

# N-pad : Neighboring Pixel-based Industrial Anomaly Detection

JunKyu Jang<sup>1</sup> Eugene Hwang<sup>2</sup> Sung-Hyuk Park \*  
Korea Advanced Institute of Science and Technology (KAIST)  
{jbjkjsm, hegene3686, sunghyuk.park}@kaist.ac.kr

## Abstract

*Identifying defects in the images of industrial products has been an important task to enhance quality control and reduce maintenance costs. In recent studies, industrial anomaly detection models were developed using pre-trained networks to learn nominal representations. To employ the relative positional information of each pixel, we present **N-pad**, a novel method for anomaly detection and segmentation in a one-class learning setting that includes the neighborhood of the target pixel for model training and evaluation. Within the model architecture, pixel-wise nominal distributions are estimated by using the features of neighboring pixels with the target pixel to allow possible marginal misalignment. Moreover, the centroids from clusters of nominal features are identified as a representative nominal set. Accordingly, anomaly scores are inferred based on the Mahalanobis distances and Euclidean distances between the target pixel and the estimated distributions or the centroid set, respectively. Thus, we have achieved state-of-the-art performance in MVTec-AD with AUROC of 99.37 for anomaly detection and 98.75 for anomaly segmentation, reducing the error by 34% compared to the next best performing model. Experiments in various settings further validate our model.*

## 1. Introduction

Humans have the inherent ability to recognize unusual or abnormal patterns that deviate from what is considered the norm [8, 9]. This trait is essential for various tasks in which inappropriate states must be detected. In particular, identifying defects in the images of industrial products is essential for enhancing quality control and reducing unnecessary maintenance costs [26, 27, 39]. Therefore, artificial intelligence models for industrial anomaly detection have been developed to more precisely identify anomalous images and segments.

In the industrial field, most products are nominal with a rare occurrence of anomalous production [5, 26, 39]. Here, an out-of-distribution classification is performed by train-

ing the distribution of nominal features using only a nominal dataset and by evaluating how the nominal and anomalous images in the test set deviate from the nominal distribution [8, 46]. Industrial anomaly detection has been challenging because some small-scale anomalous regions in products are often too small to distinguish. Moreover, anomalies in the industrial field vary from minor flaws, such as cracks, scratches, and holes, to significant irregularities, such as missing components, flips, and colors [5]. To detect these anomalies well, various models based on autoencoders (AE) [12], semi-supervised learning, generative adversarial networks (GAN) [43], and normalizing flows [17] have been developed. Recently, image representations were extracted from pre-trained models using ImageNet to learn the pixel-wise distributions of features without adaptation through transfer learning, which demonstrated state-of-the-art performances [10, 11, 32]. To successfully use pre-trained models for anomaly detection, the assumption that nominal images are perfectly aligned is necessary for accurate pixel-wise distributions. In this sense, attempts have been made to disregard positional information during detection. Nevertheless, because the inherent properties of industrial products exist primarily in their unique shapes, the positional information of each pixel cannot be overlooked.

Thus, we propose **Neighboring Pixel-based industrial Anomaly Detection (N-pad)**, which is the first attempt to employ the features of neighboring pixels to acquire positional information and minimize errors caused by misalignment. Here, two novel modules for weight application and feature aggregation of neighboring pixels are devised to estimate two nominal distributions by fully leveraging the features of neighboring pixels. Specifically, weights are applied to neighboring pixels according to the similarity values between the target pixel and its neighborhood, which are computed using the Bhattacharyya distance [19]. By integrating the two estimated distributions for the computation of the final anomaly score, we achieved state-of-the-art performance in multiple classes of the industrial dataset with a **pixel-wise area under the receiver operating characteristic curve (AUROC) of 98.75**, which is a **34% improvement** compared to the existing state-of-the-art model

. Various experiments are performed to demonstrate the robustness of the model performance.

## 2. Related Works

### 2.1. General anomaly detection

Conventional models for anomaly detection have been developed to accurately learn the representative attributes of nominal data. In this sense, existing studies have primarily implemented reconstruction- and embedding-similarity-based methods. In terms of reconstruction-based methods, models are trained to learn features that reconstruct the original data, which identifies poorly reconstructed samples as anomalies. Accordingly, extensions based on AEs [5, 7, 12, 16, 24, 37] or GANs [2, 28, 29, 36, 38, 43] have been proposed. In terms of embedding similarity-based methods, the latent features of nominal data were learned from the model to identify samples distinct from the nominal distribution as anomalies. As a reference for the nominal features, the center of constrained latent feature spaces [30, 34, 35, 47], geometric transformations [4, 15, 23, 40, 41], estimation of the probability density function using Gaussian mixture models [21, 31, 50], and kernel density estimations [20] have been employed. Hence, distance-based metrics [14, 32, 42] have been applied to assign distant samples with high anomaly scores.

### 2.2. Industrial anomaly detection

Industrial anomaly detection has developed differently from general anomaly detection because learning the unique nominal features of an industrial object or texture is essential [5]. A recent trend in industrial anomaly detection is to use a model pre-trained on an external image dataset, such as ImageNet, to learn the distribution or features of the nominal dataset without transfer learning [3]. One of the first successful applications was SPADE [10], which obtains a global feature set from the given network of the nominal data and applies a Euclidean distance-based measure of the k-nearest neighbor [14] to the feature set for image-wise anomaly detection. Another pre-trained network-based model, PaDiM [11] learned the distribution of local features at every pixel and obtained a pixel-wise anomaly map by computing the Mahalanobis distance between the pixel and its distribution [25]. Similarly, PatchCore [32] proposed an algorithm for storing a subsampled coreset [1] of the pre-trained features in a memory bank to obtain the patch-level distance between the coreset and a sample for detecting anomalies. In addition, attempts have been made to adapt the weights of the pre-trained model to identify the distribution of nominal data. FastFlow [48], FEFM [44], and CFLOW-AD [17] reported good performances by estimating the distribution of network-based features by normalizing the flow, and CFA [22] implemented

feature adaption through Coupled-hypersphere to better explain the distribution of nominal features.

However, there are some limitations in existing pre-trained feature-based models without the adaptation of pre-trained features. In particular, because PADiM utilized only the nominal data of the target pixel location to compute its anomaly score, the scores may be overestimated if all nominal industrial images are not perfectly aligned. PatchCore was developed to disregard the positional information of the pixel because the anomaly scores were computed based on the distance from the core patch-level local features that were stored in a memory bank as a whole. Nevertheless, considering the positional information of each pixel is essential for anomaly detection. When augmentations of rotated images were included for prediction in CSI [41], the predictive performance degraded, indicating that the change in position was not constructive for anomaly detection.

Thus, to overcome these limitations, the proposed model is devised to employ the information of neighboring pixels to estimate the nominal distribution of each pixel because the method of integrating the relationship between neighboring nodes with features has long been utilized in graph neural networks. Specifically, the similarities between the target pixel and its neighborhood are applied as weights to appropriately consider the information of neighboring pixels along with the target pixel. Consequently, we aim to design a model that is less affected by perfect image alignment but utilizes the positional information of pixels.

## 3. Method

### 3.1. Calculation of pixel-wise neighborhood similarity

**Feature extraction** In this study, a model architecture which implemented a pre-trained network on ImageNet as the backbone is designed for the out-of-distribution task of anomaly detection. Herein, the training set  $\mathcal{X}_{train} = \{x_k | y_k = 0\}$  consists of  $|\mathcal{X}_{train}| = N$  nominal images, and the test set  $\mathcal{X}_{test} = \{x_k | y_k = 0 \text{ or } 1\}$  consists of  $|\mathcal{X}_{test}| = N_{test}$  images that are either nominal or anomalous, where  $x_k$  denotes a single image from a set of all images  $\mathcal{X}_{train}$ , and  $y_k \in \{0, 1\}$  denotes image  $x_k$  as nominal with 0 and anomalous with 1. As in previous studies [3, 10, 11, 32], ResNet-like architectures, such as ResNet50 and WideResnet-50, were employed to extract feature maps. Within the given network  $\varphi_h$ , feature maps are extracted from the final output of the spatial resolution block at a specific hierarchy level ( $h = 1, 2, 3$ ). Because the feature map extracted from the lowest hierarchy level ( $h = 1$ ) has the largest size, the feature maps of higher hierarchy levels ( $h = 2, 3$ ) are interpolated to this size. Consequently, the pre-trained feature set  $\varphi(x_k) = [\varphi_h(x_k)]$ ,  $h = \{1, 2, 3\}$  is constructed by concatenating all channels from

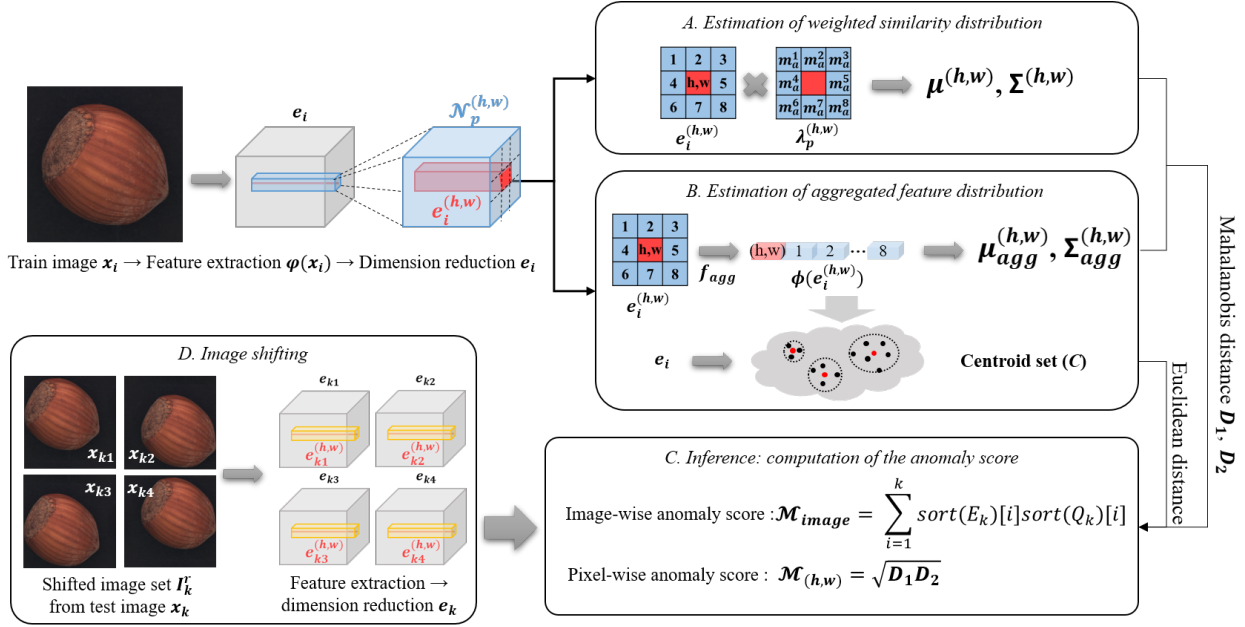


Figure 1. Overall model architecture. Two nominal distributions are estimated by applying the similarity between the target pixel and its neighboring pixels as weights (A) and by aggregating features of its neighborhood (B). Also, the  $k$ -means centroids of the aggregated features are identified as a set of representative nominal features for image-wise detection (B). Next, the pixel-wise anomaly map and image-wise anomaly score are computed by Mahalanobis distances between the estimated distributions and test features and Euclidean distance between the centroids of train features and test features (C). Lastly, a shifting technique is applied to enhance the predictive performance (D).

each level.

**Dimension Reduction** Before estimating the nominal distributions, dimension reduction is performed on the total set of concatenated features  $\varphi(x_k)$  because features extracted from the pre-trained network may infer redundant information. Although the concatenated channels at each pixel are assumed to follow a multivariate Gaussian distribution, not every channel may follow a Gaussian distribution. In this sense, when reducing the number of channels, we aim to select the channels with an approximate Gaussian distribution form. We believe that the normal distribution may be distorted when all channels with values below zero are set to zero after applying ReLU function at the end of most pre-trained networks. Accordingly, the nonzero values in the nominal features are counted for each channel, and the top- $d$  channels with the least nonzero values are selected. Consequently, the final nominal feature set reduced from  $\varphi(x_k)$  is identified and denoted as  $e_{x_k}$ .

### 3.2. Estimation of weighted similarity distribution

**Calculation of pixel-wise neighbor Bhattacharyya distance** To estimate the nominal distribution at each pixel, we propose a novel method for computing the pixel-wise similarity between a pixel and its neighborhood. In this study, we aim to calibrate possible misalignments by in-

cluding information from neighboring pixels, whereas the perfect alignment of pixels was essential for position-based estimations in existing methods. Specifically, the neighborhood of a pixel is defined as the set of  $p$  pixels that were adjacent to the target pixel:

$$\mathcal{N}_p^{(h,w)} = \{(h', w') \mid h' \in [h - \lfloor p/2 \rfloor, h + \lfloor p/2 \rfloor], w' \in [w - \lfloor p/2 \rfloor, w + \lfloor p/2 \rfloor]\} \quad (1)$$

First, based on the assumption that every pixel  $(h, w)$  in a feature map  $e_i^{(h,w)}$  of a nominal image  $i$  follows a multivariate Gaussian distribution, the sample mean  $\mu^{(h,w)}$  and covariance  $\Sigma^{(h,w)}$  of the nominal distribution are estimated. In addition, a regularization term  $\epsilon I$  is added to  $\Sigma^{(h,w)}$  to ensure full rank and invertibility.

$$\mu^{(h,w)} = \frac{1}{N} \sum_{i=1}^N e_i^{(h,w)} \quad (2)$$

$$\Sigma^{(h,w)} = \frac{1}{N-1} \sum_{i=1}^N \left( e_i^{(h,w)} - \mu^{(h,w)} \right) \left( e_i^{(h,w)} - \mu^{(h,w)} \right)^T + \epsilon I \quad (3)$$

Next, the Bhattacharyya distance  $m$ , which indicates the distance between two probability distributions, is computed

between the target pixel and all pixels within the neighborhood. Because the Bhattacharyya coefficient  $BC$  measures the overlapping degree of the two distributions, the negative exponential value of the coefficient is accepted as the similarity value. Consequently, the Bhattacharyya distance set  $\lambda_p^{(h,w)}$  between the estimated distribution of pixels  $(h, w)$ , and  $\mathcal{N}_p^{(h,w)}$  is computed as follows:

$$\begin{aligned} \lambda_p^{(h,w)} &= \{m_a | a \in \mathcal{N}_p^{(h,w)}\} \\ m_a &= \text{Batt}(\mathcal{N}_{(\mu^{(h,w)}, \Sigma^{(h,w)})}, \mathcal{N}_{(\mu^a, \Sigma^a)}), \text{Batt} = e^{-\frac{BC}{\gamma}} \\ BC(\mathcal{N}(\mu_1, \Sigma_1), \mathcal{N}(\mu_2, \Sigma_2)) &= \\ \frac{1}{8}(\mu_1 - \mu_2)^T \Sigma'^{(-1)} (\mu_1 - \mu_2) + \frac{1}{2} \log\left(\frac{\det \Sigma'}{\sqrt{\det \Sigma_1 \det \Sigma_2}}\right) \end{aligned} \quad (4)$$

where  $\mu_1$  and  $\mu_2$  denote a pair of mean values obtained from the estimated distributions and  $\Sigma'$  denotes the average of  $\Sigma_1$  and  $\Sigma_2$ . Herein, a balancing parameter  $\gamma$  is employed to modulate the degree to which the neighboring pixels are used to estimate the distributions. A  $\gamma$  of 1 is equal to the original formulation of the Bhattacharyya distance, and larger values of  $\gamma$  imply that more information is used from the neighborhood. Moreover, by assuming that the inherent information of pixels within a neighborhood, denoted as the sample covariances of  $(h, w)$  and  $(h', w')$ , are similar, the logarithm of the ratio of the determinant terms in Eq. (4) is negligible. Consequently, the final similarity with the reduced computational cost is calculated as follows:

$$BC(\mathcal{N}_{(\mu_1, \Sigma_1)}, \mathcal{N}_{(\mu_2, \Sigma_2)}) \simeq \frac{1}{8}(\mu_1 - \mu_2)^T \Sigma'^{(-1)} (\mu_1 - \mu_2) \quad (5)$$

**Learning the normality based on similarity** As the last step for learning the nominal distribution of each pixel, we aim to accentuate the features at specific locations that may infer more relevant information about the target pixel  $(h, w)$ . In this sense, weights are applied to the neighboring pixels according to their similarity to the target pixel. Accordingly, the similarity values calculated within  $\mathcal{N}_p^{(h,w)}$  are utilized to estimate the weighted sample mean  $\mu^{(h,w)}$  and covariance  $\Sigma^{(h,w)}$  to accurately train the distribution of each pixel from the nominal images. The weighted sample mean and covariance are defined as follows:

$$\begin{aligned} \mu^{(h,w)} &= \frac{1}{N} \sum_{i=1}^N \sum_{a \in \mathcal{N}_p^{(h,w)}} m'_a e_i^a, m'_a = \frac{m_a}{\sum_{a \in \mathcal{N}_p^{(h,w)}} m_a} \\ \Sigma^{(h,w)} &= \frac{1}{N - \sum_{a \in \mathcal{N}_p^{(h,w)}} (m'_a)^2} \times \\ &\sum_{i=1}^N \sum_{a \in \mathcal{N}_p^{(h,w)}} m'_a (e_i^a - \mu^{(h,w)})(e_i^a - \mu^{(h,w)})^T \end{aligned} \quad (6)$$

### 3.3. Estimation of aggregated feature distribution

**Learning the normality based on neighborhood aggregate features** To best use the information of neighboring pixels, the normality based on aggregating neighborhood features (B in Fig. 1) is learned, in addition to the normality learned with weights (A in Fig. 1). Because neighboring pixels infer unseen information from the target pixel, an anomaly map for a receptive field with higher resolution is identified by aggregating the features within a neighborhood as follows:

$$\phi(e^{(h,w)}) = f_{agg}(e^a | a \in \mathcal{N}_p^{(h,w)}) \quad (7)$$

where  $f_{agg}$  is the aggregation function for the neighborhood  $\mathcal{N}_p^{(h,w)}$ . In *N-pad*, we use adaptive average pooling for  $f_{agg}$ . Accordingly, the pixel-wise nominal distribution is learned by computing the sample mean and variance of the aggregated features at each pixel.

Because utilizing the Euclidean distance between the test feature and aggregated features has been effective in image-wise anomaly detection in existing studies, we aim to construct a memory bank consisting of a group of essential features. In this sense, the aggregated features from the nominal set are clustered using k-means, and the features identified as the centroid of each cluster are grouped into a representative set of features denoted as  $C$ . In fact, the method of retrieving centroids as key features has been highly robust for outliers and noisy features within the nominal set and reported significant performance as opposed to arbitrary feature selection [13, 45, 49]. Thus, within the memory bank of all cluster centroids  $C$ , a group of centroids near the target feature is retrieved for image-wise anomaly detection.

### 3.4. Inference: computation of anomaly score

**Pixel-wise anomaly map** The anomaly score of a pixel  $(h, w)$  is computed using the Mahalanobis distance between the target pixel and distributions estimated by the two modules, in which features highly deviated from the nominal distributions reported higher anomaly scores.

First, because the information of neighboring pixels at  $(h, w)$  is involved in estimating the weighted distribution of  $\mathcal{N}(\mu, \Sigma)$ , features extracted at  $(h, w)$  affect the values in  $\mathcal{N}_p^{(h,w)}$ . In this sense, we also employ the distributions of neighboring pixels when computing the anomaly score of the targeted position. Accordingly, Mahalanobis distances are computed between the target feature  $e^{(h,w)}$  and its neighborhood, which is defined as a collection of estimated distributions  $\mathcal{P}_q(e^{(h,w)})$  identified from  $\mathcal{N}_q^{(h,w)}$ . By applying a minimum aggregation function  $f$  to the set of Mahalanobis distances,  $\mathcal{D}_1$  is obtained for each pixel and used to calculate the anomaly score. The computation of  $\mathcal{D}_1$  proceeds as follows:

$$\mathcal{P}_q(e^{(h,w)}) = \{\mathcal{N}(\boldsymbol{\mu}^a, \boldsymbol{\Sigma}^a) \mid a \in \mathcal{N}_q^{(h,w)}\} \quad (8)$$

$$\begin{aligned} \mathcal{D}_1(e^{(h,w)}, \mathcal{P}_q(e^{(h,w)})) = \\ f(\sqrt{(e^{(h,w)} - \boldsymbol{\mu}^a)^T (\boldsymbol{\Sigma}^a)^{-1} (e^{(h,w)} - \boldsymbol{\mu}^a)} \mid a \in \mathcal{N}_q^{(h,w)}) \end{aligned} \quad (9)$$

Next, the Mahalanobis distance  $\mathcal{D}_2$  between aggregated features  $\phi(e^{(h,w)})$  of pixel  $(h, w)$  and  $\mathcal{N}(\boldsymbol{\mu}_{agg}^{(h,w)}, \boldsymbol{\Sigma}_{agg}^{h,w})$  is defined as follows:

$$\begin{aligned} \mathcal{D}_2(\phi(e^{(h,w)}), \mathcal{N}(\boldsymbol{\mu}_{agg}^{(h,w)}, \boldsymbol{\Sigma}_{agg}^{h,w})) = \\ \sqrt{(\phi(e^{(h,w)}) - \boldsymbol{\mu}_{agg}^{(h,w)})^T (\boldsymbol{\Sigma}_{agg}^{(h,w)})^{-1} (\phi(e^{(h,w)}) - \boldsymbol{\mu}_{agg}^{(h,w)})} \end{aligned} \quad (10)$$

Finally, to equalize the effects of the two pixel-wise anomaly maps  $\mathcal{D}_1$  and  $\mathcal{D}_2$  obtained from all pixels using Eq. (9) and Eq. (10), the geometric mean of the two maps is used as the final anomaly score  $\mathcal{M}_{(h,w)}$ :

$$\mathcal{M}_{(h,w)} = \sqrt{\mathcal{D}_1 \mathcal{D}_2} \quad (11)$$

**Image-wise anomaly score** The image-wise anomaly score based on the Euclidean distance between the aggregated features of a test image and refined set of aggregated nominal features has been effective in existing studies. Herein, a combination of Euclidean and Mahalanobis distances is employed to detect image-wise anomalies more accurately. First, the top- $k$  Mahalanobis distances  $\mathcal{D}_1$  between the target feature  $e^{(h,w)}$  and estimated distribution collection of  $\mathcal{P}_q(e^{(h,w)})$  are identified as a set  $\mathcal{Q}_k$ . Next, for all pixels included in  $\mathcal{Q}_k$ , the minimum Euclidean distance  $d$  between the aggregated feature of a pixel  $\phi(e^v)$  and the centroids in set  $C$  is calculated for all pixels and denoted as set  $E_k$ . Consequently, the image-wise anomaly score is defined as follows:

$$\mathcal{Q}_k = \max_{(h,w)} k(\mathcal{D}_1(e^{(h,w)}, \mathcal{P}_q(e^{(h,w)}))) \quad (12)$$

$$E_k = \{\min_C d(\phi(e^v), C) \text{ for } \forall v \in \arg \max_k \mathcal{D}_1\} \quad (13)$$

$$\mathcal{M}_{image} = \sum_{i=1}^k \text{sort}(E_k)[i] \text{sort}(\mathcal{Q}_k)[i] \quad (14)$$

where  $E_k$  and  $\mathcal{Q}_k$  are sorted in ascending order because the sizes of the two distance values at each pixel are not in accordance. Consequently, employing both the top- $k$  Euclidean and Mahalanobis distances is demonstrated to be robust for computing image-wise anomalies.

**Image-shifting** As the final inference step, target image  $x_k$  is shifted by the pixel level from size 1 to  $r$  to compute the final image-wise anomaly score and pixel-wise anomaly

map based on the anomaly scores of the shifted images. By aggregating the scores from all shifted images of  $x_k$  denoted as a set  $\mathcal{I}_k^r$ , we expect the marginal misalignment in the images to be negligible. Set  $\mathcal{I}_k^r$  is defined as follows:

$$\begin{aligned} \mathcal{I}_k^r = \{x'_k \mid x'_k[h-a, w-b] = x_k[h, w] \\ \forall (a, b) \in [[-r/2, r/2], [-r/2, r/2]]\} \end{aligned} \quad (15)$$

## 4. Experiments

### 4.1. Dataset and experimental setup

In this study, the proposed model is trained and evaluated on MVTEC Anomaly Detection dataset (MVTEC-AD) [5], which has been widely used for industrial anomaly detection tasks in existing studies. MVTEC-AD consists of ten object and five texture classes with 3,629 nominal-only images for training and 1,725 nominal and anomalous images for evaluation. Moreover, we perform additional experiments on Magnetic Tile Defects (MTD) to further validate our model, where the details are described in Sec. 4.5. All images are center-cropped from  $256 \times 256$  to  $224 \times 224$  before model training and evaluation. The proposed model with a neighborhood size  $p$  of 3 for model training, neighborhood size  $q$  of 2 for inference, shift size  $r$  of 4, balancing parameter  $\gamma$  of 0.25, dimension reduction to 550, and 10% use of the centroids from  $C$  reported the best performance.

To evaluate the performance of the proposed model, the image- and pixel-wise AUROCs are measured. In addition to the AUROC, the per-region-overlap score (PRO-score), which has been widely used in existing studies to measure anomaly detection performance, is measured for pixel-wise anomaly segmentation [5,6]. Herein, a PRO-curve is plotted using the average rates of correctly classified pixels for all connected anomalous components, with the false positive rates set between 0 and 0.3. Accordingly, the PRO-score is computed by normalizing the area under the PRO-curve.

### 4.2. Comparison with baseline methods

To validate the predictive performance of the proposed model, we have benchmarked methods from general anomaly detection, pre-trained feature-based models, and existing models with state-of-the-art performance on MVTEC-AD dataset. Since some existing models, such as Cflow-AD and PEFM, reported ensembled results with different image resolutions or without the  $224 \times 224$  crop, we standardize the image size of all models prior to model training for objective comparison.

Tab. 1 presents the average AUROC of image-wise anomaly detection and the average AUROC and PRO-score of pixel-wise anomaly segmentation over 15 classes of MVTEC-AD, where the value in parenthesis denotes the error in each score. Herein, the proposed model consistently outperforms existing models with state-of-the-art

Method	Normalizing Flow Based			Pre-trained Feature Based			
Model	FastFlow	PEFM	CFLOW-AD	SPADE	PaDiM	PatchCore	<i>N-pad</i>
Image-wise AUROC	97.52 (2.48)	98.13 (1.87)	97.24 (2.76)	-	-	99.0 (1.0)	<b>99.37 (0.63)</b>
Pixel-wise AUROC	98.03 (1.97)	97.61 (2.39)	97.57 (2.43)	96.0 (4.0)	97.5 (2.5)	98.1 (1.9)	<b>98.75 (1.25)</b>
PRO-score	93.0 (7.0)	92.4 (7.6)	91.7 (8.3)	91.7 (8.3)	92.1 (7.9)	93.5 (6.5)	<b>95.1 (4.9)</b>

Table 1. Image-wise AUROC, pixel-wise AUROC, PRO-score of various models on the MVTec-AD dataset

performance in both tasks. Specifically, the reduction of error for image-wise detection is 37% compared to the pre-trained feature-based model of PatchCore. Moreover, the proposed model achieves state-of-the-art performance in pixel-wise AUROC for 12 out of 15 classes with an average AUROC of 98.75 and PRO-score of 95.1 (Tab. 1), reducing the error by 34% and 25%, respectively, compared to the next best performing model. The detailed comparison of all scores and visual results between various models for all 15 classes of MVTec-AD are listed in the supplementary material.

We believe that the proposed method of applying similarity between the target pixel and its neighborhood as weights successfully trained the underlying relationship within the pixels. In addition, we believe that the shifting module also contributed greatly to the predictive performance by inferring the distributions of neighboring pixels when computing anomaly score.

### 4.3. Ablation study

**Evaluation of the effectiveness of key design components** The effectiveness of the four modules comprising the proposed model architecture (A, B, C, and D in Fig. 1) is evaluated by removing certain modules and comparing their predictive performances. Herein, five experiments are performed, as follows.

Experiment 1: Inference only using the weighted similarity distribution of the target pixel without the aggregated feature distribution (A).

Experiment 2: Inference only using the aggregated feature distribution without the weighted similarity distributions (B).

Experiment 3: Inference using both the weighted similarity distribution of the target pixel and aggregated feature distribution (A+B).

Experiment 4: Inference using the distributions from neighboring pixels to estimate the weighted similarity distribution (A+C).

Experiment 5: Inference using the distributions from neighboring pixels to estimate the weighted similarity and aggregated feature distributions without shifting (A+B+C).

Tab. 2 shows that all modules significantly contribute to the performance of anomaly detection. First, the degraded

performance in Experiment 5 compared with the result of *N-pad* proves that shifting the aggregated anomaly maps is superior to the sole use of the original images. Next, 0.19 increase in the AUROC from Experiment 1 to Experiment 4 demonstrates that using the distributions from the neighboring pixels was effective in estimating the weighted similarity distribution. This result suggests that aggregating the anomaly scores computed from the distributions of neighboring pixels is effective. Finally, the improved performance in Experiment 3, which integrated both modules from Experiments 1 and 2, demonstrates that the inference that uses both weighted similarity and aggregated feature distributions was effective.

Method	Exper1	Exper2	Exper3	Exper4	Exper5	<i>N-pad</i>
Pixel-wise AUROC	98.38	98.45	98.59	98.57	98.65	98.75
Error	1.62	1.58	1.41	1.43	1.35	1.25

Table 2. Evaluation of the effectiveness of key design components

**Verification of parameter efficiency in model architecture** Various parameters within the modules were tested to determine the optimal design of the proposed model. First, different neighborhood sizes ( $p$ ) for estimating the distributions of weighted similarity or aggregated features are tested. Fig. 2a demonstrates that the performance improves as the neighborhood size increases from 1, reaches the optimal level at a size of 3, and degrades with larger sizes. Thus, we believe that acquiring information from considerably close neighbors is the best, whereas distant neighbors infer excessive information with no greater relevance to the target pixel.

Second, the different numbers of distributions on neighboring pixels ( $q$ ) used to compute the anomaly map in module A (Fig. 1) are tested. Fig. 2b shows a neighbor size of 2 as the optimal value. Because applying a large neighborhood size may affect numerous pixels of the original image during interpolation, we believe that a relatively small number is optimal.

Third, different shifting sizes ( $r$ ) are tested, as shown in Fig. 2c, resulting in a shifting size of 4 being the most optimal. This approach demonstrates that calibrating imperfectly aligned industrial images through the aggregation of slightly shifted versions of the images can significantly contribute to improved performance.

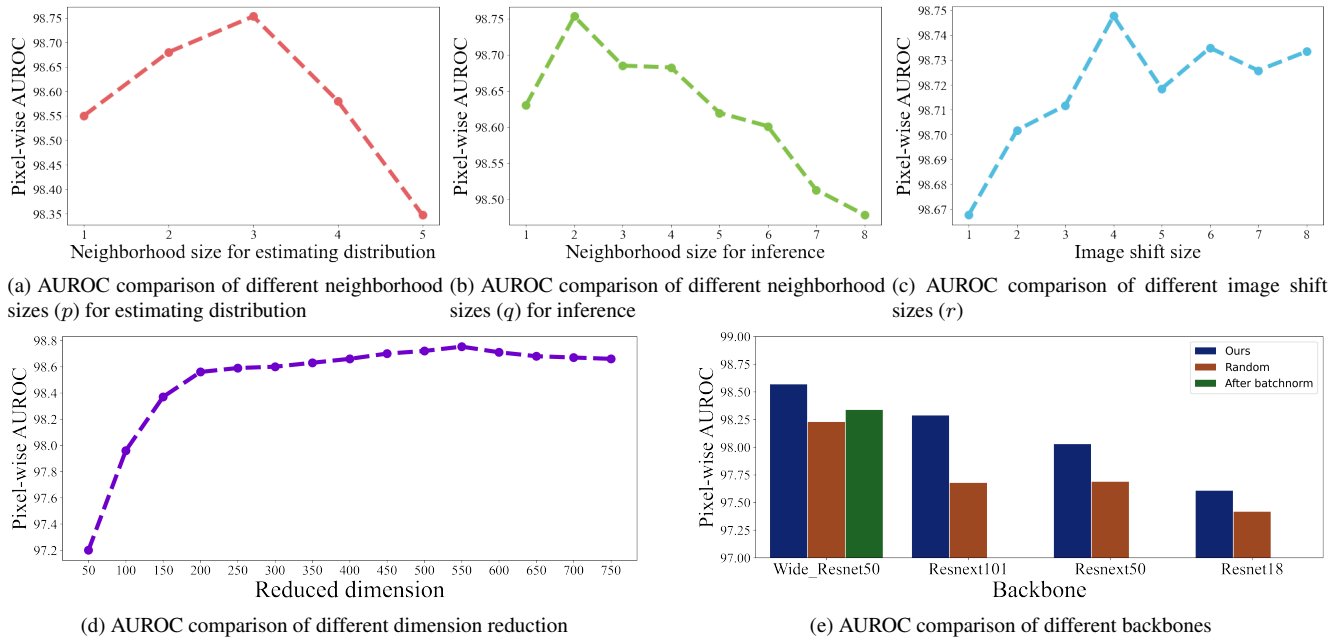


Figure 2. Verification of parameter efficiency with various sizes and backbones.

Fourth, the number of channels following the dimension reduction is tested by first using 50 channels and increasing the number up to 550 in units of 50. As shown in Fig. 2d, the proposed model outperforms PatchCore, a state-of-the-art model, with an pixel-wise AUROC of 98.1, when the number reaches 150 channels, which is only 8.37% of the total number of channels. Because the computational cost reduces quadratically with fewer channels, this result demonstrates that the proposed model can be effective with minimal computation.

Finally, additional experiments are performed with different image sizes because a few existing models have reported benchmark scores by extensively reshaping the image size or excluding image crops. Because the cropped area is mostly the edge of the image background, which may be easily identified as nominal pixels, the better predictive performance is recorded with larger image sizes without cropping. Consequently, an ensemble of models which employed images cropped by 224 and 336 reported the best pixel-wise AUROC of **98.98**, as shown in Tab. 3.

Method	256Resize 224crop	256Resize	320Resize 280Crop	384Resize 336Crop	Ensemble
Pixel-wise AUROC	98.75	98.91	98.86	98.89	<b>98.98</b>

Table 3. AUROC comparison of different image sizes and crops

**Evaluation on various ratios of k-means centroids** We have tested various ratios of the K-means centroids included for model evaluation and compared their AUROCs. In fact,

the decrease in the number of clusters did not significantly affect the results because a highly robust set of centroids was employed.

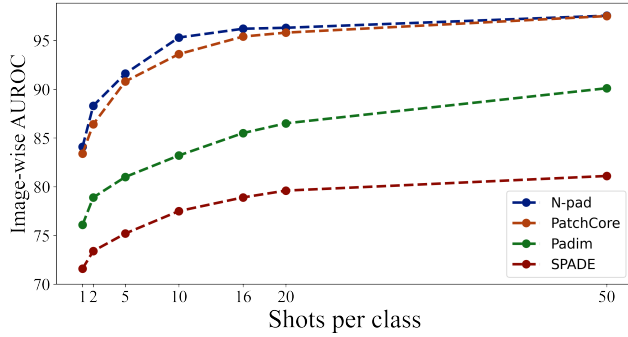
K-means ratio	0.25	0.1	0.05	0.01	0.005
Image-wise AUROC	99.39	99.37	99.34	99.31	99.10

Table 4. Evaluation of image-wise AUROC on various ratios of k-means centroids

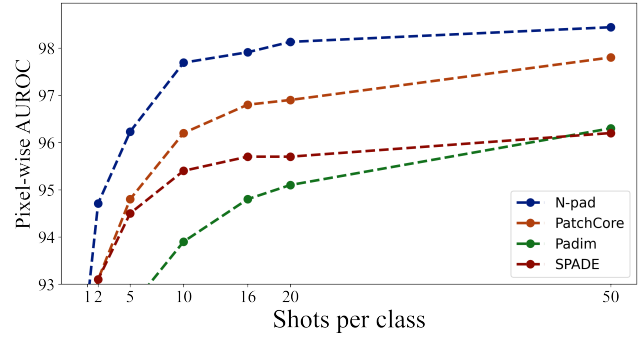
**Evaluation of the effectiveness of Bhattacharyya distance for estimating weighted similarity distributions** To demonstrate the effectiveness of the Bhattacharyya distance calculation for estimating the weighted similarity distributions in this study, uniform and random weights are tested for comparison, as presented in Tab. 5. First, weights that are randomly applied resulted in poor performance because the relationships within the pixels were not considered. Moreover, uniformly applied weights also have a minimal effect on the estimated mean and covariance of the distributions, because features from neighboring pixels may not infer significantly different information from the target pixel. Consequently, the proposed method of weighted sampling based on similarity is reported as the most effective for predictive performance.

Sampling Method	1/n	Random	Ours
Pixel-wise AUROC	98.42	98.42	<b>98.45</b>

Table 5. AUROC comparison of different sampling methods



(a) Image-wise AUROC comparison of different models.



(b) Pixel-wise AUROC comparison of different models.

Figure 3. Few shot anomaly detection performance

**Comparison of methods for reducing dimensions with various backbones** To evaluate the proposed distribution-based method for dimension reduction, results based on a random dimension reduction with different backbones are reported for comparison. First, the random selection of dimensions in the proposed model achieves an AUROC decrease of 0.34. Next, a random dimension reduction in features extracted at the batch normalization layer prior to ReLU scores AUROC that is 0.23 lower than that of the proposed model. This result demonstrates that because the channels activated greater than 0 by the activation function are more relevant for ImageNet classification, the pre-trained network features extracted from those channels may have been more effective. Furthermore, various model architectures other than the WideResNet-50 of the proposed model, such as ResNet18, ResNext50, WideResNet-101, and ResNext-101, are tested for comparison. As shown in Fig. 2e, the proposed method reports a better performance than random reduction in all architectures, demonstrating the consistency of its superiority.

#### 4.4. Few-shot Anomaly Detection

In the industrial field, anomaly detection can be required for initial production, where only a small set of nominal sample data is available. Accordingly, few-shot anomaly detection is performed to test the proposed model with limited nominal data by testing the number of training images from 1 to 50. Consequently, as demonstrated in Fig. 3, the proposed model achieves better performance than the previous state-of-the-art model using only 8% of the total dataset. Because the proposed model employs information from neighboring pixels to train the distribution of the target pixel, the augmented information from the neighborhood may have significantly contributed to few-shot learning.

#### 4.5. Evaluation on MTD dataset

In addition to the MVTec-AD dataset, we performed additional experiments with MTD (Magnetic Tile Defects)

dataset which has also been used for industrial anomaly detection in previous studies [18]. MTD dataset consists of magnetic tile images in various shapes and patterns, of which 925 are nominal and 392 are anomalous. As in previous studies, 80% of the nominal data were employed for model training and the remaining 20% and the anomalous data were employed for evaluation. Accordingly, the results were compared to existing pre-trained network-based models and DifferNet [33], which reported good performance. The results are reported as follows:

Model	DifferNet	PaDiM	PatchCore	<i>N-pad</i>
Image-wise AUROC	97.7	86.88	97.9	<b>98.22</b>
Pixel-wise AUROC		82.45	84.90	<b>85.43</b>

Table 6. Evaluation on MTD dataset

Since the shapes of nominal data are not consistent, the images may not be clearly aligned, which makes the predictions on MTD dataset challenging. Nevertheless, we have achieved superior performance by employing neighboring pixels to calculate anomaly score.

## 5. Conclusion

In this paper, we propose a novel model for industrial anomaly detection and segmentation that utilizes features from the neighborhood of the target pixel. We estimate the nominal distribution of each pixel inferring the information in the neighborhood by applying the similarity between the neighboring pixels and target pixel as weights. Moreover, another estimation of the nominal distribution based on aggregated features is proposed to employ information from various receptive fields. Various experiments evaluate the model in multiple settings and achieved state-of-the-art performances on the 15 classes of an industrial anomaly dataset. Thus, we believe that learning nominal distributions with the pre-trained features of neighboring pixels is useful and effective for improving predictive performances in industrial anomaly detection.



## References

- [1] Pankaj K Agarwal, Sariel Har-Peled, Kasturi R Varadarajan, et al. Geometric approximation via coresets. *Combinatorial and computational geometry*, 52(1), 2005. 2
- [2] Samet Akcay, Amir Atapour-Abarghouei, and Toby P Breckon. Ganomaly: Semi-supervised anomaly detection via adversarial training. In *Asian conference on computer vision*, pages 622–637. Springer, 2018. 2
- [3] Liron Bergman, Niv Cohen, and Yedid Hoshen. Deep nearest neighbor anomaly detection. *arXiv preprint arXiv:2002.10445*, 2020. 2
- [4] Liron Bergman and Yedid Hoshen. Classification-based anomaly detection for general data. *arXiv preprint arXiv:2005.02359*, 2020. 2
- [5] Paul Bergmann, Michael Fauser, David Sattlegger, and Carsten Steger. Mvtec ad—a comprehensive real-world dataset for unsupervised anomaly detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9592–9600, 2019. 1, 2, 5
- [6] Paul Bergmann, Michael Fauser, David Sattlegger, and Carsten Steger. Uninformed students: Student-teacher anomaly detection with discriminative latent embeddings. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 5
- [7] Paul Bergmann, Sindy Löwe, Michael Fauser, David Sattlegger, and Carsten Steger. Improving unsupervised defect segmentation by applying structural similarity to autoencoders. In *VISIGRAPP (5: VISAPP)*, 2019. 2
- [8] Raghavendra Chalapathy and Sanjay Chawla. Deep learning for anomaly detection: A survey. *arXiv preprint arXiv:1901.03407*, 2019. 1
- [9] Varun Chandola, Arindam Banerjee, and Vipin Kumar. Anomaly detection: A survey. *ACM computing surveys (CSUR)*, 41(3):1–58, 2009. 1
- [10] Niv Cohen and Yedid Hoshen. Sub-image anomaly detection with deep pyramid correspondences. *arXiv preprint arXiv:2005.02357*, 2020. 1, 2
- [11] Thomas Defard, Aleksandr Setkov, Angélique Loesch, and Romaric Audigier. Padim: a patch distribution modeling framework for anomaly detection and localization. In *International Conference on Pattern Recognition*, pages 475–489. Springer, 2021. 1, 2
- [12] Hanqiu Deng and Xingyu Li. Anomaly detection via reverse distillation from one-class embedding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9737–9746, 2022. 1, 2
- [13] Thanh-Toan Do, Toan Tran, Ian Reid, Vijay Kumar, Tuan Hoang, and Gustavo Carneiro. A theoretically sound upper bound on the triplet loss for improving the efficiency of deep distance metric learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10404–10413, 2019. 4
- [14] Eleazar Eskin, Andrew Arnold, Michael Prerau, Leonid Portnoy, and Sal Stolfo. A geometric framework for unsupervised anomaly detection. In *Applications of data mining in computer security*, pages 77–101. Springer, 2002. 2
- [15] Izhak Golan and Ran El-Yaniv. Deep anomaly detection using geometric transformations. *Advances in neural information processing systems*, 31, 2018. 2
- [16] Dong Gong, Lingqiao Liu, Vuong Le, Budhaditya Saha, Moussa Reda Mansour, Svetha Venkatesh, and Anton van den Hengel. Memorizing normality to detect anomaly: Memory-augmented deep autoencoder for unsupervised anomaly detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1705–1714, 2019. 2
- [17] Denis Gudovskiy, Shun Ishizaka, and Kazuki Kozuka. Cflow-ad: Real-time unsupervised anomaly detection with localization via conditional normalizing flows. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 98–107, 2022. 1, 2
- [18] Yibin Huang, Congying Qiu, and Kui Yuan. Surface defect saliency of magnetic tile. *The Visual Computer*, 36(1):85–96, 2020. 8
- [19] T. Kailath. The divergence and bhattacharyya distance measures in signal selection. *IEEE Transactions on Communication Technology*, 15(1):52–60, 1967. 1
- [20] Longin Jan Latecki, Aleksandar Lazarevic, and Dragoljub Pokrajac. Outlier detection with kernel density functions. In *International Workshop on Machine Learning and Data Mining in Pattern Recognition*, pages 61–75. Springer, 2007. 2
- [21] Kimin Lee, Kibok Lee, Honglak Lee, and Jinwoo Shin. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. *Advances in neural information processing systems*, 31, 2018. 2
- [22] Sungwook Lee, Seunghyun Lee, and Byung Cheol Song. Cfa: Coupled-hypersphere-based feature adaptation for target-oriented anomaly localization. *arXiv preprint arXiv:2206.04325*, 2022. 2
- [23] Chun-Liang Li, Kihyuk Sohn, Jinsung Yoon, and Tomas Pfister. Cutpaste: Self-supervised learning for anomaly detection and localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9664–9674, 2021. 2
- [24] Wenqian Liu, Runze Li, Meng Zheng, Srikrishna Karanam, Ziyang Wu, Bir Bhanu, Richard J Radke, and Octavia Camps. Towards visually explaining variational autoencoders. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8642–8651, 2020. 2
- [25] Prasanta Chandra Mahalanobis. On the generalized distance in statistics. National Institute of Science of India, 1936. 2
- [26] Ankush Meshram and Christian Haas. Anomaly detection in industrial networks using machine learning: a roadmap. In *Machine learning for cyber physical systems: selected papers from the international conference MLACPS 2016*, pages 65–72. Springer, 2017. 1
- [27] Takeshi Nakazawa and Deepak V Kulkarni. Anomaly detection and segmentation for wafer defect patterns using deep convolutional encoder–decoder neural network architectures in semiconductor manufacturing. *IEEE Transactions on Semiconductor Manufacturing*, 32(2):250–256, 2019. 1
- [28] Pramuditha Perera, Ramesh Nallapati, and Bing Xiang. Ocgan: One-class novelty detection using gans with constrained

- latent representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2898–2906, 2019. 2
- [29] Stanislav Pidhorskyi, Ranya Almohsen, and Gianfranco Doretto. Generative probabilistic novelty detection with adversarial autoencoders. *Advances in neural information processing systems*, 31, 2018. 2
- [30] Tal Reiss, Niv Cohen, Liron Bergman, and Yedid Hoshen. Panda: Adapting pretrained features for anomaly detection and segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2806–2814, 2021. 2
- [31] Oliver Rippel, Patrick Mertens, and Dorit Merhof. Modeling the distribution of normal data in pre-trained deep features for anomaly detection. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 6726–6733. IEEE, 2021. 2
- [32] Karsten Roth, Latha Pemula, Joaquin Zepeda, Bernhard Schölkopf, Thomas Brox, and Peter Gehler. Towards total recall in industrial anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14318–14328, 2022. 1, 2
- [33] Marco Rudolph, Bastian Wandt, and Bodo Rosenhahn. Same same but different: Semi-supervised defect detection with normalizing flows. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 1907–1916, 2021. 8
- [34] Lukas Ruff, Robert Vandermeulen, Nico Goernitz, Lucas Deecke, Shoaib Ahmed Siddiqui, Alexander Binder, Emmanuel Müller, and Marius Kloft. Deep one-class classification. In *International conference on machine learning*, pages 4393–4402. PMLR, 2018. 2
- [35] Lukas Ruff, Robert A Vandermeulen, Nico Görnitz, Alexander Binder, Emmanuel Müller, Klaus-Robert Müller, and Marius Kloft. Deep semi-supervised anomaly detection. In *International Conference on Learning Representations*, 2019. 2
- [36] Mohammad Sabokrou, Mohammad Khaloeei, Mahmood Fathy, and Ehsan Adeli. Adversarially learned one-class classifier for novelty detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3379–3388, 2018. 2
- [37] Mayu Sakurada and Takehisa Yairi. Anomaly detection using autoencoders with nonlinear dimensionality reduction. In *Proceedings of the MLSDA 2014 2nd workshop on machine learning for sensory data analysis*, pages 4–11, 2014. 2
- [38] Thomas Schlegl, Philipp Seeböck, Sebastian M Waldstein, Ursula Schmidt-Erfurth, and Georg Langs. Unsupervised anomaly detection with generative adversarial networks to guide marker discovery. In *International conference on information processing in medical imaging*, pages 146–157. Springer, 2017. 2
- [39] Barry Siegel. Industrial anomaly detection: A comparison of unsupervised neural network architectures. *IEEE Sensors Letters*, 4(8):1–4, 2020. 1
- [40] Kihyuk Sohn, Chun-Liang Li, Jinsung Yoon, Minh Jin, and Tomas Pfister. Learning and evaluating representations for deep one-class classification. In *International Conference on Learning Representations*, 2020. 2
- [41] Jihoon Tack, Sangwoo Mo, Jongheon Jeong, and Jinwoo Shin. Csi: Novelty detection via contrastive learning on distributionally shifted instances. *Advances in neural information processing systems*, 33:11839–11852, 2020. 2
- [42] Chin-Chia Tsai, Tsung-Hsuan Wu, and Shang-Hong Lai. Multi-scale patch-based representation learning for image anomaly detection and segmentation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 3992–4000, 2022. 2
- [43] Shashanka Venkataraman, Kuan-Chuan Peng, Rajat Vikram Singh, and Abhijit Mahalanobis. Attention guided anomaly localization in images. In *European Conference on Computer Vision*, pages 485–503. Springer, 2020. 1, 2
- [44] Qian Wan, Cao YunKang, Liang Gao, Shen Weiming, and Xinyu Li. Position encoding enhanced feature mapping for image anomaly detection. In *2022 IEEE 18th International Conference on Automation Science and Engineering (CASE)*, 2022. 2
- [45] Jixuan Wang, Kuan-Chieh Wang, Marc T Law, Frank Rudzicz, and Michael Brudno. Centroid-based deep metric learning for speaker recognition. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3652–3656. IEEE, 2019. 4
- [46] Jie Yang, Yong Shi, and Zhiqian Qi. Learning deep feature correspondence for unsupervised anomaly detection and segmentation. *Pattern Recognition*, 132:108874, 2022. 1
- [47] Jihun Yi and Sungroh Yoon. Patch svdd: Patch-level svdd for anomaly detection and segmentation. In *Proceedings of the Asian Conference on Computer Vision*, 2020. 2
- [48] Jiawei Yu, Ye Zheng, Xiang Wang, Wei Li, Yushuang Wu, Rui Zhao, and Liwei Wu. Fastflow: Unsupervised anomaly detection and localization via 2d normalizing flows. *arXiv preprint arXiv:2111.07677*, 2021. 2
- [49] Ye Yuan, Wuyang Chen, Yang Yang, and Zhangyang Wang. In defense of the triplet loss again: Learning robust person re-identification with fast approximated triplet loss and label distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 354–355, 2020. 4
- [50] Bo Zong, Qi Song, Martin Renqiang Min, Wei Cheng, Cristian Lumezanu, Daeki Cho, and Haifeng Chen. Deep autoencoding gaussian mixture model for unsupervised anomaly detection. In *International conference on learning representations*, 2018. 2