Learnable Pretext Task for Anomaly Detection

Loïc JÉZÉQUEL[1,2], Ngoc-Son VU[1], Jean BEAUDET[2], Aymeric HISTACE[1]

[1]ETIS UMR 8051 (CY Cergy Paris Université, ENSEA, CNRS) F-95000

[2]Idemia Identity & Security, 95520 Osny France

Résumé – La détection d'anomalie profonde est devenue une solution attrayante dans de multiples domaines et a connu récemment de nombreux développements. L'une des pistes les plus prometteuses est l'utilisation de tâches prétextes. Cependant, celles-ci sont limitées par l'absence d'échantillons anormaux et comportent un biais inductif important. Pour palier ces limitations, nous introduisons le concept de tâches prétextes apprenables, où la tâche prétexte est aussi apprise afin de réussir sur les échantillons normaux tout en échouant sur les anomalies. En appliquant le concept de tâche apprenable sur une tâche de reconnaissance de TPS, notre méthode discrimine mieux les anomalies aux cas limite et améliore considérablement les performances globales. Celle-ci surpasse l'état de l'art avec une réduction d'erreur relative jusqu'à 49% sur divers problèmes de détection d'anomalies.

Abstract – Deep anomaly detection has become an appealing solution in many fields, and has seen many recent developments. One of the most promising avenues is the use of pretext tasks. However these are limited by the lack of anomalous samples and carries an important inductive bias. To this end, we introduce the concept of learnable pretext tasks, where a pretext task itself is learned to succeed on normal samples while failing on anomalies. By applying the learnable task on a thin plate transform recognition task, our method helps discriminating harder edge-case anomalies and greatly improves anomaly detection. It outperforms state-of-the-art with up to 49% relative error reduction measured with AUROC on various anomaly detection problems.

1 Introduction

Anomaly detection is at the core of many modern machine learning applications. It is often appealing in situations where anomalies are usually expensive to obtain, and where robustness is critical. To name a few, in fraud detection [3], medical imaging [14], video surveillance [17] or manufacturing defect detection [24].

Detecting anomalies by trying to solve pretext tasks on normal data has been a successful direction in one-class anomaly detection. Such models significantly improved state-of-the-art performance. However, these methods still have many limitations especially regarding their applicable field of images and their introduction of an important inductive bias. Moreover they do not use anomalous samples during training. In reality, an additional small set of anomalies is often usable in real-life anomaly detection challenges which can help detecting harder edge-case anomalies.

In the light of these limitations, our main contributions in this paper are the following :

- We explore the concept of learnable pretext task to succeed on normal samples while failing on a small set of additional anomalies. To the best of our knowledge this is the first work in this direction.
- To this end, we propose a semi-supervised anomaly detection model based on thin plate spline transformations classification with a dynamic transformation number.
- We compare our method with state-of-the-art one-class and semi-supervised methods and improve state-of-the-art anomaly detection with up to 49% error relative improvement on object anomalies and 15% on face anti-spoofing.

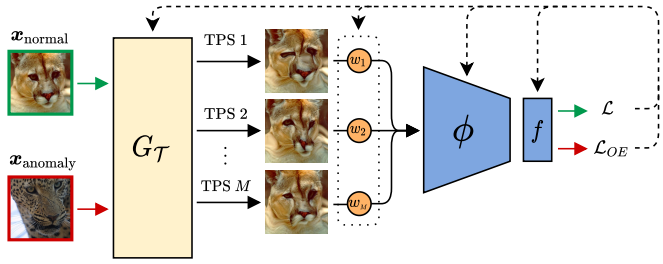

FIGURE 1 – Overview of **Sad-TPS** training. First, we generate M transformed images using our pretext image generator $G_{\mathcal{T}}$, and weight each transformation by $\boldsymbol{w}$. Then we classify each of them using the transformation detector $\phi \circ f$, and compute either our pretext task loss $\mathcal{L}$ on normal samples or the OE loss on anomalies.

2 Proposed method

2.1 Learnable pretext task

Typically, pretext tasks have only been used in a one-class framework. They have proven to be very powerful when detecting anomalies only from normal data, even sometimes competing with other semi-supervised anomaly detection methods [11]. Nevertheless, these methods tend to have two main limitations : **(i)** The choice of pretext tasks is very dependent on the dataset, both for normal samples and anomalies. These methods thus tend to have very high variance and outright failing for specific datasets. **(ii)** Edge-case samples are often poorly classified. This is quite natural since the model has never seen any anomalies during training.

In out-of-distribution literature, some efforts have been made to design training losses which incorporate out-of-distribution data to further improve the decision boundary. However to our knowledge *no work has explored using anomaly data to drive the choice of the pretext task*, and this motivates our work.

Let's consider a pretext task $\mathcal{T}$ defined by its pretext objective loss $\mathcal{L}_{pre}$ and its pretext data generation function $G_{\mathcal{T}} : \mathcal{P}(\mathcal{X}) \mapsto \mathcal{P}(\mathcal{X} \times K)$ where $\mathcal{X}$ is the set of natural images and K the set of pretext labels. Given an image $\boldsymbol{x}$ we first extract a set of features using a deep encoder ϕ, then we predict the task-specific outputs using a second network f. Our objective is to minimize the pretext objective loss $\mathcal{L}_{pre}$ on normal samples $\mathcal{X}_{\text{norm}}$, while maximizing a given Outlier Exposure (OE) loss $\mathcal{L}_{OE}$ on anomalous data $\mathcal{X}_{\text{anom}}$.

We consider a sub-family of pretext data generation function $G_{\mathcal{T}}$ parameterized by Θ, and jointly optimize the network weights of $\phi \circ f$ and Θ.

$$\phi^*, f^*, \Theta^* = \arg\min_{\phi, f, \Theta} \mathcal{L} \tag{1}$$

Our approach of learnable pretext task can be related to automatic data augmentation [15] albeit with two major differences : **(i)** transformations are also driven to perform poorly on anomalous data and **(ii)** they are kept after training and used for the anomaly detection.

In more details, we define our pretext task as a transformation classification task, where the goal is to correctly classify which transformation from a set of M transformations $T_{1:M}$ has been applied. Our task-specific network f is simply a multi-layer perceptron with softmax output and our pretext objective loss is the cross-entropy $\mathcal{L}_{CE}$. Our complete loss becomes

$$\begin{aligned}\mathcal{L} =& \mathbb{E}_{\boldsymbol{x} \sim \mathbb{P}_{\text{norm}}, T_i \sim \mathbb{P}_{\text{T}}}[\mathcal{L}_{CE}(\phi \circ f(T_i(\boldsymbol{x})), i)] + \\ & \lambda \mathbb{E}_{\boldsymbol{x}' \sim \mathbb{P}_{\text{anom}}, T_i \sim \mathbb{P}_{\text{T}}}[\mathcal{L}_{OE}(\phi \circ f(T_i(\boldsymbol{x}')), i)]\end{aligned} \tag{2}$$

where $\mathbb{P}_{\text{T}}$ is the prior distribution over the set of M transformations. Accordingly we choose each transformation to be parametric and entirely defined by a vector $\boldsymbol{\theta}_i$. Using this loss, our pretext task will essentially be designed to provide *easy classification tasks for normal samples but harder tasks for anomalies*.

Once the network has been trained, we freeze the M transformations and compute the anomaly score using the softmax truth in the standard pretext task-driven anomaly detection manner :

$$s_a(\boldsymbol{x}) = \mathbb{E}_{T_i \sim \mathbb{P}_{\text{T}}}[\phi \circ f(T_i(\boldsymbol{x}))_i] \tag{3}$$

2.2 Thin Plate Spline transformations

For our parametric transformations $T_{1:M}$ we choose Thin Plate Spline (**TPS**) transformations [2]. These transformations have several benefits : **(i)** can cover a lot of linear and non-linear transformations, **(ii)** are differentiable, **(iii)** can easily be computed, and **(iv)** can be parameterized by a few parameters.

The 2D TPS transformation T is parameterized by K input control points $\boldsymbol{\pi}^{(i)}_{1:K}$ and output control points $\boldsymbol{\pi}^{(o)}_{1:K}$. It is defined as the solution to a least square error :

$$T^* = \arg\min_{T} \left\{ \sum_{j=1}^{K} \left\| \boldsymbol{\pi}^{(o)}_j - T\left(\boldsymbol{\pi}^{(i)}_j\right) \right\|^2 \right\} \tag{4}$$

There exists a unique closed form solution T^* to this equation, where T^* is continuous in regards to its control points. In order to limit the number of parameters for each TPS transformation, we choose to fix the input control points to an evenly spaced grid of (n_w, n_h) cells and only control the $K = n_w \cdot n_h$ output control points. This reduces the total amount of pretext task parameters to $2M \cdot K$. We also employ a reflection padding on the borders, which is usually much less noticeable than zero padding.

TPS transformations classification forms a very powerful and versatile class of pretext task. It can represent any rotation or translation, generalizing previous geometrical pretext tasks [1], and to some extent can displace or focus on local parts of the images similarly to the jigsaw puzzle task [18]. This will allow our model to optimally identify low-scale and high-scale features that will discriminate normal samples from anomalies.

2.3 Automatic transformation number choice

We push our idea of automatic pretext task to the full extent and also include the number of transformations M into the learnt parameters. To avoid a dynamically sized task-specific network f, we instead turn M into the maximum number of transformations and define a set of M weights $w_1, \cdots, w_M \in [0, 1]$ where w_i represents the weighting coefficient of the i^{th} transformation for anomaly detection. Accordingly, each transformation class in the pretext objective loss and OE loss is weighted by w :

$$\mathcal{L}_{CE}(\boldsymbol{p}, i) = -\kappa \log p_i \tag{5}$$

$$\mathcal{L}_{OE}(\boldsymbol{p}, i) = -\kappa \max_j w_j \cdot p_j \tag{6}$$

in the case of the softmax flatness OE function, where $\boldsymbol{p}$ is the predicted probability for each transformation and $\kappa = \frac{M \cdot w_i}{\sum_j w_j}$ is there to keep the same range for the loss, regardless of the number of transformations. We also prevent the model from choosing too much transformations by adding an L_1 regularization term on the transformation weights, yielding our final loss

$$\begin{aligned}\mathcal{L}_{tot} =& \mathcal{L} + \lambda_{tf} \sum_i w_i \\ =& \mathbb{E}_{\boldsymbol{x} \sim \mathbb{P}_{\text{norm}}, T_i \sim \mathbb{P}_{\text{T}}}[\mathcal{L}_{CE}(\phi \circ f(T_i(\boldsymbol{x})), i)] + \\ & \lambda \mathbb{E}_{\boldsymbol{x}' \sim \mathbb{P}_{\text{anom}}, T_i \sim \mathbb{P}_{\text{T}}}[\mathcal{L}_{OE}(\phi \circ f(T_i(\boldsymbol{x}')), i)] + \lambda_{tf} \sum_{i=1}^{M} w_i\end{aligned} \tag{7}$$

where λ_{tf} is an hyper-parameter controlling the trade-off between accuracy and the number of transformations chosen. Finally, after training our model we discard all parameters including the rows of the task-specific network f weights W corresponding to transformations where $w_i < \tau$.

TABLE 1 – Comparison with the state-of-the-art methods over several datasets in the semi-supervised protocol using the AUC. The first block only contains one-class methods, whereas the second one includes semi-supervised methods. Underline indicates the overall best result, bold indicates the best semi-supervised method (We re-evaluated Elsa, DP-VAE, SSAD, GOAD and ARNet on CIFAR100).

	CIFAR-10				CIFAR-100				F-MNIST			
Models \ γ	**0%**	**1%**	**5%**	**10%**	**0%**	**1%**	**5%**	**10%**	**0%**	**1%**	**5%**	**10%**
OC-SVM [20]	64.7				62.6				74.2			
IF [16]	60.0				58.5				84.0			
PIAD [21]	79.9				78.8				94.3			
ARNet [5]	86.6				78.8				93.9			
GOAD [1]	88.2				74.5				94.1			
MHRot [10]	89.5				83.6				92.5			
PuzzleGeom [11]	88.2				85.8				92.8			
Supervised		55.6	63.5	67.7		53.8	58.4	62.5		74.4	76.8	79.0
SS-DGM [12]		49.7	50.8	52.0		-	-	-		-	-	-
SSAD [7]	62.0	73.0	71.5	70.1	57.4	65.0	67.3	68.1	92.8	92.1	88.3	85.5
DeepSAD [19]	60.9	72.6	77.9	79.8	56.3	67.7	71.6	73.2	89.2	90.0	90.5	91.3
DP VAE [4]	52.7	74.5	79.1	81.1	56.7	68.5	73.4	75.8	90.8	90.9	92.2	91.7
Elsa [8]		80.0	85.7	87.1		81.3	84.6	86.0		**92.9**	**93.4**	93.9
Sad-TPS		**89.8**	**91.8**	**92.6**		**87.8**	**88.0**	**88.5**		92.6	93.1	**94.2**

In practice we represent M by a continuous quantity m, instead of defining M independent transformation weights :

$$w_i = \begin{cases} 1 & \text{if} \quad i \leq \lfloor m \rfloor \\ m - \lfloor m \rfloor & \text{if} \quad i = \lfloor m \rfloor + 1 \\ 0 & \text{if} \quad i > \lfloor m \rfloor + 1 \end{cases} \tag{8}$$

This formulation has the advantage of keeping near-quantized weights during training while allowing our loss to be fully continuous and differentiable.

3 Experiments

3.1 Evaluation protocol

Our evaluation protocol is made of two types of anomaly detection challenges. First, we consider general object recognition datasets where the one-vs-all protocol is used. In this protocol we consider one class of a multi-classification dataset as the normal class and all the other classes as anomalous. We obtain a set of runs for each possible normal class and report the mean of all runs as the final result. Secondly, we include a dataset from face anti-spoofing (FAS) where the goal is to discriminate real faces from fake representations of someone's face. This practical anomaly detection problem incorporates object anomalies, style anomalies and local anomalies.

Unlike the one-class protocol where only normal samples are seen during training, the semi-supervised protocol provides anomalies as well. We consider various ratios γ of anomaly data in the training dataset and for each average the metrics on 10 random samples to obtain a representative and fair evaluation. As for the FAS dataset, we instead use the intra-dataset cross-type protocol where training and test data is sampled from the same dataset, albeit with one tested attack type being unseen during training.

For *object anomalies*, we use **Fashion-MNIST** [22], an harder version of MNIST with 10 classes of fashion items and **CIFAR-10** [13], an object recognition dataset composed of 10 wide classes with 6000 images per class. As for *style anomalies*, we include **CIFAR-100** [13], an extended version of CIFAR-10 with 100 classes each containing 600 images. Finally, for the *face presentation attack detection* we encompass the **WMCA** dataset [6] which contains short RGB videos of real faces and presentation attacks. We consider here the following unseen attack types : Paper Print (**PP**), Screen Recording (**SR**), Paper Mask (**PM**) and Flexible Mask (**FM**).

In all evaluations, the metric used is the area under the ROC curve (**AUROC**) or alternatively the error 1-AUROC, averaged over all possible normal classes for one-vs-all datasets.

3.2 Implementation details

We use a maximum number of TPS transformations $M = 25$ for all the evaluations. A grid of 4×4 control points is used to give enough control on finer details. λ_{tf} is fixed to 0.01, and factor λ is fixed to 0.5 as recommended in [9]. For the transformation number selection, we choose a constant quantization threshold $\tau = 0.5$.

Regarding network architecture, we use a 16-4 WideResNet [23] ($\approx 10M$ parameters with a depth of 16) for the feature extractor network ϕ, along with a dense layer of size M for the transformation recognition task. Training is performed under SGD optimizer with nesterov momentum, using a batch size of 32 and a cosine annealing learning rate scheduler.

3.3 Comparison to the state-of-the-art

We compare our Sad-TPS model with state-of-the-art anomaly detection methods. We use the semi-supervised anomaly detection protocol presented in 3.1, as well as the one-class protocol when possible. This offers a good overview of how effectively the anomaly data is used in semi-supervised methods.

For one-class methods, we include hybrid models such as **OC-SVM** [20] and **IF** [16], reconstruction error generative mo-

TABLE 2 – Comparison with the state-of-the-art methods over face anti-spoofing datasets in the cross-type protocol. The columns indicate the type of presentation attack that has not been seen during training : Paper Print (**PP**), Screen Recording (**SR**), Paper Mask (**PM**) and Flexible Mask (**FM**).

	WMCA				
Models	**All**	**PP**	**SR**	**PM**	**FM**
PIAD	76.4				
ARNet	84.5				
GOAD	86.1				
MHRot	81.3				
PuzzleGeom	85.6				
Supervised		78.3	77.1	80.7	81.9
DP VAE	53.9	-	-	-	-
DeepSAD	71.2	79.9	80.3	81.8	83.4
Elsa		86.1	84.3	**89.2**	89.1
Sad-TPS		**87.4**	**86.6**	89.0	**89.2**

dels with the **PIAD** model [21] and self-supervised methods with **ARNet** [5], **GOAD** [1], **MHRot** [10] and **PuzzleGeom** [11]. Considered semi-supervised methods are reconstruction error models with **DP VAE** [4], density estimation methods with **SS-DGM** [12], two-stage anomaly detection with **Elsa** [8], and direct anomaly distance model with **SSAD** [7] and **DeepSAD** [19]. As a baseline, we also compare a classical binary classification deep network with batch balancing between normal samples and anomalies.

The evaluation results on CIFAR-10, CIFAR-100 and Fashion-MNIST are displayed in Table 1. As we can see, our method performs best on all datasets covering simple object detection and finer object detection with an error relative improvement of up to 49% on CIFAR-10 compared to the second best performing semi-supervised model. This shows our method's great adaptability power to different training data, thanks to its learnable pretext task.

We present in Table 2 the cross-type evaluation on the face anti-spoofing challenge. We show that our method, without further tuning, improves anti-spoofing detection performances on WMCA with error relative improvements of up to 15%.

In general, all anomaly detection approaches including those with the one-class framework outdo the classical binary classification which fails to generalize to unseen anomalies. Moreover, self-supervised models with a pretext task overall excel among one-class methods further justifying the use of pretext task in a semi-supervised fashion.

4 Conclusion

In this paper, we explore the idea of learnable pretext tasks for anomaly detection. We apply this scheme to parameterizable TPS transformations, where the amount of transformations is also dynamically learned. We show through comparison with state-of-the-art methods that this is a very interesting direction that can greatly increase the anomaly detection performances with a few additional anomalous samples. Compared to other pretext task models, it can be used on a wider array of datasets and alleviates the need of an a-priori for the choice of auxiliary task.

Références

[1] L. Bergman and Y. Hoshen. Classification-based anomaly detection for general data. In *ICLR*, 2020.

[2] F. Bookstein. Principal warps : Thin-plate splines and the decomposition of deformations. *IEEE TPAMI*, 11(6) :567–585, 1989.

[3] V. Ceronmani Sharmila, R. Kiran Kumar, R. Sundaram, D. Samyuktha, and R. Harish. Credit Card Fraud Detection Using Anomaly Techniques. In *ICIICT*, pages 1–6, 2019.

[4] T. Daniel, T. Kurutach, and A. Tamar. Deep variational semi-supervised novelty detection. *CoRR*, abs/1911.04971, 2019.

[5] Y. Fei, C. Huang, C. Jinkun, M. Li, Y. Zhang, and C. Lu. Attribute restoration framework for anomaly detection. *IEEE Trans. Multimedia*, 2020.

[6] A. George, Z. Mostaani, D. Geissenbuhler, O. Nikisins, A. Anjos, and S. Marcel. Biometric face presentation attack detection with multi-channel convolutional neural network. *IEEE Trans. Inf. Forensics Secur.*, 15 :42–55, 2020.

[7] N. Görnitz, M. Kloft, K. Rieck, and U. Brefeld. Toward supervised anomaly detection. *J. Artif. Intell. Res.*, 46 :235–262, 2013.

[8] S. Han, H. Song, S. Lee, S. Park, and M. Cha. Elsa : Energy-based learning for semi-supervised anomaly detection. *CoRR*, abs/2103.15296, 2021.

[9] D. Hendrycks, M. Mazeika, and T. G. Dietterich. Deep anomaly detection with outlier exposure. In *ICLR*, 2019.

[10] D. Hendrycks, M. Mazeika, S. Kadavath, and D. Song. Using self-supervised learning can improve model robustness and uncertainty. In *NeurIPS*, pages 15637–15648, 2019.

[11] L. Jézéquel, N.-S. Vu, J. Beaudet, and A. Histace. Fine-grained anomaly detection via multi-task self-supervision. In *AVSS*, pages 1–8, 2021.

[12] D. P. Kingma, S. Mohamed, D. J. Rezende, and M. Welling. Semi-supervised learning with deep generative models. In *NeurIPS*, pages 3581–3589, 2014.

[13] A. Krizhevsky. Learning multiple layers of features from tiny images. 2009.

[14] N. Kumar and S. P. Awate. Semi-supervised robust mixture models in RKHS for abnormality detection in medical images. *IEEE Trans. Image Process.*, 29 :4772–4787, 2020.

[15] Y. Li, G. Hu, Y. Wang, T. M. Hospedales, N. M. Robertson, and Y. Yang. DADA : Differentiable automatic data augmentation. In *ECCV*, 2020.

[16] F. T. Liu, K. M. Ting, and Z.-H. Zhou. Isolation forest. In *ICDM*, pages 413–422, 2008.

[17] H. Lv, C. Zhou, Z. Cui, C. Xu, Y. Li, and J. Yang. Localizing anomalies from weakly-labeled videos. *IEEE Trans. Image Process.*, 30 :4505–4515, 2021.

[18] M. Noroozi and P. Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. In *ECCV*, volume 9910 of *Lecture Notes in Computer Science*, pages 69–84, 2016.

[19] L. Ruff, R. A. Vandermeulen, N. Görnitz, A. Binder, E. Müller, K.-R. Müller, and M. Kloft. Deep semi-supervised anomaly detection. In *ICLR*, 2020.

[20] B. Schölkopf, R. C. Williamson, A. J. Smola, J. Shawe-Taylor, and J. C. Platt. Support vector method for novelty detection. In *NeurIPS*, pages 582–588, 1999.

[21] N. Tuluptceva, B. Bakker, I. Fedulova, and A. Konushin. Perceptual image anomaly detection. In *ACPR*, pages 164–178, 2019.

[22] H. Xiao, K. Rasul, and R. Vollgraf. Fashion-mnist : A novel image dataset for benchmarking machine learning algorithms. *CoRR*, abs/1708.07747, 2017.

[23] S. Zagoruyko and N. Komodakis. Wide residual networks. In *BMVC*, 2016.

[24] Z. Zeng, B. Liu, J. Fu, and H. Chao. Reference-based defect detection network. *IEEE Trans. Image Process.*, 30 :6637–6647, 2021.