

ORIGINAL RESEARCH OPEN ACCESS

Subspace-Guided Feature Reconstruction for Unsupervised Anomaly Localization

 Katsuya Hotta¹  | Chao Zhang²  | Yoshihiro Hagihara¹ | Takuya Akashi³
¹Iwate University, Morioka-shi, Iwate, Japan | ²University of Toyama, Toyama-shi, Toyama, Japan | ³Okayama University, Kita-ku, Okayama, Japan

Correspondence: Katsuya Hotta (hotta@iwate-u.ac.jp)

Received: 5 March 2024 | **Revised:** 25 March 2025 | **Accepted:** 6 July 2025

Funding: This work was supported in part by the JSPS KAKENHI (Grant Numbers JP23K19989 and JP24K20828).

ABSTRACT

Unsupervised anomaly localization aims to identify anomalous regions that deviate from normal sample patterns. Most recent methods perform feature matching or reconstruction for the target sample with pre-trained deep neural networks. However, they still struggle to address challenging anomalies because the deep embeddings stored in the memory bank can be less powerful and informative. Specifically, prior methods often overly rely on the finite resources stored in the memory bank, which leads to low robustness to unseen targets. In this paper, we propose a novel subspace-guided feature reconstruction framework to pursue adaptive feature approximation for anomaly localization. It first learns to construct low-dimensional subspaces from the given nominal samples, and then learns to reconstruct the given deep target embedding by linearly combining the subspace basis vectors using the self-expressive model. Our core is that, despite the limited resources in the memory bank, the out-of-bank features can be alternatively “mimicked” to adaptively model the target. Moreover, we propose a sampling method that leverages the sparsity of subspaces and allows the feature reconstruction to depend only on a small resource subset, contributing to less memory overhead. Extensive experiments on three benchmark datasets demonstrate that our approach generally achieves state-of-the-art anomaly localization performance.

1 | Introduction

Anomaly localization task for visual inspection, which refers to detecting visual anomalies in images that deviate from normal sample patterns, plays a key role in many practical applications such as product quality control [1–3] and video surveillance [4–6]. In practical scenarios, anomalies are rare in production lines, rendering the acquisition of a suitable dataset for training supervised methods impractical. To tackle this problem, recent approaches have explored one-class classification (OCC) anomaly localization, leveraging only abundantly available anomaly-free images (see [7] for a review). Early works based on OCC generally train autoencoders (AEs) to reconstruct normal images only. For an arbitrary testing image, the poorly reconstructed regions localize anomalies. As such, the models with higher generalization

capacity would by contrast degrade the performance. This makes later efforts focus on tuning the generalization strength of models, such as exploring the bottleneck layer for reconstruction.

Modern OCC anomaly localization methods [8, 9] leverage the knowledge of pre-trained deep neural networks for ImageNet [10] classification. In a pioneering study, [11] attempt to extract features exclusively from normal data (i.e., nominal data) by utilizing a pre-trained neural network and subsequently keep these features in a memory bank. Identifying abnormal regions that deviate from the nominal patterns stored in the memory bank is facilitated through feature matching techniques. This concept of using feature matching lays the groundwork for state-of-the-art methods [12, 13] to achieve strong spatial localization of anomalies despite the lack of adaptation to specific feature

This is an open access article under the terms of the [Creative Commons Attribution](https://creativecommons.org/licenses/by/4.0/) License, which permits use, distribution and reproduction in any medium, provided the original work is properly cited.

© 2025 The Author(s). *IET Image Processing* published by John Wiley & Sons Ltd on behalf of The Institution of Engineering and Technology.

distributions. However, as pointed out in [12, 14], feature matching using deep feature embeddings is a non-adaptive approach. Non-adaptive approaches have limited confidence in matching unseen features because they rely heavily on the finite resources stored in a memory bank. Although this limitation can be alleviated by preparing a vast number of feature patterns, as memory bank resources increase, the anomaly localization process becomes computationally expensive, and latency increases. One straightforward strategy to address this challenge is to screen the data before storing them in a memory bank. Given the constraints imposed by computational and memory limitations, [11] and [12] employ pre-selections based on kNN [15] and coreset subsampling [16], respectively. In contrast, [9] attempts to generate synthetic anomalies in a pre-trained feature space to improve the discriminative performance for localizing anomalous features. While these approaches enhance the feature information extracted from nominal data during inference, they do not inherently tackle adapting to the feature distribution of specific data. Thus, approaches that represent specific feature distributions have shifted to a policy of reconstructing target features. Although conventional methods [17–19] can represent a wider range of normal appearances, they suffer from a shortcut learning problem in which even abnormal regions are reconstructed accurately due to their high generalization ability.

In this paper, we propose a novel subspace-guided framework to provide a better solution to anomaly localization. Our method aims to pursue a more reliable feature approximation for unseen targets by learning low-dimensional subspaces [20, 21], which can be easily derived using the nominal sample features in the memory bank. Specifically, by introducing the approximation policy of the self-expressive model [22], the unseen target features can be compactly represented by a linear combination of features from nominal samples lying on one specific underlying subspace. As such, despite the lack of target information in the memory bank, the target feature can be readily produced by resorting to the ideal subspace. Moreover, our framework addresses anomaly localization by leveraging the insight that as our subspace-guided approximation strategy only enables reproducing the anomalous-free features, the anomalous regions would induce a significant reconstruction error. We thus define the anomaly score as the difference between the target feature and the reconstructed feature via our method. Compared to prior approaches using feature matching, which relies largely on finite resources stored in a memory bank, our subspace-guided feature reconstruction yields a richer data coverage and robustness to challenging anomalies. Ideally, fully utilizing the storage in the memory bank would lead to a strong target feature approximation capacity. Nevertheless, it would cause a prohibitively large computational overhead. To address this, we leverage the sparsity [23] of subspaces and further propose a sampling method that allows the feature reconstruction to depend only on a small resource subset, improving computational efficiency. As shown in Figure 1, our method reconstructs features to mimic a given target sample to localize anomalies during inference. Extensive experiments on three public datasets demonstrate that our approach generally achieves state-of-the-art anomaly localization performance.

In summary, our paper includes the following main contributions:

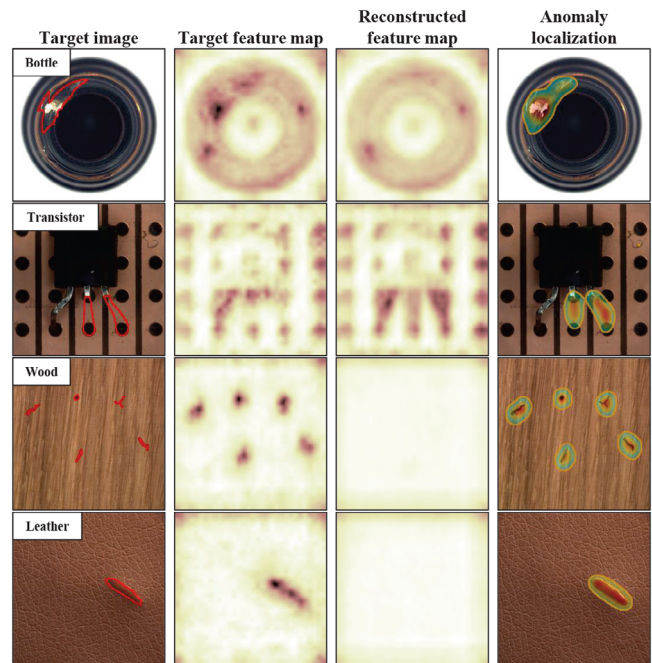


FIGURE 1 | An example of anomaly localization results from our method on the MVTeC AD benchmark dataset. The red boundary in the target image indicates anomalous regions. Each feature map is shown in red color gradients for visualization. Anomaly localization results are shown by orange for anomaly boundaries and blue-red color gradients for anomaly intensity.

- We propose a novel anomaly localization framework that incorporates the subspace-guided feature reconstruction mechanism. It adaptively mimics the out-of-memory bank features by linearly combining the features in one specific subspace derived from the given nominal samples to encourage a sufficient data coverage.
- We introduce a sparsity-inspired sampling technique for subspace basis vectors to mitigate memory expense in reconstructing target features.
- We perform extensive experiments to demonstrate that our method yields state-of-the-art anomaly localization accuracy with only very few nominal data.

2 | Related Work

Anomaly localization focuses on identifying visually anomalous regions in an image. This task is more challenging than anomaly detection [24–26], which distinguishes patterns in data that deviate from expected behavior. In the context of recent unsupervised anomaly localization methods, one of the key issues is how to efficiently identify anomalous regions during inference based on the nominal dataset available beforehand.

Early literature [27, 28] involves reconstructing normal images from anomalous images using an encoder-decoder network. The anomaly localization capability of these methods is based on the assumption that the trained network cannot accurately reconstruct anomalous regions, and anomalies can be identified

by comparing the target image with its reconstructed image. In particular, AE_{SSIM} [17] introduces a structural similarity metric into the AE model [29] to capture the structural information of the image, thus enhancing the anomaly localization ability of the model. However, these methods tend to accurately reconstruct anomalous regions due to their high generalization ability, leading to false negatives in anomaly localization as the complexity of the problem increases. To address this issue, methods starting with [25] employ pre-trained networks on extensive external natural image datasets like ImageNet [10], capable of representing fine-grained visual features.

Recent methods, which introduce the innovative concept of utilizing nominal feature representations extracted by these pre-trained models, have significantly improved anomaly localization performance compared to earlier approaches. [30] proposed a teacher-student framework based on knowledge distillation [31]. In this framework, during training, the student learns only from the knowledge of nominal samples transferred by the teacher. Following this knowledge distillation framework, STFPM [32] introduces a feature pyramid matching mechanism that alleviates the incompleteness of transferred knowledge loss by matching corresponding multi-scale features in the student-teacher network. RD [8] proposes an inverse distillation approach that extends the teacher-student framework to enhance the diversity of anomaly representations. Several studies propose training anomaly localization models in a self-supervised manner by simulating pseudo-anomalous data [33]. CutPaste [34] generates synthetic anomalies using a simple technique that involves cutting patches from an image and placing them at random locations within the image. DeSTSeg [35] uses two-dimensional Perlin noise [36, 37] to generate more realistic anomalous images. Yao et al. [38] proposed an anomaly localization framework based on edge enhancement and cascade optimization of residual features to generate augmented images that closely resemble the distribution of real anomalous images. DiffAD [39] leverages the diffusion model [40] to capture structural changes in images, achieving strong performance in anomaly localization. Similarly, RealNet [41] introduces a strength-controllable diffusion-based anomaly synthesis strategy to generate realistic yet diverse anomaly samples that closely align with natural data distributions.

Reconstruction-based approaches using deep embeddings have also been proposed recently. Unlike AE_{SSIM} , DFR [19] employs the deep features extracted by a pre-trained network as the reconstruction target. DSR [37] utilizes the latent space of pre-trained autoencoders to generate synthetic anomalies, enabling the generation of anomaly regions at the feature level. It is worth noting that the aforementioned approaches demand a large number of training data and necessitate periodic retraining whenever the training dataset is updated. In contrast, anomaly localization approaches based on the memory bank framework, which are the most relevant to our proposed method, do not require a training step. This characteristic is attractive for practical applications in industrial anomaly localization.

A representative work of the memory bank-based approach is SPADE [11], which achieves anomaly localization by storing nominal features extracted from a pre-trained neural network in a memory bank and then performing feature matching. While

storing all nominal data in the memory bank can improve anomaly localization performance, it may not be practical due to computational constraints. To enhance the computational efficiency of this feature matching process, SPADE incorporates pre-selection using kNN [15]. However, maintaining accuracy while using kNN-based pre-selection remains computationally demanding, mainly due to the limited number of reductions permissible within the memory bank. Consequently, subsequent methods have focused on maximizing the information of nominal features retained in memory banks. PaDiM [42] adopts a strategy by evaluating the Mahalanobis distance [43] at the patch level instead of using the time-intensive kNN method. PatchCore [12] proposes to build a memory bank that considers the negative effects of biased features of pre-trained networks on anomaly localization. A crucial limitation of these methods lies in their reliance on feature matching in the memory bank for anomaly localization without fundamentally addressing the task of adapting to the feature distribution of specific data.

Following this line of research, our approach introduces the concept of subspaces [44] into the task of anomaly localization, thereby affording two advantages: i) an adaptive feature representation guided by subspace-preserving properties; ii) memory bank reduction motivated by sparsity. Compared to prior approaches, our framework is less reliant on features stored in a memory bank and instead explores faithful approximation of normal features through subspace structures that compactly compress patterns of nominal features. This allows the model to reconstruct normal features while excluding abnormal regions in unseen targets and significantly alleviates the issue of shortcut learning, which is widely confirmed in conventional reconstruction-based models. Moreover, the memory bank reduction motivated by sparsity can reduce the reconstruction burden while maintaining performance by selectively retaining a small part of data that accurately represents nominal feature patterns.

3 | Anomaly Localization via Subspace Learning

In this section, we propose a framework that utilizes a self-expressive model for subspace learning to arrive at localization decisions, as illustrated in Figure 2. To this end, we present an algorithm designed to capture the similarity between a target feature and an approximated feature by a self-expressive model (Section 3.1). We then explicate the sampling technique that employs a subspace structure to decrease the inference time (Section 3.2), and we finally summarize the comprehensive algorithm up to the localization of anomalies (Section 3.3).

3.1 | Subspace-Guided Feature Reconstruction

We first describe our policy for localizing anomalous regions by approximating the features using a self-expressive model. Formally, we define the nominal data containing only normal images as $\mathcal{D} = \{I_1, \dots, I_n, \dots, I_N\}$, and the test image provided during inference, which can be either normal or abnormal, is denoted by J . Each image is characterized by its width w_* , height h_* , and number of channels c_* (i.e., $I_n \in \mathbb{R}^{w_* \times h_* \times c_*}$ and $J \in \mathbb{R}^{w_* \times h_* \times c_*}$). Our approach leverages a powerful feature extractor,

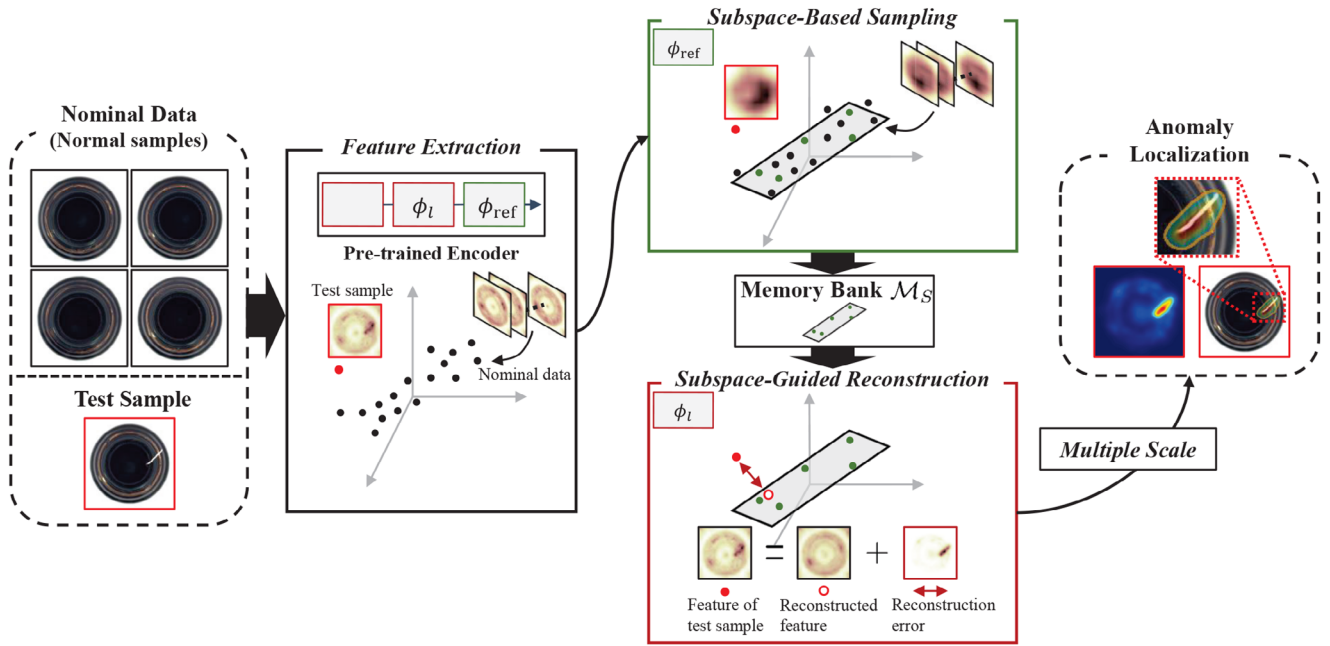


FIGURE 2 | Overview of our approach. Features are extracted from nominal data \mathcal{D} and a test sample J through a pre-trained network ϕ . To achieve reduced memory expense and computational complexity, features in hierarchy level l_{ref} are utilized to estimate the subspace in which the nominal data lie. Only a limited number of data, sufficient for recovering this subspace, is sampled and stored in the memory bank \mathcal{M}_S . By exclusively utilizing the data in \mathcal{M}_S , the reconstruction methodology based on the self-expressive model is employed to identify anomalous regions in the test sample. The proposed method performs pixel-level anomaly localization by scoring at multiple hierarchies to benefit from deeper features.

capable of capturing high-quality deep features from images, facilitating feature reconstruction at the pixel level. To realize this, as suggested in [11, 12, 25], we employ pre-trained deep neural networks on ImageNet [10] as feature extractors. This network captures meaningful features at different hierarchies, with each hierarchical feature playing an important role. As such, at the l -th hierarchical level of the pre-trained network ϕ , we denote the feature map $\phi_l(I_n)$ for image $I_n \in \mathcal{D}$, and the feature map $\phi_l(J)$ for image J . The hierarchical level l refers to the feature maps indexed from ResNet-like [45] architectures, where $l \in \{1, 2, 3, 4\}$ denotes the final output of each spatial resolution block (i.e., $\phi_l(I_n) \in \mathbb{R}^{w_l \times h_l \times c_l}$ and $\phi_l(J) \in \mathbb{R}^{w_l \times h_l \times c_l}$).

We propose a feature reconstruction mechanism by introducing a self-expressive model that exhibits enhanced feature representation capabilities compared to feature matching techniques that rely on limited resources of \mathcal{D} . We extend the previously introduced notation to formulate a feature reconstruction at the pixel level. The feature map $\phi_l(I_n)$ is defined as the feature vector $\mathbf{x}_{n,l} \in \mathbb{R}^{w_l h_l c_l}$ obtained by concatenating its dimensions into a $w_l h_l c_l$ -dimensional representation. Similarly, $\phi_l(J)$ is defined as $\mathbf{y}_l \in \mathbb{R}^{w_l h_l c_l}$. Given a data matrix $X_l = [\mathbf{x}_{1,l}, \mathbf{x}_{2,l}, \dots, \mathbf{x}_{N,l}] \in \mathbb{R}^{w_l h_l c_l \times N}$ comprising column-wise nominal feature vectors $\mathbf{x}_{n,l}$, the reconstruction of the test feature vector in the self-expressive model can be described as

$$\mathbf{y}_l = X_l \mathbf{c}_l + \mathbf{e}_l, \quad (1)$$

where $\mathbf{c}_l \in \mathbb{R}^N$ is a coefficient vector and $\mathbf{e}_l \in \mathbb{R}^{w_l h_l c_l}$ is an anomalous term. \mathbf{e}_l refers to an out-of-distribution feature that exhibits the difference from the distribution of nominal features in X_l . In an ideal context, if \mathbf{y}_l follows the distribution of X_l (i.e.,

when \mathbf{y}_l represents normal data), the anomaly term should be precisely zero. Conversely, if \mathbf{y}_l deviates from this distribution (i.e., when \mathbf{y}_l contains anomalous regions), it should be non-zero. In the optimization of Equation (1), feasible solutions are generally non-unique because the number of data that lie in a subspace is typically larger than its dimensionality. Nonetheless, at least one \mathbf{c}_l exists where $c_{i,l}$ is non-zero only if the data $\mathbf{x}_{i,l}$ and \mathbf{y}_l lie in the same subspace, which is referred to as subspace-preserving [46, 47]. It is worth mentioning that nominal features that satisfy the subspace-preserving property can represent only normal features modeled in the same low-dimensional subspace. Thus, unseen features would cause large reconstruction errors to indicate anomalies, which constitutes the core advantage of introducing low-dimensional subspaces in our framework. *In summary, our method is conceptually different from prior approaches where the models simply reconstruct a copy of target features without considering the data structure of the nominal distribution.*

Optimization. The coefficient vector \mathbf{c}_l in Equation (1), which satisfies the subspace-preserving representations, can be solved by considering the problem of minimizing the anomalous term as follows:

$$\min_{\mathbf{c}_l} \|\mathbf{y}_l - X_l \mathbf{c}_l\|_2^2 \quad \text{s.t.} \quad \|\mathbf{c}_l\|_0 \leq s, \quad (2)$$

where $\|\cdot\|_0$ represents the ℓ_0 pseudo-norm counting the number of non-zero entries in the vector, and s is a tuning parameter that controls the sparsity of the solution by selecting up to s entries in the coefficient vector \mathbf{c}_l . We utilize the orthogonal matching pursuit (OMP) algorithm [48–50] to solve this optimization problem and recover a subspace-preserving solution.

Input: \mathcal{M} , X_l , \mathbf{y}_l , s , and ϵ

- 1: Initialize $k = 0$, anomalous term $\mathbf{e}_l = \mathbf{y}_l$, and $S = \emptyset$;
- 2: **while** $k < s$ and $\|\mathbf{e}_l\|_2 > \epsilon$ **do**
- 3: Find j^* via Equation (3);
- 4: Update $S \leftarrow S \cup \{j^*\}$;
- 5: Estimate \mathbf{c}_l via Equation (4);
- 6: Update \mathbf{e}_l via Equation (5);
- 7: $k = k + 1$;
- 8: **end while**

Output: Coefficient vector \mathbf{c}_l and anomalous term \mathbf{e}_l

To efficiently solve the Equation (2), we introduce Algorithm 1 based on the OMP algorithm. Formally, we commence by initializing the support set S and the anomalous term \mathbf{e}_l as $S = \emptyset$ and $\mathbf{e}_l = \mathbf{y}_l$, respectively. Subsequently, at each iteration, we update the index set S by adding one index j^* , which is computed as follows:

$$j^* = \arg \max_{j \in \mathcal{M} \setminus S} \mathbf{x}_{j,l}^\top \mathbf{e}_l, \quad (3)$$

where the memory bank $\mathcal{M} \in \{1, \dots, N\}$ is the candidate index set of the nominal features for reconstruction. After updating S with j^* (i.e., $S \leftarrow S \cup \{j^*\}$), we solve the following problem:

$$\min_{\mathbf{c}_l} \|\mathbf{y}_l - X_l \mathbf{c}_l\|_2^2 \quad \text{s.t.} \quad \text{supp}(\mathbf{c}_l) \subseteq S. \quad (4)$$

where $\text{supp}(\cdot)$ is the support function that returns the subgroup of the domain containing elements not mapped to zero. By utilizing the minimized coefficient vector \mathbf{c}_l in iteration k , we can update the anomalous term \mathbf{e}_l through the estimation of the approximation error between \mathbf{y}_l and the reconstructed feature $X_l \mathbf{c}_l$:

$$\mathbf{e}_l \leftarrow \mathbf{y}_l - X_l \mathbf{c}_l. \quad (5)$$

We repeat the optimization process in Equations (3)–(5) until either the number of iterations k reaches the limit s or \mathbf{e}_l is smaller than the threshold ϵ .

3.2 | Subspace-Based Sampling Technique for Memory Bank Reduction

Although our model, as described in Section 3.1, is capable of predicting anomalous regions at the pixel level, utilizing a self-expressive model for reconstruction entails notable memory expense. Moreover, it substantially increases inference time, particularly when dealing with an extensive volume of high-dimensional data. We introduce a subspace-based sampling technique into our model to address this issue, motivated by [11, 12]. This technique alleviates the reconstruction burden while maintaining performance by selectively retaining a limited number of in-sample data within the memory bank \mathcal{M} , which effectively approximates the out-of-bank data. Specifically, our

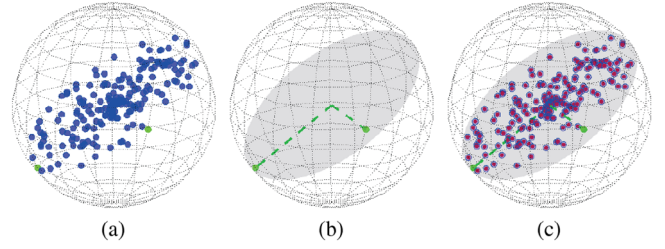


FIGURE 3 | Illustrative example of covering out-of-bank data ($\mathcal{M} \setminus \mathcal{M}_S$) from a few data (\mathcal{M}_S) chosen through subspace-based sampling in a toy dataset: consider a synthetic dataset created by randomly generating data points lying on a single subspace of \mathbb{R}^2 in the ambient space of \mathbb{R}^3 . The proposed subspace-based sampling strategy selects only very few basis vectors (green). Through a linear combination of these data points, a comprehensive approximation (red) is achieved for the out-of-bank data (blue) that lie in the same underlying subspace (gray plane).

approach aims to identify \mathcal{M} -subspace \mathcal{M}_S ($\mathcal{M}_S \in \mathcal{M}$) composed exclusively of entries that satisfy the subspace-preserving properties by selecting a few basis vectors from the subspace in the hierarchy $\text{ref} = l_{\text{ref}}$ ($l < l_{\text{ref}}$), which represents high-level features:

$$\mathbf{c}_{\text{ref}}^* = \arg \min_{\mathbf{c}_{\text{ref}}} \|\mathbf{y}_{\text{ref}} - X_{\text{ref}} \mathbf{c}_{\text{ref}}\|_2^2 \quad (6)$$

$$\text{s.t.} \quad \|\mathbf{c}_{\text{ref}}\|_0 \leq s_{\text{ref}},$$

$$\mathcal{M}_S = \text{supp}(\mathbf{c}_{\text{ref}}^*), \quad (7)$$

where s_{ref} is a tuning parameter that controls the sampling rate. The solution of Equation (6) satisfying the subspace-preserving property can be solved by Algorithm 1 as in Equation (2).

The proposed sampling technique differs significantly from [11, 12] in that its ability to effectively represent the out-of-bank features contained within \mathcal{M} through the reconstruction process. The indices encompassed within \mathcal{M}_S serve as basis vectors that form a specific subspace. In other words, the out-of-bank features $\mathbf{x}_{j,\text{ref}}, \forall j \in \mathcal{M} \setminus \mathcal{M}_S$ lying in the subspace derived from \mathcal{M}_S can be expressed as a linear combination of basis vectors that are in-sample data $\mathbf{x}_{i,\text{ref}} \in \mathcal{M}_S$:

$$\mathbf{x}_{j,\text{ref}} = \sum_{i \in \mathcal{M}_S} \alpha_i \mathbf{x}_{i,\text{ref}}, \quad (8)$$

where α_i represents an arbitrary coefficient. Figure 3 provides a visual conceptual example that demonstrates a data coverage of out-of-bank using in-sample data in a toy dataset. This expressiveness based on the subspace-preserving property facilitates a richer data coverage of the memory bank while concurrently mitigating the computational complexity associated with the reconstruction process (see experiments done in Section 4.3.1).

3.3 | Anomaly Localization via Learning Subspace Representations

We summarize the whole procedure of the proposed anomaly localization approach in Algorithm 2. Given a test image J and a nominal dataset D consisting of N normal images, our objective

ALGORITHM 2 | Anomaly Localization via Learning Subspace Representations

Input: Pre-trained network ϕ , nominal dataset \mathcal{D} , test image J , parameters $l_{\text{ref}}, l, s_{\text{ref}}, s, \epsilon$;

- 1: Set $\mathcal{M} \in \{1, \dots, N\}$;
/* Subspace-based sampling (Section 3.2) */
- 2: Given $\phi, l_{\text{ref}}, \mathcal{D}$, and J , get X_{ref} and \mathbf{y}_{ref} ;
- 3: Given $\mathcal{M}, X_{\text{ref}}, \mathbf{y}_{\text{ref}}$, and s_{ref} , solve $\mathbf{c}_{\text{ref}}^*$ via Algorithm 1;
- 4: Get \mathcal{M}_S via Equation (7)
/* Subspace-guided feature reconstruction (Section 3.1) */
- 5: Given ϕ, l, \mathcal{D} , and J , get X_l and \mathbf{y}_l ;
- 6: Given $\mathcal{M}_S, X_l, \mathbf{y}_l$, and s , compute \mathbf{e}_l via Algorithm 1;
- 7: Get anomaly scores from \mathbf{e}_l ;

Output: Anomaly score

is to identify anomalous features in J that deviate from the feature patterns observed in \mathcal{D} . Initially, we employ a network ϕ to extract X_{ref} and \mathbf{y}_{ref} from \mathcal{D} and J , respectively. We then employ the sampling technique described in Section 3.2 to reduce the number of features stored in the memory bank \mathcal{M}_S from N to s_{ref} . Subsequently, similarly extracting the l -th hierarchical level features, we perform the subspace-guided feature reconstruction process, as described in Section 3.1, utilizing \mathcal{M}_S . To maintain compatibility with the original input resolution, we resize \mathbf{e}_l to match the size of feature map $\phi_l(J)$ and employ bi-linear interpolation for upscaling. Furthermore, we apply a smoothing operation to the result using a Gaussian kernel with a width parameter $\sigma = 4$. Note that our implementation does not optimize the specific value of σ . Our algorithm is capable of utilizing anomaly scores across multiple scales (i.e., $l \in \{2, 3, 4\}$), similar to [11, 32]. For a comprehensive analysis, we kindly refer you to Section 4.3, which provides a detailed ablation study.

Complexity Analysis. Algorithm 1 necessitates a certain number of inner products, which is contingent upon the size of the memory bank \mathcal{M} . Specifically, given a memory bank $\mathcal{M} \in \{1, \dots, N\}$ containing all nominal data, N inner products are required within a single iteration, resulting in the computational complexity of $\mathcal{O}(w_l h_l c_l N s)$ for the optimization problem stated in Equation (2). The sampling technique described in Section 3.2 effectively reduces the number of indices in the memory bank from the original N to s_{ref} with the computational complexity $\mathcal{O}(w_{\text{ref}} h_{\text{ref}} c_{\text{ref}} N s_{\text{ref}})$. Thus, the computational complexity of the subspace-guided feature reconstruction with \mathcal{M}_S obtained by applying the sampling technique is reduced to $\mathcal{O}(w_l h_l c_l s_{\text{ref}} s)$. Consequently, the overall computational complexity of our approach is $\mathcal{O}(w_{\text{ref}} h_{\text{ref}} c_{\text{ref}} N s_{\text{ref}} + w_l h_l c_l s_{\text{ref}} s)$. In the case of WideResnet-50, the feature maps obtained from consecutive hierarchical levels follow the relationships $w_{l+1} = \frac{1}{2} w_l$, $h_{l+1} = \frac{1}{2} h_l$, $c_{l+1} = 2c_l$. By adhering to the memory bank reduction technique, we set $l_{\text{ref}} = l + 1$ and $s_{\text{ref}} \approx s \ll N$. Based on these considerations, the computational complexity of our algorithm can be estimated as $\mathcal{O}(w_{\text{ref}} h_{\text{ref}} c_{\text{ref}} N s_{\text{ref}} + w_l h_l c_l s_{\text{ref}} s) = \mathcal{O}(w_l h_l c_l (\frac{1}{2} N + s) s_{\text{ref}}) < \mathcal{O}(w_l h_l c_l N s)$.

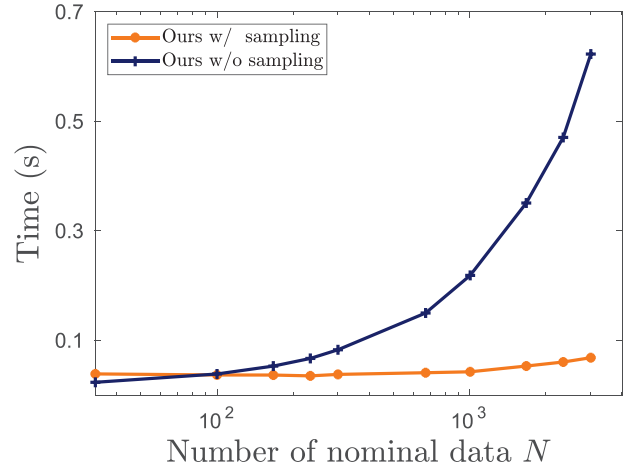


FIGURE 4 | Curves of inference time with respect to the number of nominal data N , w/ (orange) or w/o (blue) our proposed sampling.

Therefore, it is trivial to conclude that our sampling technique enjoys the advantage of an N -independent reconstruction process when using multi-scale anomaly scores. To experimentally study this property, we investigate the change of inference time with respect to the number of nominal data N , and plot the curve in Figure 4 by fixing $(w_l, h_l, c_l, s_{\text{ref}}, s)$. It can be confirmed from the orange curve in Figure 4 that our sampling method contributes to an N -independent reconstruction, which generally aligns with our theoretical analysis. Further discussion regarding other parameters is presented in Section 4.3.1.

4 | Experimental Results and Analysis

In this section, we initially introduce the datasets, provide implementation details, and specify the evaluation metrics employed in our experiments. Subsequently, we quantitatively compare our method against the state-of-the-art models on multiple benchmark datasets. Lastly, we carry out ablation studies from various perspectives to provide a comprehensive analysis of our approach.

4.1 | Implementation Details

Following [30], we employ WideResNet50 [55] as the backbone network, which is a widely adopted configuration for method comparison on anomaly localization. Input images are resized to 256×256 , and no data augmentation is applied. STFPM [32] uses different networks in the reported paper, and we tried to reproduce the results on WideResNet50 in order to provide a fair and consistent comparison. Similarly, we retrain DeSTSeg by excluding the data augmentation procedure to maintain consistency across our comparisons. All experiments are conducted on an RTX 4090 GPU.

For quantitative evaluation metrics, we evaluate our algorithm using the area under the receiver-operator curve (AUROC: %), the per-region overlap (PRO: %), and the average precision (AP: %). PRO metric uses up to a false positive rate of 30%, as recommended by [30]. These metrics compute the average accuracy of pixel-level localization per category. The PRO score, which

TABLE 1 | Anomaly localization performance on MTD dataset [51]. Comparative results are measured by AUROC and PRO.

Method	SPADE [11]	STFPM [32]	PaDiM [42]	PatchCore [12]	DSR [37]	SimpleNet [9]	DeSTSeg [35]	Ours
AUROC↑	72.8	68.5	74.1	<u>84.8</u>	57.7	78.9	74.0	85.7
PRO↑	69.4	61.4	68.5	<u>72.3</u>	40.8	71.0	<u>72.3</u>	73.1

considers the overlap and recovery of connected anomaly components, provides a more comprehensive performance assessment by accounting for variations in anomaly sizes. The AP score, computed as the area under the precision-recall curve, is more appropriate for samples with small anomalies [36].

4.2 | Anomaly Localization on Benchmark Datasets

4.2.1 | Magnetic Tile Defect (MTD) [51]

MTD contains 925 defect-free magnetic tile images and 392 anomalous images with varied illumination levels and image sizes. The test set is provided with ground-truth masks. Following [12, 56], we evaluate 20% of the defect-free images for testing, while the remaining images are used for training. We employ feature maps obtained from subsequent hierarchy levels 2, 3, 4, and the last level (where the last level is utilized only for subspace-based sampling). The parameter values are set as follows: $s_{\text{ref}} = 10$ and $s = 7$.

In Table 1, we show the results on MTD. The best result in each row is shown in bold, and the second-best result is underlined. We can observe that our method outperforms competing approaches across both evaluation metrics, particularly surpassing prior memory bank-based methods. Typically, memory bank-based methods struggle to deal with anomalies in categories with irregular texture patterns. This well demonstrates the low robustness in performing feature matching for out-of-memory bank features. Conversely, our approach is less dependent on the memory bank due to its ability to adaptively generate feature representations using subspace-guided feature reconstruction. The visual results in Figure 5 also show that the proposed method achieves robust anomaly localization against light source imbalances. However, the sample with global anomalies, as shown in the bottom result on MTD, tends to cause False Negative. Importantly, this issue is not caused by our feature reconstruction approach but rather by the low sensitivity to the overall shape of the CNN as a backbone network.

4.2.2 | Beantech Anomaly Detection (BTAD) [52]

BTAD contains three categories of industrial products, comprising 2,540 samples. The training set comprises only anomaly-free samples, while the test set comprises both anomaly-free and anomalous samples. As with the MTD, the test set is provided with ground-truth masks. We employ feature maps obtained from subsequent hierarchy levels 2, 3, and 4 (where hierarchical level 4 is utilized only for subspace-based sampling). The parameter values are set as follows: $s_{\text{ref}} = 80$ and $s = 40$.

TABLE 2 | Anomaly localization performance on BTAD dataset [52]. Comparative results are measured by AUROC / PRO.

Method	Case 01	Case 02	Case 03	Avg.
STFPM [32]	96.2 / 71.7	97.0 / 45.1	<u>99.3</u> / 96.4	97.5 / 71.0
FastFlow [53]	97.1 / 71.7	93.6 / 63.1	98.3 / 79.5	96.3 / 71.4
PatchCore [12]	<u>97.0</u> / 64.9	<u>95.8</u> / 47.3	99.2 / 67.7	<u>97.4</u> / 60.0
DSR [37]	89.1 / 61.5	78.8 / 38.5	94.1 / 84.0	87.3 / 61.3
SimpleNet [9]	96.2 / 69.6	95.0 / 53.3	99.5 / <u>97.0</u>	96.9 / 73.3
DeSTSeg [35]	96.2 / 83.2	94.8 / 42.8	<u>99.3</u> / 97.2	96.8 / <u>74.4</u>
Ours	96.2 / <u>72.6</u>	94.8 / <u>54.7</u>	<u>99.3</u> / <u>97.0</u>	96.8 / 74.8

In Table 2, we report the results on BTAD. The best result in each column is shown in bold, and the second-best result is underlined. The significant difference between AUROC and PRO scores in this dataset highlights the importance of the PRO metric, which considers the variation in anomaly size, for evaluation compared to other datasets. Notably, our method achieves state-of-the-art anomaly localization in the PRO metric. Despite its easy implementation using memory bank-based feature reconstruction, our approach outperforms sophisticated methods such as normalizing flow and knowledge distillation-based approaches. In the visual results in Figure 5, it can be observed that our method yields accurate anomaly localization, even for targets with irregular textures that do not perfectly match the nominal samples.

4.2.3 | MVTec AD [18]

MVTec AD, which is widely recognized as a standard benchmark for industrial anomaly localization tasks, consists of 15 categories of industrial products (such as carpet, tile, bottle, etc.), comprising a total of 5,354 images. Each category is divided into two sets: a training set that exclusively contains anomaly-free samples and a test set that includes both anomaly-free and anomalous samples. The test set contains various defect types associated with specific products, and corresponding ground-truth masks are provided to indicate the location of anomalies within the images. We employ feature maps obtained from subsequent hierarchy levels 2, 3, and 4 (where hierarchical level 4 is utilized only for subspace-based sampling). The parameter values are set as follows: $s_{\text{ref}} = 40$, $s = 17$, and $\epsilon = 10^{-6}$.

The results for anomaly localization on MVTec AD are listed in Table 3 and Table 4. We can confirm that our method is competitive with state-of-the-art methods regarding two evaluation metrics and particularly outperforms reconstruction-based methods AE_{SSIM} [18], DFR [19], and DSR [37]. We assume

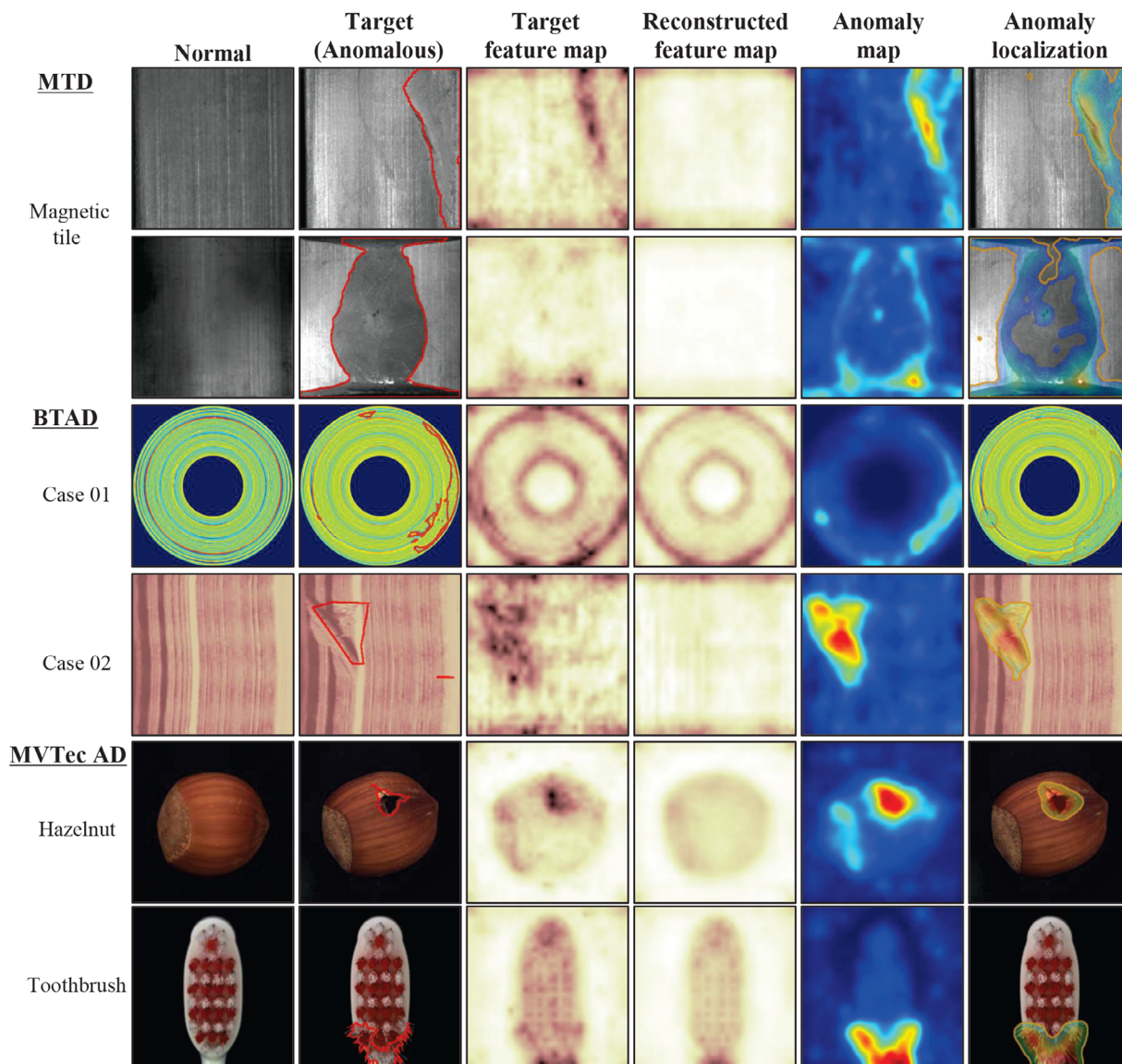


FIGURE 5 | Qualitative results of our method on three benchmark datasets. For each category, we show a nominal image (normal), the target image (abnormal), the target feature map, the reconstructed feature map, the anomaly map, and the anomaly localization result identified by our method. The boundary in red denotes the ground-truth anomalous regions.

the reason can be attributed to that the proposed feature reconstruction mechanism is designed to only reconstruct normal features that are faithful to the nominal features, thereby reducing false negatives in anomaly localization. Specifically, compared to conventional reconstruction methods, modeling the low-dimensional subspace from nominal samples enables more reliable feature approximation to unseen targets that can be represented in the same subspace. To qualitatively evaluate the reconstructed features of our method, the visual results are shown in Figure 5. Our method is able to perfectly reconstruct normal patterns even for the subtle appearance features of “Toothbrush” and assign high anomaly scores to the defective regions.

4.2.4 | Overall Anomaly Localization Performance

Table 5 reports the overall anomaly localization performance of each method in terms of AP. The proposed method achieves

strong anomaly localization performance on the MTD and BTAD datasets and outperforms PatchCore on the MVTec AD. Moreover, the proposed method surpasses state-of-the-art methods regarding the average performance over the three datasets, including memory bank-based methods (SPADE [11] and PatchCore [12]) and anomaly simulation-based methods (DeSTSeg [35] and SimpleNet [9]). Our subspace-guided feature reconstruction achieves high accuracy in industrial product anomaly localization, even without any dataset-specific training.

4.3 | More Evaluations

To emphasize the advantages of our proposed model, we conduct a comprehensive analysis to evaluate the effects of our model in the following aspects: (1) Ablation study for the subspace-based sampling technique, (2) Evaluation of the expressiveness

TABLE 3 | Anomaly localization performance in terms of AUROC on MV7ec AD dataset [18].

Method	Bottle	Cable	Capsule	Carpet	Grid	Hazelnut	Leather	Metal Nut	Pill	Screw	Tile	Toothbrush	Transistor	Wood	Zipper	Avg.
AE _{SSIM} [18]	93.0	82.0	94.0	87.0	94.0	97.0	78.0	89.0	91.0	96.0	59.0	92.0	90.0	73.0	88.0	87.0
SPADE [11]	98.4	97.2	99.0	97.5	93.7	<u>99.1</u>	97.6	98.1	96.5	98.9	87.4	97.9	94.1	88.5	96.5	96.0
PaDiM [42]	98.3	96.7	98.5	<u>99.1</u>	97.3	98.2	99.2	97.2	95.7	98.5	94.1	98.8	98.5	94.9	98.5	97.5
STFPM [32]	<u>98.8</u>	95.5	98.3	98.8	<u>99.0</u>	98.5	99.3	97.6	97.8	98.3	97.4	98.9	82.5	97.2	98.5	92.1
PatchCore [12]	98.6	98.5	<u>98.9</u>	<u>99.1</u>	98.7	98.7	99.3	98.4	97.6	99.4	95.9	98.7	<u>96.4</u>	95.1	<u>98.9</u>	98.1
RIAD [54]	98.4	84.2	92.8	96.3	98.8	96.1	99.4	92.5	95.7	98.8	89.1	98.9	87.7	85.8	97.8	94.2
DFR [19]	97.0	92.0	99.0	97.0	98.0	99.0	98.0	93.0	97.0	99.0	87.0	99.0	80.0	93.0	96.0	95.0
DSR [37]	98.7	97.2	91.0	95.8	99.5	98.8	99.4	93.0	94.0	97.9	98.7	<u>99.3</u>	78.0	91.6	98.2	95.4
SimpleNet [9]	98.2	<u>97.9</u>	99.0	98.2	98.7	98.2	99.1	<u>98.7</u>	<u>98.7</u>	<u>99.2</u>	96.8	98.4	97.5	91.8	98.6	<u>97.9</u>
DeSTSeg [35]	99.2	96.2	98.5	95.0	98.6	99.6	99.8	98.8	99.2	97.4	<u>97.6</u>	99.4	93.5	<u>96.1</u>	99.1	<u>97.9</u>
Ours	98.6	97.5	98.5	99.2	95.9	98.7	<u>99.5</u>	97.8	93.1	98.4	94.0	<u>99.3</u>	98.5	95.6	98.7	97.6

TABLE 4 | Anomaly localization performance in terms of PRO on MV7ec AD dataset [18].

Method	Bottle	Cable	Capsule	Carpet	Grid	Hazelnut	Leather	Metal Nut	Pill	Screw	Tile	Toothbrush	Transistor	Wood	Zipper	Avg.
AE _{SSIM} [18]	83.4	47.8	86.0	64.7	84.9	91.6	56.1	60.3	83.0	88.7	17.5	78.4	72.5	60.5	66.5	69.4
SPADE [11]	95.5	<u>90.9</u>	93.7	94.7	86.7	95.4	97.2	94.4	94.6	<u>96.0</u>	75.6	93.5	87.4	87.4	92.6	91.7
PaDiM [42]	94.8	88.8	93.5	96.2	94.6	92.6	97.8	85.6	92.7	94.4	86.0	93.1	84.5	91.1	95.9	92.1
STFPM [32]	95.1	87.7	92.2	95.8	<u>96.6</u>	94.3	98.0	<u>94.5</u>	96.5	93.0	92.1	92.2	69.5	93.6	95.2	92.1
PatchCore [12]	<u>96.1</u>	92.6	<u>95.5</u>	<u>96.6</u>	95.9	93.9	<u>98.9</u>	91.3	94.1	97.9	87.4	91.4	83.5	89.6	97.1	93.5
DFR [19]	93.0	81.0	97.0	93.0	93.0	<u>97.0</u>	97.0	90.0	96.0	<u>96.0</u>	79.0	93.0	79.0	91.0	90.0	91.0
DSR [37]	94.8	85.3	82.9	93.4	98.4	92.1	97.7	89.8	92.5	89.8	97.0	93.7	77.2	86.4	94.1	91.0
SimpleNet [9]	92.6	90.8	92.9	93.1	94.1	89.8	97.2	87.8	94.1	95.9	92.7	91.3	<u>90.4</u>	77.9	95.1	91.7
DeSTSeg [35]	97.1	84.6	94.0	90.7	94.3	97.2	99.3	95.5	<u>96.4</u>	87.8	<u>93.0</u>	95.8	89.3	91.0	<u>96.6</u>	93.5
Ours	<u>96.1</u>	89.2	93.6	97.1	85.7	95.4	98.7	92.7	88.4	92.2	<u>87.6</u>	<u>93.9</u>	95.2	93.6	<u>95.7</u>	<u>93.0</u>

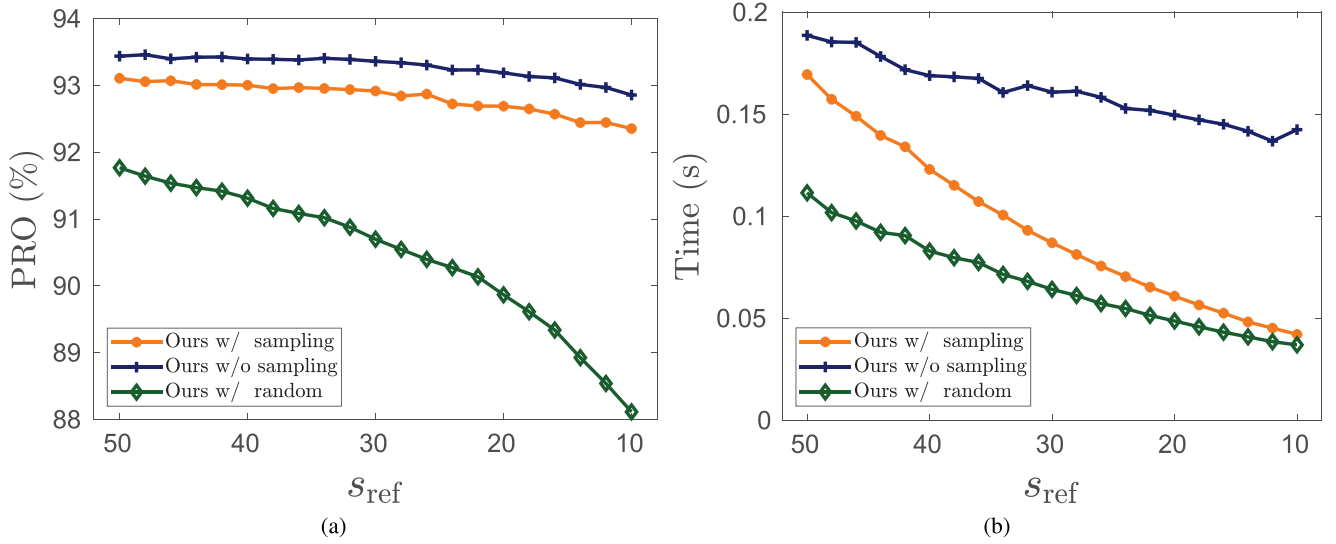


FIGURE 6 | Comparison of anomaly localization performance of different sampling mechanisms: (a) PRO and (b) inference time per image.

TABLE 5 | Mean anomaly localization performance using AP per dataset collection.

Method	MTD [51]	BTAD [52]	MVTec	
			AD [18]	Avg.
SPADE [11]	23.5	46.3	52.0	40.6
STFPM [32]	31.0	51.9	49.8	44.2
PatchCore [12]	33.6	50.8	57.1	47.2
DSR [37]	8.9	21.2	71.2	33.8
SimpleNet [9]	33.0	41.9	50.1	41.7
DeSTSeg [35]	31.5	39.1	73.2	47.9
Ours	34.1	53.0	63.1	50.1

of subspace-guided feature reconstruction, (3) Evaluation of the method generalization across different backbones.

4.3.1 | Effectiveness of Subspace-Based Sampling Technique

To further support the computational complexity analysis in Section 3.3, we investigate the anomaly localization performance on MVTec AD by ablating or adopting the proposed sampling technique. In addition, we compare the case of replacing the proposed sampling method with simple random sampling. To confirm the statistical results, we conducted the experiments by varying the number of data stored in the memory bank (s_{ref}) from 50 to 10, and the parameter determining the sparsity in the reconstruction step is fixed at $s = s_{ref}/2$. The reported inference times encompass the forward pass through the backbone network. As shown in Figure 6a, we can confirm that the PRO score of the proposed sampling technique consistently stays within a 0.5% range of that achieved without the sampling technique, regardless of the value of s_{ref} . In particular, Figure 6b shows that when s_{ref} is less than 20, the PRO score drops significantly for the random sampling method, while the proposed sampling method maintains the PRO score, and the inference time remains comparable to that of the random sampling.

4.3.2 | Expressiveness of Subspace-Guided Feature Reconstruction

In approaches based on the memory bank framework, while storing all nominal data in a memory bank can enhance anomaly localization performance, the computational complexity poses a significant impediment, rendering it impractical. Thus, the performance of such an approach depends on the ability to alternatively mimic out-of-bank features with limited in-sample features, even to encourage a sufficient data coverage. To verify the data coverage for out-of-bank data, we conduct a comparative analysis of the representation error $\|\mathbf{e}\|_2$ of out-of-bank features with the proposed reconstruction methodology and the feature matching employed by SPADE [11] and PatchCore [12] as follows:

$$\|\mathbf{e}\|_2 = \frac{1}{N} \sum_{j \in \{1, \dots, N\}} \|\mathbf{x}_j - \mathbf{x}_j^*\|_2, \quad (9)$$

where \mathbf{x}_j^* approximates or is the nearest vector to \mathbf{x}_j . In our reconstruction methodology, \mathbf{x}_j^* denotes the data point that is represented through the linear combination in Equation (8). In the feature matching-based method, the top- s_{ref} data that exhibit proximity in spatial distance to \mathbf{x}_j are stored in M , and \mathbf{x}_j^* denotes the nearest neighbor data point as follows: $\mathbf{x}_j^* = \min_{\mathbf{x}_i \in M} \|\mathbf{x}_j - \mathbf{x}_i\|_2$. In Figure 7, we can observe that our subspace-guided feature reconstruction in the three settings in MVTec AD provides a better representation of out-of-bank data than the feature matching-based method. Particularly in the texture category, the feature patterns of the out-of-bank data cannot be well represented by feature matching because they are diverse for the data coverage of the in-sample data. Conversely, our approach is able to mimic the out-of-bank features using limited data by the benefit of the self-expressive model. This result supports the concept that our methodology enables adaptive feature representation even with a limited number of in-sample data, which leads to improved anomaly localization performance for the texture categories presented in Section 4.2.

We also investigate the expressiveness of our feature reconstruction methodology on limited nominal data. In particular, we

TABLE 6 | Anomaly localization performance for different backbones on MVTec AD dataset [18]. Comparative results are measured by AUROC/PRO/inference time per image (s).

Backbone	Ours (w/o sampling)	Ours (w/ sampling)
ResNet50 [45]	97.3 / 92.3 / 0.180	97.2 / 91.8 / 0.115
ResNet101 [45]	97.6 / 92.9 / 0.183	97.5 / 92.5 / 0.116
WideResNet50 [55]	97.6 / 93.4 / 0.184	97.6 / 93.0 / 0.117
WideResNet101 [55]	97.8 / 93.7 / 0.183	97.7 / 93.3 / 0.123
EfficientNet-B1 [57]	97.5 / 93.2 / 0.068	97.3 / 92.9 / 0.065
EfficientNet-B7 [57]	97.3 / 93.2 / 0.106	97.1 / 92.9 / 0.084

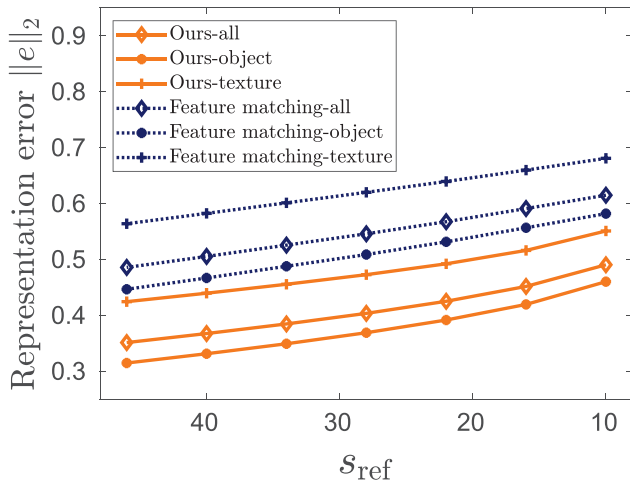


FIGURE 7 | Evaluation of data coverage for out-of-bank data: representation error of out-of-bank data as a function of s_{ref} in the three settings of object, texture, and both in MVTec AD dataset [18].

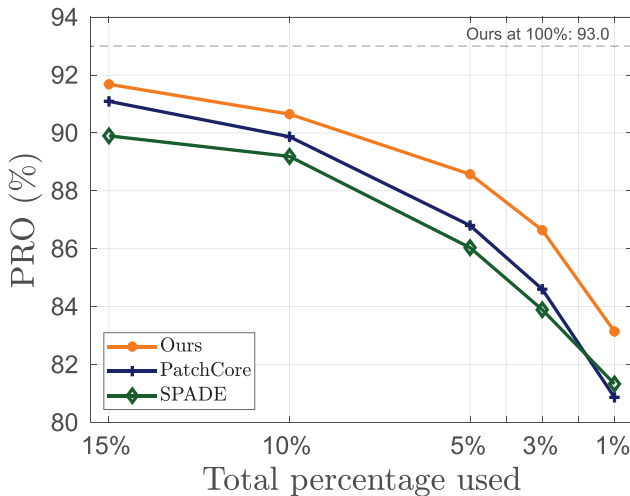


FIGURE 8 | Curves of the change of PRO score with respect to the total percentage used in the overall nominal training data under low-shot configuration on MVTec AD dataset [18].

vary the amount of training samples from 15% to 1% of the total nominal training data on the MVTec AD dataset and compare it to memory bank-based approaches using WideResNet50. As shown in Figure 8, even when utilizing only 15% of the total

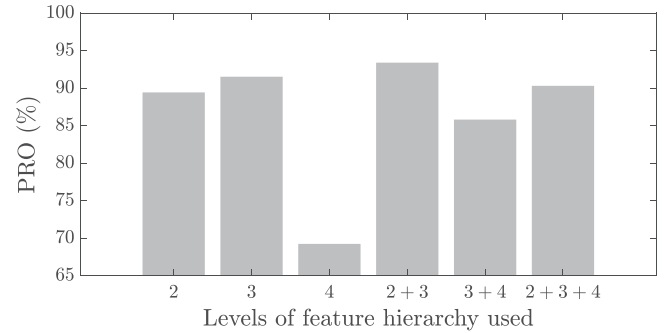


FIGURE 9 | Evaluation of network feature hierarchies and anomaly localization performance in terms of PRO on MVTec AD dataset [18].

nominal training data, our method remains within a 1.5% range of the performance compared to the case of fully using the training data (Figure 8, dashed line). Furthermore, we can see that our method consistently outperforms all competitive methods when the number of training samples is reduced to 1%. We can thus confirm that our method suffices to present rich expressiveness even on a limited number of nominal training data.

4.3.3 | Method Generalization Across Different Hierarchies and Backbones

We investigate the generalization capability of the proposed reconstruction methodology (Section 3.1) for features stored in the memory bank. Figure 9 shows the single or multi-scale localized predictions obtained by indexing WideResNet50 blocks, where $l \in \{2, 3, 4\}$. Obviously, the result using the features from hierarchy level 4 is significantly lower than those using other features. This is partly due to the fact that high-level features, which are very deep and abstract, lose localized information due to their low resolution. Moreover, these features in ImageNet pre-trained networks are biased toward natural image classification tasks [12]. Consequently, a practical choice would involve utilizing at least a combination of low-level features with a high resolution in our feature reconstruction methodology.

Table 6 shows the performance of the proposed method (with and without a sampling technique) using different backbones. We observe that the anomaly localization performance with the proposed approach remains consistently stable across different backbones. In terms of the timings, we can see that our sampling method significantly improves the inference time on the ResNet

family, while the gain on the EfficientNet family is comparably small. This is expected to be due to the small difference in the dimensionality of feature maps at successive hierarchical levels, which causes the cost for sampling to dominate the overall computational expense.

5 | Conclusion

In this paper, we proposed a novel subspace-guided feature reconstruction framework for anomaly localization. Moreover, we proposed a subspace-based sampling technique that aggregates the basis vectors of forming a subspace to retain high inference speeds. Extensive experiments on several public datasets have validated the efficiency and effectiveness of our proposed method for localization accuracy and dependence on the number of data.

Although our method is designed to efficiently localize anomalies using subspace-based sampling, its reliance on the memory bank may induce some computational overhead imposed by the calling procedure. In the future, we would like to devise a memory bank-free yet end-to-end trainable reconstruction model using unsupervised self-expressive networks. In addition, our method currently assumes the data is distributed in linear subspaces. We would also like to focus on developing a stronger reconstruction mechanism to explore the non-linear subspaces for anomaly localization.

Author Contributions

Katsuya Hotta: conceptualization, data curation, formal analysis, funding acquisition, methodology, project administration, resources, software, validation, visualization, writing – original draft, writing – review & editing. **Chao Zhang:** conceptualization, funding acquisition, writing – review & editing. **Yoshihiro Hagihara:** funding acquisition, writing – review & editing. **Takuya Akashi:** writing – review & editing.

Acknowledgements

This work was supported in part by the JSPS KAKENHI (Grant Numbers JP23K19989 and JP24K20828).

Conflicts of Interest

The authors declare no conflicts of interest.

Data Availability Statement

The data that support the findings of this study are available from the corresponding author upon reasonable request.

References

1. P. Bergmann, K. Batzner, M. Fauser, D. Sattlegger, and C. Steger, “Beyond Dents and Scratches: Logical Constraints in Unsupervised Anomaly Detection and Localization,” *International Journal of Computer Vision* 130, no. 4 (2022): 947–969.
2. Y. Zou, J. Jeong, L. Pemula, D. Zhang, and O. Dabeer, “Spot-the-Difference Self-Supervised Pre-Training for Anomaly Detection and Segmentation,” in *European Conference on Computer Vision* (Springer, 2022), 392–408.
3. R. A. Shah, O. Urmonov, and H. Kim, “Two-Stage Coarse-to-Fine Image Anomaly Segmentation and Detection model,” *Image and Vision Computing* 139 (2023): 104817.

4. W. Li, V. Mahadevan, and N. Vasconcelos, “Anomaly Detection and Localization in Crowded Scenes,” *IEEE Transactions on Pattern Analysis and Machine Intelligence* 36, no. 1 (2013): 18–32.
5. W. Liu, W. Luo, D. Lian, and S. Gao, “Future Frame Prediction for Anomaly Detection—a New Baseline,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (IEEE, 2018), 6536–6545.
6. H. Chen, X. Mei, Z. Ma, X. Wu, and Y. Wei, “Spatial–Temporal Graph Attention Network for Video Anomaly Detection,” *Image and Vision Computing* 131 (2023): 104629.
7. X. Tao, X. Gong, X. Zhang, S. Yan, and C. Adak, “Deep Learning for Unsupervised Anomaly Localization in Industrial Images: A Survey,” *IEEE Transactions on Instrumentation and Measurement* 71 (2022): 1–21.
8. H. Deng and X. Li, “Anomaly Detection via Reverse Distillation From One-Class Embedding,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (IEEE, 2022), 9737–9746.
9. Z. Liu, Y. Zhou, Y. Xu, and Z. Wang, “SimpleNet: A Simple Network for Image Anomaly Detection and Localization,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (IEEE, 2023), 20402–20411.
10. J. Deng, W. Dong, R. Socher, L. J. Li, K. Li, and L. Fei-Fei, “Imagenet: A Large-Scale Hierarchical Image Database,” in: *IEEE Conference on Computer Vision and Pattern Recognition* (IEEE, 2009), 248–255.
11. N. Cohen and Y. Hoshen, “Sub-Image Anomaly Detection With Deep Pyramid Correspondences,” CoRR: abs/2005.02357 (2020).
12. K. Roth, L. Pemula, J. Zepeda, B. Schölkopf, T. Brox, and P. Gehler, “Towards Total Recall in Industrial Anomaly Detection,” in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (IEEE, 2022), 14298–14308.
13. Y. Sun, H. Wang, Y. Hu, H. Jiang, and B. Yin, “MBMF: Constructing Memory Banks of Multi-Scale Features for Anomaly Detection,” *IET Computer Vision* 18, no. 3 (2023): 355–369.
14. S. Lee, S. Lee, and B. C. Song, “Cfa: Coupled-Hypersphere-Based Feature Adaptation for Target-Oriented Anomaly Localization,” *IEEE Access* 10 (2022): 78446–78454.
15. E. Eskin, A. Arnold, M. Prerau, L. Portnoy, and S. Stolfo, “A Geometric Framework for Unsupervised Anomaly Detection: Detecting Intrusions in Unlabeled Data,” in *Applications of Data Mining in Computer Security* (Springer, 2002), 77–101.
16. P. K. Agarwal, S. Har-Peled, K. R. Varadarajan, et al., “Geometric Approximation via Coresets,” *Combinatorial and Computational Geometry* 52, no. 1 (2005): 1–30.
17. P. Bergmann, S. Löwe, M. Fauser, D. Sattlegger, and C. Steger, “Improving Unsupervised Defect Segmentation by Applying Structural Similarity to Autoencoders,” CoRR:abs/1807.02011 (2018).
18. P. Bergmann, M. Fauser, D. Sattlegger, and C. 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* (IEEE, 2019), 9592–9600.
19. J. Yang, Y. Shi, and Z. Qi, “Dfr: Deep Feature Reconstruction for Unsupervised Anomaly Segmentation,” CoRR:abs/2012.07122 (2020).
20. K. Hotta, H. Xie, and C. Zhang, “Component-Based Nearest Neighbour Subspace Clustering,” *IET Image Processing* 16, no. 10 (2022): 2697–2708.
21. K. Hotta, T. Akashi, S. Tokai, and C. Zhang, “PMSSC: Parallelizable Multi-Subset Based Self-Expressive Model for Subspace Clustering,” *Computational Visual Media* 9, no. 3 (2023): 479–494.
22. E. Elhamifar and R. Vidal, “Sparse Subspace Clustering: Algorithm, Theory, and Applications,” *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35, no. 11 (2013): 2765–2781.
23. D. L. Donoho, “For Most Large Underdetermined Systems of Linear Equations the Minimal ℓ_1 -Norm Solution is Also the Sparsest Solution,”

24. R. P. de Monteiro, M. C. Lozada, D. R. C. Mendieta, R. V. S. Loja, and C. J. A. Bastos Filho, “Bastos-Filho A Hybrid Prototype Selection-Based Deep Learning Approach for Anomaly Detection in Industrial Machines,” *Expert Systems with Applications* 204 (2022): 117528.
25. L. Bergman, N. Cohen, and Y. Hoshen, “Deep Nearest Neighbor Anomaly Detection,” CoRR:abs/2002.10445 (2020).
26. M. Rudolph, T. Wehrbein, B. Rosenhahn, and B. Wandt, “Fully Convolutional Cross-Scale-Flows for Image-Based Defect Detection,” in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision* (IEEE, 2022), 1088–1097.
27. S. Akçay, A. Atapour-Abarghouei, T. P. Breckon, “Ganomaly: Semi-Supervised Anomaly Detection via Adversarial Training,” in *Proceedings of the Asian Conference on Computer Vision* (Springer, 2019), 622–637.
28. S. Akçay, A. Atapour-Abarghouei, and T. P. Breckon, “Skip-Ganomaly: Skip Connected and Adversarially Trained Encoder-Decoder Anomaly Detection,” in *2019 International Joint Conference on Neural Networks (IJCNN)* (IEEE, 2019), 1–8.
29. D. P. Kingma and M. Welling, “Auto-Encoding Variational Bayes,” in *Proceedings of the International Conference on Learning Representations* (IEEE Information Theory Society, 2014).
30. P. Bergmann, M. Fauser, D. Sattlegger, and C. Steger, “Uninformed Students: Student-Teacher Anomaly Detection With Discriminative Latent Embeddings,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (IEEE, 2020), 4183–4192.
31. G. Hinton, O. Vinyals, and J. Dean, “Distilling the Knowledge in a Neural Network,” CoRR:abs/1503.02531 (2015).
32. G. Wang, S. Han, E. Ding, and D. Huang, “Student-Teacher Feature Pyramid Matching for Anomaly Detection,” in *Proceedings of the British Machine Vision Conference* (The British Machine Vision Association, 2021).
33. L. Capogrosso, F. Girella, F. Taioli, et al., “Diffusion-Based Image Generation for In-Distribution Data Augmentation in Surface Defect Detection,” in *19th International Conference on Computer Vision Theory and Applications* (IEEE, 2024).
34. C. L. Li, K. Sohn, J. Yoon, and T. Pfister, “Cutpaste: Self-Supervised Learning for Anomaly Detection and Localization,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (IEEE, 2021), 9664–9674.
35. X. Zhang, S. Li, X. Li, P. Huang, J. Shan, and T. Chen, “DeSTSeg: Segmentation Guided Denoising Student-Teacher for Anomaly Detection,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (IEEE, 2023), 3914–3923.
36. V. Zavrtnik, M. Kristan, and D. Skočaj, “Draem-a Discriminatively Trained Reconstruction Embedding for Surface Anomaly Detection,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision* (IEEE, 2021), 8330–8339.
37. V. Zavrtnik, M. Kristan, and D. Skočaj, “Dsr—a Dual Subspace Re-Projection Network for Surface Anomaly Detection,” in *European Conference on Computer Vision* (Springer, 2022), 539–554.
38. N. Yao, Y. Zhao, Y. Guo, and S. G. Kong, “Few-Sample Anomaly Detection in Industrial Images With Edge Enhancement and Cascade Residual Feature Refinement,” *IEEE Transactions on Industrial Informatics* 20, no. 12 (2024): 13975–13985.
39. X. Zhang, N. Li, J. Li, T. Dai, Y. Jiang, and S. T. Xia, “Unsupervised Surface Anomaly Detection With Diffusion Probabilistic Model,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision* (IEEE, 2023), 6782–6791.
40. R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, “High-Resolution Image Synthesis With Latent Diffusion Models,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (IEEE, 2022), 10684–10695.
41. X. Zhang, M. Xu, and X. Zhou, “RealNet: A Feature Selection Network With Realistic Synthetic Anomaly for Anomaly Detection,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (IEEE, 2024), 16699–16708.
42. T. Defard, A. Setkov, A. Loesch, and R. Audigier, “Padim: A Patch Distribution Modeling Framework for Anomaly Detection and Localization,” in *International Conference on Pattern Recognition* (Springer, 2021), 475–489.
43. P. C. Mahalanobis, “On the Generalized Distance in Statistics,” *Sankhyā: The Indian Journal of Statistics, Series A (2008-)* 80 (2018): S1–S7.
44. R. Vidal, “Subspace Clustering,” *IEEE Signal Processing Magazine* 28, no. 2 (2011): 52–68.
45. K. He, X. Zhang, S. Ren, and J. Sun, “Deep Residual Learning for Image Recognition,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (IEEE, 2016), 770–778.
46. C. You, and R. Vidal, “Geometric Conditions for Subspace-Sparse Recovery,” in *International Conference on Machine Learning* (IEEE Information Theory Society, 2015), 1585–1593.
47. C. You, D. Robinson, and R. Vidal, “Scalable Sparse Subspace Clustering by Orthogonal Matching Pursuit,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (IEEE, 2016), 3918–3927.
48. Y. C. Pati, R. Rezaifar, and P. S. Krishnaprasad, “Orthogonal Matching Pursuit: Recursive Function Approximation With Applications to Wavelet Decomposition,” in *Proceedings of Asilomar Conference on Signals, Systems and Computers* (IEEE, 1993), 40–44.
49. J. A. Tropp, “Greed is Good: Algorithmic Results for Sparse Approximation,” *IEEE Transactions on Information Theory* 50, no. 10 (2004): 2231–2242.
50. M. A. Davenport and M. B. Wakin, “Analysis of Orthogonal Matching Pursuit Using the Restricted Isometry Property,” *IEEE Transactions on Information Theory* 56, no. 9 (2010): 4395–4401.
51. Y. Huang, C. Qiu, and K. Yuan, “Surface Defect Saliency of Magnetic Tile,” *The Visual Computer* 36 (2020): 85–96.
52. P. Mishra, R. Verk, D. Fornasier, C. Piciarelli, G. L. Foresti, “VT-ADL: A vision Transformer Network for Image Anomaly Detection and Localization,” in *Proceedings of the IEEE 30th International Symposium on Industrial Electronics (ISIE)* (IEEE, 2021), 01–06.
53. J. Yu, Y. Zheng, X. Wang, et al., “Fastflow: Unsupervised Anomaly Detection and Localization via 2D Normalizing Flows,” CoRR:abs/2111.07677 (2021).
54. V. Zavrtnik, M. Kristan, and D. Skočaj, “Reconstruction by Inpainting for Visual Anomaly Detection,” *Pattern Recognition* 112 (2021): 107706.
55. S. Zagoruyko, and N. Komodakis, “Wide Residual Networks,” in *Proceedings of the British Machine Vision Conference* (The British Machine Vision Association, 2016), 87.1–87.12.
56. M. Rudolph, B. Wandt, and B. Rosenhahn, “Same Same but Differnet: Semi-Supervised Defect Detection With Normalizing Flows,” in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision* (IEEE, 2021), 1907–1916.
57. M. Tan, and Q. Le, “Efficientnet: Rethinking Model Scaling for Convolutional Neural Networks,” in *International Conference on Machine Learning* (PMLR, 2019), 6105–6114.