIL pooling functions, all uncertainties were calculated using MaxLogit, considering the other uncertainty measures cannot be combined trivially with the embedding-based approach. Max and Mean pooling were implemented using the instance-based MIL approach, while the Attention, Distribution, Distribution with attention, and TransMIL pooling were implemented using the embedding-based approach. For Attention pooling, we followed the implementation provided by Ilse et al. (2018). For Distribution and Distribution with Attention pooling, we used the implementation provided by Oner et al. (2023) with 128 features in the attention layers and 11 bins in the distribution pooling layer. For TransMIL pooling, we used the implementation provided by Shao et al. (2021). The deep ensembles contained 5 models, as this was shown to be sufficient for uncertainty estimation (Ovadia et al., 2019). For the MC-DO models, we used $T = 32$ stochastic forward passes, which is consistent with what was done by Linmans et al. (2023). For TTA we used $T = 10$ different forward passes. For the experiments with MaxLogit, we used the maximum value in the output of the final layer of the instance-level classifier, while only considering this maximum value for the first model in case a deep ensemble was used. The UBIX hyperparameters were separately optimized on the validation set H_{val} for each unique combination of MIL pooling function and uncertainty estimation approach.

5.4. Metrics

For all models, to evaluate the classification performance, we calculated the area under the receiver operating characteristic curve (AUC), where intermediate and advanced AMD stages belonged to the positive class and the remaining stages to the negative class. Additionally, we computed Cohen's kappa score. For the datasets with more than two classes, the quadratic weighted kappa score (κ_w) was calculated to consider the class order. Otherwise, unweighted kappa metric was used (κ).

To quantify artificial artifact detection performance for different uncertainty measures, we used the AUC as well, where the score was the uncertainty measure and the labels were the dichotomous variable of whether an instance had an artificial artifact or not. Furthermore, to estimate how well the uncertainty values were separated, we evaluated the separability of the two groups, with and without artificial artifacts, based on the uncertainty score. For this, the Xie-Beni index (XB) is calculated, defined as the ratio between cluster separation (*i.e.*, the minimum squared distance between cluster centers) and cluster compactness (*i.e.*, the mean squared distance between each data point and its cluster center (Xie and Beni, 1991). The lower XB, the better the data is clustered. Statistical significance was determined using non-parametric bootstrapping with 1000 iterations (Rutter, 2000). We applied a Bonferroni correction to account for the number of comparisons we made.

5.5. Training and optimization

The network weights were optimized with the Adam optimizer (Kingma and Ba, 2014) and a learning rate of 10^{-4} using the cross entropy loss. All images were normalized between 0 and 1. As a means of regularization, we employed online data augmentation. With a 15% probability, random affine transformations were applied of ±20° rotation within the B-scan plane, ±10% shearing within the B-scan plane, ±10% zooming within the B-scan plane, ±20 voxels translation in the horizontal and vertical direction within the B-scan plane, ±2 voxels translation in the B-scan direction. B-scans were also horizontally flipped with a 15% probability. With a 30% probability, random

additive Gaussian noise with a mean of 0 and a standard deviation of 0.1 was applied. Also with a 30% probability, we applied brightness modifications using the power law, varying the power between 0.75 and 3. To account for class imbalance, images were sampled during training based on their class, such that each class was sampled with an equal probability. Because of GPU memory restrictions and given that the input images were large, each batch contained one bag during training and inference. We used early stopping, based on κ_w and with a patience of 10 000 batches.

The hyperparameters τ, δ and γ were optimized on the internal validation set H_{val}. The values for τ in the grid search for $UBIX_{hard}$ were $\{P_{U,80}, P_{U,80.1}, \dots, P_{U,100}\}$, where $P_{U,i}$ is the percentile i of all uncertainties in the validation set. For the UBIX grid search, the values for δ were $\{1, 5, 10, \dots, 5 \cdot 10^3, 10^4\}$ and the values for γ were $\{-0.5, -0.45, \dots, 1.5\}$. For $UBIX_{soft}$ implemented using deep ensembles, ordinal entropy and Max pooling, the performance was optimal at $\delta = 5$ and $\gamma = 1.05$. For $UBIX_{hard}$ implemented using deep ensembles, ordinal entropy and Max pooling, the optimal value of the hyperparameter τ was $P_{U,93.3} = 2.20$.

5.6. Normalization type

All MIL and UBIX models (which do not differ from MIL models during training) described in this section so far used batch normalization (BatchNorm) (Ioffe and Szegedy, 2015) followed by a rectified linear unit (ReLU) activation layer (Nair and Hinton, 2010) after every convolutional layer. This BatchNorm layer was implemented such that the mean and variance were computed along the instance axis (rather than the bag axis), spatial, and channel axes. This had the effect that the instance-level classifier in these models used some information from other instances than the instance it was classifying, specifically when calculating the mean and variance values that are used in this layer. We studied the effect of replacing these BatchNorm layers with InstanceNorm (Ulyanov et al., 2016) layers on the generalization of both the MIL and UBIX models. For InstanceNorm, the mean and variance values were only calculated along the spatial and channel axes, so not along the bag or instance axes. Following the default implementation of PyTorch, the BatchNorm models included learnable shift and scale parameters, while the InstanceNorm models did not. The 3D model used in this paper also had a batch size of 1, so the normalization layer was effectively InstanceNorm with learnable shift and scale parameters. In all cases, the means and variances of the current sample, rather than the moving averages, were used during both training and inference.

6. Results

6.1. Baseline comparison

In this section, we compare the performance of the two UBIX variants with three baseline models on the internal dataset. Additionally, we explore how these models differ in performance when applied to external datasets. Fig. 4 provides an overview of these comparisons.

On the internal dataset, we observe that most differences between models are relatively small compared to the external datasets. When evaluating on external datasets, there is a notable drop in performance for all methods. However, this drop is consistently smaller for the UBIX models when considering κ_w as the performance metric.

UBIX exhibits superior performance in scenarios characterized by the presence of artifacts. For instance, on the T_{blink} dataset, UBIX achieves a κ_w of 0.708, surpassing the scores of 0.479 for only MIL, 0.346 for MIL (no ensemble), and 0.084 for the 3D model. To illustrate the interpretability of the model at the B-scan level, we provide visual examples of $UBIX_{soft}$ predictions in Fig. 5.

In Fig. 6, we investigate the impact of introducing artificial artifacts when using UBIX, compared to MIL. We evaluate performance while varying the percentage of OCT volumes affected by artificial artifacts within the T_{test} dataset.

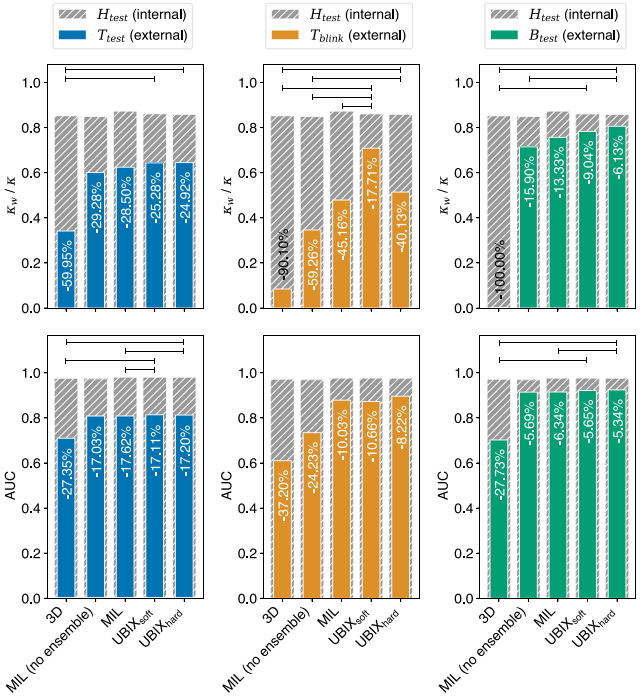


Fig. 4. Performance of UBIX, compared to the 3D and MIL baselines, evaluated on internal and external datasets. The models were all trained and validated on Heidelberg data (H_{train} and H_{val}). The UBIX methods in this figure use deep ensembles with ordinal entropy to estimate uncertainty. The percentages displayed within the bars indicate the difference in performance when transferring the model from the internal to the external test set. Max pooling was used for each of the reported results. The horizontal black bars above the bar plots indicate whether there was a statistically significant difference ($p < 0.05$) between the performance on the external datasets of either of the UBIX variants and the other three models.

6.2. Applicability to several MIL settings and uncertainty estimation approaches

We explored the applicability of $UBIX_{soft}$ in different MIL settings and uncertainty estimation approaches. Figs. 7 and 8 show the generalizability of $UBIX_{soft}$ when using different uncertainty measures. The latter figure show that for $T_{test}$and κ_w, ordinal entropy emerges as the top-performing measure. For B_{test}, the two ordinal variants also show the highest performance.

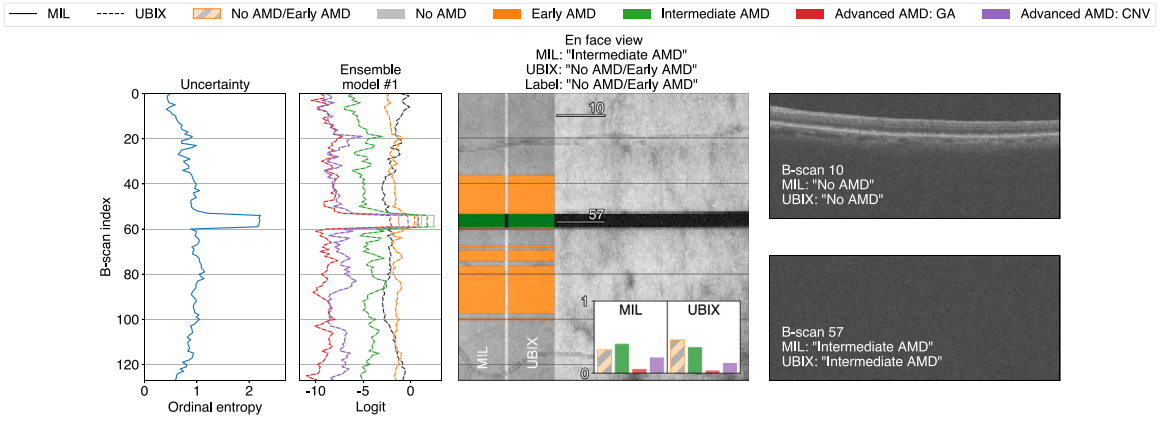
Several uncertainty estimation methods were compared in Fig. 9. For T_{blink}, the biggest impact of UBIX could be observed when using deep ensembles as the uncertainty estimation method. Fig. 10 shows the generalizability of several MIL pooling functions and the effect of UBIX in combination with these different MIL pooling functions.

6.3. Normalization type

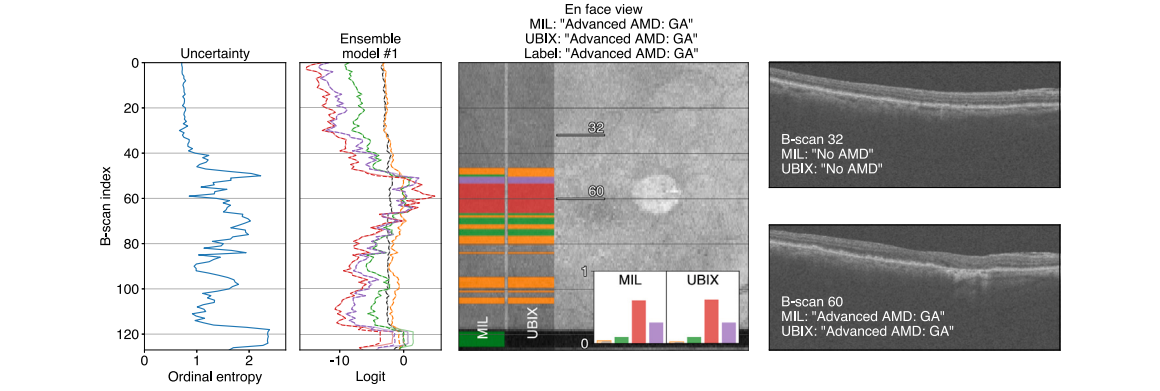
The effect of using InstanceNorm instead of BatchNorm on generalizability for MIL and UBIX is shown in Fig. 11. Since the drop in performance is already substantially less when transferring to the external datasets for the MIL models with InstanceNorm than with BatchNorm, the effect of UBIX is less visible in this figure. However, when considering images with different types of artifacts than blinking artifacts, we found that UBIX was able to recover better than MIL, as illustrated in Fig. 12.

7. Discussion

We proposed a method using MIL with OOD detection to improve the generalizability of deep learning models for classification



(a) UBIX correctly predicts "No AMD/Early AMD", while MIL incorrectly predicts "Intermediate AMD". UBIX suppresses the instance-level outputs at the location of the artifact around B-scan 57, causing it to be robust to that artifact, in contrast to MIL.



(b) Correct GA volume-level classification. Central GA is also well visible in the en face image as a white circular object around B-scan 60, which is pointed out by the instance-level output in red.

Fig. 5. Examples where $\text{UBIX}_{\text{soft}}$ corrects volume-level and instance-level predictions. The figure also illustrates instance-level interpretability. Each subfigure shows, from left to right, the uncertainty per instance, the instance-level logits of the first model in the ensemble (only showing one model for clarity), the en face image (the volume averaged over the y-axis), and two B-scans of interest. The uncertainty, logit and en face plots correspond spatially in the horizontal direction. The left and right banner in the en face view indicate the instance-level outputs for MIL and UBIX, respectively. The B-scans on the right are highlighted in the en face view. In the bottom right of the en face views, volume-level probabilities are shown.

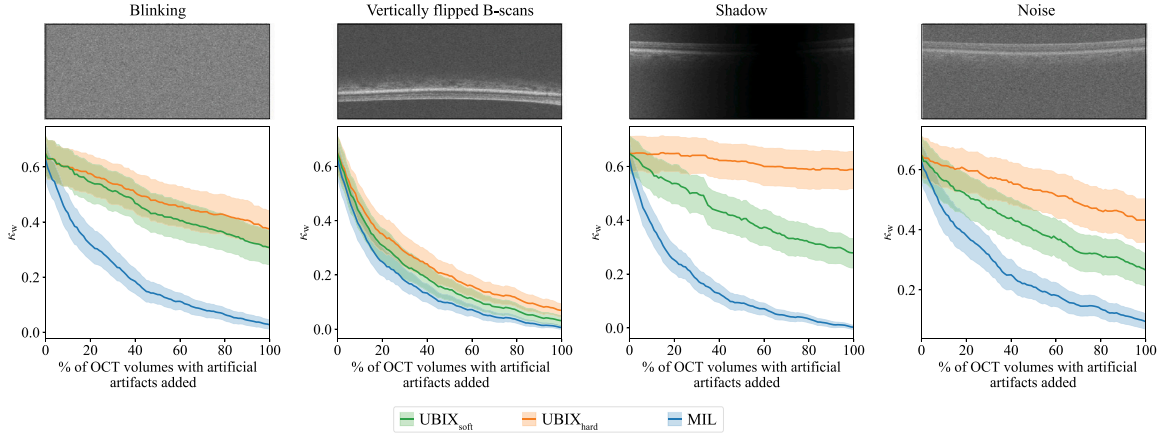


Fig. 6. Robustness to different artificial artifacts of the UBIX, compared to MIL, on T_{test}. The top image in each column shows an example B-scan of the artificial artifact. The plots show the relation between the performance and the percentage of OCT volumes in the dataset that contain these artifacts. The shaded areas indicate 95% confidence intervals, obtained using bootstrapping with 1000 iterations.

of 3D medical images. The model aims to reduce the effect of on-the-fly detected OOD instances in the final classification of the bag. By suppressing the contribution of OOD instances, UBIX maintains performance on unseen data distributions, particularly images coming from different scanners.

The robustness of the proposed approach was demonstrated by transferring UBIX models and baseline models to external datasets from different vendors. As shown in Fig. 4, UBIX variants are less prone to substantial performance drops than the other models. On all external datasets, either $\text{UBIX}_{\text{soft}}$ or $\text{UBIX}_{\text{hard}}$ showed better results than the

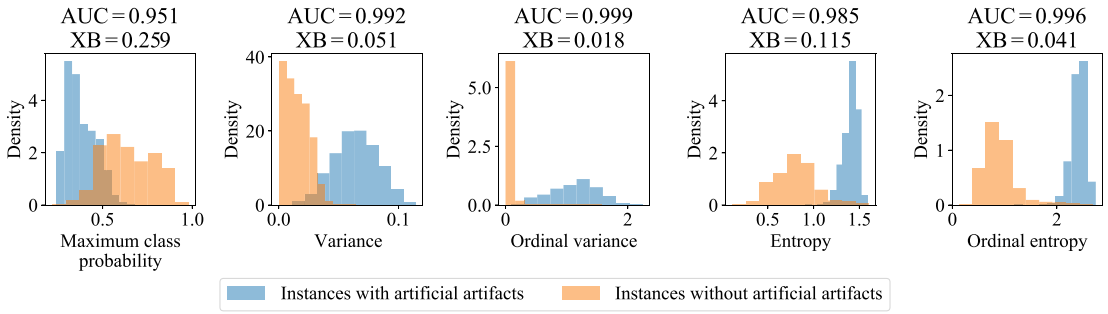


Fig. 7. Density plots of uncertainty estimates at instances with artificial blinking artifacts compared to uncertainty estimates at instances without artificial artifacts, applied to T_{test}. The AUC for artificial artifact detection performance and the clustering metric XB (the lower, the better) are shown above the density plots.

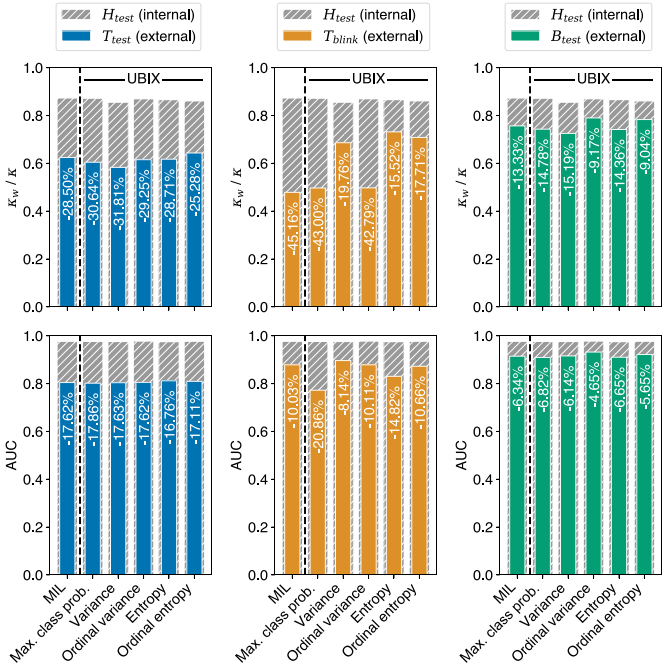


Fig. 8. Performance metrics for different uncertainty measures used for $UBIX_{soft}$, evaluated on the internal and external test sets. The MIL method performance (without UBIX) is indicated on the left of each plot. The models were all trained and validated on Heidelberg data (H_{train} and H_{val}). Deep ensembles were used in combination with the reported uncertainty measures. Max pooling was used for each of the reported results. Max. class prob. = Maximum class probability.

baseline models in terms of absolute performance. The performance drop was most notable on T_{blink}, where $UBIX_{soft}$ maintained a κ_w of 0.708, while the best and worst performing baseline models (MIL and 3D, respectively) had a κ_w of 0.479 and κ_w of 0.084, respectively. It was expected that these performance differences were more notable on T_{blink}, which only contained OCTs with blinking artifacts, because UBIX was designed to be robust to vendor-specific artifacts.

It was noted that, depending on the inference data, it was preferable to fully exclude the outputs of uncertain instances ($UBIX_{hard}$) or to only suppress them ($UBIX_{soft}$). From Fig. 6, we found that $UBIX_{hard}$ showed better robustness than $UBIX_{soft}$ for OCTs with artificial artifacts. A possible reason for this could be that some of the artificial artifacts highly corrupted the information, resulting in a notably strong incorrect signal and the requirement for full exclusion of uncertain instances. $UBIX_{hard}$ seemed to be especially robust to shadow artifacts, given that the performance barely decreased when introducing more OCT volumes with artificial artifacts. For some external datasets and metrics, however, $UBIX_{soft}$ achieved better results than $UBIX_{hard}$, *e.g.*, for κ_w on T_{blink} and AUC on T_{test}.

The performed data augmentation might also have an effect on generalizability. Since signal-to-noise ratios differ per scanner, noise augmentation probably aided our models to generalize. Although we did resample all images to have the same pixel spacing within B-scans, the original spacings that we had could have been slightly inaccurate. Therefore, zooming could potentially also have a positive effect on generalizability. The same type of data augmentation was applied in all experiments and measuring its effect was considered out of the scope for this paper.

Large variability in B-scan spacing between scanners can also cause features learned by 3D CNNs to be poorly generalizable. MIL, which processes B-scans individually and combines B-scan level outputs using a MIL pooling function to get a volume level output, improves the robustness to this variability in slice spacing with respect to 3D models. We observed that performance differences are minimal between a 3D CNN and MIL when evaluated on data from the same vendor used during training. When evaluating on data from a different vendor, a performance drop was observed for both methods, although this drop is much larger for the 3D method (60.0% drop in κ_w on T_{test}) than for MIL (29.3% in κ_w on T_{test}).

One of the advantages of using MIL as base of the UBIX model is that instance-level annotations are not required for training, while the model is able to produce a classification output at this level (B-scan level in our case) as well as calculating instance-level uncertainty. This introduces model explainability, increasing the transparency of our method, and allowing surveillance of its behavior.

The extent to which UBIX improves generalizability depends highly on the quality of its underlying OOD detection. Therefore, we compared three different commonly used uncertainty measures in combination with deep ensembles, and we proposed two ordinal variants. On T_{test}, entropy and its ordinal variant had the highest performance in terms of κ_w and AUC, respectively. The ordinal variants seemed to distinguish the B-scans with and without artificial artifacts the best. This can be seen in the density plots of Fig. 7, and this is also reflected in the AUC and XB values, which were best for the two ordinal variants. Hence, the ordinal variants led to higher performances on artificial artifacts, although in the external test sets, we find mixed observations (see Fig. 8). A possible reason for this could be that fewer evaluation data with real artifacts were available, resulting in a less accurate performance measurement than when using artificial artifacts.

Furthermore, uncertainty measures for which we found the most competitive performance in terms of κ_w did not always perform the best in terms of AUC. A possible explanation for this inconsistency could be that the properties these two metrics measure are quite different. κ_w measures the grading performance (*i.e.*, the capability of distinguishing – in case of T_{test} and T_{blink} – four different grades), while the AUC only measures binary classification performance (*i.e.*, determining whether the sample has a grade higher or lower than *Intermediate AMD*). Therefore, we believe that for models with varying results like these, it should be carefully considered which metrics are most relevant in which clinical settings. In screening scenarios, for example, this binary

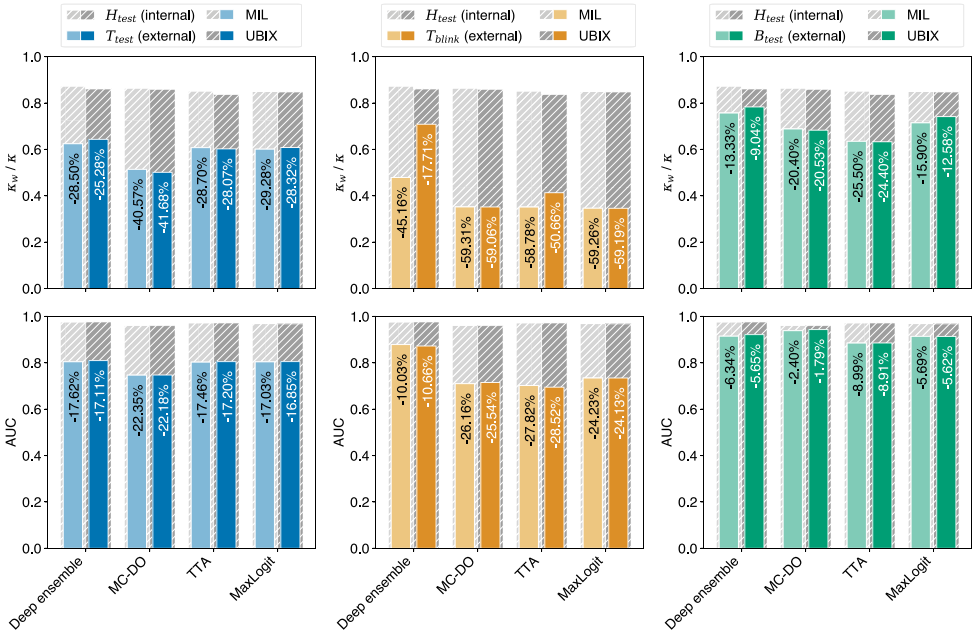


Fig. 9. Performance metrics for different uncertainty estimation methods used for $UBIX_{soft}$, compared to MIL (without UBIX), evaluated on the internal and external test sets. The models were all trained and validated on Heidelberg data (H_{train} and H_{val}). Max pooling was used for each of the reported results. For the MC-DO, TTA, and MaxLogit experiments, one model was used, instead of an ensemble.

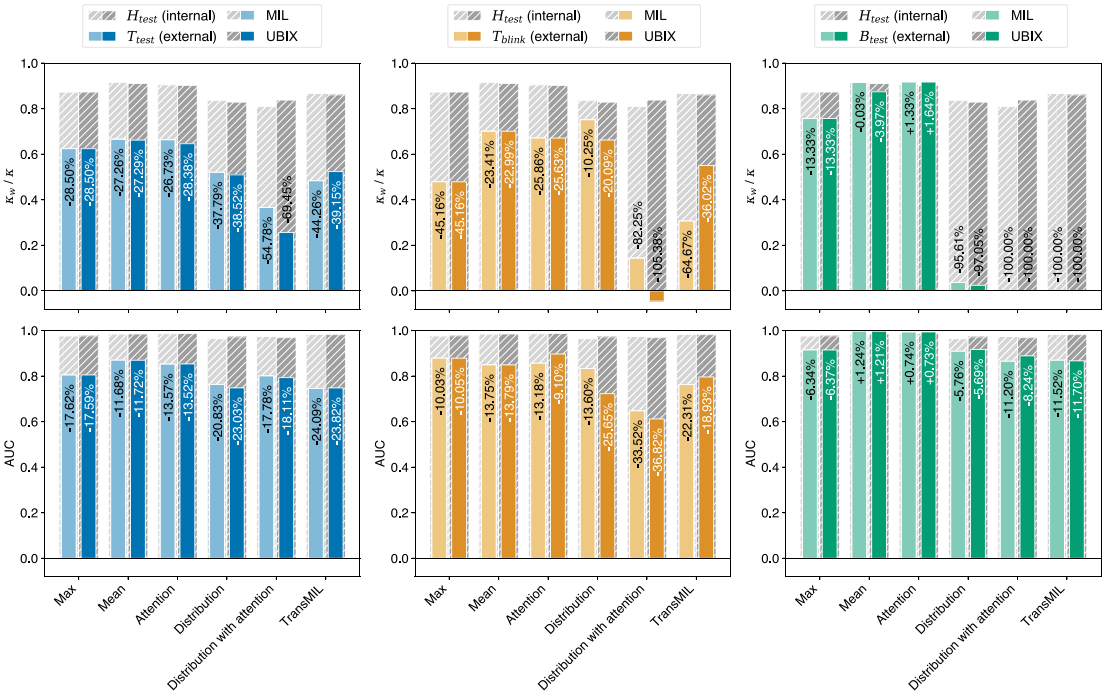


Fig. 10. Performance metrics for different MIL pooling functions when using $UBIX_{hard}$, compared to MIL (without UBIX), evaluated on the internal and external test sets. The models were all trained and validated on Heidelberg data (H_{train} and H_{val}). $UBIX_{hard}$ was used because $UBIX_{soft}$ was only designed for MIL models that utilize Max pooling. Uncertainties were calculated using MaxLogit in all reported UBIX models for this experiment. Ensembles of models were used for prediction in this experiment.

classification task (using the AUC as metric) may be more relevant. Conversely, in secondary care, a more granular grading (using κ_w as metric) may be desired.

In addition to the integration of various uncertainty measures with UBIX, we also examined its performance with uncertainty estimation techniques other than deep ensembles. As depicted in Fig. 9, deep ensembles generally demonstrated the least performance degradation when assessed on external datasets. This was particularly evident in the subset featuring only blinking artifacts, denoted as T_{blink}. Notably, the effectiveness of UBIX was most pronounced when paired with deep ensembles. This suggests that deep ensembles may yield more precise uncertainty estimations compared to the other three methods. This would allow UBIX to more effectively diminish the logits for instances in which this is needed (*i.e.,* where the model is prone to incorrect outputs due to OOD characteristics). This aligns with findings in the literature (Ovadia et al., 2019), where deep ensembles were identified as superior in detecting data shifts, outperforming other OOD detection techniques.

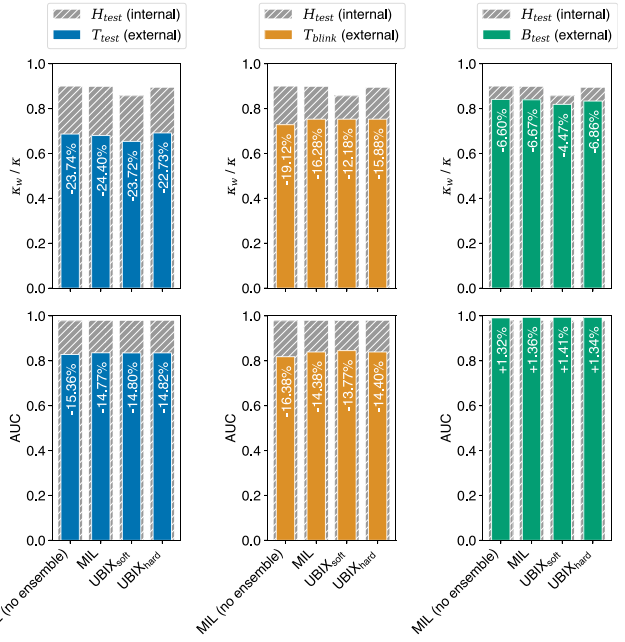


Fig. 11. Performance of models using InstanceNorm instead of BatchNorm. Results are shown for UBIX, compared to the MIL alternatives without UBIX, evaluated on internal and external datasets. The models were all trained and validated on Heidelberg data (H_{train} and H_{val}). The UBIX methods in this figure use deep ensembles with ordinal entropy to estimate uncertainty and Max pooling as a MIL pooling function. The percentages displayed within the bars indicate the difference in performance when transferring the model from the internal to the external test set.

The observed variation in internal performance among different MIL pooling functions was relatively small, particularly compared to their external performance (see Fig. 10). In our comparative study of these MIL pooling functions, incorporating UBIX showed minimal effect. This minimal effect is likely attributable to our choice of MaxLogit for uncertainty estimation, a method identified as having limited influence (Fig. 9). Since not all MIL pooling functions could be implemented with the instance-based approach, we were restricted to MaxLogit in this experiment. We leave it to future research to explore other uncertainty estimation techniques that may be more compatible with embedding-based MIL pooling functions. In conjunction with Attention pooling on the T_{blink} dataset, adding UBIX did show a small increase in performance. Both Mean and Attention pooling outperformed others on the internal test set and the external B_{test} set. The unexpected efficacy of Mean pooling, despite initial assumptions about its inadequacy in aggregating instance scores (Ilse et al., 2018), might be attributed to the nature of the reference standard, which relies on counting drusen across multiple instances in the bag. Mean pooling may be able to do this more effectively than Max pooling. The Distribution and Distribution with attention pooling functions showed no substantial inferiority compared to the other functions on the internal test set. However, on external test sets, they demonstrated underperformance in terms of κ_w. Interestingly, these effects were less pronounced for the AUC than for κ_w, suggesting a consistent shift in underlying probabilities.

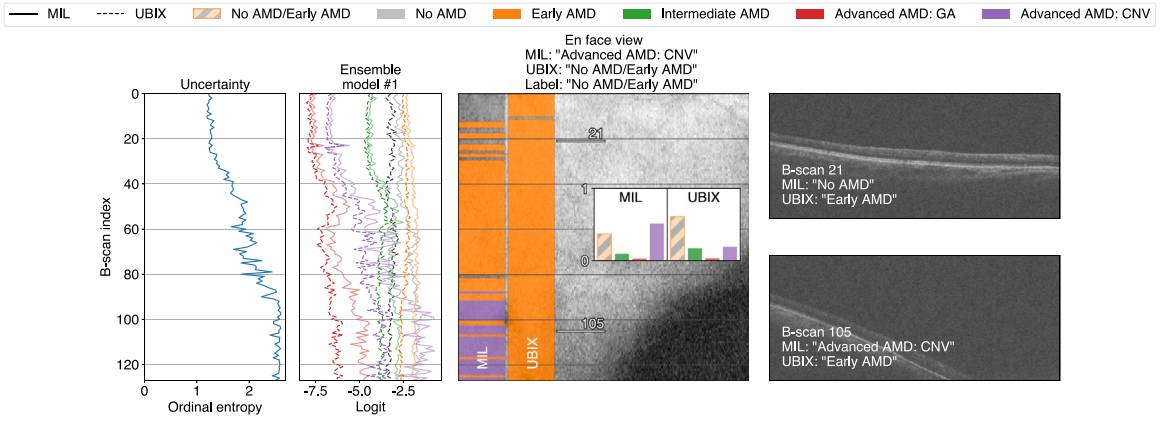
Given the current experimental setup, it is challenging to recommend a single general strategy for future practice regarding the best combination of uncertainty estimation technique and MIL pooling function for UBIX. In our experiments, we found that the optimal combination depended on multiple factors, such as whether these approaches were used with UBIX$_{soft}$ or UBIX$_{hard}$, the evaluation dataset, and the metric of interest. However, based on the experiments presented in Fig. 9 and Fig. 10, certain combinations frequently performed best across various datasets and metrics. Specifically, for UBIX$_{soft}$ with Max pooling, deep ensembles as the uncertainty estimation technique came out best most often. Furthermore, for UBIX$_{hard}$ with deep ensembles, Mean or Attention as the MIL pooling function frequently yielded the best results. Therefore, we expect these should generally be good options, even though more research is needed to provide more general recommendations.

As discussed in Section 5.6, the BatchNorm instance-level classifiers used in this work may use information from multiple instances, not just the ones currently being classified. We hypothesized that this characteristic might negatively impact model robustness. To test this, we also trained models using InstanceNorm, which does not share this characteristic. We found that replacing BatchNorm with InstanceNorm already improved the generalizability substantially (see Fig. 11). Because this generalizability without UBIX was already high, the effect of adding UBIX on the dataset-level metrics was negligible. However, even with the InstanceNorm model, we found rare individual imaging artifacts in T_{test}, to which the default model was not robust. An example of such artifacts is shown in Fig. 12(a). We observed that with the addition of UBIX, the model was able to recover. Moreover, as shown before in this section, we showed that UBIX, for certain model choices, did enhance robustness to OOD artifacts in terms of dataset-level metrics. This suggests that UBIX is beneficial in cases of low generalizability and neutral otherwise.

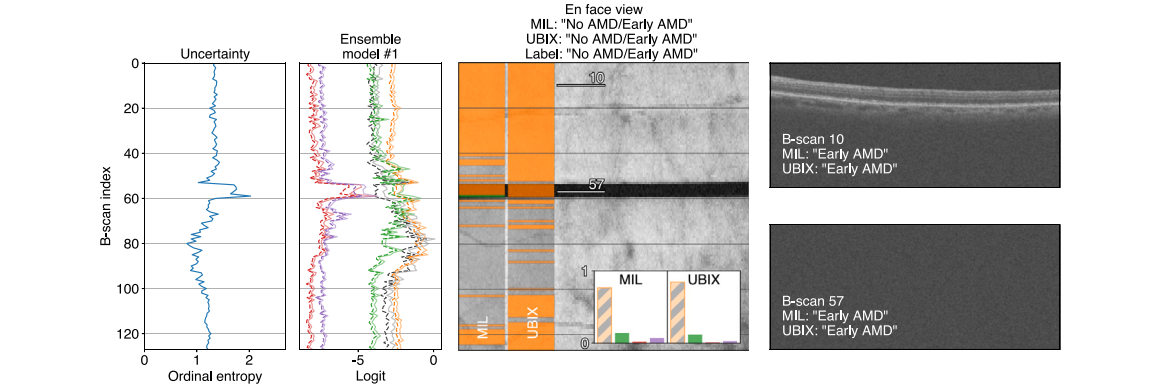
Since the three datasets were acquired and annotated at different sites with varying protocols, the reference standards were set differently among these datasets. To minimize the effect of this discrepancy, we merged the first two classes of the CIRCL grading when evaluating with the WARMGS system which was available for T_{test}. Moreover, when evaluating on B_{test} for which only the binary labels *No AMD* and *Intermediate AMD* were available, we also binarized the CIRCL systems, where the positive class started at *Early AMD*. This harmonization approach, however, was not perfect, causing the resulting class definitions to still not be completely equal. Despite this discrepancy, we think measuring performance differences between methods is still well possible. Nevertheless, the absolute performances can be underestimated because of these differences in reference standards.

As a potential undesired side effect, it should be noted that difficult cases, which are assumed to be more uncertain, could be excluded which are in fact necessary for making a correct prediction. It will depend on the setting of model deployment whether the benefits of robustness to OOD data outweigh this drawback. In screening, for example, a high specificity is generally considered more important than a high sensitivity. Instances that are falsely excluded by UBIX will often contain abnormalities, in which case especially sensitivity would suffer, but specificity will not be affected in that case. When the task is disease staging (as is the case in this work) and UBIX is implemented with an ordinal uncertainty measure, this potential undesired side-effect is unlikely to occur. Difficulty in such a case will usually lie in the uncertainty between two classes that are close together in the staging scale (such as *Early AMD* and *Intermediate AMD*), resulting in a low estimate of the ordinal uncertainty measure. As artifacts are not likely to cause uncertainties between two classes that are close on the staging scale, these artifacts will probably have a much higher ordinal uncertainty measure. To give an indication of which instances were assigned the highest uncertainties, we manually analyzed the B-scans with the highest ordinal entropies in T_{test} in Section A of the Appendix. There we found that more than half of the B-scans in the first percentile of most uncertain ones contained one of the nine different types of artifacts that were seemingly related to image acquisition.

If relevant structures are entirely excluded because they are difficult for the model to classify (for example because of an unseen lesion type or ambiguity), this is likely to be because there is something atypical with the whole scan or setting in which the model is used and the user should be alerted. So instead of only silently excluding instances, future work could analyze a method for combining UBIX with alerting the user if there are too many OOD instances detected. In future work, UBIX could also be adapted to work with patches as instances instead



(a) A case where B-scans corresponding to the lower part in the en face view suffer from an artifact that causes the retina to be displayed diagonally, falling out of view on the right. This artifact was likely out-of-distribution, resulting in the classification of these B-scans as "Advanced AMD: CNV", while they actually do not show any signs of neovascularization. The ordinal entropy was also high for these cases with artifacts, allowing UBIX to correct the bag-level prediction.



(b) The same case was displayed in Fig. 5a. As could be observed in that figure, the blinking artifact caused an incorrect classification of the MIL model with BatchNorm, which was corrected by UBIX. However, the MIL model with InstanceNorm, as shown in the current subfigure, already showed robustness to this artifact without UBIX.

Fig. 12. Examples of $\text{UBIX}_{\text{soft}}$ predictions when using InstanceNorm instead of BatchNorm. The subfigures follow the same structure as Fig. 5.

of slices. If an artifact is only locally present within a slice, the entire slice would not be excluded and potentially useful information would not be ignored.

We did not compare our method to any other domain adaptation methods which often require additional supervised or unsupervised training. Such a comparison would be unfair, as our method does not require any additional training. Nevertheless, our approach could potentially be further improved with the incorporation of domain adaptation methods such as those proposed by Seeböck et al. (2019) and Romo-Bucheli et al. (2020).

Future work could also investigate different OOD detection methods to be incorporated in UBIX, as UBIX is theoretically compatible with any OOD detection method. Performing a systematic analysis for OOD detection methods was beyond the scope of this paper, so we only applied deep ensembles in this study. Such a comparison might lead to valuable insights and performance improvements.

We only evaluated our approach for AMD grading in OCT. The method is expected to be applicable in more problem settings, such as the classification of other features and retinal diseases in OCT, but potentially also in other medical image analysis applications. Potential applications include histopathology, in which MIL is already common practice (Quellec et al., 2017; Ilse et al., 2018) and many types of artifacts can occur (Schömig-Markiefka et al., 2021; Kanwal et al., 2024; Foucart et al., 2018), and CT scans, which can contain metal artifacts (Wellenberg et al., 2018).

8. Conclusion

We showed that the generalizability of classification models to unseen scenarios can be improved by UBIX, an approach that seamlessly suppresses the contribution of OOD instances to the final classification during test-time based on the uncertainty associated with these instances, in the context of MIL for AMD classification in OCT. Our proposed approach alleviates the need for retraining on new data, which is an expensive process in terms of data acquisition, model development, and human annotation time. This increases reliability by improving the applicability of artificial intelligence models for OCT classification in broader scopes than the settings in which they were developed.

CRediT authorship contribution statement

Coen de Vente: Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Bram van Ginneken:** Writing – review & editing, Supervision, Funding acquisition. **Carel B. Hoyng:** Writing – review & editing, Data curation. **Caroline C.W. Klaver:** Writing – review & editing, Data curation. **Clara I. Sánchez:** Writing – review & editing, Validation, Supervision, Resources, Project administration, Methodology, Investigation, Funding acquisition, Formal analysis, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The code is available on GitHub (referenced in the manuscript's abstract), including model weights and a link to the already public data. We do not have permission to share the private data.

Acknowledgments

This research was funded by Eurostars grant E12712.

Appendix A. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.media.2024.103259.

References

Apostolopoulos, S., Ciller, C., De Zanet, S., Wolf, S., Sznitman, R., 2017. RetiNet: Automatic AMD identification in OCT volumetric data. Invest. Ophthalmol. Vis. Sci. 58 (8), 387.

Ayhan, M.S., Kühlewein, L., Aliyeva, G., Inhoffen, W., Ziemssen, F., Berens, P., 2020. Expert-validated estimation of diagnostic uncertainty for deep neural networks in diabetic retinopathy detection. Med. Image Anal. 64, 101724.

Bazvand, F., Ghassemi, F., 2020. Artifacts in macular optical coherence tomography. J. Curr. Ophthalmol. 32 (2), 123.

Calli, E., Sogancioglu, E., Scholten, E.T., Murphy, K., van Ginneken, B., 2019. Handling label noise through model confidence and uncertainty: application to chest radiograph classification. In: Medical Imaging. In: Proceedings of the SPIE, (1), http://dx.doi.org/10.1117/12.2514290.

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

Chew, E.Y., Clemons, T., SanGiovanni, J.P., Danis, R., Domalpally, A., McBee, W., Sperduto, R., Ferris, F.L., AREDS2 Research Group, et al., 2012. The age-related eye disease study 2 (AREDS2): study design and baseline characteristics (AREDS2 report number 1). Ophthalmology 119 (11), 2282–2289.

Chikontwe, P., Kim, M., Nam, S.J., Go, H., Park, S.H., 2020. Multiple instance learning with center embeddings for histopathology classification. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. Springer, pp. 519–528.

De Fauw, J., Ledsam, J.R., Romera-Paredes, B., Nikolov, S., Tomasev, N., Blackwell, S., Askham, H., Glorot, X., O'Donoghue, B., Visentin, D., et al., 2018. Clinically applicable deep learning for diagnosis and referral in retinal disease. Nature Medicine 24 (9), 1342–1350.

European Commission, 2019. Ethics guidelines for trustworthy AI. URL: https://digital-strategy.ec.europa.eu/en/library/ethics-guidelines-trustworthy-ai.

Farsiu, S., Chiu, S.J., O'Connell, R.V., Folgar, F.A., Yuan, E., Izatt, J.A., Toth, C.A., 2014. Quantitative classification of eyes with and without intermediate age-related macular degeneration using optical coherence tomography. Ophthalmology 121 (1), 162–172.

Fauser, S., Smailhodzic, D., Caramoy, A., van de Ven, J.P.H., Kirchhof, B., Hoyng, C.B., Jeroen Klevering, B., Liakopoulos, S., den Hollander, A.I., 2011. Evaluation of serum lipid concentrations and genetic variants at high-density lipoprotein metabolism loci and TIMP3 in age-related macular degeneration. Invest. Ophthalmol. Vis. Sci. 52 (8), 5525–5528.

Foucart, A., Debeir, O., Decaestecker, C., 2018. Artifact identification in digital pathology from weak and noisy supervision with deep residual networks. In: 2018 4th International Conference on Cloud Computing Technologies and Applications. Cloudtech, IEEE, pp. 1–6.

Gal, Y., Ghahramani, Z., 2016. Dropout as a Bayesian approximation: Representing model uncertainty in deep learning. In: Proceedings of the 33rd International Conference on Machine Learning. ICML-16, pp. 1050–1059.

González-Gonzalo, C., Thee, E.F., Klaver, C.C., Lee, A.Y., Schlingemann, R.O., Tufail, A., Verbraak, F., Sánchez, C.I., 2021. Trustworthy AI: Closing the gap between development and integration of AI systems in ophthalmic practice. Prog. Retin. Eye Res. 101034.

Han, Z., Wei, B., Hong, Y., Li, T., Cong, J., Zhu, X., Wei, H., Zhang, W., 2020. Accurate screening of COVID-19 using attention-based deep 3D multiple instance learning. IEEE Trans. Med. Imaging 39 (8), 2584–2594.

He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep residual learning for image recognition. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition. CVPR, IEEE, pp. 770–778. http://dx.doi.org/10.1109/cvpr.2016.90.

Hendrycks, D., Basart, S., Mazeika, M., Zou, A., Kwon, J., Mostajabi, M., Steinhardt, J., Song, D., 2019. Scaling out-of-distribution detection for real-world settings. arXiv preprint arXiv:1911.11132.

Hendrycks, D., Gimpel, K., 2016. A baseline for detecting misclassified and out-of-distribution examples in neural networks. arXiv preprint arXiv:1610.02136.

Ho, J., Castro, D.P., Castro, L.C., Chen, Y., Liu, J., Mattox, C., Krishnan, C., Fujimoto, J.G., Schuman, J.S., Duker, J.S., 2010. Clinical assessment of mirror artifacts in spectral-domain optical coherence tomography. Invest. Ophthalmol. Vis. Sci. 51 (7), 3714–3720.

Hsu, Y.C., Shen, Y., Jin, H., Kira, Z., 2020. Generalized ODIN: Detecting out-of-distribution image without learning from out-of-distribution data. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 10951–10960.

Ikram, M.A., Brusselle, G.G., Murad, S.D., van Duijn, C.M., Franco, O.H., Goedegebure, A., Klaver, C.C., Nijsten, T.E., Peeters, R.P., Stricker, B.H., et al., 2017. The Rotterdam Study: 2018 update on objectives, design and main results. Eur. J. Epidemiol. 32 (9), 807–850.

Ilse, M., Tomczak, J., Welling, M., 2018. Attention-based deep multiple instance learning. In: International Conference on Machine Learning. PMLR, pp. 2127–2136.

Ioffe, S., Szegedy, C., 2015. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In: International Conference on Machine Learning. pmlr, pp. 448–456.

Kanwal, N., López-Pérez, M., Kiraz, U., Zuiverloon, T.C., Molina, R., Engan, K., 2024. Are you sure it's an artifact? Artifact detection and uncertainty quantification in histological images. Comput. Med. Imaging Graph. 112, 102321.

Kingma, D., Ba, J., 2014. Adam: A method for stochastic optimization. arXiv:1412.6980.

Klein, R., Davis, M.D., Magli, Y.L., Segal, P., Klein, B.E.K., Hubbard, L., 1991. The wisconsin age-related maculopathy grading system. Ophthalmology 98 (7), 1128–1134.

Kurmann, T., Yu, S., Márquez-Neila, P., Ebneter, A., Zinkernagel, M., Munk, M.R., Wolf, S., Sznitman, R., 2019. Expert-level automated biomarker identification in optical coherence tomography scans. Sci. Rep. 9 (1), 1–9.

Lakshminarayanan, B., Pritzel, A., Blundell, C., 2017. Simple and scalable predictive uncertainty estimation using deep ensembles. In: Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R. (Eds.), In: Advances in Neural Information Processing Systems, vol. 30, Curran Associates, Inc., URL: https://proceedings.neurips.cc/paper/2017/file/9ef2ed4b7fd2c810847ffa5fa85bce38-Paper.pdf.

Lee, C.S., Baughman, D.M., Lee, A.Y., 2017. Deep learning is effective for classifying normal versus age-related macular degeneration OCT images. Ophthalmol. Retina 1 (4), 322–327.

Linmans, J., Elfwing, S., van der Laak, J., Litjens, G., 2023. Predictive uncertainty estimation for out-of-distribution detection in digital pathology. Med. Image Anal. 83, 102655.

Linmans, J., van der Laak, J., Litjens, G., 2020. Efficient out-of-distribution detection in digital pathology using multi-head convolutional neural networks. In: Medical Imaging with Deep Learning. PMLR, pp. 465–478, URL: https://openreview.net/forum?id=hRwB2BTRNu.

Liu, W., Wang, X., Owens, J., Li, Y., 2020. Energy-based out-of-distribution detection. Adv. Neural Inf. Process. Syst. 33, 21464–21475.

Mehrtash, A., Wells, W.M., Tempany, C.M., Abolmaesumi, P., Kapur, T., 2020. Confidence calibration and predictive uncertainty estimation for deep medical image segmentation. IEEE Trans. Med. Imaging 39 (12), 3868–3878.

Nair, V., Hinton, G., 2010. Rectified linear units improve restricted Boltzmann machines. In: International Conference on Machine Learning. pp. 807–814.

Neppalli, S., Kessell, M.A., Madeley, C.R., Hill, M.L., Vlaskovsky, P.S., Taylor, D.B., 2021. Artifacts in contrast-enhanced mammography: are there differences between vendors? Clinical Imaging 80, 123–130.

Oner, M.U., Kye-Jet, J.M.S., Lee, H.K., Sung, W.K., 2020. Studying the effect of mil pooling filters on mil tasks. arXiv preprint arXiv:2006.01561.

Oner, M.U., Kye-Jet, J.M.S., Lee, H.K., Sung, W.K., 2023. Distribution based MIL pooling filters: Experiments on a lymph node metastases dataset. Med. Image Anal. 87, 102813.

Ovadia, Y., Fertig, E., Ren, J., Nado, Z., Sculley, D., Nowozin, S., Dillon, J.V., Lakshminarayanan, B., Snoek, J., 2019. Can you trust your model's uncertainty? Evaluating predictive uncertainty under dataset shift. arXiv preprint arXiv:1906.02530.

Patil, A., Tamboli, D., Meena, S., Anand, D., Sethi, A., 2019. Breast cancer histopathology image classification and localization using multiple instance learning. In: 2019 IEEE International WIE Conference on Electrical and Computer Engineering. WIECON-ECE, IEEE, pp. 1–4.

Puzyeyeva, O., Lam, W.C., Flanagan, J.G., Brent, M.H., Devenyi, R.G., Mandelcorn, M.S., Wong, T., Hudson, C., et al., 2011. High-resolution optical coherence tomography retinal imaging: a case series illustrating potential and limitations. J. Ophthalmol. 2011.

Qiu, Z., Pan, Y., Wei, J., Wu, D., Xia, Y., Shen, D., 2021. Predicting symptoms from multiphasic MRI via multi-instance attention learning for hepatocellular carcinoma grading. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. Springer, pp. 439–448.

Qu, L., Wang, M., Song, Z., et al., 2022. Bi-directional weakly supervised knowledge distillation for whole slide image classification. Adv. Neural Inf. Process. Syst. 35, 15368–15381.

Quellec, G., Cazuguel, G., Cochener, B., Lamard, M., 2017. Multiple-instance learning for medical image and video analysis. IEEE Rev. Biomed. Eng. 10, 213–234.

Rasti, R., Rabbani, H., Mehridehnavi, A., Hajizadeh, F., 2017. Macular OCT classification using a multi-scale convolutional neural network ensemble. IEEE Trans. Med. Imaging 37 (4), 1024–1034.

Ren, Q., Zhao, Y., He, B., Wu, B., Mai, S., Xu, F., Huang, Y., He, Y., Huang, J., Yao, J., 2023. IIB-MIL: Integrated instance-level and bag-level multiple instances learning with label disambiguation for pathological image analysis. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. Springer, pp. 560–569.

Romo-Bucheli, D., Seeböck, P., Orlando, J.I., Gerendas, B.S., Waldstein, S.M., Schmidt-Erfurth, U., Bogunović, H., 2020. Reducing image variability across OCT devices with unsupervised unpaired learning for improved segmentation of retina. Biomed. Optics Express 11 (1), 346–363.

Rutter, C.M., 2000. Bootstrap estimation of diagnostic accuracy with patient-clustered data. Academic Radiol. 7, 413–419.

Schömig-Markiefka, B., Pryalukhin, A., Hulla, W., Bychkov, A., Fukuoka, J., Madabhushi, A., Achter, V., Nieroda, L., Büttner, R., Quaas, A., et al., 2021. Quality control stress test for deep learning-based diagnostic model in digital pathology. Mod. Pathol. 34 (12), 2098–2108.

Seeböck, P., Romo-Bucheli, D., Waldstein, S., Bogunovic, H., Orlando, J.I., Gerendas, B.S., Langs, G., Schmidt-Erfurth, U., 2019. Using CycleGANs for effectively reducing image variability across OCT devices and improving retinal fluid segmentation. In: 2019 IEEE 16th International Symposium on Biomedical Imaging. ISBI 2019, IEEE, pp. 605–609.

Shao, Z., Bian, H., Chen, Y., Wang, Y., Zhang, J., Ji, X., et al., 2021. Transmil: Transformer based correlated multiple instance learning for whole slide image classification. Adv. Neural Inf. Process. Syst. 34, 2136–2147.

Shin, S.Y., Lee, S., Yun, I.D., Kim, S.M., Lee, K.M., 2018. Joint weakly and semi-supervised deep learning for localization and classification of masses in breast ultrasound images. IEEE Trans. Med. Imaging 38 (3), 762–774.

Swanson, E.A., Fujimoto, J.G., 2017. The ecosystem that powered the translation of OCT from fundamental research to clinical and commercial impact. Biomed. Opt. Express 8 (3), 1638–1664.

Tack, J., Mo, S., Jeong, J., Shin, J., 2020. Csi: Novelty detection via contrastive learning on distributionally shifted instances. Adv. Neural Inf. Process. Syst. 33, 11839–11852.

Tomczak, J.M., Ilse, M., Welling, M., Jansen, M., Coleman, H.G., Lucas, M., de Laat, K., de Bruin, M., Marquering, H., van der Wel, M.J., et al., 2018. Histopathological classification of precursor lesions of esophageal adenocarcinoma: A deep multiple instance learning approach. In: Medical Imaging with Deep Learning. pp. 1–3.

Ulyanov, D., Vedaldi, A., Lempitsky, V., 2016. Instance normalization: The missing ingredient for fast stylization. arXiv preprint arXiv:1607.08022.

van de Ven, J.P.H., Smailhodzic, D., Boon, C.J.F., Fauser, S., Groenewoud, J.M.M., Victor Chong, N., Hoyng, C.B., Jeroen Klevering, B., den Hollander, A.I., 2012. Association analysis of genetic and environmental risk factors in the cuticular drusen subtype of age-related macular degeneration. Mol. Vis. 18, 2271–2278.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I., 2017. Attention is all you need. Adv. Neural Inf. Process. Syst. 30.

Venhuizen, F.G., van Ginneken, B., van Asten, F., van Grinsven, M.J., Fauser, S., Hoyng, C.B., Theelen, T., Sánchez, C.I., 2017. Automated staging of age-related macular degeneration using optical coherence tomography. Invest. Ophthalmol. Vis. Sci. 58 (4), 2318–2328. http://dx.doi.org/10.1167/iovs.16-20541.

Wang, X., Tang, F., Chen, H., Luo, L., Tang, Z., Ran, A.R., Cheung, C.Y., Heng, P.A., 2020. UD-MIL: uncertainty-driven deep multiple instance learning for OCT image classification. IEEE J. Biomed. Health Inf. 24 (12), 3431–3442.

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

Wellenberg, R., Hakvoort, E., Slump, C., Boomsma, M., Maas, M., Streekstra, G., 2018. Metal artifact reduction techniques in musculoskeletal CT-imaging. Eur. J. Radiol. 107, 60–69.

Xie, X.L., Beni, G., 1991. A validity measure for fuzzy clustering. IEEE Trans. Pattern Anal. Mach. Intell. 13 (8), 841–847.

Xu, C., Qi, S., Feng, J., Xia, S., Kang, Y., Yao, Y., Qian, W., 2020. DCT-MIL: deep CNN transferred multiple instance learning for COPD identification using CT images. Phys. Med. Biol. 65 (14), 145011.

Xu, G., Song, Z., Sun, Z., Ku, C., Yang, Z., Liu, C., Wang, S., Ma, J., Xu, W., 2019. Camel: A weakly supervised learning framework for histopathology image segmentation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 10682–10691.

Yanagihara, R.T., Lee, C.S., Ting, D.S.W., Lee, A.Y., 2020. Methodological challenges of deep learning in optical coherence tomography for retinal diseases: a review. Transl. Vis. Sci. Technol. 9 (2), 11.

Yin, S., Peng, Q., Li, H., Zhang, Z., You, X., Liu, H., Fischer, K., Furth, S.L., Tasian, G.E., Fan, Y., 2019. Multi-instance deep learning with graph convolutional neural networks for diagnosis of kidney diseases using ultrasound imaging. In: Uncertainty for Safe Utilization of Machine Learning in Medical Imaging and Clinical Image-Based Procedures. Springer, pp. 146–154.

Zhu, W., Sun, L., Huang, J., Han, L., Zhang, D., 2021. Dual attention multi-instance deep learning for Alzheimer's disease diagnosis with structural MRI. IEEE Trans. Med. Imaging 40 (9), 2354–2366.