Margin-aware intraclass novelty identification for medical images

Xiaoyuan Guo[a,*] **Judy W. Gichoya**[b] **Saptarshi Purkayastha**[c] **and Imon Banerjee**[d,e]

[a]Emory University, Department of Computer Science, Atlanta, Georgia, United States
[b]Emory University, Department of Radiology and Imaging Sciences, Atlanta, Georgia, United States
[c]Indiana University-Purdue University Indianapolis, School of Informatics and Computing, Indianapolis, Indiana, United States
[d]Mayo Clinic, Department of Radiology, Phoenix, Arizona, United States
[e]Arizona State University, Department of Computer Engineering, Tempe, Arizona, United States

Abstract

Purpose: Existing anomaly detection methods focus on detecting interclass variations while medical image novelty identification is more challenging in the presence of intraclass variations. For example, a model trained with normal chest x-ray and common lung abnormalities is expected to discover and flag idiopathic pulmonary fibrosis, which is a rare lung disease and unseen during training. The nuances of intraclass variations and lack of relevant training data in medical image analysis pose great challenges for existing anomaly detection methods.

Approach: We address the above challenges by proposing a hybrid model—transformation-based embedding learning for novelty detection (TEND), which combines the merits of classifier-based approach and AutoEncoder (AE)-based approach. Training TEND consists of two stages. In the first stage, we learn in-distribution embeddings with an AE via the unsupervised reconstruction. In the second stage, we learn a discriminative classifier to distinguish in-distribution data and the transformed counterparts. Additionally, we propose a margin-aware objective to pull in-distribution data in a hypersphere while pushing away the transformed data. Eventually, the weighted sum of class probability and the distance to margin constitutes the anomaly score.

Results: Extensive experiments are performed on three public medical image datasets with the one-vs-rest setup (namely one class as in-distribution data and the left as intraclass out-of-distribution data) and the rest-vs-one setup. Additional experiments on generated intraclass out-of-distribution data with unused transformations are implemented on the datasets. The quantitative results show competitive performance as compared to the state-of-the-art approaches. Provided qualitative examples further demonstrate the effectiveness of TEND.

Conclusion: Our anomaly detection model TEND can effectively identify the challenging intraclass out-of-distribution medical images in an unsupervised fashion. It can be applied to discover unseen medical image classes and serve as the abnormal data screening for downstream medical tasks. The corresponding code is available at https://github.com/XiaoyuanGuo/TEND_MedicalNoveltyDetection.

 [DOI: 10.1117/1.JMI.9.1.014004]

Keywords: anomaly detection; out-of-distribution (OOD) detection; novelty identification; intraclass OOD; medical image.

Paper 21139RRRR received Jun. 4, 2021; accepted for publication Jan. 13, 2022; published online Feb. 3, 2022.

*Address all correspondence to Xiaoyuan Guo, xiaoyuan.guo@emory.edu

2329-4302/2022/$28.00 © 2022 SPIE

1 Introduction

With recent prominent developments of machine learning techniques in computer vision, integrating machine learning tools to solve medical image problems is becoming more and more popular due to the powerful computation and efficiency.[1] However, when deploying machine learning models in real-world applications, models trained on in-distribution (ID) data may fail to deal with out-of-distribution (OOD) inputs and assign incorrect probabilities.[2] This can severely contaminate the reliability of artificial intelligence models, especially in medical areas as the safety in clinical decisions is much more critical than other fields. For example, a classifier trained on existing bacterial classes wrongly classified a new type of bacteria as one of the classes from the training data with high confidence,[3] which could be concerning for clinical usage but may be avoided by combining an OOD detection model. Thus a successful open-world deployment with OOD detection should be sensitive to unseen classes and distribution-shifted samples and also be resilient to potential adversarial attacks.[4]

However, medical OOD detection poses great challenges due to the heterogeneity and unknown data characteristics of medical data. (1) Mutations can happen. Different from natural objects with fixed attributes, known diseases may progress to other mutated versions and generate anomalous data. (2) Heterogeneous data are a big concern. Medical images collected from different race groups can introduce heterogeneity. (3) Distribution shifting always exists. Data scanned with different machines or institutes may have distribution shifting. (4) Data with defects are common. Medical images can be overexposed or scanned with incorrect positions/angles.

OOD data, also called anomaly, outlier, usually refer to data that shows dissimilarity from the training distribution. Given an image x, the goal of OOD detection is to identify whether x is from ID dataset D_{in} or OOD dataset D_{out}. There are two types of OOD data commonly targeted to identify—(i) intraclass data: OOD data belonging this type, which is also called novelty data, often shares severe similarity with the ID classes and is extremely challenging to distinguish, e.g., the pneumonia chest x-ray presents close appearance with the normal images. (ii) Interclass data: these data are significantly different from ID samples, e.g., a head CT image is very different in shape and color from the skin cancer image. Even though many anomaly detection methods have been proposed,[5,6] most of them focus on natural images and follow the one-vs-rest setup[7] for benchmark natural image datasets (e.g., MNIST,[8] CIFAR-10,[9] and ImageNet.[10]). Thus the performance reported on the benchmark datasets is actually for interclass prediction due to the clear class variation and is often trivial to detect. In contrast, the anomaly detection in medical images is more of an intraclass identification problem, which can be also called novelty detection.

To train a novelty detector with only ID data available, learning high-quality "normality" features is the fundamental step to identify the OOD samples during inference. AutoEncoder (AE)[11] architecture, as an unsupervised model to learn efficient data features through reconstruction, is the most straightforward way to extract features for ID data.[12] For anomaly detection, the reconstruction error is treated as the score of outliers based on the assumption that the AE[11] is unable to reconstruct the anomalies well and causes large reconstruction errors. However, in the intraclass detection where the variations among the in-class and out-of-class medical images of the same category are very subtle, the AE[11] often fails owing to the lack of discriminative ability for intraclass detection (see Sec. 2).

To enhance the discriminative ability of the AE,[11] we propose transformation-based embedding learning of novelty detection (TEND) to distinguish intraclass OOD inputs in an unsupervised fashion. Based on the vanilla AE[11] model to learn the "normality" of ID data in the first stage and function as a feature extractor in the second stage, TEND utilizes distorted images generated by adding transformations on the ID data and treats the data as non-ID data (marginal OOD, see Sec. 3.2). A binary classifier of TEND is trained with the ID data as normal class and the non-ID data as OOD class. Hence, the classifier is aware of the existence of outliers and gains certain identification ability of true outliers during inference without being trained on any true OOD data. To further separate OOD data from the ID ones, we learn a distance metric objective to encourage clustering of ID data during training and enforce a margin between OOD

versus ID data in the embedding space. In summary, the main contributions of our paper are as follows.

(1) We propose a novelty detection model TEND that utilizes the AE's feature extraction and adds discrimination ability for outliers with transformations of in-distribution data and embedding distance as auxiliary. No out-of-distribution data are required for training the model.
(2) Although there have been a lot of anomaly detection research work done, the accurate detection performance results are lacking. We compare and report the novelty detection performance details of the unsupervised TEND model with state-of-the-art anomaly detection models and one supervised model on three public medical image datasets following two experimental settings—one-vs-rest and rest-vs-one.
(3) We validate our method on diverse image datasets and demonstrate our model's effectiveness. Extensive evaluations include the detection of intraclass out-of-distribution data from the original datasets and the corresponding generated with unused transformations on in-distribution data. Given the experimental observations, our model will be beneficial in discovering new anomaly cases in medical applications without any preconceived OOD training data.

2 Background

There have been a lot of research works that summarize state-of-the-art anomaly detection methods,[13–18] generally the methods aiming for anomalous image data detection can be divided into the following three categories:

2.1 *AutoEncoder-Based Methods*

AE[11] models can help extract significant embedding features by reconstructing the original images unsupervised. Trained with ID data, the architectures learn the "normality" and should lead to large reconstruction error when working on OOD dataset. Thus the reconstruction error acts as the anomaly score to separate ID and OOD data.[19–21] However, AE risks learning the identity function by simply outputting the original inputs, which largely limits its discriminative ability of anomalies. Other improved versions of AE are also used for anomaly detection,[22–25] e.g., variational autoencoders (VAE)[25] provide probabilistic way of describing the latent space to reconstruct input data. Nevertheless, the reconstruction is often blurry and not good enough for clear discrimination of outliers. Since TEND is designed based on AE, we take the vanilla AE[11] as a baseline. In addition, we also compare the performance with an extension of AE that adds a Gaussian mixture model (GMM) head on the AE backbone, (AE_GMM for simplicity's sake) and the standard VAE[25] model. Similar with VAE, UAV-AdNet[26] uses the Kullback–Leibler divergence to regularize losses for anomaly detection but focuses on autonomous surveillance systems with GPS label used, which does not apply to this work.

2.2 *Generative Adversarial Network-Based Methods*

Similar to the AE models, GAN[27] framework can also learn latent feature representations by training a fake image generator and a real-vs-fake image discriminator.[28,29] With the adversarial feature learning, GAN-based anomaly detectors can acquire discriminative latent features that can be used for separating the ID data from the OOD data. To further improve the discriminative ability of latent representations, BiGAN[30] adopts a bidirectional mapping learning. GANomaly[31] minimizes the distance of the ID data and the generated ones in latent feature space to detect the OOD data with large distance. Even so, the performance of GAN-based anomaly detectors largely depends on the training of GAN models, which always require large amounts of training data for OOD and often fail to handle inputs with large image size. Instead of selecting AnoGAN,[32] which detects pixel-wise anomalies rather than in image level, we compare TEND with GANomaly[31] and f-AnoGAN,[33] AnoGAN's extension, for experiments given the better performance.

2.3 *Classifier-Based Methods*

As the novelty detection in medical images can be reduced to a one-class classification[34] problem with the one-vs-rest setup, one-class classifiers are often used for identifying unseen classes, e.g., OC-SVM,[35] FCDD,[6] DOC,[36] and DeepSVDD.[12] With only ID data as training inputs, one-class classifiers often optimize a kernel-based objective function and minimize a hypersphere to threshold out the anomaly data based on distance. The one-class classifiers exploit in-distribution data with specific object functions to threshold out anomalies. Nonetheless, their detection abilities on intraclass OOD data are not effective as the intraclass OOD data share a lot of similarity with the ID data. Except for the one-class classifiers, ODIN[37] works on multiclasses datasets by adding perturbations of the input and temperature scaling to the score function to distinguish in-distribution and OOD data. Despite the efficiency and sophisticated methodology, the prerequisites of multiple OOD classes of the dataset are not typical in the medical image area and thus classifier-based methods have limited applicability in healthcare. To showcase the performance difference, we choose DeepSVDD,[12] which is a representative model, to compare with TEND.

3 Method

TEND focuses on novelty identification for medical images. By following the one-vs-rest setup[6] and its reversed version—the rest-vs-one setup, one or more certain classes of the datasets in use are treated as normal classes. Unsupervised learning of feature embeddings for the normal classes is the fundamental step for outlier detection. GANs and AEs are all good options for this work. Nonetheless, GANs often require large amounts of data for training and are unstable for large images, we choose the vanilla AE[27] to encode the ID data. Moreover, as introduced in Sec. 2, AEs are designed for compressing inputs and have no strong discriminative ability, which makes them inappropriate for medical novelty detection because of the minute intraclass variations of medical image datasets. Thus to enhance the discriminative ability of TEND, we train a binary classifier and a margin-aware objective function (also called margin learner) jointly to separate the normal class data from the anomalies.

3.1 *Architecture*

Figure 1 shows the network architecture of TEND, which is a two-stage novelty detector with an AE[11] as the feature extractor backbone. In order to train the feature extractor with only ID data, the AE[11] model (shown in the dotted blue box of Fig. 1) is optimized with a reconstruction loss function L_{rec}. The learnt bottleneck section will be frozen as indicated by the purple lock in Fig. 1 and used for encoding/extracting image features in the second stage. To train the following binary discriminator without OOD data available, we add transformations on the original images to construct distribution-shifted OOD samples based on the observation that some augmentations can be useful for OOD detection by considering them as fake OOD data.[7] The details of how to construct the transformations are explained in Sec. 3.2. The generated OOD data should be first fed to the trained encoder to obtain the corresponding deep features. Both of the encoded features of normal and transformed data are fed to the classifier simultaneously. With a convolutional (conv) layer and a fully connected layer (FCN), the classifier learns to identify the in-distribution data as normal class and the transformed images as outliers. A latent decision boundary between the two classes is optimized, the detection on true anomaly data is still not promising given the fact that the transformed images cannot represent the true outliers' distribution. The decision boundary may not work for the anomalies in the feature space. To solve this problem, TEND adopts the margin-aware learning idea of DeepSVDD[12] to optimize a distance objective function simultaneously. Different from the objectives only for ID data,[12] TEND works on both the ID data and the fake OOD data by enforcing the embeddings of ID data to cluster around a voted center O (see Sec. 3.3 for more details) and pushing away the fake abnormal data to at least a certain distance R (a predefined margin).

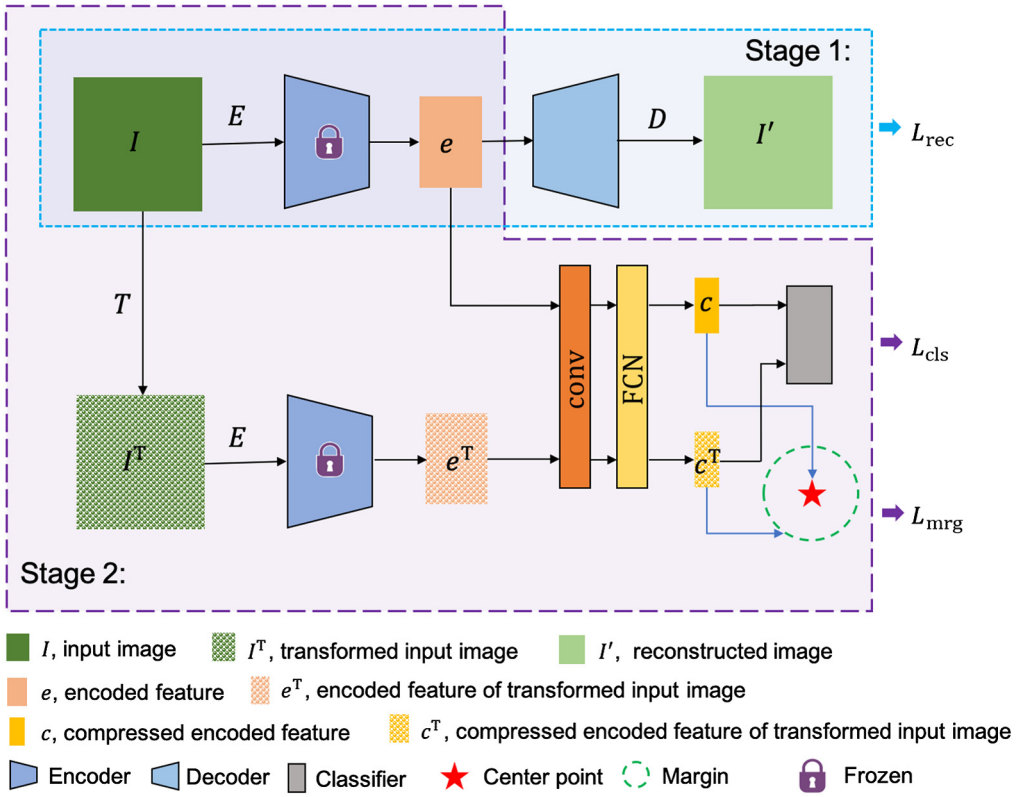

Fig. 1 Network architecture of TEND. Stage 1: training AE with in-distribution data and stage 2: joint training of the classifier and the margin learner.

3.2 *Transformations for Generating Fake OOD Data*

SimCLR[38] has performed an extensive study in which family of augmentations leads to a better self-supervised learning, i.e., transformations should be considered as positives. The authors report that some of the examined augmentations (e.g., rotation) could lead to degraded performance. Based on the observation, such augmentations can be useful for OOD detection by considering them as fake OOD data. Therefore, we leverage a family of transformations and utilize more complex transformations and distortion functions that will change the visual features of the original inputs to generate fake abnormal data for training in OOD model. The generated auxiliary data are fed to the forehead of the TEND backbone and then to the classifier, which helps separate the embedding features of the ID data from those of the unknown OOD data. Different from the most common transformations, e.g., rotation, used in classic data augmentation, we adopt a range of different distortions, i.e., barrel, perspective, arc, polar, tile, affine defined in the Image.distort method of Wand package. The blue box in the middle part of Fig. 2 shows the six different transformations on the three datasets. These transformations bring significant difference to the original inputs and generate intraclass OOD samples. We treated these extreme distortions of ID data as outliers for training. Except for the six distortions used in this paper, there are more transformations worthwhile being explored. To further demonstrate the benefits of training the TEND model using extreme transformations, we use moderate distortions, such as randomly cutting, randomly cropping and resizing, addition of noises, and Gaussian blurring only for validation (shown in the right yellow box of Fig. 2). The package usage and parameters selection for the six training distortions and the four validation transformation are present in our code repository.

3.3 *Joint Training*

With an AE[11] as the backbone, TEND incorporates a classifier and a margin-aware embedding mapping to gain discriminative ability for anomalies. In the first stage, the backbone is trained

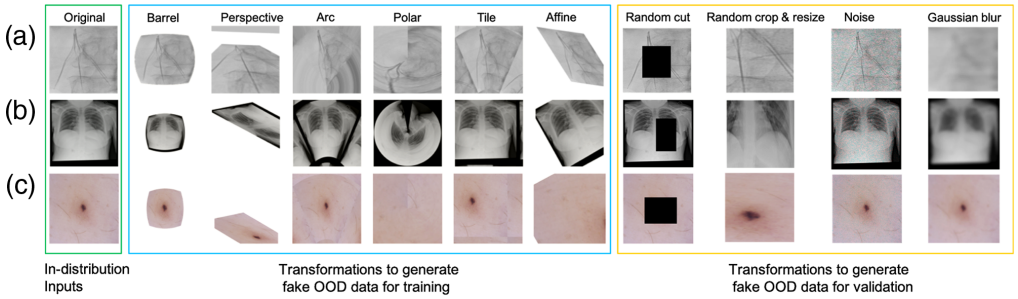

Fig. 2 Examples of transformations used for generating fake OOD data. Three image examples from (a) IVC-filter, (b) RSNA, and (c) ISIC2019 datasets are presented. The original data in the green box are inputs from in-distribution class, the transformed in-distribution images in the blue box are auxiliary data as anomalies feed to TEND's classifier during training, other possible transformations shown in the yellow box are for validation.

only on ID data. Suppose that the input image I and reconstructed image I' is with size of $M \times N$, a reconstruction objective f_{rec} defined in Eq. (1) is used to optimize the learning embedding representations of the normal class. This first-stage training ensures the feature extractor to focus on learning the "normality" of in-class data:

$$f_{\text{rec}} = \min \frac{1}{M}\frac{1}{N}\sum_{i=1,j=1}^{M,N} \|I_{ij} - I'_{ij}\|^2, \tag{1}$$

$$L_{\text{cls}} = \frac{1}{S}\sum_{i=1}^{S} y_i \cdot \log(p(y_i)) + (1 - y_i) \cdot \log(1 - p(y_i)). \tag{2}$$

With the distorted ID data as anomalies in the second stage, the binary discriminator is able to train with a final output indicating the data class. Notably, the inputs of this classifier are the encoded features extracted by the backbone. Here the AE model is fully frozen and only used for extracting image features. The encoded features e, e^{T} are processed by a following convolutional layer (conv) and an FCN of the classifier. Thus the embeddings learnt by the encoder are mapped to a new compressed space as c, c^{T} with size of K (512 in our case). The classifier enables the separation of the compressed features of the ID data and the distorted data. A binary cross entropy (BCE) loss function L_{cls} shown in Eq. (2) is utilized for optimizing, with the S to be the total number of the training data, y_i representing the i'th data's binary label and $p(y_i)$ being the corresponding probability of the prediction. Nonetheless, the transformations T can only introduce limited class variations, hence the identification for real OOD data is still not ideal. Thus a margin-aware objective is jointly trained to force the clustering of the compressed features of the ID data and the surrounding of the transformed ID data outside the margin as illustrated by Fig. 2.

In experiments, we test three margin R values (150, 250, and 500). Similar to DeepSVDD,[12] the compressed feature center O is calculated by the mean of all the ID data's compressed features. Before calculation, TEND's classifier block is trained with several warm-up epochs, (e.g., 10 epochs), then the center O is defined with the same size of K as the compressed feature c. Since then, the margin learner of TEND is trained together with the discriminator. Importantly, the margin learner has different learning objectives for the normal class (g_{in}) shown in Eq. (3) and the generated abnormal class (g_{out}) shown in Eq. (4):

$$g_{\text{in}} = \min \frac{1}{K}\sum_{i=1}^{K} \|c_i - O\|^2, \tag{3}$$

$$g_{\text{out}} = \min \frac{1}{K} \sum_{i=1}^{K} \max(R - \|c_i^{\text{T}} - O\|^2, 0). \tag{4}$$

In summary, TEND has two stage-wise losses. The first-stage loss is for the reconstruction of the AE training, i.e., $L_{1\text{st}} = L_{\text{rec}}$. The second-stage loss includes the binary classifier and the margin learner, i.e., $L_{2\text{nd}} = L_{\text{cls}} + L_{\text{mrg}}$. In experiments, we use mean-square-error loss for L_{rec} and BCE loss for L_{cls}. Marginal loss L_{mrg} equals the summation of the mean of distance errors for ID data and the mean of the errors for distorted data.

3.4 *Implementation Details*

An AE architecture is trained as our baseline, the trained model later on is treated as the backbone of TEND. We report the encoder, decoder, Conv, and FCN parts of TEND in Table 1. FC is FCN, Conv stands for the convolutional layer, TConv means the transposed convolutional layer, and channel indicates the image channel. All the Conv and TConv layers use kernel filter size 4, stride 2, and padding 1. The encoder encodes input images as e, whereas the Conv layer compresses e to c with smaller sizes. Each Conv and TConv is followed by a standard batch-normalization layer and a ReLU function.

In our experiments, we use Adam optimizer with a learning rate of 0.001 for model training. Each network is trained with 50 to 150 epochs depending on the dataset size and the data complexity as datasets with more complex data or large amounts of samples often take more time to get the loss decreased to a satisfactory level. When training with the margin-aware metric, we run 10 warm-up epochs first and then calculate the embedding center O. The pipelines are developed using Pytorch 1.5.0, Python 3.0., and Cuda compilation tools V10.0.130 on a machine with 3 NVIDIA Quadro RTX 6000 with 24 GB memory.

3.5 *Anomaly Score*

As a standard evaluation procedure for anomaly detectors, the ID and outliers are mixed for computing the accuracy while different detectors have different anomaly score definitions. For the baseline AE model, we set the reconstruction error as the OOD data score. TEND does not focus on the reconstruction, therefore, the final anomaly score of TEND is the classification probability adding the marginal distance. Given the fact that the classification probability p is in range $[0-1]$ and the distance value d is in $[0, +\infty)$, we scale down the distance value d by dividing the predefined margin R, i.e., $d' = \frac{d}{R}$. Therefore, the final anomaly score for TEND is $S_i = \lambda p_i + (1-\lambda) d_i'$. The value of λ is set as 0.5 in our experiments as default. To further demonstrate the effectiveness of each component of TEND, we have done the ablation study of TEND and reported the results in Sec. 4.4. TEND without the binary classifier is called margin learner (the anomaly score is d').

3.6 *Evaluation Metrics*

Having the anomaly prediction score, the detection accuracy largely depends on the threshold setting. To be fair, the detection evaluation should be threshold invariant. Following the standard

Table 1 TEND architecture details.

Dataset	Encoder	Decoder	Conv	FCN
IVC-filter/RSNA/ISIC	Conv (channel, 16) Conv (16, 32) Conv (32, 64) Conv (64, 128) Conv (128, 256)	TConv (256, 128) TConv (128, 64) TConv (64, 32) TConv (32, 16) TConv (16, channel)	Conv (256, 512)	FC (2048, 512) FC (512, 1)

evaluation metrics used in other works,[37,39] we adopt AUROC (AUC in short) to showcase the performance difference among the models. AUROC is the area under the receiver operating characteristic (ROC) curve, which is a threshold-independent metric. The AUROC can be interpreted as the probability that a positive example is assigned a higher detection score than a negative example. To find an optimal threshold for ROC curve by tuning the decision thresholds, we use the geometric mean (G-Mean) as the metric to determine the best threshold values and report the resulted true positive rate ($\text{TPR} = \frac{\text{TP}}{\text{TP+FN}}$) and false positive rate ($\text{FPR} = \frac{\text{FP}}{\text{FP+TN}}$). The difference between the TPR and FPR given the optimal selection, $\text{DIFF} = \text{TPR} - \text{FPR}$, is also reported for model comparison. Large differences stand for better true and false positive predictions. Moreover, we measure the uncertainty of models' performance with 10 rounds of bootstrapping estimations, by randomly sampling the predictions to the same amount of test samples with replacement. The resulting standard deviation values are present in Tables 3–6.

4 Experiments

In this section, we perform empirical evaluations of TEND on publicly available medical image datasets with varying complexity. For evaluating the accuracy in identifying novel class data, we compare our results with state-of-the-art unsupervised OOD models, starting from simple vanilla AE[11] model and a VAE,[25] to DeepSVDD,[12] GANomaly,[31] and f-AnoGAN[33] models. We also compare our unsupervised TEND model against a supervised binary classifier, which was trained on both ID and OOD data for the detection task.

4.1 *Datasets*

In our experiments, we have three medical datasets in use, including inferior vena cava filters on radiographs[40] and RSNA chest x-ray dataset,[41] ISIC2019.[42] IVC-filter dataset has 14 classes in total. The details are ALN (73 images), BardSimonNitinol (59 images), Optease (129 images), BardDenali (50 images), Celect (75 images), Option (196 images), BardEclipseG2X (84 images), CelectPlatinum (48 images), Trapease (100 images), BardG2 (45 images), Greenfield12Fr (122 images), Tulip (99 images), BardMeridian (55 images), and GreenfieldTitanium (101 images). RSNA has three classes—normal, with opacity, and not normal in total. ISIC2019[42] consists of 8 classes, i.e., melanoma (MEL, 4148 images), melanocytic nevus (NV, 11,559 images), basal cell carcinoma (BCC, 3323 images), actinic keratosis (867 images), benign keratosis (2240 images), dermatofibroma (239 images), vascular lesion (253 images), and squamous cell carcinoma (628 images). The IVC-filter and ISIC2019 image are with varying sizes, with the width size ranging from 150 to 1500, height size ranging from 150 to 1500 roundly, e.g., $469 \times 365 \times 3$. The RSNA dataset is in dicom format, each dicom file has the pixel array of size 1024×1024. To unify the training pipeline, we resize all the IVC-filter, RSNA, and ISIC data in $256 \times 256 \times$ channel.

For the one-vs-rest setting, the in-class and rest classes data details are summarized in Table 2. Due to the data imbalance, we usually pick the class with the most data as our

Table 2 Three publicly available dataset used in the study—total number of images in the dataset. In-distribution data (D_{in}) and out-of-distribution data (D_{out}) with one-vs-rest setting.

		D_{in}		D_{out}	
Dataset	Total classes	Class	#Images	Class	#Images
IVC-filter[40]	14	Option	196	BardSimonNitinol, ALN...	1040
RSNA[41]	3	Normal	8851	With opacity, not normal	21,376
ISIC[42]	8	NV	11,559	MEL, BCC...	11,698

in-distribution data and all the left classes as intraclass OOD data. For IVC-filter, we select the option type as the normal class; for RSNA dataset, we treat the normal class as ID data; for ISIC2019 dataset, we choose the NV class with the most samples as ID inputs. The total numbers of ID and OOD data for each dataset are reported in the column of #images in Table 2. Notably, the rest-vs-one setting experiments treat the classes conversely.

4.2 *Training and Evaluation Settings*

To train and evaluate OOD detectors' performance, we split the in-distribution data with 80% as training set $D_{\text{in}}^{\text{tr}}$ and 20% as test set $D_{\text{in}}^{\text{te}}$ and use all the left classes as D_{out}. For OOD detection evaluation, we mixed $D_{\text{in}}^{\text{te}}$ and D_{out} by assigning the ID data with label 0 and OOD data with label 1. Since this paper focuses on intraclass OOD detection, we will report the OOD detection results within the same dataset instead of crossing different datasets.

4.3 *Quantitative Results*

4.3.1 *One-vs-rest results*

Following the one-vs-rest setting, Table 3 presents the AUC scores and the corresponding FPR, TPR values determined by the optimal thresholds for AE,[11] VAE,[25] DeepSVDD,[12] GANomaly[31], f-AnoGAN,[33] and TEND models with margin 150 (i.e., TEND_150), 250 (i.e., TEND_250), and 500 (i.e., TEND_500). The difference between the TPR and FPR is also reported in the *DIFF* column in Table 3. ↓ means that the lower the value of the better the model is, whereas ↑ stands for the higher the value of the better the model performs. Thus we expect the model to have high AUC score and prefer low FPR and high TPR values when deploying the models with the optimal threshold as decision boundary, which means the larger the difference between TPR and FPR the better. The best and second best DIFF and AUC results are highlighted by bold and italics, respectively. Among the unsupervised anomaly detectors, our model TEND_150 attains the optimal DIFF result 0.531 and AUC score 0.772 for IVC-filter dataset and second best AUC score 0.615 for RSNA dataset; TEND_250 achieves the second highest DIFF 0.524 for IVC-filter dataset and the second highest AUC score 0.615 for RSNA dataset. Meanwhile, TEND_250 reaches the best DIFF 0.343 and AUC score 0.717 for ISIC2019 dataset compared to other methods. TEND_500 reaches the suboptimal AUC score 0.760 for IVC-filter dataset, has the largest DIFF value 0.178 and AUC score 0.627 for RSNA dataset, and obtains the second best AUC score 0.678 for ISIC2019. GANomaly performs better than DeepSVDD on IVC-filter and RSNA datasets with higher DIFF and AUC values, whereas DeepSVDD exceeds GANomaly on ISIC2019 dataset. Across the three datasets, f-AnoGAN generally outperforms GANomaly and its performance gradually improves as the training dataset becomes larger. Nevertheless, our model TENDs show certain advantages in acquiring better accuracy and exhibits competitive performances compared with other unsupervised models. Notably, we implement TEND with three different margins to show the difference with changing settings. By observing our results in Table 3, no unique margin in TEND provides the optimal result on all the datasets and thus it needs to be tuned for specific experiments. The effects of applying different radii are present in Sec. 4.5. The margin learner and the supervised model binary classifier are also discussed in ablation study (see Sec. 4.4).

4.3.2 *Rest-vs-one results*

To further compare the models' performances, the complementary experimental setting—rest-vs-one is implemented with the results reported in Table 4. Identical to the one-vs-rest experiments, we keep the tested models consistent and change the in-distribution class as OOD classes and the previous OOD data as our in-distribution data. The training and testing processes are the same as reported in Sec. 4.2. Our model TEND_150 gets the best DIFF 0.291 and AUC score 0.650 for IVC-filter dataset and obtains the suboptimal DIFF 0.126 and AUC score 0.584 for RSNA dataset. GANomaly performs the best for RSNA dataset. TEND_250

Table 3 FPR, TPR values, difference of TPR and FPR values, and AUC scores of various OOD detection methods trained on IVC-filter,[40] RSNA,[41] and ISIC2019[42] datasets with the one-vs-rest setting.

	IVC-filter				RSNA				ISIC2019			
Methods	↓FPR	↑TPR	↑DIFF	↑AUC	↓FPR	↑TPR	↑DIFF	↑AUC	↓FPR	↑TPR	↑DIFF	↑AUC
AE[11]	0.198 ± 0.104	0.350 ± 0.075	0.152 ± 0.067	0.436 ± 0.040	0.318 ± 0.014	0.461 ± 0.009	0.143 ± 0.010	0.566 ± 0.004	0.833 ± 0.060	0.186 ± 0.059	−0.648 ± 0.025	0.096 ± 0.003
AE_GMM	0.224 ± 0.138	0.153 ± 0.008	−0.071 ± 0.134	0.464 ± 0.067	0.496 ± 0.012	0.321 ± 0.003	−0.175 ± 0.013	0.412 ± 0.006	0.083 ± 0.006	0.211 ± 0.003	0.128 ± 0.006	0.564 ± 0.003
VAE[25]	0.489 ± 0.097	0.707 ± 0.076	0.218 ± 0.117	0.542 ± 0.080	0.473 ± 0.001	0.462 ± 0.001	−0.011 ± 0.012	0.487 ± 0.001	0.351 ± 0.011	0.395 ± 0.007	0.045 ± 0.007	0.471 ± 0.005
Margin learner	0.426 ± 0.099	0.549 ± 0.033	0.123 ± 0.098	0.568 ± 0.055	0.475 ± 0.016	0.478 ± 0.013	0.003 ± 0.010	0.491 ± 0.005	0.517 ± 0.020	0.584 ± 0.024	0.067 ± 0.010	0.530 ± 0.005
DeepSVDD[12]	0.503 ± 0.106	0.672 ± 0.042	0.170 ± 0.130	0.500 ± 0.075	0.508 ± 0.021	0.413 ± 0.023	−0.095 ± 0.015	0.421 ± 0.009	0.348 ± 0.021	0.621 ± 0.021	*0.273±0.006*	0.677 ± 0.003
GANomaly[31]	0.446 ± 0.172	0.627 ± 0.227	0.181 ± 0.200	0.518 ± 0.103	0.524 ± 0.005	0.678 ± 0.015	0.154 ± 0.009	0.576 ± 0.005	0.396 ± 0.030	0.481 ± 0.027	0.086 ± 0.012	0.551 ± 0.009
f-AnoGAN[33]	0.419 ± 0.077	0.511 ± 0.070	0.092 ± 0.045	0.544 ± 0.022	0.365 ± 0.033	0.541 ± 0.029	*0.176±0.008*	0.614 ± 0.005	0.366 ± 0.007	0.600 ± 0.007	0.234 ± 0.005	0.647 ± 0.003
TEND_150 (ours)	0.219 ± 0.077	0.749 ± 0.086	**0.531±0.071**	**0.772±0.030**	0.425 ± 0.029	0.590 ± 0.026	0.165 ± 0.010	*0.615±0.006*	0.377 ± 0.016	0.596 ± 0.015	0.220 ± 0.009	0.650 ± 0.006
TEND_250 (ours)	0.160 ± 0.091	0.684 ± 0.035	*0.524±0.082*	0.752 ± 0.051	0.389 ± 0.045	0.561 ± 0.043	0.172 ± 0.009	*0.615±0.006*	0.326 ± 0.017	0.669 ± 0.020	**0.343±0.011**	**0.717±0.006**
TEND_500 (ours)	0.122 ± 0.099	0.639 ± 0.095	0.517±0.042	*0.760±0.028*	0.438 ± 0.040	0.616 ± 0.041	**0.178±0.008**	**0.627±0.005**	0.351 ± 0.012	0.618 ± 0.011	0.268 ± 0.009	*0.678±0.006*
Binary classifier*	0.280 ± 0.006	0.847 ± 0.003	**0.567±0.006**	**0.853±0.003**	0.417 ± 0.007	0.589 ± 0.006	0.172 ± 0.008	0.593 ± 0.003	0.497 ± 0.023	0.340 ± 0.015	−0.157 ± 0.010	0.363 ± 0.004

Note: Bold numbers are the best results and italic numbers are the second best. Models with * are supervised and those without * are unsupervised.

Table 4 FPR, TPR values, difference of TPR and FPR values, and AUC scores of various OOD detection methods trained on IVC-filter,[40] RSNA,[41] and ISIC2019[42] datasets with the rest-vs-one setting.

	IVC-filter				RSNA				ISIC2019			
Methods	↓FPR	↑TPR	↑DIFF	↑AUC	↓FPR	↑TPR	↑DIFF	↑AUC	↓FPR	↑TPR	↑DIFF	↑AUC
AE[11]	0.706 ± 0.163	0.312 ± 0.159	−0.394 ± 0.059	0.165 ± 0.027	0.760 ± 0.022	0.544 ± 0.046	−0.216 ± 0.024	0.321 ± 0.005	0.593 ± 0.023	0.383 ± 0.024	−0.210 ± 0.012	0.353 ± 0.007
AE_GMM	0.728 ± 0.053	0.748 ± 0.022	0.020 ± 0.058	0.510 ± 0.029	0.600 ± 0.008	0.584 ± 0.004	−0.016 ± 0.009	0.492 ± 0.005	0.159 ± 0.005	0.059 ± 0.002	−0.100 ± 0.006	0.450 ± 0.003
VAE[25]	0.359 ± 0.082	0.464 ± 0.088	0.105 ± 0.063	0.560 ± 0.035	0.518 ± 0.029	0.453 ± 0.027	−0.065 ± 0.007	0.461 ± 0.036	0.518 ± 0.032	0.658 ± 0.045	0.140 ± 0.016	0.575 ± 0.005
Margin learner	0.617 ± 0.022	0.619 ± 0.045	0.003 ± 0.043	0.484 ± 0.025	0.510 ± 0.018	0.527 ± 0.016	0.017 ± 0.006	0.514 ± 0.004	0.510 ± 0.018	0.527 ± 0.016	0.017 ± 0.006	0.514 ± 0.004
DeepSVDD[12]	0.514 ± 0.043	0.475 ± 0.045	−0.039 ± 0.065	0.439 ± 0.043	0.514 ± 0.028	0.552 ± 0.032	0.038 ± 0.007	0.522 ± 0.004	0.530 ± 0.007	0.540 ± 0.013	0.010 ± 0.011	0.487 ± 0.007
GANomaly[31]	0.595 ± 0.060	0.622 ± 0.040	0.027 ± 0.051	0.449 ± 0.036	0.396 ± 0.014	0.638 ± 0.014	**0.242±0.004**	**0.656±0.003**	0.462 ± 0.0166	0.583 ± 0.019	0.121 ± 0.009	0.570 ± 0.005
f-AnoGAN[33]	0.419 ± 0.077	0.511 ± 0.070	0.092 ± 0.045	0.544 ± 0.022	0.295 ± 0.029	0.276 ± 0.012	−0.019 ± 0.019	0.406 ± 0.005	0.276 ± 0.004	0.677 ± 0.006	**0.401±0.007**	**0.718±0.004**
TEND_150 (ours)	0.359 ± 0.057	0.640 ± 0.031	**0.291±0.051**	**0.650±0.028**	0.452 ± 0.022	0.578 ± 0.024	*0.126±0.007*	*0.584±0.003*	0.336 ± 0.015	0.501 ± 0.006	0.164 ± 0.014	0.608 ± 0.007
TEND_250 (ours)	0.427 ± 0.061	0.582 ± 0.071	0.155 ± 0.058	*0.573±0.039*	0.492 ± 0.016	0.577 ± 0.015	0.084 ± 0.006	0.549 ± 0.004	0.386 ± 0.014	0.623 ± 0.011	*0.237±0.011*	*0.637±0.008*
TEND_500 (ours)	0.428 ± 0.069	0.584 ± 0.081	*0.156±0.038*	*0.573±0.025*	0.487 ± 0.015	0.550 ± 0.014	0.063 ± 0.008	0.541 ± 0.005	0.412 ± 0.018	0.533 ± 0.016	0.121 ± 0.013	0.582 ± 0.009
Binary classifier*	0.617 ± 0.022	0.619 ± 0.045	0.003 ± 0.043	0.484 ± 0.025	0.510 ± 0.018	0.527 ± 0.016	0.017 ± 0.006	0.514 ± 0.004	0.471 ± 0.014	0.599 ± 0.017	0.128 ± 0.005	0.584 ± 0.004

Note: Bold numbers are the best results and italic numbers are the second best. Models with * are supervised and those without * are unsupervised.

reaches the suboptimal results for ISIC2019 dataset, whereas f-AnoGAN can achieve the best. Generally, the detection of anomalies under rest-vs-one setting is more challenging than the one-vs-rest setting and nearly no model can work well for all the situations. Still, TEND has satisfactory performances across the three datasets with the rest-vs-one setting.

4.4 *Ablation Studies*

To further explore the effectiveness of each module in TEND, we perform the ablation studies with the settings of removing the binary classifier from TEND (margin learner) and training a supervised binary classifier (binary classifier), respectively. For the one-vs-rest setting, the results are shown as margin learner with radius setting 150 in Table 3, with slight DIFF and AUC improvements compared to the baseline AE on IVC-filter and ISIC2019 datasets. Comparatively, TEND_150 enlarges the DIFF with 0.379, 0.022, and 0.868 improvements and increases the AUC scores by 0.336, 0.049, and 0.554, respectively, on IVC-filter, RSNA, and ISIC2019 datasets. For the rest-vs-one setting, compared with the margin learner, TEND_150 achieves the DIFF with 0.288, 0.109, and 0.147 improvements for IVC-filter, RSNA, and ISIC2019 dataset, respectively, and enhances the AUC score with 0.166, 0.070, and 0.094 for the three datasets. These observations indicate the effectiveness of TEND's architecture.

We also report the performance of an AE extension, AE_GMM, which clusters the embeddings from the AE backbone and predicts the data classes—ID or OOD. From both Tables 3 and 4, a GMM head can improve the discriminative ability of AE to a certain extent; however, when testing on transformed OOD data in Tables 5 and 6, the advantages fail to remain. In comparison, TEND's heads on AE have more generalization ability and demonstrate consistent detection performance.

Instead of training the binary classifier of TEND model in an unsupervised fashion, we include partial true OOD data in training data. Since IVC-filter and ISIC2019 datasets have multiple classes, we randomly select 2-3 OOD classes for training and the left classes for validation.

One-vs-rest setting. For RSNA datasets, we use the class not normal (see Table 2 for details) for known OOD data and test the model on the left with opacity data. The supervised binary classifier is also evaluated with quantitative results appended in the end of Table 3. With prior knowledge about OOD data, the binary classifier can achieve very high AUC scores for IVC-filter (+0.081 compared to the best of unsupervised results). Nonetheless, this advantage fails to remain on other datasets, which indicates the benefits from prior knowledge are limited.

Rest-vs-one setting. For RSNA datasets, we use the class normal as known OOD data and not normal as ID data, the left class is used for evaluation. Different from the observation above, the corresponding results in Table 4 for binary classifier fail to exceed the unsupervised models, and more results can be observed in Table 6. In conclusion, the supervised binary classifier may lack generalization ability when dealing with unexpected data (refer Sec. 4.6 for more experimental results and discussions).

4.5 *Qualitative Results*

As our model TEND has a margin learner module (see the L_{mrg} part of Fig. 1) to enforce ID data inside of a predefined margin R (illustrated as the green dotted circle in Fig. 1) as to the voted center O (represented as the red star in Fig. 1) and OOD data outside of the region, we hereby visualize the data samples based on the obtained distance output by the margin learner. Take one-vs-rest setup results for illustration, the voted center O, whose calculation details were introduced in Sec. 3.3, is located at the origin of the 2D coordinate system. To visualize each data sample, we utilize their distance to the voted center O as their corresponding radius values to the origin. Each sample is represented by randomly picking one point along the circle that is defined with its corresponding radius. The x axis and y axis values help indicate how far the point is from the origin. Given an example with a distance value d_i, its corresponding coordinate (x_i, y_i) satisfies that $d_i^2 = x_i^2 + y_i^2$. The data samples with in-distribution labels are marked in green and the left data with OOD labels are in red. We draw the defined margin of the model with a blue circle for reference (refer to the Appendix code snippet for the visualization implementation details).

Table 5 Accuracy of various OOD detection methods trained on IVC-filter,[40] RSNA,[41] and ISIC2019[42] with the one-vs-rest setting.

	IVC-filter				RSNA				ISIC2019			
Methods	Random cut	Random crop and resize	Noise	Gaussian blur	Random cut	Random crop and resize	Noise	Gaussian blur	Random cut	Random crop and resize	Noise	Gaussian blur
AE[11]	**1.000±0.000**	0.371 ± 0.036	0.988 ± 0.007	0.064 ± 0.009	0.001 ± 0.000	0.029 ± 0.002	0.422 ± 0.004	0.000 ± 0.000	0.252 ± 0.004	0.581 ± 0.005	0.428 ± 0.004	0.187 ± 0.002
AE_GMM	0.110 ± 0.001	0.151 ± 0.000	0.142 ± 0.001	0.142 ± 0.001	0.660 ± 0.003	0.023 ± 0.001	0.577 ± 0.007	0.402 ± 0.007	0.055 ± 0.001	0.028 ± 0.001	0.086 ± 0.002	0.087 ± 0.002
VAE[25]	0.013 ± 0.006	0.137 ± 0.031	0.020 ± 0.013	0.008 ± 0.007	0.990 ± 0.001	0.288 ± 0.004	0.438 ± 0.005	0.424 ± 0.005	0.027 ± 0.001	0.241 ± 0.004	0.434 ± 0.003	0.364 ± 0.004
DeepSVDD[12]	**1.000±0.000**	0.735 ± 0.039	0.607 ± 0.024	0.044 ± 0.018	0.604 ± 0.003	0.120 ± 0.005	0.642 ± 0.006	0.455 ± 0.005	0.985 ± 0.001	0.740 ± 0.003	0.567 ± 0.003	0.190 ± 0.004
GANomaly[31]	**1.000±0.000**	0.792 ± 0.017	0.727 ± 0.030	0.690 ± 0.031	0.959 ± 0.003	0.910 ± 0.003	0.330 ± 0.005	0.313 ± 0.005	0.919 ± 0.003	0.608 ± 0.005	0.306 ± 0.002	0.348 ± 0.003
f-AnoGAN[33]	0.888 ± 0.024	0.699 ± 0.034	0.583 ± 0.035	0.501 ± 0.052	0.726 ± 0.005	0.729 ± 0.007	0.386 ± 0.003	0.413 ± 0.005	0.665 ± 0.007	0.431 ± 0.004	0.410 ± 0.005	0.391 ± 0.004
TEND_150 (ours)	0.951 ± 0.007	0.988 ± 0.006	0.921 ± 0.017	**1.000±0.000**	**1.000±0.000**	**1.000±0.000**	**1.000±0.000**	**1.000±0.000**	**1.000±0.000**	**1.000±0.000**	**0.997±0.000**	**0.997±0.000**
TEND_250 (ours)	**1.000±0.000**	**1.000±0.000**	**1.000±0.000**	**1.000±0.000**	**1.000±0.000**	**1.000±0.000**	**1.000±0.000**	**1.000±0.000**	0.996 ± 0.001	0.942 ± 0.003	0.799 ± 0.005	0.741 ± 0.005
TEND_500 (ours)	0.752 ± 0.026	0.861 ± 0.026	0.797 ± 0.029	0.984 ± 0.008	**1.000±0.000**	**1.000±0.000**	**1.000±0.000**	**1.000±0.000**	0.950 ± 0.002	0.976 ± 0.001	0.905 ± 0.002	0.905 ± 0.003
Binary classifier*	0.963 ± 0.003	0.963 ± 0.005	0.509 ± 0.001	0.899 ± 0.001	0.499 ± 0.006	0.680 ± 0.003	0.281 ± 0.004	0.215 ± 0.004	0.271 ± 0.006	0.762 ± 0.004	0.498 ± 0.005	0.491 ± 0.006

Note: Bold denotes the best results and * indicates the model is supervised.

Table 6 Accuracy of various OOD detection methods trained on IVC-filter,[40] RSNA,[41] and ISIC2019[42] with the rest-vs-one setting.

	IVC-filter				RSNA				ISIC2019			
Methods	Random cut	Random crop and resize	Noise	Gaussian blur	Random cut	Random crop and resize	Noise	Gaussian blur	Random cut	Random crop and resize	Noise	Gaussian blur
AE[11]	**1.000±0.000**	0.116 ± 0.009	0.627 ± 0.014	0.032 ± 0.005	0.999 ± 0.003	0.705 ± 0.004	0.901 ± 0.002	0.001 ± 0.000	0.782 ± 0.004	0.250 ± 0.005	0.388 ± 0.004	0.368 ± 0.004
AE_GMM	0.131 ± 0.010	0.206 ± 0.011	0.212 ± 0.010	0.220 ± 0.010	0.361 ± 0.003	0.319 ± 0.004	0.383 ± 0.005	0.396 ± 0.005	0.067 ± 0.002	0.054 ± 0.002	0.158 ± 0.003	0.157 ± 0.003
VAE[25]	0.036 ± 0.002	0.460 ± 0.008	0.476 ± 0.011	0.487 ± 0.010	0.188 ± 0.002	0.849 ± 0.003	0.603 ± 0.005	0.596 ± 0.004	0.174 ± 0.003	0.627 ± 0.006	0.555 ± 0.007	0.544 ± 0.007
DeepSVDD[12]	0.858 ± 0.011	0.529 ± 0.006	0.495 ± 0.008	0.496 ± 0.008	0.905 ± 0.001	0.415 ± 0.004	0.494 ± 0.004	0.425 ± 0.003	0.827 ± 0.003	0.294 ± 0.004	0.524 ± 0.005	0.541 ± 0.005
GANomaly[31]	0.785 ± 0.008	0.583 ± 0.009	0.577 ± 0.013	0.629 ± 0.009	0.999 ± 0.000	0.682 ± 0.003	0.792 ± 0.003	0.238 ± 0.005	0.979 ± 0.001	0.694 ± 0.003	0.464 ± 0.004	0.476 ± 0.004
f-AnoGAN[33]	0.934 ± 0.008	0.594 ± 0.013	0.361 ± 0.014	0.344 ± 0.012	0.380 ± 0.004	0.373 ± 0.004	0.716 ± 0.003	0.300 ± 0.004	0.989 ± 0.001	0.825 ± 0.002	0.460 ± 0.005	0.464 ± 0.006
TEND_150 (ours)	**1.000±0.000**	**1.000±0.000**	**1.000±0.000**	**1.000±0.000**	**1.000±0.000**	**1.000±0.000**	**1.000±0.000**	**1.000±0.000**	**1.000±0.000**	**1.000±0.000**	**1.000±0.000**	**1.000±0.000**
TEND_250 (ours)	**1.000±0.000**	**1.000±0.000**	**1.000±0.000**	**1.000±0.000**	**1.000±0.000**	**1.000±0.000**	**1.000±0.000**	**1.000±0.000**	0.995 ± 0.001	**1.000±0.000**	0.998 ± 0.000	0.997 ± 0.001
TEND_500 (ours)	**1.000±0.000**	0.997 ± 0.002	0.999 ± 0.001	**1.000±0.000**	**1.000±0.000**	**1.000±0.000**	**1.000±0.000**	**1.000±0.000**	0.984 ± 0.001	0.941 ± 0.002	0.902 ± 0.004	0.760 ± 0.002
Binary classifier*	0.025 ± 0.005	0.796 ± 0.010	0.659 ± 0.009	0.644 ± 0.012	0.927 ± 0.002	0.972 ± 0.001	0.984 ± 0.001	0.816 ± 0.003	0.100 ± 0.003	0.849 ± 0.003	0.470 ± 0.003	0.477 ± 0.003

Note: Bold denotes the best results and * indicates the model is supervised.

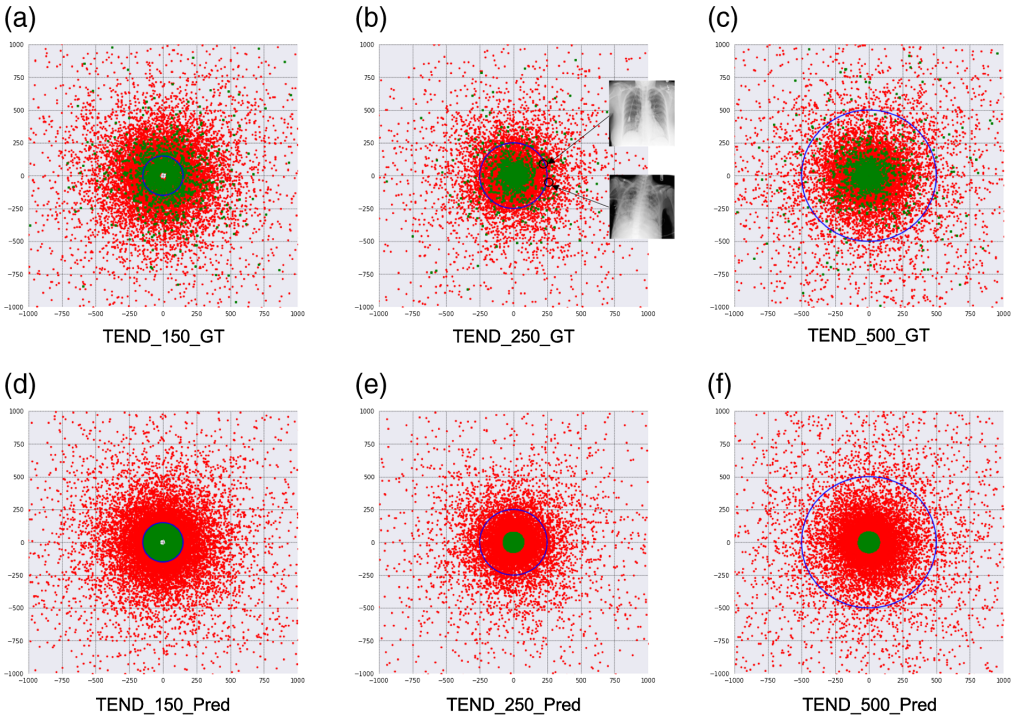

Fig. 3 2D visualization of ID (green points) and OOD (red points) data distance distributions for RSNA dataset learnt by TEND's margin learner module with radius: (a), (d) 150; (b), (e) 250; and (c), (f) 500 under the one-vs-rest setting. (a)–(c) Distance distribution with ground-truth labels and (d)–(f) the predicted results with the optimal threshold values. Blue circles are plotted based on the radius in each subfigure for reference.

Take RSNA dataset for example, in Fig. 3, the voted center O is represented by the point with coordinates (0, 0) and the area defined by radius R is present with the plotted blue circles in each subfigure. For better visualization and comparison, each subfigure has both the x axis and y axis ranging from -1000 to 1000, those data points that have larger distance out of range will be ignored. Figures 3(a)–(c) show the distance distribution of data with ground-truth labels [i.e., ID (in green) and OOD (in red)] learnt by TEND with radius of (a), (d) 150, (b), (e) 250, and (c), (f) 500, whereas (d)–(f) indicate the predictions after thresholding, with the green points for samples predicted as ID and red points for samples predicted as OOD. To help inspect the data points around the boundary, two cases based on the ground-truth information are illustrated for TEND_250_GT, with the upper one as an ID data and the lower case for OOD class. From Figs. 3(a)–(c), the learnt distance distributions for ID and OOD data are similar for TEND with different radii values. But the ID data can be outside the circle with radius 150 [Fig. 3(a)] and will be inside the circle regions with radius 250 [Fig. 3(b)] and 500 [Fig. 3(c)], which suggests that when using larger margin to divide ID and OOD data, ID samples will be easier to be included while more OOD data will be inside the region, leading to more false positive predictions. Therefore, it is not the larger the margin, the better the performance is. After having the distance values predicted by the margin learner module, we apply the Gmeans method to find the optimal threshold considering both the distance predictions and the binary possibility. Figures 3(d)–(f) illustrate the ID and OOD predictions of TEND after thresholding. We can see that the boundary of predicted ID data samples is very close to the margin circle of radius 150 [Fig. 3(d)], but much smaller compared to radius 250 [Fig. 3(e)] and 500 [Fig. 3(f)]. As they are in the same scale, we can observe that the thresholding areas for ID are smaller when the margin values increase.

To further analyze the OOD detection ability of TEND, we take the RSNA dataset for example and inspect part of the predictions. As shown in Fig. 4, four kinds of predictions, namely true positive (TP), true negative (TN), false positive (FP), and false negative (FN) predictions,

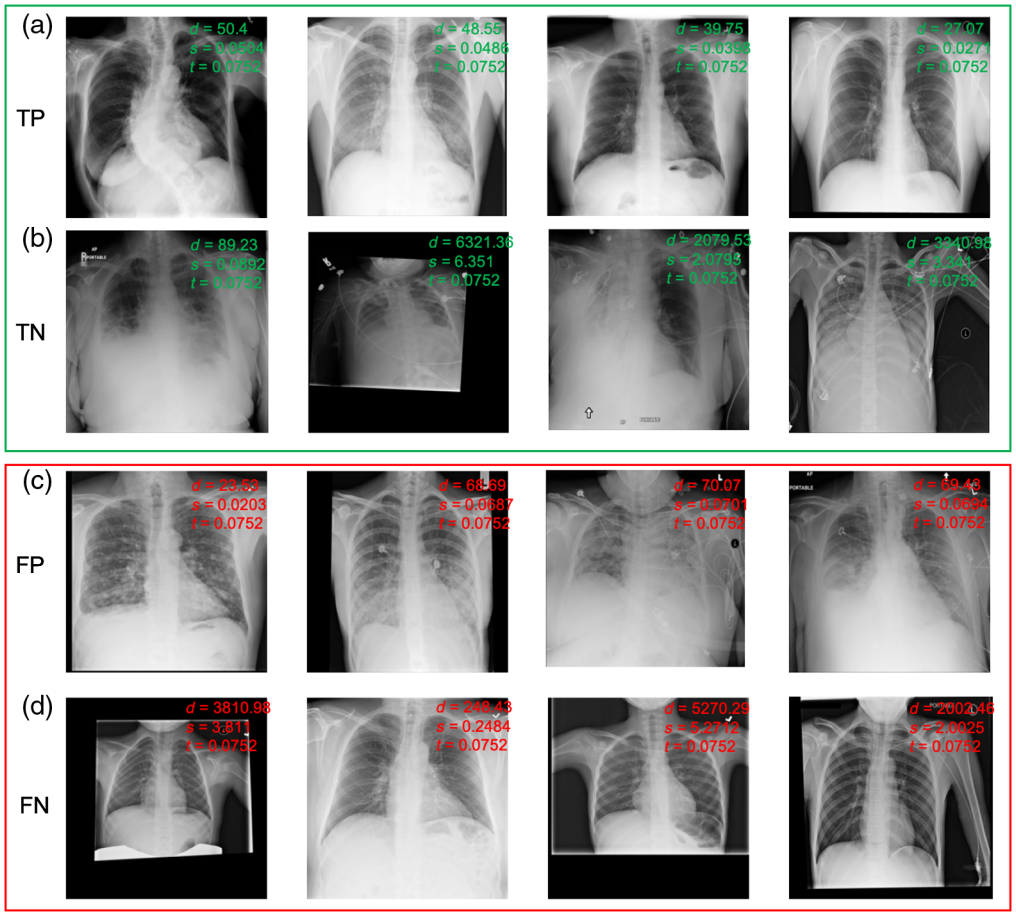

Fig. 4 (a) TP, (b) TN, (c) FP, and (d) FN predictions of TEND_500 on RSNA datasets following the one-vs-rest setting. *d*, distance value from the margin learner module; *p*, probability outputted by the binary discriminator module; *s*, final score; and *t*, optimal threshold (ID: $s < t$, OOD: $s \geq t$).

predicted by TEND_500 are present, with four representative cases for each situation. TP means that true ID samples are correctly identified and TN is for correct identification of OOD samples. FP refers to the OOD data that is misclassified as ID data and FN stands for wrongly classified OOD data. From Fig. 3, data points close to the center are more confident of being in the ID category, which means the smaller the distance is, the higher possibility of the data being an ID sample is. Observing the TP cases in Fig. 4, most of them are with distance values <50, which is relatively small compared to the predefined margin 500. whereas the TN cases are often with larger distances. The first chest x-ray image of TN cases has a final score 0.0892, close to the threshold 0.0752, which indicates this case is a challenging case. Figure 4(c) shows the hard FP cases for TEND_500 to identify as they are all with both small distance values and probabilities. The FN cases shown in Fig. 4(d) can be those ID data with irregular format or position shifting. With imperfections, TEND_500 will treat them as outliers and assign larger distance values by the margin learner module. Compared with others, the second FP case is much more challenging as the data are inside the predefined margin but classified wrongly due to the threshold setting. We also present the 2D distance visualization and detection results with examples for ISIC2019 datasets in Figs. 5 and 7 and IVC-filter in Figs. 6 and 8 respectively (see Appendix).

4.6 *Effects of Transformations*

To further compare the intraclass OOD detection ability, we generate validation data by applying four unseen transformations to all the ID data defined in Sec. 3.2 and shown in the right yellow

box in Fig. 2. As we have two experimental settings—the one-vs-rest and the rest-vs-one, we report them in Tables 5 and 6, respectively. The best and the second best accuracy results are bolded and italics, respectively. As all the validation data are in OOD category, we calculate the OOD detection accuracy based on the optimal threshold t determined in Table 3 (corresponding to Table 5) and Table 4 (corresponding to Table 6) for each model and each dataset. Those data with score $s \geq t$ are labeled as OOD (which are true negative samples, TN in short) and the data having score $s < t$ are classified as ID class (which are false positive samples, FP in short). Accordingly, the detection accuracy is formulated as $\mathrm{ACC}_{\mathrm{val}} = \mathrm{TN}/(\mathrm{TN} + \mathrm{FP})$.

4.6.1 *One-vs-rest results of transformations*

Table 5 shows the accuracy of detecting the generated validation OOD data with different models with the one-vs-rest experimental setting. Among all the models present in Table 5, the AE,[11] VAE,[25] DeepSVDD,[12] GANomaly,[31] f-AnoGAN,[33] and our TENDs are all unsupervised methods, whereas the binary classifier marked with an asterisk is a supervised model that is trained with both ID data and partial true OOD data. Random Cut is relatively easy to distinguish compared to other transformations as multiple methods including DeepSVDD, GANomaly, and f-AnoGAN can detect most of them all for the three datasets. In contrast, the random crop and resize, noise, and Gaussian Blur transformations are much more difficult for them to handle. Nonetheless, TEND architectures with different margins nearly achieve all the best and the second best accuracy for the test datasets. In summary, although TEND is an unsupervised model, it can still obtain stronger intraclass OOD identification ability and even outperform other state-of-the-art models and the supervised model binary classifier on both IVC-filter and RSNA datasets. This advantage is due to the benefits of transformations during training.

4.6.2 *Rest-vs-one results of transformations*

Table 6 presents the accuracy of detecting the generated validation OOD data with different models following the rest-vs-one experimental setting. AE partially retains its sensitivity in random cut and noise transformations for both IVC-filter and RSNA datasets. In general, VAE shows little advantages in transformed OOD detection except for the noise and gaussian blur OOD detection for ISIC2019 dataset. DeepSVDD, GANomaly, f-AnoGAN occasionally show advanced performance for different situations. Comparatively, TENDs show more stable results in accurate detection of the transformed OOD data, especially for both IVC-filter and RSNA datasets. This stability for such intraclass OOD detection benefits from the learning process of training with transformation.

5 Discussion and Limitations

We implement TEND with three different margins and show our results across various medical datasets under different settings. Although our models show competitive performance and surpass other methods under certain situations, the margin parameter has to be tuned for specific usages. Depending on the data complexity and variance across classes of a dataset, 250 is a good starting point. The ability of separating OOD from ID does not always improve as the margin increases due to the data complexity. For datasets with clear class variations, the margin can be set larger accordingly and vice versa. In addition, TEND utilizes transformation to generate fake OOD samples for discriminative learning. Due to the large amount of possibilities, this work only exploits a limited number of possible transformations.

6 Conclusion

In this paper, we introduced an unsupervised novelty detector—TEND, which can detect intraclass OOD data for medical applications in an open-world environment. TEND is a two-stage anomaly detector with a vanilla AE trained on in-distribution data in the first stage to serve as

feature extractors in the second stage and two modules—a margin learner module and a binary discriminator module—jointly trained in the second stage for separating in-distribution inputs from the non-linearly transformed counterparts. With no OOD data used in training, TEND is able to learn nuances from intraclass variations in medical image analysis problems and provide a stepping stone for developing rare disease diagnosis models with no sample images. Extensive results with the one-vs-rest and rest-vs-one experimental settings on multiple public medical image datasets demonstrate the effectiveness of our model. More general evaluations on data with unseen transformations further evince our model's generalization ability and robustness. In summary, an efficient novelty detection method for medical images has been developed that can be applied to discover unknown classes with only predefined normal data. We plan to extend this work by integrating TEND into real time imaging pipelines for inference of medical imaging models.

7 Appendix

Below is the code for plotting the 2D visualization figure of the data samples according to obtained distances.

Table 7

```
import numpy as np
import matplotlib as mpl
import matplotlib.pyplot as plt
mpl.style.use('seaborn')
def generate_point(R):
  # given a radius R, generate the coordinates x, y
  # of a random point in the cricle
  theta = np.random.uniform(0, np.pi * 2)
  x = R * np.cos(theta)
  y = R * np.sin(theta)
  return x, y
def plot_point(anomaly_score, dist, threshold, R, K = 1000):
  # anomaly_score: list of anomaly scores, (N,)
  # dist: list of distance values output by the MRG part of TEND, (N,)
  # threshold: anomaly score threshold, a float number,
  # samples with anomaly score smaller than the threshold are classified as ID,
  # equal to or greater than the threshold are in OOD category
  # K: float number, deciding the x-axis and y-axis range for showing data
  Xs = [], Ys = [], Xs2 = [], Ys2 = []
for i in range(0, len(anomaly_score)):
  x, y = generate_point(dist[i])
  if anomaly_score[i] < threshold:
    Xs.append(x)
```

Table 7 (*Continued*).

```
    Ys.append(y)
  else:
    Xs2.append(x)
    Ys2.append(y)
fig = plt.figure(figsize=(8,8))
plt.scatter(Xs, Ys, c ="green", linewidths = 2, marker ="s", s = 2)
plt.scatter(Xs2, Ys2, c ="red", linewidths = 2, marker ="o", s = 2)
plt.xlim([-k, k])
plt.ylim([-k, k])
theta = np.linspace(0 , 2 * np.pi , 300)
plt.grid(color = 'black', linestyle = '-', linewidth = 0.5)
plt.plot(R * np.cos(theta), R * np.sin(theta), color='blue')
return fig
```

Here we show more 2D visualizations of ID and OOD data distance distribution for ISIC2019 dataset in Fig. 5 and IVC-filter dataset in Fig. 6. Different predictions including TP, TN, FP, FN are also present with the examples of RSNA dataset in Fig. 7 and IVC-filter dataset in Fig. 8. Due to the limited FN predictions of IVC-filter dataset, only one FP case is reported.

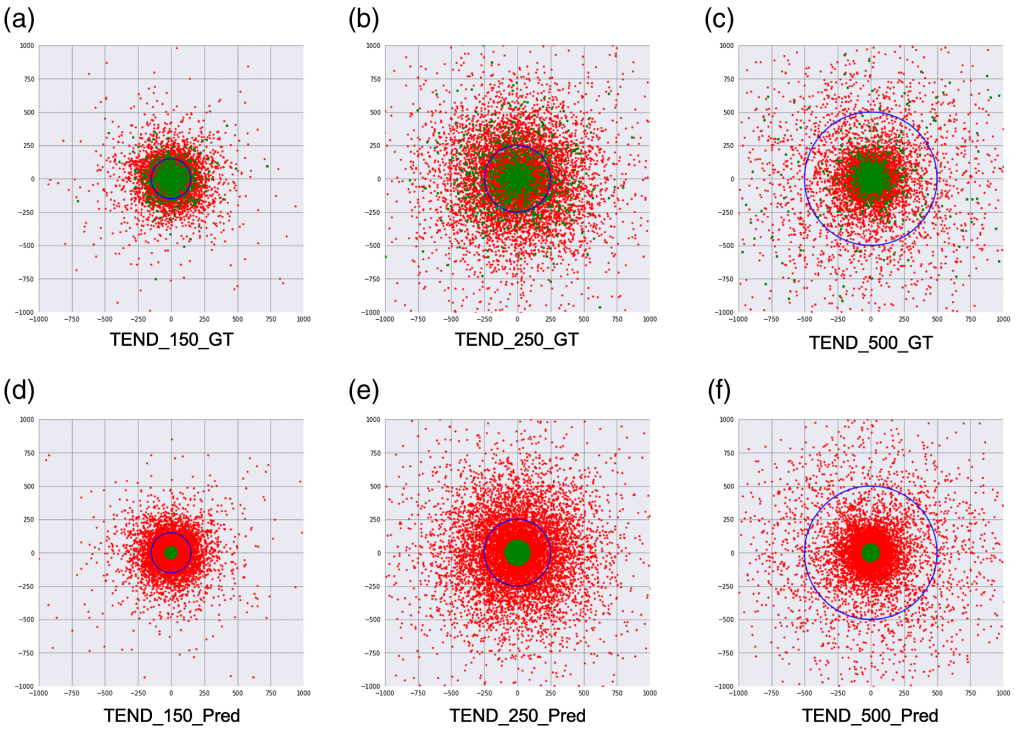

Fig. 5 2D visualization of ID (green points) and OOD (red points) data distance distributions for ISIC2019 dataset learnt by TEND's margin learner module with radius (a), (d) 150; (b), (e) 250; and (c), (f) 500 following the one-vs-rest setting. (a)–(c) Distance distribution with ground-truth labels and (d)–(f) the predicted results with the optimal threshold values. Blue circles are the plotted based on the radius in each subfigure for reference.

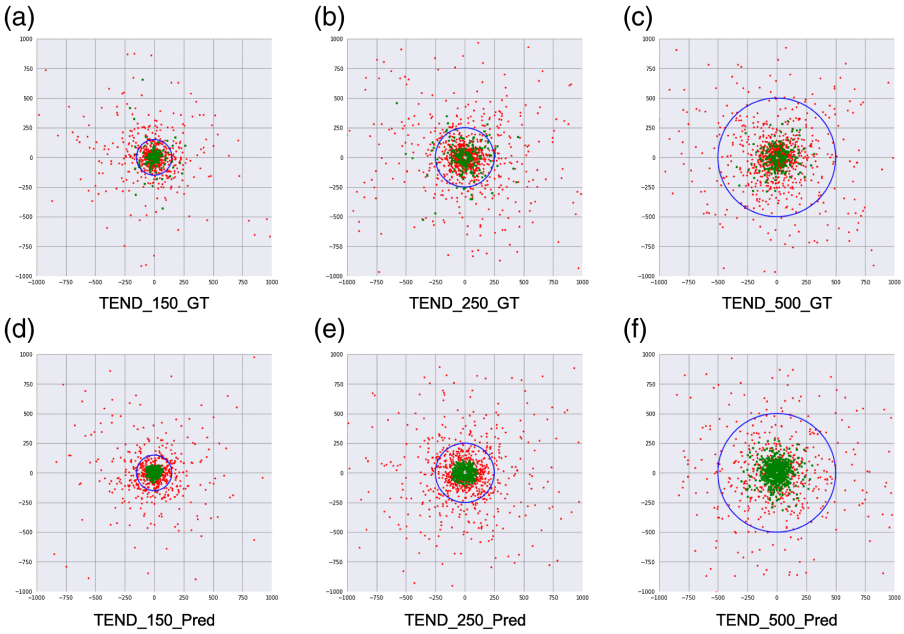

Fig. 6 2D visualization of ID (green points) and OOD (red points) data distance distributions for IVC-filter dataset learnt by TEND's margin learner module with radius (a), (d) 150; (b), (e) 250; and (c), (f) 500 under the one-vs-rest setting. (a)–(c) Distance distribution with ground-truth labels and (d)–(f) the predicted results with the optimal threshold values. Blue circles are the plotted based on the radius in each subfigure for reference.

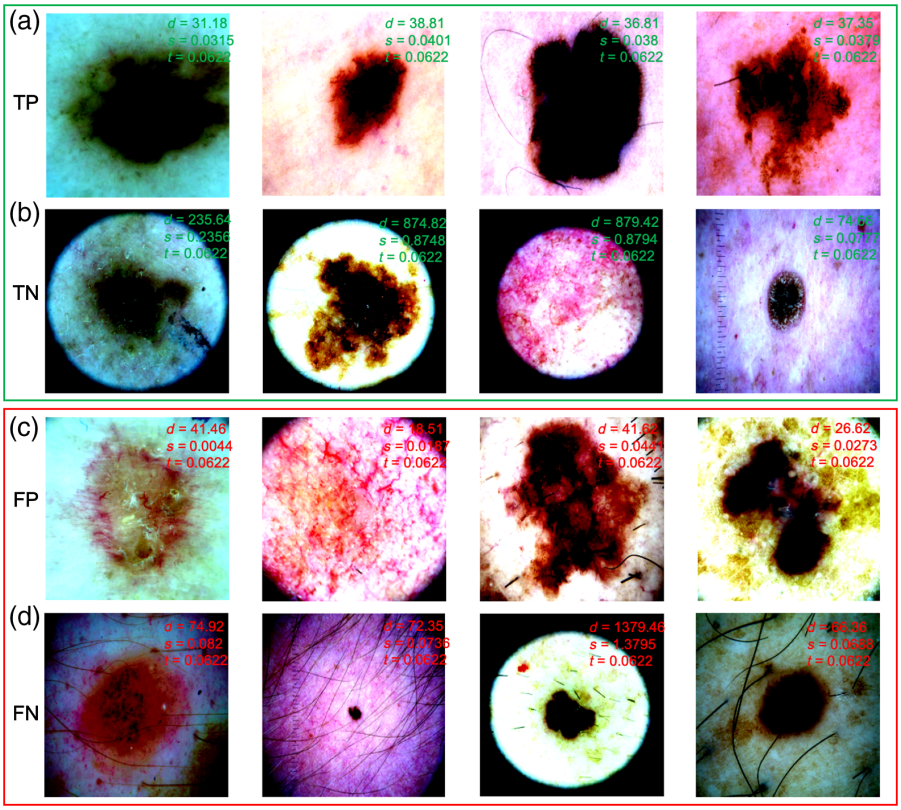

Fig. 7 (a) TP, (b) TN, (c) FP, and (d) FN predictions of TEND_500 on ISIC2019 datasets with the one-vs-rest setting. d, distance value from the margin learner module; p, probability outputted by the binary discriminator module; s, final score; and t, optimal threshold (ID: $s < t$, OOD: $s \geq t$).

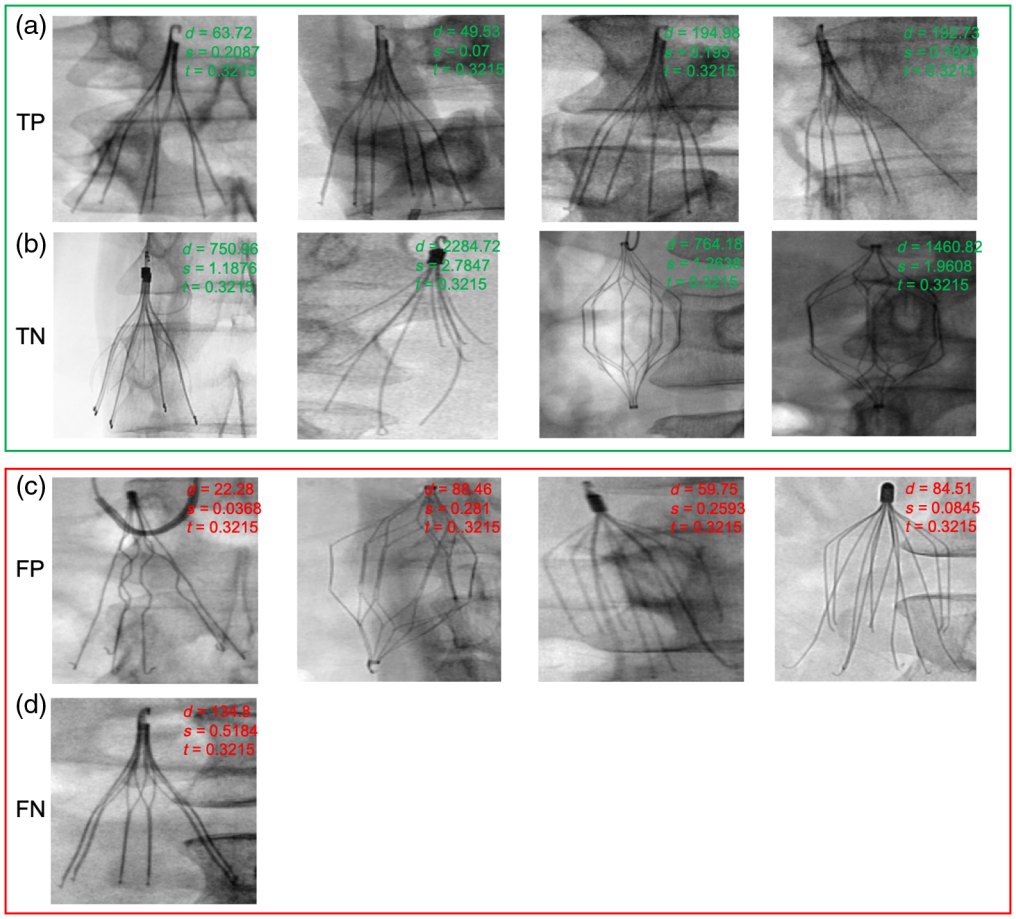

Fig. 8 (a) TP, (b) TN, (c) FP, and (d) FN predictions of TEND_500 on IVC-filter datasets following the one-vs-rest setting. d, distance value from the margin learner module; p, probability outputted by the binary discriminator module; s, final score; and t, optimal threshold (ID: $s < t$, OOD: $s \geq t$).

Disclosures

No conflicts of interests, financial or otherwise, are declared by the authors.

Acknowledgments

The work was supported by the National Institute of Biomedical Imaging and Bioengineering MIDRC grant of the National Institutes of Health under Contract Nos. 75N92020C00008 and 75N92020C00021 and the US National Science Foundation (No. #1928481) from the Division of Electrical, Communication, and Cyber Systems.

References

1. G. Litjens et al., "A survey on deep learning in medical image analysis," *Med. Image Anal.* **42**, 60–88 (2017).
2. V. Sehwag et al., "Analyzing the robustness of open-world machine learning," in *Proc. 12th ACM Workshop on Artif. Intell. and Security*, pp. 105–116 (2019).
3. J. Ren et al., "Likelihood ratios for out-of-distribution detection," in *Adv. Neural Inf. Process. Syst.*, pp. 14707–14718 (2019).
4. V. Sehwag et al., "Analyzing the robustness of open-world machine learning," in *Proc. 12th ACM Workshop Artif. Intell. and Secur.*, pp. 105–116 (2019).

5. P. Schlachter, Y. Liao, and B. Yang, "Deep one-class classification using intra-class splitting," in *IEEE Data Science Workshop (DSW)*, pp. 100–104 (2019).
6. P. Liznerski et al., "Explainable deep one-class classification," in *Int. Conf. Learn. Represent.* (2021).
7. J. Tack et al., "CSI: novelty detection via contrastive learning on distributionally shifted instances," in *Adv. Neural Inf. Process. Syst.* (2020).
8. L. Deng and C. Cortes, "The MNIST database of handwritten digit images for machine learning research," *IEEE Sig. Process. Magazine* **29**(6), 141–142 (2012).
9. A. Krizhevsky, V. Nair, and G. Hinton, "CIFAR-10 (Canadian Institute for Advanced Research)," 2020, http://www.cs.toronto.edu/kriz/cifar.html5.
10. J. Deng et al., "Imagenet: a large-scale hierarchical image database," in *IEEE Conf. Comput. Vision and Pattern Recognit.*, IEEE, pp. 248–255 (2009).
11. R. J. W. David Rumelhart and G. Hinton, *Parallel Distributed Processing Explorations in the Microstructure of Cognition*, Vol. **1**, MIT Press, Cambridge, Massachusetts (1986).
12. L. Ruff et al., "Deep one-class classification," in *Int. Conf. Mach. Learn.*, pp. 4393–4402 (2018).
13. R. Chalapathy and S. Chawla, "Deep learning for anomaly detection: a review," *ACM Computing Surveys (CSUR),* **54**(2), 1–38 (2021).
14. K. Lee et al., "A simple unified framework for detecting out-of-distribution samples and adversarial attacks," in *Adv. Neural Inf. Process. Syst.*, Vol. 31, pp. 7167–7177 (2018).
15. Y. Ouyang and V. Sanchez, "Video anomaly detection by estimating likelihood of representations," in *25th Int. Conf. Pattern Recognit. (ICPR)*, IEEE, pp. 8984–8991 (2021).
16. D. Hendrycks, M. Mazeika, and T. Dietterich, "Deep anomaly detection with outlier exposure," in *Int. Conf. Learn. Represent.* (2018).
17. A. TVyas et al., "Out-of-distribution detection using an ensemble of self supervised leave-out classifiers," in *Proc. Eur. Conf. Comput. Vision (ECCV)* (2018).
18. R. Wang et al., "Deep learning for anomaly detection," in *Proc. 13th Int. Conf. Web Search and Data Mining*, pp. 894–896 (2020).
19. M. Sakurada and T. Yairi, "Anomaly detection using autoencoders with nonlinear dimensionality reduction," in *Proc. MLSDA 2014 2nd Workshop Mach. Learn. for Sens. Data Anal.*, pp. 4–11 (2014).
20. C. Zhou and R. C. Paffenroth, "Anomaly detection with robust deep autoencoders," in *Proc. 23rd ACM SIGKDD Int. Conf. Knowl. Discov. and Data Mining*, pp. 665–674 (2017).
21. L. Beggel, M. Pfeiffer, and B. Bischl, "Robust anomaly detection in images using adversarial autoencoders," in *Joint Eur. Conf. Mach. Learn. and Knowl. Discov. in Databases*, Springer, pp. 206–222 (2019).
22. T. Tagawa, Y. Tadokoro, and T. Yairi, "Structured denoising autoencoder for fault detection and analysis," in *Asian Conf. Mach. Learn.*, PMLR, pp. 96–111 (2015).
23. A. A. Pol et al., "Anomaly detection with conditional variational autoencoders," in *18th IEEE Int. Conf. Mach. Learn. Appl. (ICMLA)*, IEEE, pp. 1651–1657 (2019).
24. R. Yao et al., "Unsupervised anomaly detection using variational auto-encoder based feature extraction," in *IEEE Int. Conf. Prognostics and Health Management (ICPHM)*, *IEEE*, pp. 1–7 (2019).
25. A. A. Pol et al., "Anomaly detection with conditional variational autoencoders," in *18th IEEE iInt. Conf. Mach. Learn. Appl. (ICMLA)*, pp. 1651–1657 (2019).
26. I. Bozcan and E. Kayacan, "UAV-AdNet: unsupervised anomaly detection using deep neural networks for aerial surveillance," in *IEEE/RSJ Int. Conf. Intell. Rob. and Syst. (IROS)*, IEEE, pp. 1158–1164 (2020).
27. I. Goodfellow et al., "Generative adversarial networks," *Commun. ACM* **63**(11), 139–144 (2020).
28. H. Zenati et al., "Adversarially learned anomaly detection," in *IEEE Int. Conf. Data Mining (ICDM)*, IEEE, pp. 727–736 (2018).
29. P. Perera, R. Nallapati, and B. Xiang, "OCGAN: one-class novelty detection using GANs with constrained latent representations," in *Proc. IEEE Conf. Comput. Vision and Pattern Recognit.*, pp. 2898–2906 (2019).

30. J. Donahue, P. Krähenbühl, and T. Darrell, "Adversarial feature learning," in *Int. Conf. Learn. Represent. (ICLR)* (2017).
31. S. Akcay, A. Atapour-Abarghouei, and T. P. Breckon, "Ganomaly: semi-supervised anomaly detection via adversarial training," in *Asian Conf. Comput. Vision*, Springer, pp. 622–637 (2018).
32. T. Schlegl et al., "Unsupervised anomaly detection with generative adversarial networks to guide marker discovery," *Lect. Notes Comput. Sci.* **10265**, 146–157 (2017).
33. T. Schlegl et al., "f-AnoGAN: fast unsupervised anomaly detection with generative adversarial networks," *Med. Image Anal.* **54**, 30–44 (2019).
34. S. S. Khan and M. G. Madden, "One-class classification: taxonomy of study and review of techniques," *Knowl. Eng. Rev.* **29**(3), 345–374 (2014).
35. B. Schölkopf et al., "Estimating the support of a high-dimensional distribution," *Neural Comput.* **13**(7), 1443–1471 (2001).
36. P. Perera and V. M. Patel, "Learning deep features for one-class classification," *IEEE Trans. Image Process.* **28**(11), 5450–5463 (2019).
37. S. Liang, Y. Li, and R. Srikant, "Enhancing the reliability of out-of-distribution image detection in neural networks," in *Int. Conf. Learn. Represent. (ICLR)* (2018).
38. T. Chen et al., "A simple framework for contrastive learning of visual representations," in *Int. Conf. Mach. Learn.*, PMLR, pp. 1597–1607 (2020).
39. Q. Yu and K. Aizawa, "Unsupervised out-of-distribution detection by maximum classifier discrepancy," in *Proc. IEEE/CVF Int. Conf. Comput. Vision*, pp. 9518–9526 (2019).
40. J. C. Ni et al., "Deep learning for automated classification of inferior vena cava filter types on radiographs," *J. Vasc. Interv. Radiol.* **31**(1), 66–73 (2020).
41. X. Wang et al., "Chestx-ray8: hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases," in *Proc. IEEE Conf. Comput. Vision and Pattern Recognit.*, pp. 2097–2106 (2017).
42. N. C. Codella et al., "Skin lesion analysis toward melanoma detection: a challenge at the 2017 international symposium on biomedical imaging (ISBI), hosted by the international skin imaging collaboration (ISIC)," in *IEEE 15th Int. Symp. Biomed. Imaging (ISBI 2018)*, IEEE, pp. 168–172 (2018).

Xiaoyuan Guo is a computer science PhD student at Emory University. Her primary research interests are computer vision and medical image processing, especially improving medical image segmentation, classification, and object detection accuracy with mainstream computer vision techniques.

Judy W. Gichoya is an assistant professor in the Department of Radiology and Imaging Sciences at Emory University School of Medicine. She is also a member of the Cancer Prevention and Control Research Program at Winship Cancer Institute. She holds professional memberships with Radiological Society of North America, American College of Radiology, Society of Interventional Radiology, Society of Imaging Informatics in Medicine, and American Medical Informatics Association.

Saptarshi Purkayastha is an assistant professor working on data science and health informatics. He is also the director of undergraduate education and research in the Department of BioHealth Informatics at IUPUI. He currently is investigating methods for improving engagement in online education, using guided inquiry learning in the study of health information management.

Imon Banerjee is an associate faculty at Mayo Clinic and associate faculty at Arizona State University. Her current research is focused on unstructured medical data analysis and integration of multisource medical data from varying hospital systems for building predictive models.