S2VAD: A Self-Supervised Framework for Unsupervised Visual Anomaly Detection

Mariam Ishtiaq[1,2], Jongun Won[1,2]
[1]Railroad Physical AI Research Department, Korea Railroad Research Institute (KRRI), Uiwang, South Korea
[2]Transportation System Engineering, University of Science and Technology (UST), Daejeon, South Korea
{mariam16, juwon}@krri.re.kr

***Abstract*—In open world environments, vision anomaly detection (VAD) is inherently complex due to diverse and unpredictable anomaly manifestations. To overcome the limitation of the availability of an inclusive training data, unsupervised anomaly detection (UAD) methods provide a solid baseline. Principally, using only normal distribution patterns for training, any deviations can be flagged as anomalies during inference. Following the same analogy, we propose a self-supervised framework for unsupervised anomaly detection termed S2VAD. We explore the potential of DINOv2 as an encoder for normal feature extraction, a bottleneck as a compression head, and a transformer decoder using self-attention for VAD. We use a global cosine loss for comparing the test image to the learned normality and detecting anomalies. Using the benchmark MVTec-AD dataset, our work shows state-of-the-art (SOTA) performance on texture classes with an image AUROC of 99.8%. To explore the model further, we share interesting future work directions.**

***Index Terms*—unsupervised anomaly detection, self-supervised learning, transformers, autoencoders, foundation models.**

I. Introduction

Vision anomaly detection (VAD) is one of the key open research problems in computer vision due to its increasing demand in automation. However, the diverse and evolutionary nature of anomalies poses limitations to the development of a generalizable solution. Conventionally, class and category-wise anomaly detection (AD) models have been developed, like semi-supervised GANomaly [1] and fully supervised YOLO [2]. But these methods require accurate data labels, are memory inefficient, and hinder scalability. Therefore, a cumulatively trained model is essential for not only cross-domain compatibility but also to address one-class variety. To address this problem, foundation models have been used for self-supervised pretraining in unsupervised anomaly detection [3].

Formally, AD can be defined as the simultaneous anomaly classification and localization problem. Recent literature has focused on increasing the generalizability of applications, improving classification and localization accuracy, and overcoming challenges in dataset availability. This work aims to address these challenges by making the following key contributions:

- The proposed **self-supervised** framework for unsupervised **VAD**, *S2VAD* is a multi-class *VAD* pipeline. Built on a pre-trained foundation model encoder, it leverages self-supervised feature representations for both texture and object categories. The reconstruction-based framework gives strong generalization across diverse visual domains without class-specific fine-tuning.
- For stable reconstruction of normal patterns, our method incorporates **feature-space regularization** using **bottleneck**, **multilevel consistency constraints** using a **multilayer decoder**, and a **cosine similarity loss**. This pipeline ensures a sharper separation between normal and anomalous regions and enhances anomaly localization accuracy.
- A detailed analysis of **the proposed S2VAD's strengths and weaknesses** provides a strong baseline for future enhancements and work directions.

This research was supported by a grant from the R&D Program: 'Development of an AI-based Integrated Railway Digital Twin Platform,' grant number PK2501D1, of the Korea Railroad Research Institute (KRRI).

II. Literature Review

VAD refers to the detection of unusual and undesired patterns in images, which, given the diverse nature of anomalies, motivates the use of unsupervised learning methods to overcome inherent data limitations and labeling overhead. VAD finds applications in myriad domains, including industry [4], medicine [5], and surveillance [6]. Recent literature shows a growing interest in developing a unified VAD solution.

By definition, anomalies are rare and do not conform to the model's understanding of normality. Therefore, reconstruction-based methods [7] have been developed to identify data points that significantly deviate from the learned patterns. When the underlying structure of training data is normal, such abnormalities are easier to flag.

Methodologies using autoencoders, like segmentation-guided denoising student-teacher for anomaly detection (DeSTSeg) [8], use a pre-trained teacher network and a denoising student encoder-decoder. The student learns to denoise the input (or generate feature representations that are robust to anomalies) by aligning it with the 'normal' features extracted by the teacher.

Denoising diffusion probabilistic models (DDPM) [9], [10] have also been used for AD. In these methods, the training phase uses normal data, and inverts (denoises/ reconstructs) a progressive noising process. The model's failure or deviation in this inversion is used as the anomaly score.

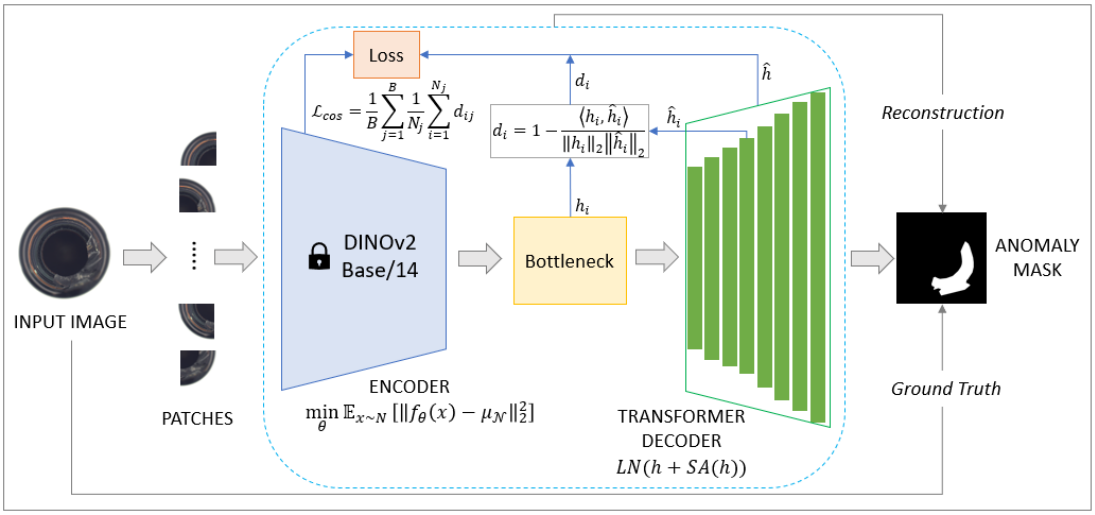

Fig. 1. **The proposed S2VAD Framework.** The pipeline uses a DINOv2-Base/14 encoder for feature extraction, followed by a bottleneck for feature compression. A transformer decoder with self-attention is used for feature reconstruction. We use a global cosine feature alignment loss for anomaly mapping and mask generation.

III. Proposed Model - S2VAD

This section gives a technical detail about the model and the components of the proposed pipeline. A detailed schematic of the S2VAD is shown in Figure 1.

A. Pipeline

We lay the following four building blocks to create the S2VAD pipeline:

1) **DINOv2-ViT-Base/14 encoder [11]:** DINOv2 is a family of foundation models readily invoked in literature for feature extraction in a wide range of applications [3]. Since foundation models leverage millions of images for training, they can be used for universal downstream tasks without requiring explicit fine-tuning. This also broadens their applicability to visual tasks beyond VAD on both *image-level*, like image and object classification, instance retrieval, and video interpretation, and *pixel-level*, like depth estimation, dense matching, and semantic segmentation.

   The use of transferable frozen features for context understanding presents a case for self-supervised learning as a solution using the out-of-distribution approach. This essentially realizes the rationale behind the proposed reconstruction-based AD model, S2VAD, where the encoder can learn rich 'normal' features using the frozen DINOv2 layers.

   In S2VAD, for an input image x from the dataset D, the pretrained DINOv2-Base/14 encoder, f_θ extracts a set of patch embeddings $\{z_i\}_{i=1}^{N} = f_\theta(x)$, where N is the total number of extracted patches. The model learns normal distributions as consistent representations by minimizing their gap, which can be given by (1):

$$\min_\theta \ \mathbb{E}_{x\sim\mathcal{N}}\left[\|f_\theta(x) - \mu_\mathcal{N}\|_2^2\right], \tag{1}$$

   Here $f_\theta(x)$ denotes the feature embedding of input x, $\mu_\mathcal{N}$ is the mean feature vector of all normal samples, and $\mathbb{E}_{x\sim\mathcal{N}}[\cdot]$ shows the expectation taken over the distribution of normal data. $\|\cdot\|_2^2$ is the mean squared distance between each encoded normal feature and the centroid of the normal feature distribution, $\mu_\mathcal{N}$. The resulting minimization in (1) facilitates the separation of normal from anomalous patterns, offering efficient learning.

2) **Bottleneck:** The high-dimensional and semantically rich features extracted from the encoder are compressed using the bottleneck. This gives a compact and discriminative subspace of tightly clustered normal patterns, helpful for the effective separation of anomalies in task-agonistic applications. We used the MLP bottleneck as (2):

$$h = g_\phi(z) = \sigma(W_2\,\delta(W_1 z + b_1) + b_2), \tag{2}$$

   where, $g_\phi(\cdot)$ is the MLP projection parameterized by layer-wise weights and biases, $\phi = \{W_1, W_2, b_1, b_2\}$, $\delta(\cdot)$ represents a non-linear activation function such as ReLU, and $\sigma(\cdot)$ denotes regularization, which we introduced in the form of jitter. The output $h \in \mathbb{R}^{d'}$ (with d' being the latent feature vector dimension) is a compact latent representation that serves as the normality-aware feature used to score downstream anomalies. The resulting close clustering of the normal subspace is analogous to a higher sensitivity of the model to deviations introduced by anomalous inputs.

3) **Transformer Decoder:** To reconstruct the input features, a transformer decoder is used to generate predictions from the compact latent representations h, from the bottleneck block. We use *self-attention* (SA$(\cdot)$) to capture dependencies among feature tokens in the decoder, allowing the model to learn contextual relationships and

correlations between different spatial regions (patches) of the input image. With an accurate reconstruction of normal patterns, the decoder enables the detection of anomalies as deviations between reconstructed and original features. We define a transformer decoder layer in (3):

$$\hat{h} = \text{DecLayer}(h) = \text{LN}\big(h + \text{SA}(h)\big), \tag{3}$$

where, $\text{LN}(\cdot)$ denotes *layer normalization*, such that the residual connection $h + \text{SA}(h)$ stabilizes training and preserves original feature information. In our multi-layer setup, the decoder consists of $L = 8$ stacked self-attention layers, which is a design choice rather than a hyperparameter.

4) **Global cosine feature alignment loss:** To enforce compactness of normal features in the latent space, we define a cosine similarity-based loss between reference features h (detached normal features) and the reconstructed decoder features $\hat{h}$ (trainable features). The point-wise cosine distance is given by (4):

$$d_i = 1 - \frac{\langle h_i, \hat{h}_i \rangle}{\|h_i\|_2 \, \|\hat{h}_i\|_2}, \quad i = 1, \ldots, N, \tag{4}$$

where $\langle \cdot, \cdot \rangle$ denotes the dot product.

The overall global cosine loss is then computed as the mean over all batches and samples using (5):

$$\mathcal{L}_{\text{cos}} = \frac{1}{B} \sum_{j=1}^{B} \frac{1}{N_j} \sum_{i=1}^{N_j} d_{ij}, \tag{5}$$

where B is the batch size and N_j is the number of patches in sample j. To define an anomaly, we scale the gradients of distances below a percentile threshold to focus on more significant deviations. This loss encourages normal features to cluster around a compact latent distribution, making deviations introduced by anomalous inputs more prominent and hence, more easily detectable.

IV. Implementation Details

This section gives S2VAD's implementation environment, the dataset and hyperparameters.

A. Development environment

We used NVIDIA RTX 3070 with 8GB RAM to implement S2VAD. The environment was built on a Windows 10 platform using python 3.8.12 and torch2.0.1+cu118. The code and pretrained model weights of DINOv2 are released under the Apache License 2.0.

B. Dataset

The MVTec-AD dataset [12] is a benchmark dataset for the development and evaluation of industrial VAD and anomaly localization methods. It contains 5,354 high-resolution images with 15 different object and texture categories. Each category of the dataset includes:

- Normal (defect-free) images for training.
- Anomalous (defective) images for testing, with various real-world defects such as scratches, dents, contaminations, or misprints.

TABLE I
COMPREHENSIVE PERFORMANCE OF MVTEC-AD DATASET USING THE PROPOSED AD MODEL. THE CLASS-WISE AND AVERAGE RESULTS FOR IMAGE AND PIXEL-BASED EVALUATION PARAMETERS ARE SHOWN.

Class	**Image**			**Pixel**			
	I-AUROC	**I-AP**	**I-F1**	**P-AUROC**	**P-AP**	**P-F1**	**P-AUPRO**
				Texture			
carpet	1.0000	1.0000	1.0000	0.9923	0.6917	0.6774	0.9661
grid	0.9967	0.9988	0.9912	0.9834	0.3041	0.3727	0.9336
leather	1.0000	1.0000	1.0000	0.9917	0.4499	0.4950	0.9703
tile	0.9978	0.9992	0.9881	0.9591	0.5669	0.6821	0.8503
wood	0.9947	0.9984	0.9836	0.9545	0.5451	0.5533	0.9042
Average (Texture)	**0.9979**	**0.9993**	**0.9906**	**0.9768**	**0.5182**	**0.5561**	**0.9224**
				Object			
bottle	0.9905	0.9972	0.9839	0.9650	0.6675	0.6691	0.8720
cable	0.7701	0.8726	0.7745	0.8636	0.2808	0.3560	0.5748
capsule	0.7475	0.9315	0.9145	0.9522	0.3870	0.4311	0.8179
hazelnut	0.9929	0.9959	0.9787	0.9848	0.6165	0.6154	0.9239
metal_nut	0.9306	0.9844	0.9286	0.9146	0.6503	0.6496	0.7729
pill	0.9048	0.9830	0.9156	0.9331	0.4107	0.4361	0.7204
screw	0.8096	0.9215	0.8863	0.9619	0.1243	0.2018	0.8401
toothbrush	0.8861	0.9551	0.8955	0.9717	0.4736	0.5181	0.8080
transistor	0.8575	0.8522	0.7671	0.8199	0.3663	0.3745	0.6196
zipper	0.9758	0.9934	0.9669	0.9624	0.4635	0.5321	0.8994
Average (Object)	**0.8875**	**0.9488**	**0.9018**	**0.9422**	**0.4091**	**0.4514**	**0.7845**
Overall Average	**0.9236**	**0.9656**	**0.9322**	**0.9548**	**0.4497**	**0.4870**	**0.8291**

- Pixel-accurate ground-truth masks for evaluating anomaly localization performance.

C. Hyperparameters

We use an input image size of 256 with a center crop size of 224. Middle layers 2 to 9 of DINOv2-Base/14 are used for feature extraction in the encoder. StableAdamW optimizer is used with a learning rate of 2e-3. The network is trained for 10,000 iterations with a batch size of 8.

V. Results and Analysis

We share S2VAD's performance, its limitations and future work directions in this section.

A. Model Performance

A detailed result of the class and category-wise performance of S2VAD on MVTec-AD dataset is shown in Table I.

We report the performance in terms of standard anomaly detection metrics including AUROC, AP and F1 for both image and pixel levels and AUPRO for pixel-level performance. This determines the anomaly classification and localization capability of S2VAD.

*1) **Strengths**:* The overall image-level AUROC of 92.4% and I-AP of 96.6% demonstrate strong discrimination between normal and anomalous samples across all classes. The consistency of pixel-level defect classification is also ensured with the average pixel-level AUROC (P-AUROC) of 95.5%.

A summary of the model performance using image and pixel-wise metrics is shown in Fig. 2. Overall, the texture categories (average I-AUROC: 99.8%) outperform object categories (average I-AUROC: 88.8%), suggesting that the model captures repetitive structural regularities well. This strength of S2VAD is evident from the closed cluster of texture categories.

*2) **AUROC performance**:* Table II shows the performance of the model compared to literature in terms of I-AUROC and P-AUROC. S2VAD outprforms other reconstruction-based methods like AE-SSIM and RIAD from [13] and SOTA AD methods from anomalib [14] like efficient AD (EffAD), student-teacher feature pyramid matching for anomaly detection (STFPM) and FastFlow

*3) **AUPRO performance**:* The pixel-level localization performance using the AUPRO comparison of S2VAD with recent literature is given in Table III. Few-shot methods like window-based contrastive language-image pretraining (WinCLIP) [15], denoising diffusion probabilistic models (DDPM), latent diffusion model (LDM) [16], diffusion-based anomaly detection (DiAD) [17], and an autoencoder-based method, DeSTSeg [17] are used as comparison baselines.

B. Limitations

We observe and report the following limitations in S2VAD.

1) **Confidence calibration:** From Table I, an average P-AP of 44.9% indicates that while detection is accurate globally (AUROC), the confidence calibration of anomaly scores needs improvement.
2) **Imbalanced texture-object generalization:** The significant performance gap between texture and object subsets implies that the model's learned representations may lean more toward pattern-based regularity rather than shape or structure-based variations, which are common in objects.

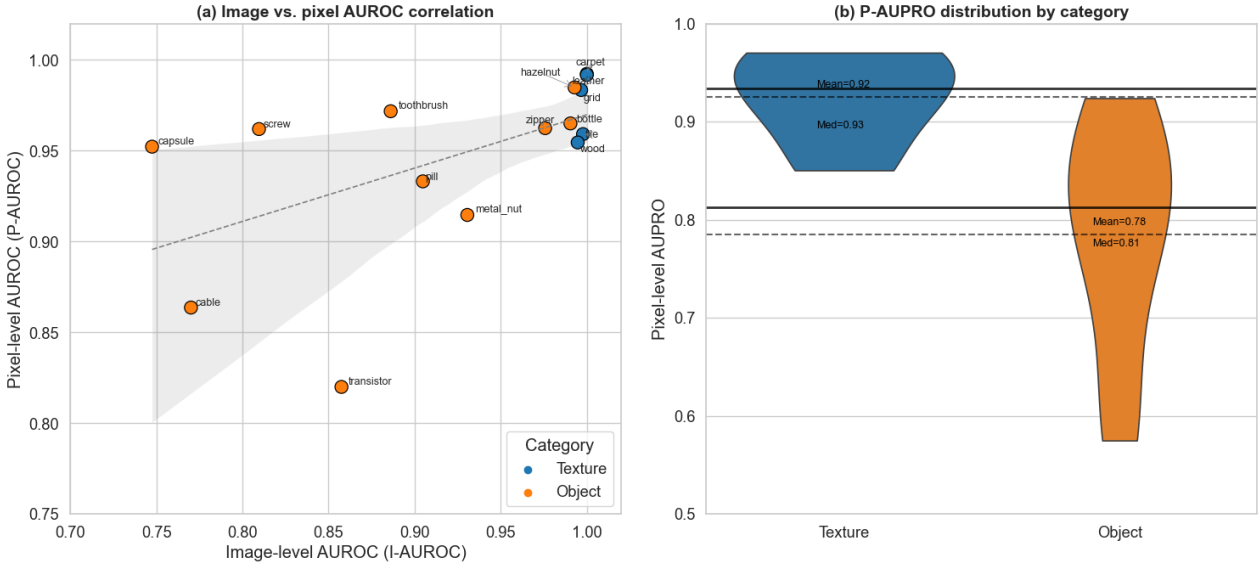

Fig. 2. **Image and pixel-wise performance comparison across texture and object categories.** Figure (a) shows the image vs. pixel AUROC correlation. The texture classes are closely clustered, while object classes are relatively sparse. Figure (b) shows the pixel-wise AUPRO distribution by category. Texture classes show a higher mean of 92%, while object classes have a mean of 78%.

TABLE II
COMPARISON OF I-AUROC AND P-AUROC FOR S2VAD AGAINST DIFFERENT MODELS ON TEXTURE, OBJECT, AND OVERALL CATEGORIES OF MVTECAD.

Model	I-AUROC	P-AUROC
Texture		
AE-SSIM	78.0	56.7
RIAD	95.1	93.7
STFPM	96.3	97.2
EffAD	82.9	85.9
FastFlow	87.7	91.4
S2VAD (Ours)	99.8	97.7
Object		
AE-SSIM	91.0	75.8
RIAD	89.9	94.3
STFPM	87.2	92.4
EffAD	85.1	88.6
FastFlow	76.8	91.9
S2VAD (Ours)	88.8	94.2
Overall		
AE-SSIM [13]	87.0	69.4
RIAD [13]	91.7	94.2
STFPM [18]	90.3	93.9
EffAD [18]	84.3	87.7
FastFlow [18]	80.4	91.7
S2VAD (Ours)	92.3	95.5

TABLE III
COMPARISON OF AUPRO FOR S2VAD AGAINST DIFFERENT MODELS USIGN MVTEC-AD.

Model	AUPRO
WinCLIP [15]	64.6
DDPM [16]	49.0
LDM [16]	66.3
DeSTSeg [17]	82.6
DiAD [17]	64.4
S2VAD (Ours)	82.9

C. Future Work Directions

The results in previous section are encouraging enough to use this baseline approach for improvements in the following directions:

- A detailed ablation analysis of encoder base and transformer decoder architecture, is required to improve the model's object detection capability. Options can be explored to avoid over-smoothing spatial features or underestimating the local variance in defect boundaries.
- Decoder architecture and loss functions can be evaluated to better handle irregular object geometries, complex textures, and subtle, small-scale anomalies. Defect sensitivity can be investigated via multi-scale attention and boundary-aware loss functions.

The weak anomaly localization capacity of the model, particularly for object classes, can be addressed with hybrid models. An extended version of this work [19] overcomes these limitations using self-attention with masking for local and linear attention for global AD, using a hybrid attention decoder.

VI. CONCLUSION

In this paper, we lay the foundation work to analyze the potential of the DINOv2 encoder with a self-attention based transformer decoder for VAD using S2VAD. Results on the benchmark dataset MVTec-AD show comparable performance to recent work. A high image and pixel performance shows that S2VAD has the potential to be applied to numerous surface anomaly detection problems in autonomous driving, railway track state monitoring, and industrial anomaly detection.

REFERENCES

[1] Z. Li, Y. Yan, X. Wang *et al.*, "A survey of deep learning for industrial visual anomaly detection," *Artificial Intelligence Review*, vol. 58, June 2025. [Online]. Available: https://doi.org/10.1007/s10462-025-11287-7

[2] M. Ishtiaq and J.-U. Won, "Yolo-sifd: Yolo with sliced inference and fractal dimension analysis for improved fire and smoke detection," *Computers, Materials & Continua*, vol. 82, no. 3, pp. 5343–5361, 2025. [Online]. Available: http://www.techscience.com/cmc/v82n3/59939

[3] J.-H. Kim and G.-R. Kwon, "Unsupervised visual anomaly detection using self-supervised pre-trained transformer," *IEEE Access*, vol. 12, pp. 127 604–127 613, 2024.

[4] S. Yang, S. Liu, P. Shang, and H. Wang, "An overview of methods of industrial anomaly detection," in *2024 7th International Conference on Robotics, Control and Automation Engineering (RCAE)*. IEEE, 2024, pp. 603–607.

[5] J. Wang, K. Byeon, J. Song, A. Nguyen, S. Ahn, S. H. Lee, and J. T. Kwak, "Pathology-informed latent diffusion model for anomaly detection in lymph node metastasis," 2025. [Online]. Available: https://arxiv.org/abs/2508.15236

[6] Y. Liu, H. Wang, Z. Wang, X. Zhu, J. Liu, P. Sun, R. Tang, J. Du, V. C. Leung, and L. Song, "Crcl: Causal representation consistency learning for anomaly detection in surveillance videos," *IEEE Transactions on Image Processing*, 2025.

[7] D.-C. Hoang, P. X. Tan, A.-N. Nguyen, D.-T. Tran, V.-H. Duong, A.-T. Mai, D.-L. Pham, K.-T. Phan, M.-Q. Do, T. H. A. Duong, T.-M. Huynh, S.-A. Bui, D.-M. Nguyen, V.-A. Trinh, K.-D. Tran, and T.-U. Nguyen, "Unsupervised visual-to-geometric feature reconstruction for vision-based industrial anomaly detection," *IEEE Access*, vol. 13, pp. 3667–3682, 2025.

[8] X. Zhang, S. Li, X. Li, P. Huang, J. Shan, and T. Chen, " DeSTSeg: Segmentation Guided Denoising Student-Teacher for Anomaly Detection ," in *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. Los Alamitos, CA, USA: IEEE Computer Society, Jun. 2023, pp. 3914–3923. [Online]. Available: https://doi.ieeecomputersociety.org/10.1109/CVPR52729.2023.00381

[9] C. Li, G. Feng, Y. Li, R. Liu, Q. Miao, and L. Chang, "Difftad: Denoising diffusion probabilistic models for vehicle trajectory anomaly detection," *Knowledge-Based Systems*, vol. 286, p. 111387, 2024. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0950705124000224

[10] S. Akshay, N. L. Narasimhan, J. George, and V. N. Balasubramanian, "A unified latent schrodinger bridge diffusion model for unsupervised anomaly detection and localization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2025, pp. 25 528–25 538.

[11] M. Oquab, T. Darcet, T. Moutakanni, H. Vo, M. Szafraniec, V. Khalidov, P. Fernandez, D. Haziza, F. Massa, A. El-Nouby, M. Assran, N. Ballas, W. Galuba, R. Howes, P.-Y. Huang, S.-W. Li, I. Misra, M. Rabbat, V. Sharma, G. Synnaeve, H. Xu, H. Jegou, J. Mairal, P. Labatut, A. Joulin, and P. Bojanowski, "Dinov2: Learning robust visual features without supervision," *arXiv preprint arXiv:2304.07193*, Feb. 2024. [Online]. Available: https://doi.org/10.48550/arXiv.2304.07193

[12] P. Bergmann, K. Batzner, M. Fauser, D. Sattlegger, and C. Steger, "The mvtec anomaly detection dataset: A comprehensive real-world dataset for unsupervised anomaly detection," *International Journal of Computer Vision*, vol. 129, no. 4, pp. 1038–1059, Apr. 2021. [Online]. Available: https://doi.org/10.1007/s11263-020-01400-4

[13] 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*, 2023, pp. 20 402–20 411.

[14] S. Akcay, D. Ameln, A. Vaidya, B. Lakshmanan, N. Ahuja, and U. Genc, "Anomalib: A deep learning library for anomaly detection," in *2022 IEEE International Conference on Image Processing (ICIP)*, 2022, pp. 1706–1710.

[15] J. Jeong, Y. Zou, T. Kim, D. Zhang, A. Ravichandran, and O. Dabeer, "Winclip: Zero-/few-shot anomaly classification and segmentation," in *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023, pp. 19 606–19 616.

[16] H. He, J. Zhang, H. Chen, X. Chen, Z. Li, X. Chen, Y. Wang, C. Wang, and L. Xie, "Diad: A diffusion-based framework for multi-class anomaly detection," 2023.

[17] Z. Zhou, J. Wang, Z. Yu, Z. Wang, X. Liu, L. Qiu, and S. Zhang, "Featdae: Introducing features with denoising autoencoder for anomaly detection," *IEEE Transactions on Instrumentation and Measurement*, vol. 74, pp. 1–14, 2025.

[18] J. Jang, H. Lee, and Y. Lee, "Disentangled knowledge distillation for unified multi-class anomaly detection," in *2024 IEEE International Conference on Image Processing (ICIP)*, 2024, pp. 312–318.

[19] M. Ishtiaq and J. Won, "Glocal-had: Hybrid attention decoder for reconstruction-based vision anomaly detection framework," *IEEE Access*, vol. 13, pp. 208 681–208 695, 2025.