COS 521 - Project Report

Yuxuan Zhang, Qiang Zhang, Weicong Dong

December 14, 2021

Abstract

Abstract: *Image Anomaly detection is an important research topic in computer vision due to its wide application in real-world scenarios (i.e., detecting defective products in manufacturing pipelines.) In this project, we aim to improve the PADIM method [8], one of the state-of-the-art image anomaly detection algorithms, using the Johnson-Lindenstrauss dimension reduction algorithm learned in the course. Our proposed approach demonstrates significant advantages: using the same computational budget as the PADIM method reported in their paper, our proposed method achieves significantly better performance; with the same performance requirement, our proposed method uses much less memory and allows for faster inference.*

1 Background and Introduction

Anomaly detection is a crucial task for computer vision applications due to its wide applications in real-world scenarios. To be more specific, typical use cases include 1. Addressing the safety issue in certain scenarios, like human-robot interactions; if there is something wrong with the camera sensor, we should detect the anomaly image immediately. Otherwise, the robot might be out of control and hurt humans nearby. Or to say, for autonomous driving vehicles, any missing in anomaly detection might cause a severe car accident. 2. Monitoring the quality of the product produced in the manufacturing factory, which requires a robust and efficient anomaly detection algorithm to find the inferior-quality product. 3. Controlling food safety, which exploits image anomaly detection methods to find overdue or degenerative food.

One of the SOTA solutions, PADIM [8], aims to capture the distribution of clean images by exploiting the features extracted by the convolutional neural network (CNN). However, the feature space is high dimensional, forcing the algorithm to perform sub-sampling and operate on a subspace of lower dimensions. This practice, as we show, is suboptimal as it leads to information loss. Towards solving this issue, in this report, we introduce a novel solution by using JL lemma to project the high-dimensional original data point into a low-dimension one. We first formulate the image anomaly detection problem as a high-dimensional distance measurement problem. Then we project the data item into low-dimensional space via JL projection, whose relative new distances among different points can be guaranteed by the JL lemma. After projection, we apply the Mixture of the Gaussian model to fit the parameters; the likelihood from this distribution is used to detect the anomaly images. With extensive experiments, we show that our method can be much faster than the original one to achieve the same detection performance. Moreover, within the same computation budget, our model can achieve consistently better performance.

2 Related Work

While image anomaly detection can be framed as a binary classification problem and approached using classification models, such practice is undesirable because positive samples (anomalies) are typically extremely rare in the real-world setting. With limited positive samples in training data, a classification model would fail to capture all existing patterns in minor class (anomaly class) and thus fail to generalize in the deployment environment. With such consideration, the mainstream anomaly detection methods would use no abnormal image for training. In other words, the models are trained with normal images only, aiming to capture the distribution of normal images and then compare it with the normal distribution to identify the anomaly. Current image anomaly detection methods can be categorized into four types: Reconstruction-based method, feature embedding-based method, Self-supervised learning method, and SVDD-based method.

1. **Reconstruction-based methods:** Reconstruction-based methods are widely used in practice due to their simplicity and scalability [6, 4, 10, 11, 12, 22, 14, 24] , where the anomaly is identified based on reconstruction error. Typically, an auto-encoder is trained (on normal images only) to capture the normal distribution. At inference time, the input image is reconstructed using the learned auto-encoder, and the reconstructed loss is calculated as the anomaly score. The intuition behind such methods is that abnormal images would lead to high reconstruction loss as the auto-encoder is trained only on normal images. Moreover, the pixel-based reconstruction loss can be used to localize the anomalies patterns, making these methods highly interpretable.

   Recent works also proposed adopting GAN (generative adversarial network) to exploit its strong ability to model image distribution [21, 17, 1]. At the inference time, reconstruction is performed through latent code optimization and, similarly, reconstruction loss is calculated and used as anomaly score. While these methods also lead to good detection performance, it's much more costly compared to the auto encoder-based methods, as the iterative steps must be performed to find the reconstruction latent code of the GAN. Because of the high inference latency and weak scalability, GAN-based methods are less commonly used in practice.

2. **Feature embedding-based methods:** While the distribution of the image itself is hard to model, meaningful image features can be extracted with deep learning approaches. With such inspiration, the feature embedding-based methods [19, 18, 2, 3, 5, 7, 26, 16] typically use a pre-trained CNN (i.e., on ImageNet) to extract features embedding, whose distribution is then modeled with an n-sphere or multivariate Gaussian. At the inference time, embedding for the input image is extracted, and its distance to the center of the normal embedding is measured and used as the anomaly score. Again, abnormal images are expected to have high distances to the center of normal distribution. As such methods essentially require no real "learning" process, a good performance can be achieved with limited training samples, which is common in industrial practices.

3. **Self-supervised learning methods:** Self-supervised learning methods aim to learn meaning patterns on unlabeled data using self-defined, programming-based labels. Such methods also inspire anomaly detection practices. For example, Golan et al. [9] randomly flips, rotates, and translates unlabeled images, and then trains a classifier to predict the particular type of transformation performed. At the inference time, the input image is predicted to be abnormal when the classifier does not provide a confident and correct prediction. The underlying assumption is that the learned classifiers lose confidence in the abnormal input images. Such methods work best when a large amount of unlabeled data is available.

4. **SVDD-based methods:** SVDD [23] is a classical anomalies detection method where the normal data is mapped to predefined kernel space. After that, the algorithm searches for the smallest hypersphere enclosing the normal data, primarily to ensure anomalies fall outside the learned hypersphere. Recent works [20] extend the method to image data by replacing kernel function with deep learning models, both enhancing detection performance and allowing anomaly localization.

3 JL Lemma Review

Much of the real-world data can be represented as high-dimensional points, such as images, sentences, audios, etc. The motivation behind JL lemma in the high-dimensional analysis is the fact that the algorithm complexity usually explodes exponentially as the dimension grows. According to [13], even if for the simple nearest neighbor search algorithm, it suffers from the high-dimension very much. Consequently, how to project the point into low-dimension space is very critical in many scenarios.

According to Wikipedia[25], the Johnson–Lindenstrauss lemma is a result concerning low-distortion embeddings of points from high-dimensional into low-dimensional Euclidean space, proposed by William B. Johnson and Joram Lindenstrauss. The lemma states that a set of points in a high-dimensional space can be embedded into a space of much lower dimension in such a way that distances between the points are nearly preserved. Here we rewrite this lemma as follows:

If $m = O(\log(1/\delta)/\epsilon^2$, then for any vector x, with probability (1-δ), we have:

$$(1-\epsilon)||x||_2^2 \leq ||\Pi x||_2^2 \leq (1+\epsilon)||x||_2^2 \tag{1}$$

This distance reservation property has many potential applications. In this report, we connect it with the anomaly detection task. The anomaly examples are defined by the distance between the selected sample and the center of the left other samples. Since the distance is preserved after the projection to the low-dimensional, it allows us to directly detect the anomaly examples in low-dimension space, with high probability.

There are some other formulations, one of them is its extension to the subspace embedding. This lemma says that, there is one d-dimensional linear subspace $\mathcal{U} \subset \mathbb{R}^n$, Π is any matrix satisfying the JL lemma, then if $m = O(\frac{d\log(1/\epsilon)+log(1/\delta)}{\epsilon^2})$, with probability $1-\delta$, we have:

$$(1-\epsilon)||x||_2^2 \leq ||\Pi x||_2^2 \leq (1+\epsilon)||x||_2^2 \tag{2}$$

The difference between this subspace embedding version and the original one lies in two aspects: one is that the projection matrix Π should be shared by all the points in the subspace, another one is that the low-dimension m is larger than the original one. For this report, we choose the subspace embedding version of the JL lemma.

When it comes to the projection matrix, there are many possible candidates. One common projection matrix is chosen from Gaussian distribution: firstly, choose the random unit vector from the sphere S^{m-1}; secondly, choose random unit vector from the space orthogonal to the first row, thirdly choose it from the space orthogonal to the first two rows... repeat this process until we finally get the whole projection matrix with shape $d \times m$. By sampling the matrix this way, some properties like the Spherical symmetry, Orthogonality, Normality can be satisfied. In this report, we choose this projection matrix as default.

4 Method Introduction

4.1 PADIM algorithm

We start by introducing the PADIM algorithm [8], one of the SOTA methods for anomaly detection, which we aim to improve. The method works by first modeling the distribution of clean images and then measuring the distance between the test image and the clean images to decide whether the input image is an anomaly or not. Specifically, it adopts a pre-trained convolutional neural network to extract image features and then models the distribution of image features with multivariate Gaussian distributions. At the testing time, it calculates the Mahalanobis distance between test image features and the learned distribution to decide the anomaly score.

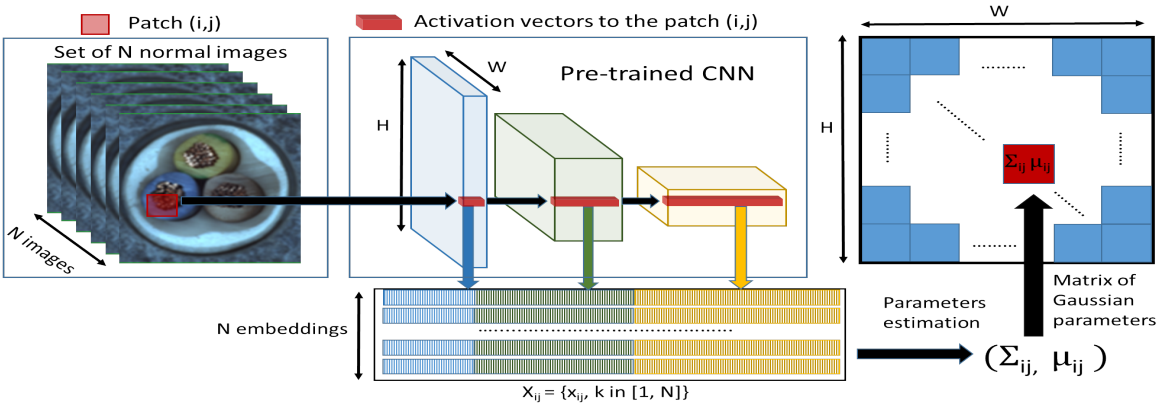

Figure 1: For each image patch corresponding to position (i, j) in the largest CNN feature map, PaDiM learns the Gaussian parameters (μ_{ij}, Σ_{ij}) from the set of N training embedding vectors $X_{ij} = \{x_{ij}^k, k \in [\![1, N]\!]\}$, computed from N different training images and three different pretrained CNN layers. The figure is extracted from PADIM paper

4.1.1 Training stage details

At the training stage, a pre-trained convolutional neural networks (i.e., Wide-ResBet pre-trained on ImageNet) is first used to extract the feature embeddings from images. Let $X_{ij} = \{x_{ij}^k, k \in [\![1, N]\!]\}$ denote the patch embedding vectors at position (i, j) for normal training image $1, 2, 3, ..., N$ as shown on Figure 1. The PADIM method models the distribution using the multivariate Gaussian distribution $\mathcal{N}(\mu_{ij}, \boldsymbol{\Sigma_{ij}})$, where the population mean μ_{ij} and covariance Σ_{ij} are estimated using the formula below :

$$\mu_{ij} = \frac{1}{N} \sum_{k=1}^{N} x_{ij}^k \tag{3}$$

$$\Sigma_{ij} = \frac{1}{N-1} \sum_{k=1}^{N} (x_{ij}^k - \mu_{ij})(x_{ij}^k - \mu_{ij})^{\mathrm{T}} + \epsilon I \tag{4}$$

where ϵI is the regularization term to ensure the estimated covariance matrix Σ_{ij} is of full rank and invertible. Note that the method fits a separate multivariate Gaussian distribution for each patch at different position (i, j). Note that the patch embedding vectors carry semantic information from different hierarchical levels. Therefore, the estimated multivariate Gaussian distribution $\mathcal{N}(\mu_{ij}, \boldsymbol{\Sigma_{ij}})$ captures hierarchical semantic information and $\boldsymbol{\Sigma_{ij}}$ captures the inter-level correlations.

4.1.2 Inference stage details

We adopt the Mahalanobis distance [15] $M(x_{ij})$ to measure the distance between the test patch embedding x_{ij} and learned distribution $\mathcal{N}(\mu_{ij}, \boldsymbol{\Sigma_{ij}})$, which is then used as the anomaly score. Specifically, $M(x_{ij})$ is computed as:

$$M(x_{ij}) = \sqrt{(x_{ij} - \mu_{ij})^T \Sigma_{ij}^{-1} (x_{ij} - \mu_{ij})} \tag{5}$$

Higher value $M(x_{ij})$ means that the test patch embedding is far away from the normal distribution, indicating a higher possibility of anomalous areas. The image-level anomaly score (anomaly of the entire image) is the maximum of anomaly map M (i.e., $\max\{M(x_{i,j}), \forall i, j\}$). The method is scalable as the anomaly score of the different patch (i, j) can be computed parallelly.

4.2 Proposed Improvement

While rich image features can be extracted with pre-trained CNN, the features space is of very high dimension (e.g., 6048 for wide-ResNet). Accordingly, the PADIM fails to exploit all these features due to time and physical constraints (i.e., inference time constraints, CPU & GPU memory constraints). Because of this, the PADIM method works by randomly sampling approximately 1/10 dimension of features space to use (550 randomly selected dimension in specific), meaning that the potentially useful information in the remaining 9/10 dimensions remains unexploited.

Motivated by this, we plan to improve the method using the dimensional reduction algorithm (Johnson-Lindenstrauss) learned in the course, allowing the algorithm to exploit all information in feature space to detect the anomaly patterns. Specifically, we prepared a transformation matrix Π as discussed in the course,

$$\Pi = \frac{1}{\sqrt{m}} G, \;\; G \sim \mathcal{N}(0, 1) \tag{6}$$

,where Π is of size (reduced_dimension, original_dimension). The original_dimension is 6048 for wide-ResNet and the reduced_dimension is set to be 550 to keep the same computation budget as the original paper. We then use Π to reduce the dimension of x_{ij}^k to get new ${x'_{ij}}^k$, specifically:

$${x'_{ij}}^k = \Pi x_{ij}^k \tag{7}$$

we then fit the multivariate Gassian distribution $\mathcal{N}(\mu'_{ij}, \boldsymbol{\Sigma'_{ij}})$ with

$$\mu'_{ij} = \frac{1}{N} \sum_{k=1}^{N} {x'_{ij}}^k \tag{8}$$

$$\Sigma'_{ij} = \frac{1}{N-1} \sum_{k=1}^{N} ({x'_{ij}}^k - \mu'_{ij})({x'_{ij}}^k - \mu'_{ij})^{\mathrm{T}} + \epsilon I \tag{9}$$

at the inference time, the distance is calculated as

$$M(x'_{ij}) = \sqrt{(x'_{ij} - \mu'_{ij})^T \Sigma_{ij}^{-1} (x'_{ij} - \mu'_{ij})} \tag{10}$$

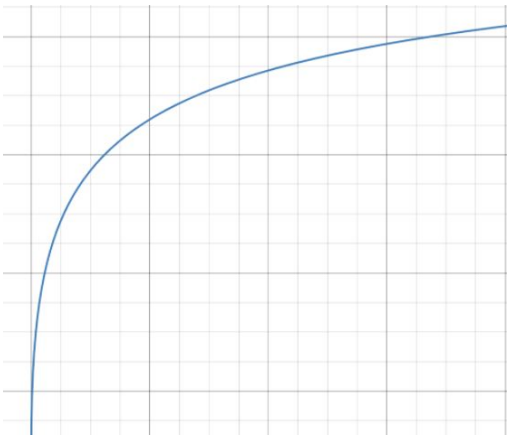

Figure 2: **Curve shape from our theory prediction.** X-axis represents the projected dimension, y-axis represents the anomaly detection performance. This curve provides the lowerbound for our method after dimension reduction.

4.3 Theoretical Prediction for Model Performance

Here we provide a brief proof for the theoretical guarantees of our proposed method. It is obvious to see that after we project the subspace into low dimension one, the performance drops. Here in the following, we deduct the theory to predict the relationship between the anomaly detection performance y and low-space dimension x, which guarantees that the performance drop is limited by a lower bound. We denote the low-space dimension as $x = O(\frac{d\log(1/\epsilon)+\log(1/\sigma)}{\epsilon^2})$. For each anomaly item in the original space, we define its distance to the dataset center as D_1. For each clean item in the original space, we define the distance as D_2. After projection, we define the corresponding distance in the low-dimensional space as d_1 and d_2. Then from JL subspace embedding, with a high probability $1-\delta$, we can have the following:

$$(1-\epsilon)D_1 < d_1 < (1+\epsilon)D_1, (1-\epsilon)D_2 < d_2 < (1+\epsilon)D_2 \tag{11}$$

The model will predict correctly if $D_1 > D_2$ is in the original space. Similarly, the model will confuse the output when $d_1 < d_2$. This only happens when there is an overlap between the interval of d_1 and the interval of d_2. For this case, we can estimate the new performance value via the following equation:

$$r = 1 - \frac{\max\{d_2\} - \min\{d_1\}}{\max\{d_1\} - \min\{d_2\}} = 1 - \frac{(1+\epsilon)D_2 - (1-\epsilon)D_1}{(1+\epsilon)D_1 - (1-\epsilon)D_2} \tag{12}$$

Define $k = (D_1 - D_2)/(D_1 + D_2)$, then we can get: $r = 2k/(\epsilon + k)$. Define p as the probability for the case that there is an overlap between interval for d_1 and the interval for d_2, then the new performance y can be calculated via expectation:

$$y = p \cdot r + (1-p) \cdot 1 = \frac{2pk}{\epsilon + k} + 1 - p,\ \epsilon \in (k, 1) \tag{13}$$

From the following equation

$$x = O(\frac{d\log(1/\epsilon) + \log(1/\sigma)}{\epsilon^2}) = \frac{q}{\epsilon^2} \tag{14}$$

We can get $\epsilon = \sqrt{q/x}$, then place it into the equation 13, we can get:

$$y = \frac{2pk}{\sqrt{q/x} + k} + 1 - p,\ x \in (q, \frac{q}{k^2}) \tag{15}$$

	(PADIM [8],Wide-ResNet)	(Proposed,Wide-ResNet)	(PADIM [8],ResNet18)	(Proposed,ResNet18)
Carpet	99.1	99.2	98.9	99.0
Grid	97.3	97.8	94.9	95.2
Leather	99.2	99.4	99.1	99.3
Tile	94.1	95.1	91.2	91.2
Wood	94.9	95.2	93.6	93.8
Bottle	98.3	98.7	98.1	98.4
Cable	96.7	97.3	95.8	96.0
Capsule	98.5	98.8	98.3	98.4
Hazelnut	98.2	98.4	97.7	97.8
Metal Nut	97.2	97.9	96.7	97.2
Pill	95.7	96.0	94.7	94.9
Screw	98.5	98.8	97.4	97.5
Toothbrush	98.8	99.1	98.7	98.9
Transistor	97.5	98.0	97.2	97.6
Zipper	98.5	98.7	98.2	98.3

Table 1: **Quantitative Comparison of PADIM vs. Proposed Method:** We quantitatively compare the proposed method with the PADIM baseline for all classes in MVTec AD [4] dataset. As results demonstrate, our proposed method shows consistent improvement.

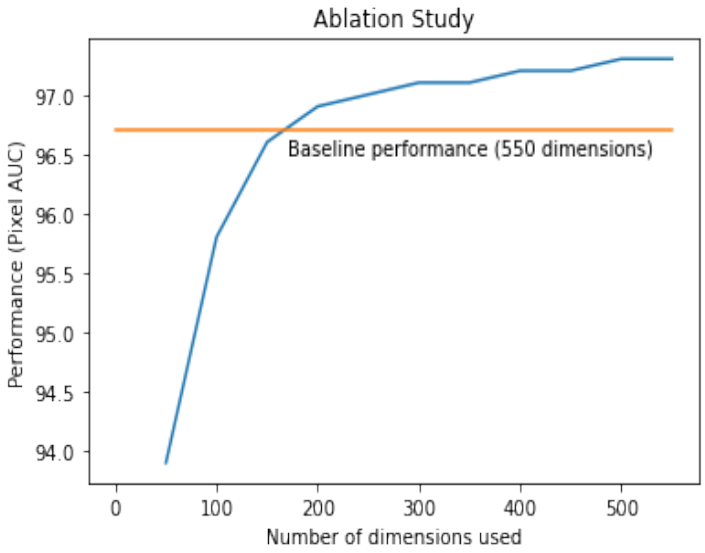

Figure 3: **Ablation Study:** In this ablation study, we show the number of dimensions used affects the performance of the proposed method. As the graph shows, the performance improves rapidly as the dimension used increases from 50 to 200, while the improvement is relatively stagnant after the dimension goes above 200. Note that for our method, we only need to use 150 dimensions to achieve the performance of the original PADIM method, which in comparison uses 550 dimensions. Also, using the same computational budget (using 550 dimensions), the proposed method improves the PADIM performance from 96.7% to 97.3%, which is a significant improvement considering that the maximum AUC score metric is 1.

Here p, q, k are some hyper-parameters determined by the property of the dataset. We can draw the predicted curve shape, which is visualized in Fig.2.

5 Experiment Results

Datasets. We evaluate the proposed method on the MVTec AD [4], the same dataset used in the PADIM paper. The dataset is collected from the real-world setting of industrial quality control. It contains 15 object classes, where each class has approximately 200-300 images. The image resolution is between 700×700 and 1024×1024. To facilitate anomaly localization, objects in each image are well-centered and aligned similarly across the dataset.

Qualitative Comparison & Analysis. We first set the reduced dimension to be 550, the same as the dimension size used in the PADIM paper. This practice ensures the comparison is fair as we use the same computation budget. We evaluate the model performance and report the pixel

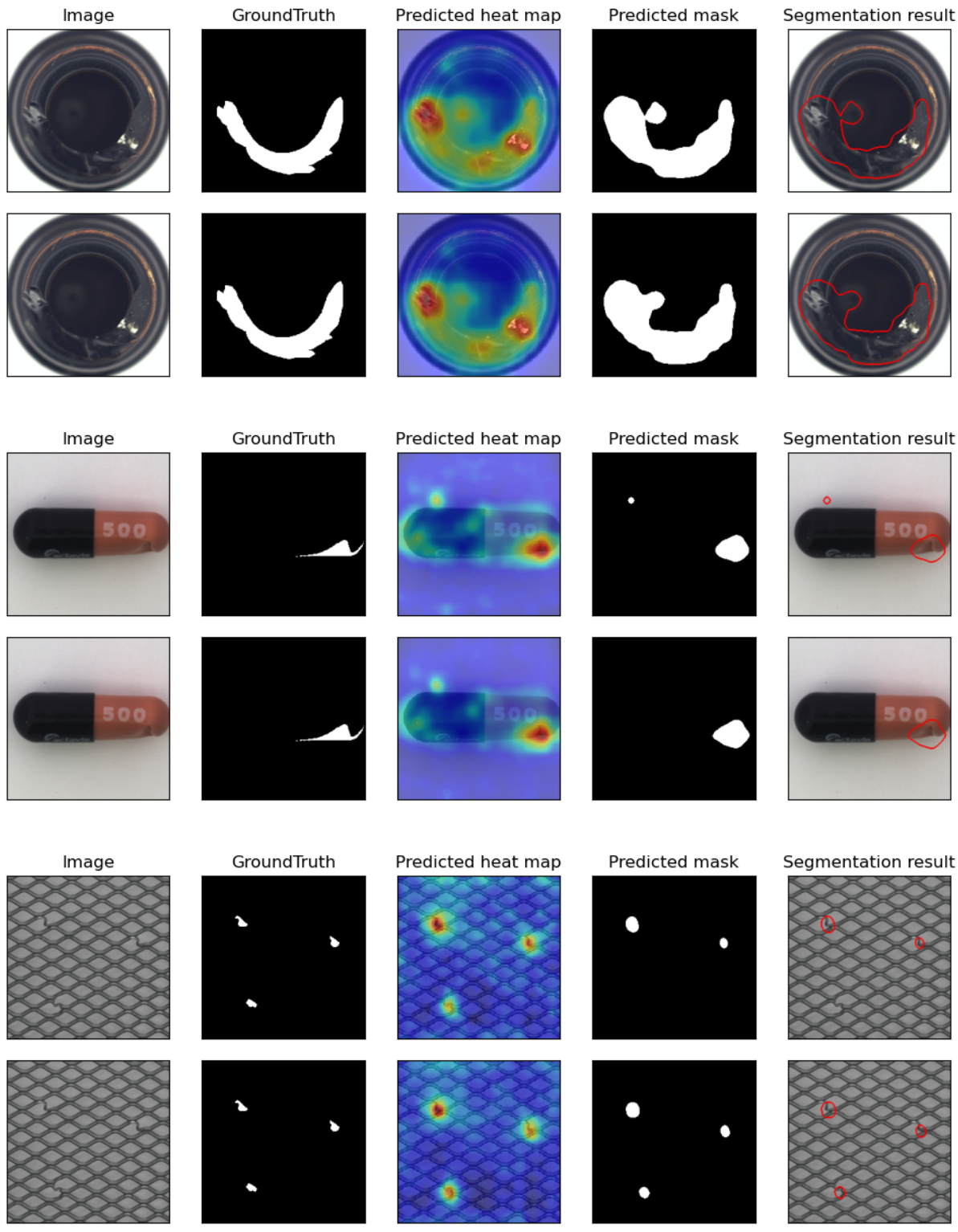

Figure 4: **Qualitative Comparison of PADIM vs. Proposed Method:** For each class, we show the predictions of the PADIM baseline in the first line and the predictions of the proposed method in the second line. As the result shows, our method provides more accurate localization for the anomaly.

AUC score in the table 1. As the results show, the proposed method significantly outperforms the original PADIM method in all classes. Note that the maximum value of AUC is 1 (100%), so the improvement from 97.3% to 97.8% for Grid class or 99.2% to 99.4% for Leather class should be considered as significant.

Quantitative Comparison & Analysis. We also qualitatively compare our method with baseline PADIM in figure 4. For each class, we show the predictions of the PADIM baseline in the first line and the predictions of the proposed method in the second line. As the result shows, our method provides more accurate localization. Specifically, our predicted mask is more accurately aligned with the ground truth mask and our predicted heatmaps fire less on the non-anomaly region.

Ablation study. We further ablate on the number of dimensions used for the reduced feature embedding to evaluate how the model performance changes with the increase/decrease of feature dimension. A larger dimension would enable to less information loss and accordingly better performance, but also lead to slower inference as more computation is needed. As the figure 3 shows, the performance improves rapidly as the dimension used increases from 50 to 200, while the improvement is relatively stagnant after the dimension goes above 200. Note that for our method, we only need to use 150 dimensions to achieve the performance of the original PADIM method, which in comparison uses 550 dimensions. This leads to approximately $10\times$ saving for memory space and $3\times$ improvement for inference speed. Also, using the same computational budget (using 550 dimensions), the proposed method improves the PADIM performance from 96.7% to 97.3%, which is a significant improvement considering that the maximum AUC score metric is 1.

6 Conclusion

In this report, we propose a novel method to improve the PADIM algorithm, one of the state-of-the-art algorithms for image anomaly detection, by exploiting the JL Lemma. The extensive experiment shows that the proposed algorithm significantly outperforms the original PADIM method for both detection performance and inference speed.

References

[1] Samet Akcay, Amir Atapour-Abarghouei, and Toby P. Breckon. Ganomaly: Semi-supervised anomaly detection via adversarial training. *ACCV*, 2018.

[2] Liron Bergman, Niv Cohen, and Yedid Hoshen. Deep nearest neighbor anomaly detection. In *arXiv, 2002.10445*, 2020.

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

[4] P. Bergmann, M. Fauser, D. Sattlegger, and C. Steger. Mvtec ad–a comprehensive real-world dataset for unsupervised anomaly detection. In *CVPR*, 2019.

[5] P. Bergmann, M. Fauser, D. Sattlegger, and C. Steger. Uninformed students: Student-teacher anomaly detection with discriminative latent embeddings. In *CVPR*, 2020.

[6] P. Bergmann, S. Löwe, .M. Fauser, D. Sattlegger, and C. Steger. Improving unsupervised defect segmentation by applying structural similarity to autoencoders. In *VISIGRAPP*, 2019.

[7] N. Cohen and Y. Hoshen. Sub-image anomaly detection with deep pyramid correspondences. In *arXiv, 2005.02357*, 2020.

[8] Thomas Defard, Aleksandr Setkov, Angelique Loesch, and Romaric Audigier. Padim: a patch distribution modeling framework for anomaly detection and localization, 2020.

[9] Izhak Golan and Ran El-Yaniv. Deep anomaly detection using geometric transformations. In *NeurIPS*, 2018.

[10] D. Gong, L. Liu, V. Le, B. Saha, M. R. Mansour, S. Venkatesh, and A. van den Hengel. Memorizing normality to detect anomaly: Memory-augmented deep autoencoder for unsupervised anomaly detection. In *ICCV*, 2019.

[11] Chaoqin Huang, Fei Ye, Jinkun Cao, Maosen Li, Ya Zhang, and Cewu Lu. Attribute restoration framework for anomaly detection. In *arXiv, 1911.10676*, 2019.

[12] D. P Kingma and M. Welling. Auto-encoding variational bayes. In *ICLR*, 2014.

[13] Jon M Kleinberg. Two algorithms for nearest-neighbor search in high dimensions. In *Proceedings of the twenty-ninth annual ACM symposium on Theory of computing*, pages 599–608, 1997.

[14] W. Liu, R. Li, M. Zheng, S. Karanam, Z.Wu, B. Bhanu, R. J. R., and O. Camps. Towards visually explaining variational autoencoders. In *CVPR*, 2020.

[15] P.C. Mahalanobis. On the generalized distance in statistics. In *National Institute of Science of India*, 1936.

[16] Schettini R. Napoletano P, Piccoli F. Anomaly detection in nanofibrous materials by cnn-based self-similarity. In *Sensors.*, volume 18, page 209, 2018.

[17] S. Pidhorskyi, R. Almohsen, D. A Adjeroh, and G. Doretto. Generative probabilistic novelty detection with adversarial autoencoders. In *NIPS*, 2018.

[18] O. Rippel, P. Mertens, and D. Merhof. Modeling the distribution of normal data in pre-trained deep features for anomaly detection. In *arXiv, 2005.14140*, 2020.

[19] L. Ruff, R. Vandermeulen, N. Goernitz, L. Deecke, S. A. Siddiqui, A Binder, E. Müller, and M. Kloft. Deep one-class classification. In *ICLM*, 2018.

[20] Lukas Ruff, Robert Vandermeulen, Nico Goernitz, Lucas Deecke, Shoaib Ahmed Siddiqui, Alexander Binder, Emmanuel Müller, and Marius Kloft. Deep one-class classification. In *ICML*, 2018.

[21] M. Sabokrou, M. Khalooei, M. Fathy, and E. Adeli. Adversarially learned one-class classifier for novelty detection. In *CVPR*, 2018.

[22] K. Sato, K. Hama, T. Matsubara, and K. Uehara. Predictable uncertainty-aware unsupervised deep anomaly segmentation. In *IJCNN*, 2019.

[23] David MJ Tax and Robert PW Duin. Support vector data description. *Machine learning*, 54(1):45–66, 2004.

[24] S. Venkataramanan, K-C Peng, R. V. Singh, and A. Mahalanobis. Attention guided anomaly localization in images. In *arXiv, 1911.08616*, 2019.

[25] Wikipedia contributors. Johnson–lindenstrauss lemma — Wikipedia, the free encyclopedia. https://en.wikipedia.org/w/index.php?title=Johnson%E2%80%93Lindenstrauss_lemma&oldid=1056837475, 2021. [Online; accessed 9-December-2021].

[26] J. Yi and S. Yoon. Patch svdd: Patch-level svdd for anomaly detection and segmentation. In *arXiv, 2006.16067*, 2020.