Adapted-MoE: Mixture of Experts with Test-Time Adaption for Anomaly Detection

Tianwu Lei[1‡], Silin Chen[1‡], Bohan Wang[1], Zhengkai Jiang[3] and Ningmu Zou[1,2 §]

[1] School of Integrated Circuits, Nanjing University
{tianwulei,silin.chen,bohanwang}@smail.nju.edu.cn
nzou@nju.edu.cn
[2] Interdisciplinary Research Center for Future Intelligent Chips (Chip-X), Nanjing University Suzhou, China
[3] Hong Kong University of Science and Technology
kaijiang.jzk@gmail.com

Abstract. Most unsupervised anomaly detection methods learn a single decision boundary to distinguish samples in the training dataset, neglecting the variation for normal samples even in the same category and a distribution bias exists between the test set and the train set in the real world. Therefore, we propose an Adapted-MoE which contains a routing network and a series of expert models to handle multiple distributions of same-category samples by divide and conquer. Specifically, we propose a routing network to route same-category samples into the subclasses feature space. Then, a series of expert models are utilized to construct several independent decision boundaries. We propose the test-time adaption to eliminate the bias between the unseen test sample representation and the feature distribution learned by the expert model. Our experiments are conducted on a dataset that provides multiple subclasses from three categories, namely Texture AD. The Adapted-MoE significantly improves the performance of the baseline model, achieving 2.18%-7.20% and 1.57%-16.30% increase in I-AUROC and P-AUROC, which outperforms the current state-of-the-art methods. Our code is available at `https://github.com/ray3572/AdaptedMoE`.

Keywords: Anomaly Detection, Mixture of Experts, Test-Time Adaption, Unseen Sample, Distribution Bias.

1 Introduction

Anomaly detection recognizes anomalous images and detects anomalous regions, which is a essential method in industrial quality applications [2, 9]. Because obtaining and labeling anomalous samples is difficult in the real world, unsupervised anomaly detection (UAD) which discriminates outliers by learning normal sample

---

‡ These authors contributed equally to this work.

§ Corresponding Author

features has gradually become the focus of research [4, 8, 16]. Motivated by the fact that normal samples are easy to collect, many methods learn the features distribution of normal samples by reconstructing them recently [13,19].These methods assume that the reconstruction network can distinguish between representations of anomalous samples based on distributions learned from normal samples, thereby establishing the decision boundary. Other methods leverage synthetic anomalous images to train deep learning models, enabling them to learn discriminative image features effectively [18]. These methods intensely depend on the quality of the synthetic anomaly images as well as on more empirical knowledge about the defect patterns. Some methods also use memorybank [5,11,14] to store features of normal samples and discriminate anomalous samples by calculating feature similarity. These methods ignore the existence of unseen samples within the testing process. We summarize these current methods as shown in Fig. 1, where the methods uniformly learn representations distribution for normal samples and build a single decision boundary in the same category based on the distribution. In the test time, samples outside the decision boundary are considered anomalous samples.

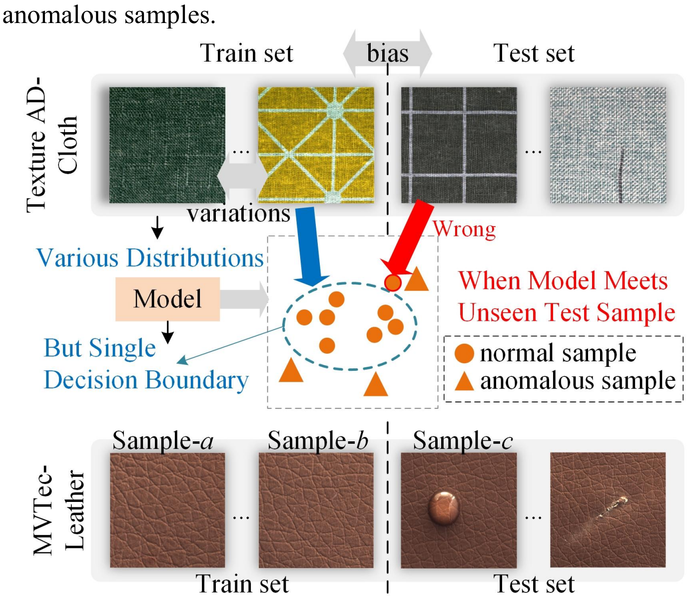

Fig. 1. Existing methods construct single decision boundary by learning representations of normal samples, ignoring variations in the feature distribution of samples within the same category as shown in the Texture AD-Cloth [7]. Moreover, the test dataset still has a massive distribution of unseen samples. Existing datasets (e.g., MVTec AD dataset [2]) in which similar samples are all in the same distribution are illustrated by Sample-a, Sample-b, and Sample-c.

The aforementioned methods demonstrate optimal performance due to the consistency in the training datasets and exhibit minimal distribution bias between the train set and test set (e.g. Sample-a, Sample-b, and Sample-c from MVTec in Fig.1). However, the real samples are affected by variations in the lighting conditions, equipment, camera position, and other factors during the acquisition process. It results in a variation in the distribution of the samples used to learn the representations, as well as the samples to be detected. As shown in Fig. 1, practical applications suffer from a large number of samples in the same category that are still "novel type" (e.g. different color, material in Texture AD-Cloth) exacerbating the variation in the train set. Furthermore, it is possible that the test data and the training data, which belong to the same category, may exhibit distribution bias. Unseen normal samples may be projected outside the single decision boundary, potentially leading to significant inaccuracies. In this paper, we formulate the mentioned issue in terms of two definitions. (1) Various complicated feature distributions exist in the training samples. As shown in Fig. 1, samples in Texture AD-Cloth are collected from the same category (cloth), but each sample is in a completely independent data distribution due to color and material differences. It indicates that a single decision boundary in the training process is not sufficient to distinguish all samples of the same category.(2) Distribution bias in the test set and train set for normal and anomalous samples. As shown in Fig. 1, the test set samples are unseen compared to the training set in Texture AD-Cloth. The application of the decision boundary derived from the training samples has been demonstrated to result in inconsistencies and inaccuracies.

As a significant number of real samples are excluded within the dataset, we propose a new method called Adapted Mixture of Experts (Adapted-MoE) to solve the above issue. We assume that a random normal sample has a distribution with a certain pattern in the feature space. We leverage the mean and variance of the normal samples to unify the feature embeddings under the same distribution as the learned specific pattern before inputting them into the expert model via a normalization method. The major contributions of this paper are summarized as follows:

- To our knowledge, our proposed Adapted-MoE firstly investigates the challenging problem of variation in the train set and bias between the train set and test set for anomaly detection.
- We propose a MoE model for learning normal sample feature distribution for different subclasses. Moreover, we also designed a routing network based on representation learning to distinguish normal samples. A simple and effective test-time adaption is proposed to solve the unseen sample bias in the testing process.
- We conduct extensive experiments to confirm the effectiveness of the Adapted-MoE on the Texture AD benchmark. The experimental results show that the proposed method significantly outperforms the previous state-of-the-art.

2 Proposed Method

The proposed Adapted MoE is elaborately introduced in this section. As shown in Fig. 2, Adapted-MoE consists of a feature extractor, a routing network, test-time adaption, and several expert models. Specifically, We adopt fixed pretrained CNNs on ImageNet [3] as the feature extractor. The features from several stages are collected. Then these features are resized to the same size and concatenated across channel dimensions to restructure the feature maps. Subsequently, the expert model with the highest correlation is assigned via the routing network. The test-time adaptation method is then employed to transfer the feature to the space that can be handled by the selected expert model. Finally, anomaly detection is achieved by the expert model.

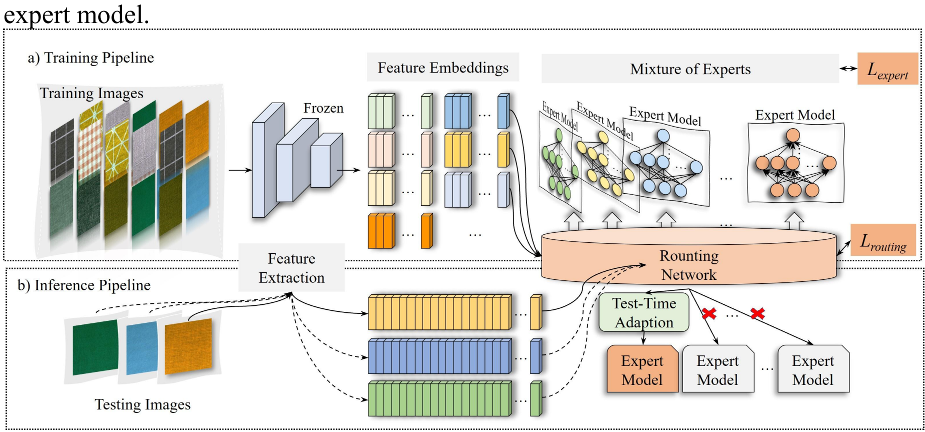

Fig. 2. Overview of Adapted-MoE. First a frozen backbone is employed to conduct feature extraction on the samples. Subsequently, the extracted feature embeddings are divided into different expert models for training through a routing network, where the training loss consists of the routing loss $L_{routing}$ and the loss of the expert model L_{expert}. In the testing phase, Test-Time Adaption calibrates the routed features to eliminate distribution bias before anomaly detection.

2.1 Mixture of Experts

Most anomaly detection methods construct the feature space based on normal samples. However, the feature distribution of normal samples in the same category is still diverse, and a single decision boundary will lead to an inaccurately determined outcome. Therefore, we propose a mixture expert model to divide normal samples from the same category into multiple expert models to learn the different feature distributions of multiple subclasses during the training process.Firstly, given the i_{th} training sample's feature maps $X_i \in R^{C\times H\times W}$ where C, H and W represent the channels, height and width of feature maps, feature embedding xi are firstly obtained by a projection layer and a global average pooling(GAP) layer f^{GAP} . The projection layer is composed of a 3 × 3 convolution W_i, b_i to projection feature maps from ImageNet to the anomaly detection feature space:

$$x_i = f^{GAP}(\sigma((\mathbf{W}_i X_i + \mathbf{b}_i)) \in \mathbb{R}^C \tag{1}$$

Inspired by [15], we classify m training samples using a designed center loss in our routing network.

$$L_{routing} = \sum_{i=0}^{m} \alpha \parallel x_i - c_k \|^2 - (1-\alpha) y_i \log\left(\frac{e^{w_i x_i}}{\sum_j^n e^{w_j x_i}}\right) \quad (2)$$

where m represents the mini−batch in the training process, c_k denotes the center of the k_{th} subclass in the training set and is updated per steps, y_i represents the subclass label for i_{th} normal sample, n is the number of subclass in the training set and also the number of experts, $w \in R^{C\times n}$ is the classifier matrix and α is weight adjustment parameter, with a value range of 0 to 1. As shown in Fig. 3, the samples in the same subclasses are converged to the center of the subclasses by minimizing the above objective function, and the samples from different subclasses will be far away from each other in the feature space. During the inference process, x_i is routed to the expert model with maximize $xi * c_k$ which denotes the cosine distance between x_i and c_k, and the final score will be calculated by the softmax function.

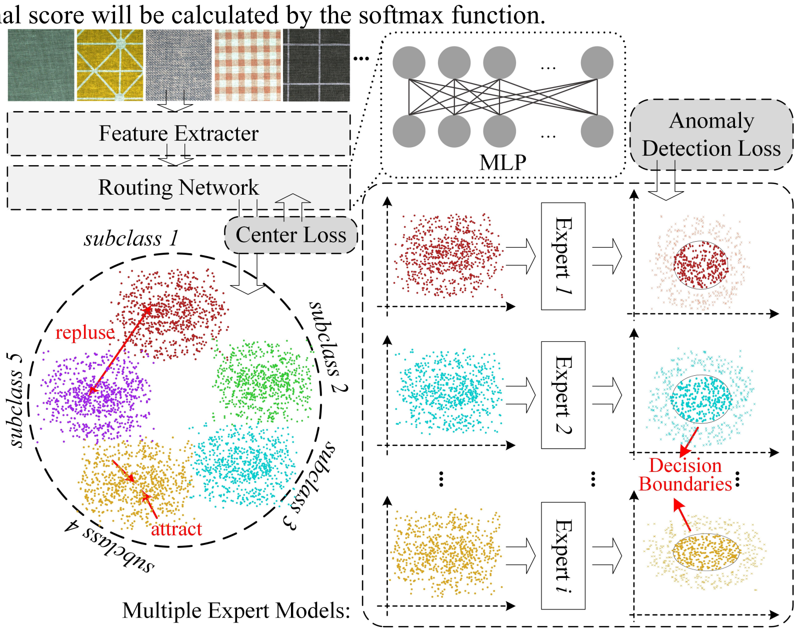

Fig. 3. Mixture of Experts. For a mini-batch of feature embeddings, the center loss is utilized in the routing network to divide them into different subclasses during the training process. Simple expert models construct multiple decision boundaries in independent feature spaces for different subclasses.

After obtaining the feature embedding of the subclasses, we simply design a multi-layer perceptron as an expert model to construct decision boundary for independent subclass. We use feature embedding to randomly generate noise vectors and train expert models based on synthetic anomaly detection methods and the loss of expert L_{expert} same as [10]. The total loss L_{total} is described by:

$$L_{total} = L_{routing} + \sum_{i=1}^{m} L_{expert}^{i} \tag{3}$$

Ultimately, the final anomaly detection score is obtained by aggregating the results of multiple expert models as follows:

$$result = \frac{\sum_{i}^{k} w_i x_{test}^{i} Expertt(x_{test}^{i})}{\sum_{i}^{k} w_i x_{test}^{i}} \tag{4}$$

Normalization. It is worth emphasizing that due to the similarity of the anomaly detection samples, the feature distribution of the different subclasses that are projected into the feature space is not uniform [12]. Therefore, we adopt the normalization to constrain the value range of the feature embedding x_i. It effectively separates the feature of different subclasses so that they can be more evenly distributed in the feature space, which can be expressed as $x_i = \frac{x_i}{\|x_i\|}$. As mentioned above, the routing network is scored by cosine similarity and softmax of the classifier matrix w and feature embedding x_i. Benefiting from the monotonicity of softmax, the normalized feature embedding does not affect the routing score.

2.2 Test-Time Adaption

As shown in Fig. 4, the feature space learned by the expert model based on the existing training set still suffers from a bias in the feature distribution of the unseen subclasses. We assume that the feature distributions of the unseen subclasses have a certain feature distribution in feature space. Thus their decision boundaries can be obtained by simply eliminating the inconsistency of the feature distributions in the inference process. In this paper, we define this bias as distribution distance and propose a test-time adaptation method to eliminate the bias between unseen samples and training samples.

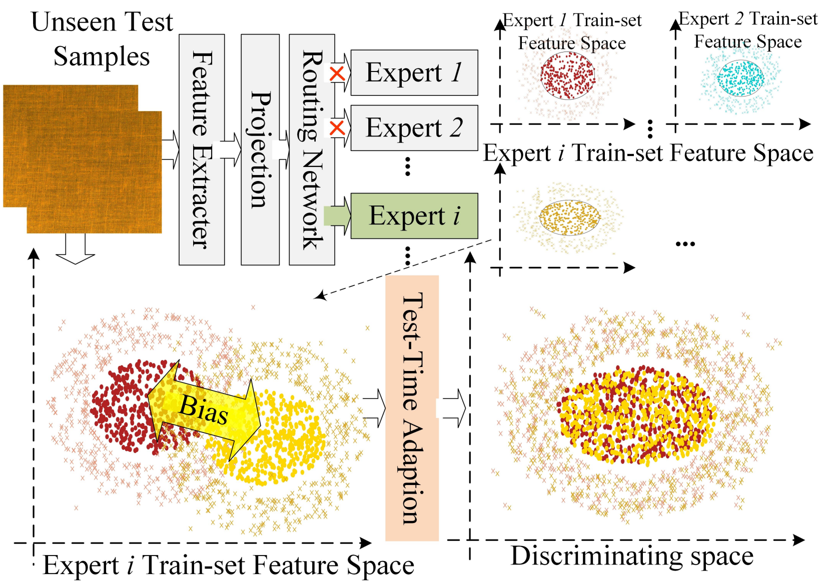

Fig. 4. Test-Time Adaptation. Since the test samples do not appear in the training phase, the distribution of the samples at the testing has bias with the distribution of the samples learned by the expert model. We eliminate the distance between the two distributions by Test-Time Adaptation, to unify the position of the decision boundary.

Firstly, given the feature embedding x_{test} of the test sample, the closest subclass center embedding c_k of the test sample in the feature space can be found by the routing network. As shown in Fig. 4, the test sample distribution and the learned distribution are similar but have a distance gap. Since the k_{th} expert model is based on training data with center c_k and standard deviation which is denoted as *std* to construct the decision boundary. Therefore, we calibrate the distribution of test embeddings x_{test} to the feature space of the k_{th} expert model to unify the decision boundary:

$$\mathrm{x'}_{\mathrm{test}} = \frac{(\mathrm{x}_{\mathrm{test}} - \mathrm{mean}(\mathrm{x}_{\mathrm{test}})) \times \mathrm{std}}{\mathrm{std}(\mathrm{x}_{\mathrm{test}})} + \mathrm{c}_{\mathrm{k}} \tag{5}$$

$$\mathrm{x'}_{\mathrm{test}} = \frac{\mathrm{x}_{\mathrm{test}}}{\|\mathrm{x}_{\mathrm{test}}\|} \tag{6}$$

We use the center of the training data to make the feature distributions with the same measure by mean and variance and subsequently normalize the corrected embeddings x'_{test} to obtain the final decision boundary.

3 Experiments

3.1 Datasets and Metrics

Datasets As shown in Fig. 1, existing datasets sampling data are similarly distributed in the same category (e.g., MVTec [2]). To validate the proposed Adapted-MoE, we use the dataset named Texture AD benchmark [7] in the experiments. The Texture AD benchmark§ is an anomaly detection dataset, which contains sampled images and defect annotations for three categories, cloth, metal, and wafer. Significantly, the Texture AD dataset provides multiple different types in subclasses under each category providing samples of various distributions. To validate our method, we choose 10 subclasses for training and 5 unseen subclasses for testing in the cloth dataset, 4 unseen subclasses in the wafer dataset and 3 unseen subclasses in the metal dataset. All images in this dataset are captured using a high-resolution industrial camera (MV-CS200-10 GC) at 5472 × 3648 pixels and cropped to 1024 × 1024.

Metrics For anomaly detection results, we use the Area Under the Receiver Operating Curve (AUROC) to evaluate our proposed model comprehensively same as other works. Image-level anomaly detection performance is measured via the standard AUROC, denoted as I-AUROC. Moreover, a pixel-level AUROC (P-AUROC) is used to evaluate the anomaly localization.

3.2 Implementation Details

All experiment codes are implemented based on the Pytorch framework and all the models are trained with one NVIDIA GeForce RTX 4080 (16 GB memory) for acceleration. We validated the effectiveness of the Adapted-MOE using SimpleNet [10] as our baseline. For the baseline, a pretrained WideResNet50 [17] is used already as a feature extractor which is frozen in both training and testing processes. For fair comparisons, the SimpleNet with Adapted-MoE is trained for 160 epochs with a batch size of 8 and the learning rate is from 0.0001 to 0.0002. In Gaussian noise $N(0, \sigma^2)$, σ is set by default to 0.015.

3.3 Comparisons with State-Of-The-Arts

We compare the proposed Adapted-MoE with a number of state-of-the-art approaches on Texture AD benchmark, including SimpleNet [10], EfficientAD [1], PyramidFlow [6], DREAM [18], Mean-shifted [12] and MSFlow [20]. Firstly, we compared the performance of anomaly detection. As shown in Table 1, excellent results are achieved by our Adapted-MoE on most of the unseen subclasses in three categories. Moreover, our proposed method outperforms other methods in average I-AUROC accuracy on the test set of cloth, wafer, and metal by 67.53%, 58.58%, and 66.12%, respectively. To further demonstrate the excellence of our method, we secondly compare the capability of anomaly localization on novel unseen data. We compare the values of P-AUROC with state-of-the-art methods on three categories in Texture AD. The results show that our method outperforms existing methods in unseen subclass performance for each category as well as average accuracy. The average P-AUROC of our proposed method is 76.05%, 63.40%, and 73.76% for cloth, wafer, and metal. The results of visualization compared with SOTA are detailed in the Appendix.

Table 1. Image-AUROC (%)/Pixel-AUROC (%) comparison with the state-of-the-art methods on Texture AD dataset.

Category	subclass	SimpleNet *(2023)*	PyramidFlow *(2023)*	DRAEM *(2021)*	Mean-Shift *(2023)*	MSFlow *(2024)*	EfficientAD *(2024)*	Ours
Cloth	*subclass1*	65.08/58.30	57.88/68.00	57.58/60.99	**66.22**/-	50.00/56.11	65.65/62.76	57.98/**80.94**
	subclass2	59.26/51.52	63.18/57.06	50.21/65.36	33.66/-	54.01/63.14	**76.98**/58.92	62.31/**68.41**
	subclass3	58.83/63.48	60.74/60.74	55.44/56.91	66.21/-	50.00/51.66	55.69/47.08	**84.61**/**74.43**
	subclass4	**70.40**/70.68	59.39/57.26	58.01/53.45	65.69/-	50.00/47.44	42.38/38.75	60.77/**76.50**
	subclass5	68.47/54.47	49.72/34.84	55.95/77.03	39.54/-	50.14/42.23	**72.20**/61.77	71.96/**79.95**
	Average	64.41/59.69	58.18/55.58	55.44/62.75	54.26/-	50.83/52.12	62.58/53.86	**67.53**/**76.05**
Wafer	*subclass1*	52.11/57.18	55.54/51.23	55.69/44.91	52.83/-	51.19/44.91	50.28/55.76	**67.30**/**60.74**
	subclass2	**59.66**/**66.16**	43.35/39.47	57.09/34.10	53.29/-	49.78/34.10	42.25/33.98	55.95/60.81
	subclass3	53.66/57.58	52.76/51.52	**59.22**/35.01	55.44/-	53.64/35.01	50.23/51.53	51.71/**65.40**
	subclass4	50.68/53.40	46.36/44.63	52.46/43.59	48.28/-	50.00/43.59	45.51/40.02	**59.36**/**66.63**
	Average	54.03/58.58	49.50/46.71	56.12/39.40	52.47/-	51.15/39.40	47.07/39.40	**58.58**/**63.40**
Metal	*subclass1*	59.07/62.27	52.87/53.42	52.07/58.31	44.34/-	62.90/65.37	65.27/59.69	**65.60**/**73.98**
	subclass2	59.87/58.33	48.74/48.86	56.32/51.53	47.39/-	53.54/57.34	55.46/51.04	**66.19**/**69.48**
	subclass3	57.83/58.97	58.92/57.67	51.48/57.31	45.04/-	59.78/60.37	**68.73**/54.91	66.57/**77.81**
	Average	58.92/59.86	53.51/53.31	53.29/55.75	45.59/-	58.74/61.02	63.30/55.21	**66.12**/**73.76**

3.4 Ablation Studies

In this section, we present ablation studies on the proposed method, including the structure of Adapted-MoE, the *top k* number of MoE and the choice of the loss function in the routing network. The baseline for all ablation experiments in this section is SimpleNet.

Table 2. Ablation Study of Structure for Adapted-MoE.

+MoE	+TTA	+Norm	Average P-AUROC(%)		
			Cloth	Wafer	Metal
-	-	-	59.69	58.58	59.86
√	-	-	53.93(-5.76)	58.10(-0.48)	60.27(+0.41)
-	√	-	65.96(+6.27)	59.26(+0.68)	**60.15(+1.57)**
√	-	√	56.88(-2.81)	56.23(-2.35)	60.56(+0.70)
-	√	√	74.36(+14.67	59.73(+1.15)	63.17(+3.31)
√	√	-	61.45(+1.76)	54.42(-4.16)	56.74(-3.12)
√	√	√	**76.05(+16.3)**	**60.15(+1.57)**	69.10(+9.24)

Table 3. P-AUROC(%) / I-AUROC(%) for Loss Choices.

Loss	Cloth	Wafer	Metal
Softmax	74.92/55.48	58.97/53.24	**69.14**/54.92
CenterLoss	**76.05/67.53**	**60.15/56.21**	69.10/**55.50**

The Structure of Adapted-MoE. To verify the validity of our proposed method, we conducted ablation experiments on cloth data, wafer data and metal data in the Texture AD dataset. As shown in Table 2, using the MoE individually ignores the bias between the distribution of test samples and the distribution of samples that have been learned, leading to shortcomings in anomaly detection and anomaly location. In cloth data and wafer data, independent usage of Test-Time Adaption for feature embeddings can improve anomaly location performance, but the anomaly detection capability is greatly reduced due to the feature embeddings are not well assigned to the corresponding subclass space. Due to the small inter-class differences of the subclasses in the metal data(proved by visualization in the Appendix), MoE and normalization will lead to the wrong division of the subclasses into subspaces and only Test-Time Adaptation is needed to bring accuracy increase, 13.9%/7.20% of average P-AUROC and I-AUROC. Using both MoE and Test-Time Adaption will make the distribution of test samples not normalized correctly. Therefore, we introduced all the proposed methods into the baseline, which eventually improved 16.3%/3.12% and 1.57%/2.18% of average P-AUROC and I-AUROC on the cloth and wafer datasets, respectively. More details ablation results of subclasses can be found in the Appendix.Furthermore, we provide the ablation experimental results of the final proposed method for all subclasses in the cloth dataset. As shown in Fig. 5, the implementation of our proposed method improves the performance of anomaly detection in most of the subclasses by up to 25.48%. For anomaly location, our approach improves the performance of all subclasses with a maximum improvement of 25.48% and a minimum improvement of 5.82%.

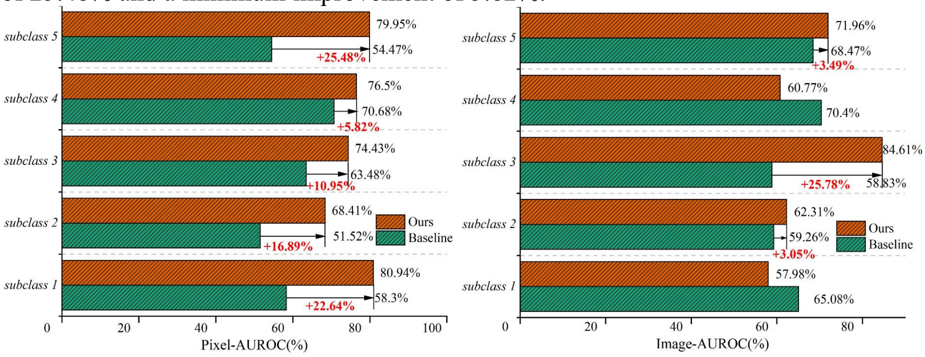

Fig. 5. Ablation experiments for subclasses on cloth dataset. For the I-AUROC metric, our method improves on some unseen subclasses by 3.05%-25.78%. For the P-AUROC metric, our method improves on all unseen subclasses by 5.82%-25.48%.

The *Top k* for Mixture of Experts. The routing network identifies the expert model that is most closely associated with the test data, thereby minimizing the distance between the test samples. Furthermore, this approach can be employed to select the *Top k* expert models that are most closely aligned with the test data. As shown in Fig. 6, we perform ablation experiments on the cloth dataset for *Top k* expert model choices. The results show that in selecting Top4 expert models is more beneficial to the overall model performance.

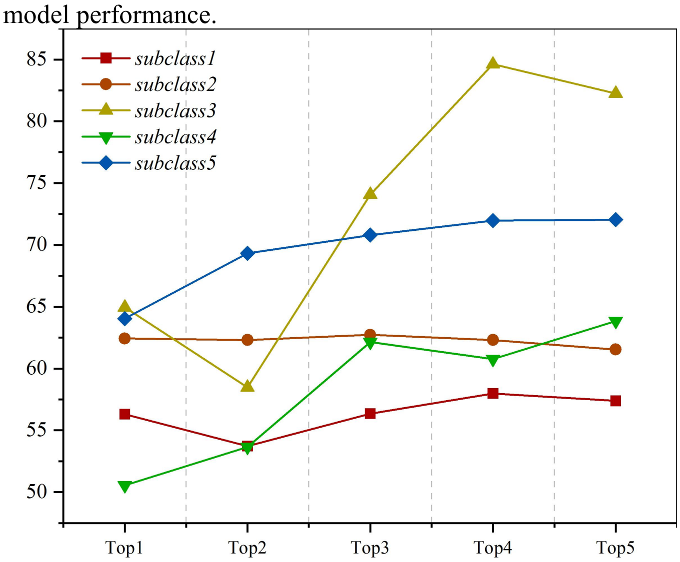

Fig. 6. Ablation experiments for *Top k* on cloth dataset. The results show that our method is optimal in choosing Top4.

Choice of Loss Function in Routing Network. Due to the small scale of variation within the same category of data, the loss function determines for routing networks whether they can better distinguish between different subclasses. We compared the effect of softmax loss and center loss on the average performance of the three categories of datasets, as shown in Table 3. The results show that center loss can better improve the performance of the routing network.

4 Conclusion

In this paper, we propose an Adapted-MoE for addressing the data variation and bias in the same category for anomaly detection. We propose a Mixture of Experts that divides same-category samples into different feature spaces via a routing network, with each expert model constructing its own independent decision boundary. We use normalization to make the samples more uniformly distributed in the feature space. In addition, we propose a Test-Time Adaption to eliminate the bias between the distribution of test samples and learned features. Extensive experiments on Texture AD demonstrate that Adapted-MoE can be simply and efficiently implemented for anomaly detection and localization.

Acknowledgments. This work was supported in part by Postgraduate Research Practice Innovation Program of Jiangsu Province (Grant Number: KYCX25_0394).

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

References

1. Batzner, K., Heckler, L., König, R.: Efficientad: Accurate visual anomaly detection at millisecond-level latencies. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. pp. 128–138 (2024)
2. Bergmann, P., Fauser, M., Sattlegger, D., Steger, C.: Mvtec ad–a comprehensive real-world dataset for unsupervised anomaly detection. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 9592–9600 (2019)
3. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A largescale hierarchical image database. In: 2009 IEEE conference on computer vision and pattern recognition. pp. 248–255. Ieee (2009)
4. Heckler, L., König, R., Bergmann, P.: Exploring the importance of pretrained feature extractors for unsupervised anomaly detection and localization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition.pp. 2917–2926 (2023)
5. Hu, J., Chen, X., Gan, Z., Peng, J., Zhang, S., Zhang, J., Wang, Y., Wang, C., Cao, L., Ji, R.: Dmad: Dual memory bank for real-world anomaly detection. arXiv preprint arXiv:2403.12362 (2024)
6. Lei, J., Hu, X., Wang, Y., Liu, D.: Pyramidflow: High-resolution defect contrastive localization using pyramid normalizing flow. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 14143–14152 (2023)
7. Lei, T., Wang, B., Chen, S., Cao, S., Zou, N.: Texture-ad: An anomaly detection dataset and benchmark for real algorithm development. arXiv preprint arXiv:2409.06367 (2024)

8. Liu, B., Tan, P.N., Zhou, J.: Unsupervised anomaly detection by robust density estimation. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 36,pp. 4101–4108 (2022)
9. Liu, J., Xie, G., Wang, J., Li, S., Wang, C., Zheng, F., Jin, Y.: Deep industrial image anomaly detection: A survey. Machine Intelligence Research 21(1), 104–135 (2024)
10. Liu, Z., Zhou, Y., Xu, Y., Wang, Z.: Simplenet: A simple network for image anomaly detection and localization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 20402–20411 (2023)
11. Park, H., Noh, J., Ham, B.: Learning memory-guided normality for anomaly detection. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 14372–14381 (2020)
12. Reiss, T., Hoshen, Y.: Mean-shifted contrastive loss for anomaly detection. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 37, pp. 2155– 2162 (2023)
13. Ristea, N.C., Madan, N., Ionescu, R.T., Nasrollahi, K., Khan, F.S., Moeslund, T.B., Shah, M.: Self-supervised predictive convolutional attentive block for anomaly detection. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 13576–13586 (2022)
14. Wang, Y., Peng, J., Zhang, J., Yi, R., Wang, Y., Wang, C.: Multimodal industrial anomaly detection via hybrid fusion. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8032–8041 (2023)
15. Wen, Y., Zhang, K., Li, Z., Qiao, Y.: A discriminative feature learning approach for deep face recognition. In: Computer vision–ECCV 2016: 14th European conference, amsterdam, the netherlands, October 11–14, 2016, proceedings, part VII 14. pp. 499–515. Springer (2016)
16. Wu, D., Fan, S., Zhou, X., Yu, L., Deng, Y., Zou, J., Lin, B.: Unsupervised anomaly detection via masked diffusion posterior sampling. arXiv preprint arXiv:2404.17900 (2024)
17. Zagoruyko, S., Komodakis, N.: Wide residual networks. arXiv preprint arXiv:1605.07146 (2016)
18. Zavrtanik, V., Kristan, M., Skočaj, D.: Draem-a discriminatively trained reconstruction embedding for surface anomaly detection. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 8330–8339 (2021)
19. Zhang, H., Wang, Z., Wu, Z., Jiang, Y.G.: Diffusionad: Denoising diffusion for anomaly detection. arXiv preprint arXiv:2303.08730 4 (2023)
20. Zhou, Y., Xu, X., Song, J., Shen, F., Shen, H.T.: Msflow: Multiscale flow-based framework for unsupervised anomaly detection. IEEE Transactions on Neural Networks and Learning Systems (2024)