Industrial Anomaly Detection Focusing on Individual Anomalies

金　容民，王　永東，谷島　諒丞，ルイ笠原　純ユネス，永谷　圭司，
淺間　一，安　琪，山下　淳（東京大学）

Kim Yongmin, Yongdong Wang, Ryosuke Yajima, Jun Younès Louhi Kasahara,
Keiji Nagatani, Hajime Asama, Qi An, and Atsushi Yamashita (The University of Tokyo)

Abstract: In Industrial Anomaly Detection (IAD), The existing methods can generate anomaly scores, segmentation maps, and can interact with users about detetected anomalies. However, they cannot provide descriptions of the designated anomaly in an input image, making it challenging for users to interpret to improve manufacturing processes. To overcome these limitations, we propose a method that can output descriptions of the designated anomaly area. Specifically, we constructed a multi-modal IAD dataset and then enhanced the existing model. The evaluation results show that our method improves IAD accuracy by 52.37% compared to the existing VLM in our test scenario.

1. Introduction

Industrial Anomaly Detection (IAD) has made significant progress, including feature embedding-based methods[1), 2)] and reconstruction-based methods[1), 3), 4)] . These methods can generate an anomaly score and a segmentation map for each sample.

Recently, methods[5), 6)] that can interact with users about anomalies were proposed. They utilize Vision and Language Models (VLM), which have various abilities to offer descriptions of anomalies.

However, while they can capture the primary anomalies when multiple anomalies exist within the data, they struggle to detect less prominent anomalies. This issue stems from the fact that the CLIP[7)] is used in existing VLMs. CLIP[7)] is a model that can extract image features aligned with languages. It captures only the most salient features when attempting to recognize multiple features within an image, leading to less prominent features being ignored[8)]. Therefore, to effectively represent each anomaly, it is necessary to be able to extract features that are specific to the corresponding anomaly region.

Recently, Sun et al.[9)] has addressed this issue by utilizing masks in addition to CLIP[7)], allowing it to extract features corresponding to masked regions. This approach mitigated the problem of CLIP's inability to effectively extract features from specific areas and improved image recognition performance compared to the original CLIP.

However, the AlphaCLIP training datasets do not include industrial anomaly datasets, resulting in poor performance on industrial anomaly detection tasks. Moreover, because the training dataset of AlphaCLIP only includes data referring to objects themselves, AlphaCLIP lacks the ability to identify abnormal regions within objects, which is essential for IAD.

In this study, we aim to build a model with referring capabilities that can provide detailed descriptions for each anomaly in IAD. Specifically, we construct a multi-modal training IAD dataset including input images, masks, and the corresponding text description. We separated masks into each anomaly region because the masks are filled in the entire anomaly region. With our multi-modal IAD dataset, we enhance the AlphaCLIP referring capability and inject industrial anomaly knowledge into it.

To validate our method, we create a test multi-modal IAD dataset from MVTec[10)] with the same process in the training dataset. The evaluation of this test dataset shows that our model can provide the anomaly descriptions with higher accuracy than the original AlphaCLIP model even if there are multiple anomalies.

2. Related work

2.1 Zero-shot anomaly detection

Zero-Shot Anomaly Detection (ZSAD) aims to detect anomalies without training on anomaly data. Feature embedding-based methods[1), 2)] extract features from normal samples using pre-trained models, with Patchcore[2)] using core-set sub-sampling for efficiency and SimpleNet[1)] addressing domain biases through feature mapping. Reconstruction-based methods[11) ~14)] identify anomalies by reconstructing normal images and comparing them to originals pixel-by-pixel. While some approaches require separate models per class[1), 2), 11), 14)], others[15), 16)] use unified models for multi-class detection. Additionally, Hu et al.[17)] and Zhang, Xu, and Zhou[18)] use the Latent Diffusion Model (LDM)[19)] to synthesize anomaly data from limited normal samples, which enables models to train on synthesizing anomaly data in the ZASD setting and results in enhanced detecting capabilities.

Language-guided methods[3)] leverage pre-trained multi-modal models such as CLIP[7)], Region CLIP[20)], and Image-bind[21)] for visual-language integration. WinCLIP[3)] and April-GAN[22)] construct prompts for normal and abnormal samples, comparing pixel-wise vision-textual features to provide anomaly scores. AnoVL[23)] employs Test-Time Adaptation (TTA) to refine features and enhance anomaly localization. These models exhibit strong zero-shot or few-shot capabilities due to the generalization ability of multimodal models.

Recently, AnomalyCLIP[4)] and AdaCLIP[24)] showed high performance on ZSAD with the text prompt adaption. They made their models that can learn text prompts training on auxiliary data.

However, although they can detect anomalies with high accuracy, none of these methods can provide descriptions of anomalies. It makes users difficult to understand detected anomalies and identify anomaly reasons.

2.2 Vision and language models on IAD

Lately, methods leveraging VLMs have been proposed for explainable IAD. AnomalyGPT[5)] designs an LLM-based im-

SY0009/24/0000 - 0375

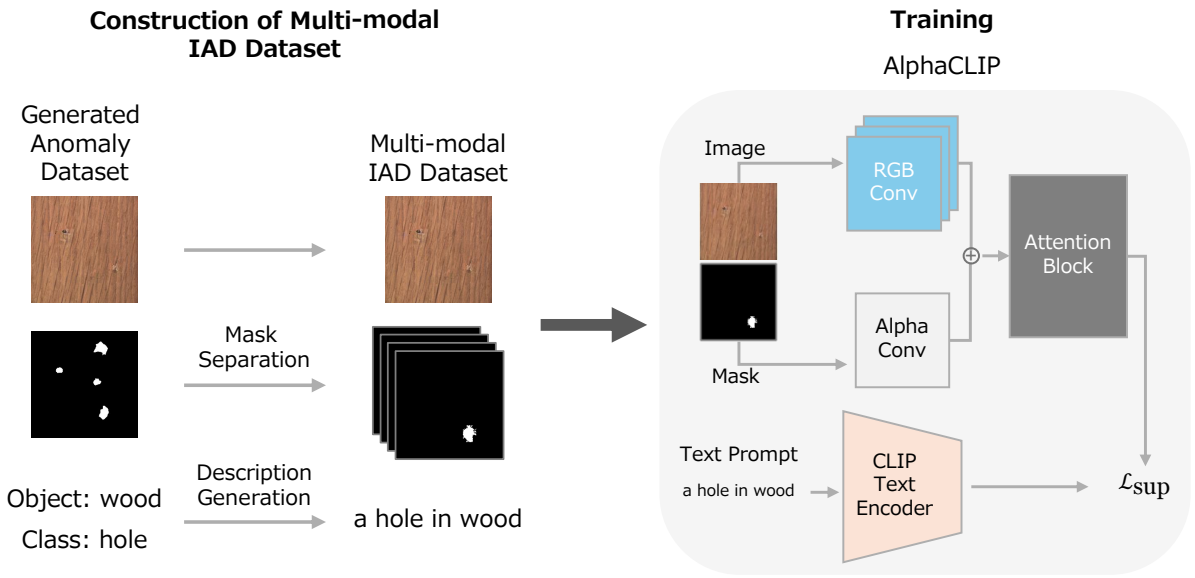

Fig. 1 Overview of Our approach. Left) we construct a multi-modal IAD dataset including images, masks, and text descriptions. We separate the existing anomaly masks so that they refer to each anomaly region, and make descriptions based on the object name and the class name. Right) we fine-tune the AlphaCLIP model on our Multi-modal IAD dataset to enhance its abilities in IAD.

age decoder to generate anomaly maps and uses prompt embedding to incorporate domain knowledge into large language models (LLMs). Myriad[6] further utilizes anomaly maps to identify anomalies in an input image from the existing IAD models that can output anomaly maps.

However, they struggle to provide descriptions of each anomaly region respectively. Moreover, they cannot offer descriptions on a designated region by users.

Recent VLMs[9), 25), 26)] have improved grounding capabilities by referencing objects or regions of interest using coordinates or specialized tokens instead of detailed textual descriptions. However, the application of these VLMs in IAD faces challenges due to the lack of domain-specific knowledge and different levels of referring capabilities.

In this study, to solve these limitations, we propose a method that can provide descriptions of a specific region designated by users. To do this end, we construct multi-modal IAD datasets and enhance AlphaCLIP abilities in IAD.

3. Method

We first explain how AlphaCLIP can extract features corresponding to an area in Sec. 3.1. Next, we describe the way that we create multi-modal IAD datasets where there are images, masks, and class-specific text description corresponding to the mask area in Sec. 3.2. Lastly, we explain the way to train AlphaCLIP on the multi-modal training IAD datasets in Sec.3.3.

3.1 Background: AlphaCLIP

To describe each anomaly in a specific area, we employ AlphaCLIP AE$(\cdot)$, which can extract image anomaly features from a designated area using a mask. AlphaCLIP enables a model to extract the image features for each designated region by inserting an additional convolution layer along with the existing RGB convolution layers (denoted as 'Alpha Conv' and 'RGB Conv' respectively in the right part of Fig 1). With this extra convolution layer for the alpha channel, AlphaCLIP can extract the image information $\mathbb{E}_{I,i}$ for each specified anomaly region from the input image $\mathbb{I}$ and the mask $\mathbb{M}'_i$ as follows:

$$\mathrm{AE}(I, \mathbb{M}'_i) = \mathbb{E}_{I_i}, \tag{1}$$

where $\mathbb{E}'_{I,i} \in \mathbb{R}^{D_{IE}}, 1 \leq i \leq N$ and with D_{IE} is the image embedding dimension of AlphaCLIP.

The text embedding is obtained using the text encoder TE$(\cdot)$ in AlphaCLIP. This encoder converts the textual description $\mathbb{T}_i$ into a corresponding embedding representation, expressed as:

$$\mathrm{TE}(\mathbb{T}_i) = \mathbb{E}_{T_i}, \tag{2}$$

where $\mathbb{E}_{T_i} \in \mathbb{R}^{D_{TE}}$, with D_{TE} being the text embedding dimension, which is the same as the image embedding dimension D_{IE}.

AlphaCLIP can output the description text T_o corresponding to the input image $\mathbb{I}$ and the mask $\mathbb{M}'_i$ with the following operation,

$$T_o = \arg\max_{T_j} \left(\mathbb{E}_{I_i} \cdot \mathbb{E}_{T_j} \right). \tag{3}$$

This operation selects the text description T_j with the highest similarity score as the output description T_o corresponding to the designated region.

3.2 Construction of multi-modal IAD datasets

In a ZASD setting, it is not allowed to train a model with anomaly data in existing datasets. Therefore, we used Anomaly-Diffusion[17)] to synthesize anomaly datasets from normal images. Anomaly-Diffusion is a segmentation dataset that contains anomaly images, masks, object names, and object classes. However, the existing masks cover the entire anomaly region. We separate the existing masks so that each mask indicates individual anomaly regions. Specifically, We separate the original mask $\mathbb{M}$ into N individual masks $\mathbb{M}'_i$ using DBSCAN[27)]. It can be formulated as follows:

Table 1 The number of each object in the constructed IAD multi-modal dataset. We create 'Train' and 'Validation' dataset from the synthesis MVTec dataset[17], and 'Test' from the MVTec test dataset[10].

Object	Train	Validation	Test
Bottle	1,318	574	67
Grid	3,092	1,337	124
Pill	3,320	1,437	165
Wood	7,861	3,425	156
Leather	1,937	735	95
Carpet	2,284	957	94
Zipper	4,450	1,972	176
Screw	1,224	520	127
Cable	3,733	1,581	125
Metal nut	1,926	843	118
Capsule	1,800	723	111
Hazelnut	1,791	815	91
Tile	1,928	835	86
Transistor	1,711	711	44
Toothbrush	428	200	49
Total	40,803	16,665	1,522

$$\mathrm{DBSCAN}(\mathbb{M}) = \mathbb{M}_s = (\mathbb{M}'_1, \mathbb{M}'_2, \ldots, \mathbb{M}'_N), \quad (4)$$

where $\mathbb{M}'_i \in \mathbb{R}^{H \times W}, 1 \leq i \leq N$.

DBSCAN[27] is a clustering algorithm that groups points that are closely packed together while marking points that lie alone in low-density regions as outliers.

Next, we create anomaly text descriptions for each mask. Specifically, we utilize object names and anomaly class names in existing datasets. Because the original class names were not easily readable, we utilized GPT-4o[28] to make them more comprehensible. Specifically, we made a more readable text description with the following prompt format:

```
Anomaly Image: <Image>
Object name: <Object name>
Anomaly class name: <Class name>
You are given an image along with
an object name and an anomaly class name.
Based on these inputs, generate a natural and concise description
that describes the anomaly occurring with the object.
Make sure the description clearly conveys the relationship
between the object and the anomaly.

Output:
```

In the text prompt, `<Image>`, `<Object name>` and `<Class name>` refer to an input image, the object name, and the class name respectively in the existing dataset.

We constructed multi-modal training and validation datasets including images, separated masks, and corresponding text prompts from the MVTec dataset[10] and Anomaly-Diffusion[17].

The left side of Fig 1 illustrates real data samples in our multi-modal IAD datasets from the synthesis MVTec dataset[17]. We create a test dataset from the original MVTec[10] dataset with the same process. Tables 1 and 2 indicate the overall dataset statics including the number of each object and the total number of masks in a single instance. It indicates that many instances contain multiple anomalies.

Table 2 The total mask number of each data in the constructed IAD multi-modal dataset.

Number of Masks	Train	Validation	Test
1	19,721	8,461	1,002
2	11,186	4,682	346
3	5,460	2,490	192
4+	2,436	1,032	88
Total	38,803	16,665	1,628

3.3 Training AlphaCLIP on multi-modal IAD datasets

To train AlphaCLIP, we employ the supervised contrastive loss[29], This loss function is effective when we fine-tune a CLIP model on small-scale training datasets where there are many positive and negative samples in a single training batch. It aligns the image features $\mathbb{E}_{I_i}$ and text features $\mathbb{E}_{T_i}$ by leveraging both positive and negative samples.

The supervised contrastive loss $\mathcal{L}$sup is defined as:

$$\mathcal{L}_{\text{sup}} = \frac{1}{N} \sum_{i=1}^{N} \left[-\frac{1}{|P(i)|} \sum_{p \in P(i)} \log \frac{\exp(\mathbb{E}_{I,i} \cdot \mathbb{E}_{T,p} / \tau)}{\sum_{a \in A(i)} \exp(\mathbb{E}_{I,i} \cdot \mathbb{E}_{T,a} / \tau)} \right], \quad (5)$$

where $P(i)$ represents the set of indices corresponding to positive samples for the i-th data point, $A(i)$ denotes the set of all samples excluding i itself, τ is a temperature parameter.

By minimizing this supervised contrastive loss on the multi-modal IAD datasets, the model can learn to effectively align the image and text embedding of a designated area with a mask. It ensures that features corresponding to the same anomaly description are brought closer in the shared embedding space, while features from different anomalies are pushed apart.

The right illustration in Fig 1 shows fine-tuning AlphaCLIP on the multi-modal IAD dataset. This allows AlphaCLIP can output the text description of a designated anomaly area in industrial images.

4. Experiments

4.1 Experiment setting

We fine-tuned the original AlphaCLIP's ViT-L/14@336px model on the MVTec multi-modal IAD training dataset described in Sec 3.2. For the fine-tuning process, it is crucial to ensure that AlphaCLIP retains its original capabilities while learning from the IAD training dataset. LoRA[30] demon-

Table 3 Hyperparameter values used for training AlphaCLIP with LoRA.

Hyperparameter	Value
LoRA rank (r)	64
LoRA α	64
Dropout rate	0.1%
Learning rate (Alpha Convolution)	1×10^{-4}
Learning rate (RGB Convolution)	1×10^{-6}
Weight decay	1×10^{-4}

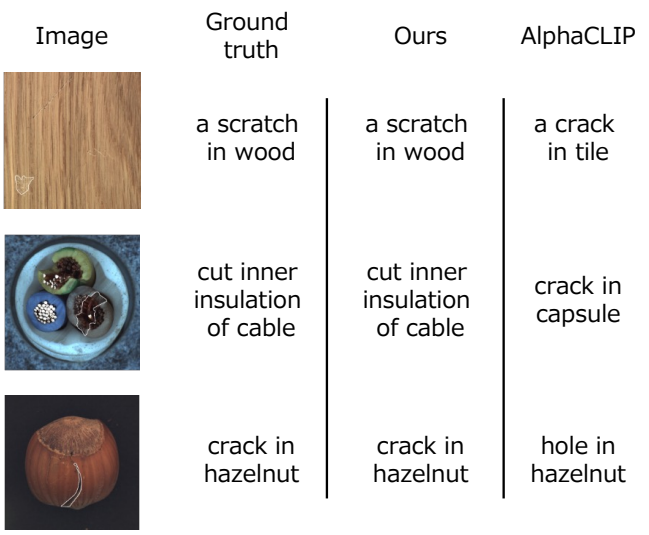

Fig. 2 Qualitative evaluation between the original AlphaCLIP and our model on the multi-modal MVTec test dataset. The white solid lines in each image refer to the areas used in the masks.

strated that by training only a small subset of parameters, it can achieve performance similar to full-parameter fine-tuning. Moreover, because it does not train the entire parameter, it can help prevent over-fitting. For these reasons, we determined to use LoRA for fine-tuning.

Table 3 shows our hyperparameter's values used in the training. For LoRA, we set $r = 64$ and $\alpha = 64$, and applied a dropout rate of 0.1%. During training, we followed the original AlphaCLIP training configuration, setting the learning rate for the Alpha Convolution to $1e^{-4}$ and for the RGB Convolution to $1 \times e^{-6}$ individually. We used the AdamW optimizer with $\beta = (0.9, 0.999)$ and a weight decay of $1 \times e^{-4}$.

For data augmentation, we applied random horizontal flips and random vertical flips. The total batch size was set to 64 for training, and a batch size of 256 was used for evaluation. All experiments were conducted on two GTX 3080 GPUs.

4.2 Experiment results

Our model can detect an anomaly with a descriptive prompt on a designated region. Figure 2 illustrates the accuracy of the multi-modal MVTec test dataset per object in AlphaCLIP and our fine-tuned model. The original AlphaCLIP model failed to detect object anomalies and recognize the kinds of objects due to the lack of knowledge of the MVTec dataset. On the other hand, our model can identify anomalies given a mask and an image even if multiple anomalies exist in an image.

Table 4 Accuracy comparison between AlphaCLIP and our model for each mask number. The 'Total accuracy' row shows the average performance on the entire dataset.

Mask Number	Accuracy (%) ↑	
	AlphaCLIP	Ours
1	17.47	74.15 (+56.68)
2	14.45	66.18 (+51.73)
3	32.29	68.23 (+35.94)
4	16.67	50.00 (+33.33)
5	6.67	46.67 (+40.00)
6	16.67	61.11 (+44.44)
7	0.00	71.43 (+71.43)
Total accuracy	18.37	70.64 (+52.27)

Table 5 Object-wise accuracy comparison between AlphaCLIP and our model, with the gap shown next to our model's values.

Object	Accuracy (%) ↑	
	AlphaCLIP	Ours
Bottle	0.00	85.07 (+85.07)
Grid	25.81	60.48 (+34.67)
Pill	9.09	60.61 (+51.52)
Wood	41.67	69.87 (+28.20)
Leather	24.21	56.84 (+32.63)
Carpet	0.00	64.89 (+64.89)
Zipper	15.34	71.02 (+55.68)
Screw	18.90	55.91 (+37.01)
Cable	0.00	67.20 (+67.20)
Metal nut	16.10	79.66 (+63.56)
Capsule	18.92	63.96 (+45.04)
Hazelnut	18.68	87.91 (+69.23)
Tile	13.95	94.19 (+80.24)
Transistor	36.36	88.64 (+52.28)
Toothbrush	57.14	100.00 (+42.86)
Total	18.37	70.64 (+52.27)

Table 5 indicates the quantitative evaluation per object. The original AlphaCLIP model totally failed to detect anomalies on 'Bottle', 'Cable', and 'Carpet' objects, the performances on others are low. On the other hand, Our fine-tuned model can detect anomalies on the objects that AlphaCLIP failed. It can identify anomalies in 'Toothbrush' the most. Our model struggled to detect anomalies on 'Leather', and 'Screw' because their anomalies are small and hard to be visually notified. It performed with much higher accuracy (+50.27%) than the original AlphaCLIP model. This demonstrated that our model can identify industrial anomalies with detailed descriptions.

Our model can focus on individual anomalies in multiple anomalies. Table 4 presents the accuracy of the multi-modal MVTec test dataset per the total number of the anomalies in data. AlphaCLIP shows lower accuracy on those data where more anomalies exist, but our model can detect anomalies. This indicates that our model can reliably detect anomalies even in data with numerous anomalies.

5. Conclusion

In this work, we proposed a method that can provide descriptions of a designated anomaly region. To do this end, we constructed the multi-modal IAD dataset in the MVTec dataset[10] and fine-tuned the AlphaCLIP model on it. In creating the dataset, we ensured that the masks indicated each

individual anomaly region, and we used object names and anomaly class names to create easy-to-understand descriptions for each anomaly region. We demonstrated that our model can provide descriptions about a designated anomaly area, and can consistently perform even in data where there are many anomalies.

In this study, we only conduct experiments in MVTec dataset[10], due to computational constraints. Adapting our approach to anomaly detection in other domains, such as the medical domain[31), 32)] , is a future work.

References

[1] Z. Liu et al. SimpleNet: A Simple Network for Image Anomaly Detection and Localization. *CVPR*. (2023), pp. 20402–20411.

[2] K. Roth et al. Towards Total Recall in Industrial Anomaly Detection. *CVPR*. (2022), pp. 14318–14328.

[3] J. Jeong et al. WinCLIP: Zero-/Few-Shot Anomaly Classification and Segmentation. *CVPR*. Vancouver, BC, Canada: IEEE, (2023), pp. 19606–19616.

[4] Q. Zhou et al. AnomalyCLIP: Object-agnostic Prompt Learning for Zero-shot Anomaly Detection. (2024). URL: https://arxiv.org/abs/2310.18961.

[5] Z. Gu et al. AnomalyGPT: Detecting Industrial Anomalies Using Large Vision-Language Models. *AAAI*. (2023).

[6] Y. Li et al. Myriad: Large Multimodal Model by Applying Vision Experts for Industrial Anomaly Detection. (2023). URL: http://arxiv.org/abs/2310.18961.

[7] A. Radford et al. Learning transferable visual models from natural language supervision. *ICML*. (2021), pp. 8748–8763.

[8] T. Chen, C. Luo, and L. Li. Intriguing Properties of Contrastive Losses. NIPS 34, (2021).

[9] Z. Sun et al. Alpha-CLIP: A CLIP Model Focusing on Wherever You Want. CVPR, pp. 13019–13029, (2023).

[10] P. Bergmann et al. The MVTec Anomaly Detection Dataset: A Comprehensive Real-World Dataset for Unsupervised Anomaly Detection. IJCV 129.4, pp. 1038–1059, (2021).

[11] H. Yin. LafitE: Latent Diffusion Model with Feature Editing for Unsupervised Multi-class Anomaly Detection. (2023). URL: http://arxiv.org/abs/2307.08059.

[12] F. Lu et al. Removing Anomalies as Noises for Industrial Defect Localization. *ICCV*. (2023), pp. 16166–16175.

[13] H. Zhang et al. DiffusionAD: Norm-guided One-step Denoising Diffusion for Anomaly Detection. (2023). URL: https://arxiv.org/abs/2310.18961.

[14] X. Zhang et al. Unsupervised Surface Anomaly Detection with Diffusion Probabilistic Model. *ICCV*. (2023), pp. 6759–6768.

[15] X. Yao et al. Focus the Discrepancy: Intra- and Inter-Correlation Learning for Image Anomaly Detection. *ICCV*. IEEE, (2023), pp. 6780–6790.

[16] Z. You et al. A Unified Model for Multi-class Anomaly Detection. *NIPS*. Vol. 35. (2022), pp. 4571–4584.

[17] T. Hu et al. AnomalyDiffusion: Few-Shot Anomaly Image Generation with Diffusion Model. *AAAI*. (2023).

[18] X. Zhang, M. Xu, and X. Zhou. RealNet: A Feature Selection Network with Realistic Synthetic Anomaly for Anomaly Detection. (2024).

[19] R. Rombach et al. High-Resolution Image Synthesis with Latent Diffusion Models. CVPR, pp. 10674–10685, (2021).

[20] Y. Zhong et al. RegionCLIP: Region-Based Language-Image Pretraining. *CVPR*. (2022), pp. 16793–16803.

[21] R. Girdhar et al. ImageBind: One Embedding Space To Bind Them All. *CVPR*. (2023), pp. 15180–15190.

[22] X. Chen, Y. Han, and J. Zhang. APRIL-GAN: A Zero-/Few-Shot Anomaly Classification and Segmentation Method for CVPR 2023 VAND Workshop Challenge Tracks 1&2: 1st Place on Zero-shot AD and 4th Place on Few-shot AD. (2023).

[23] H. Deng et al. Bootstrap Fine-Grained Vision-Language Alignment for Unified Zero-Shot Anomaly Localization. (2024). URL: https://arxiv.org/abs/2308.15939.

[24] Y. Cao et al. AdaCLIP: Adapting CLIP with Hybrid Learnable Prompts for Zero-Shot Anomaly Detection. (2024). URL: http://arxiv.org/abs/2407.15795.

[25] Z. Peng et al. Kosmos-2: Grounding Multimodal Large Language Models to the World. (2023). URL: https://arxiv.org/abs/2306.14824.

[26] H. You et al. Ferret: Refer and Ground Anything Anywhere at Any Granularity. (2023). URL: https://arxiv.org/abs/2310.07704.

[27] M. Ester et al. A density-based algorithm for discovering clusters in large spatial databases with noise. *kdd*. Vol. 96. 34. (1996), pp. 226–231.

[28] J. Achiam et al. Gpt-4 technical report. arXiv, (2023). URL: https://arxiv.org/abs/2303.08774.

[29] P. Khosla et al. Supervised Contrastive Learning. (2021). URL: https://arxiv.org/abs/2303.08774.

[30] E. J. Hu et al. LoRA: Low-Rank Adaptation of Large Language Models. *ICLR*. (2021).

[31] N. C. F. Codella et al. Skin Lesion Analysis toward Melanoma Detection: A Challenge at the 2017 International Symposium on Biomedical Imaging, Hosted by the International Skin Imaging Collaboration. *ISBI*. (2018), pp. 168–172.

[32] J. Bernal et al. WM-DOVA Maps for Accurate Polyp Highlighting in Colonoscopy: Validation vs. Saliency Maps from Physicians. CMIG 43, pp. 99–111, (2015).