AnomalyGPT: Detecting Industrial Anomalies Using Large Vision-Language Models

Zhaopeng Gu[1,2*], Bingke Zhu[1,3*], Guibo Zhu[1,2†], Yingying Chen[1,3†], Ming Tang[1,2], Jinqiao Wang[1,2,3]

[1]Foundation Model Research Center, Institute of Automation, Chinese Academy of Sciences, Beijing, China
[2]School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing, China
[3]Objecteye Inc., Beijing, China
guzhaopeng2023@ia.ac.cn, {bingke.zhu, gbzhu, yingying.chen, tangm, jqwang}@nlpr.ia.ac.cn

Abstract

Large Vision-Language Models (LVLMs) such as MiniGPT-4 and LLaVA have demonstrated the capability of understanding images and achieved remarkable performance in various visual tasks. Despite their strong abilities in recognizing common objects due to extensive training datasets, they lack specific domain knowledge and have a weaker understanding of localized details within objects, which hinders their effectiveness in the Industrial Anomaly Detection (IAD) task. On the other hand, most existing IAD methods only provide anomaly scores and necessitate the manual setting of thresholds to distinguish between normal and abnormal samples, which restricts their practical implementation. In this paper, we explore the utilization of LVLM to address the IAD problem and propose AnomalyGPT, a novel IAD approach based on LVLM. We generate training data by simulating anomalous images and producing corresponding textual descriptions for each image. We also employ an image decoder to provide fine-grained semantic and design a prompt learner to fine-tune the LVLM using prompt embeddings. Our AnomalyGPT eliminates the need for manual threshold adjustments, thus directly assesses the presence and locations of anomalies. Additionally, AnomalyGPT supports multi-turn dialogues and exhibits impressive few-shot in-context learning capabilities. With only one normal shot, AnomalyGPT achieves the state-of-the-art performance with an accuracy of 86.1%, an image-level AUC of 94.1%, and a pixel-level AUC of 95.3% on the MVTec-AD dataset.

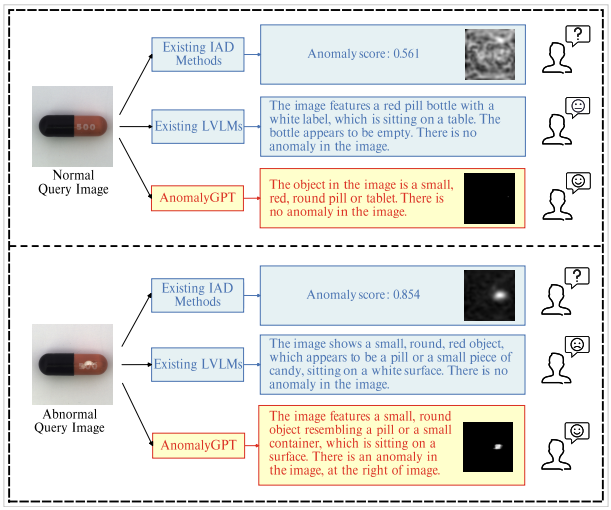

Figure 1: Comparison between our AnomalyGPT, existing IAD methods and existing LVLMs. Existing IAD methods can only provide anomaly scores and need manually threshold setting, while existing LVLMs cannot detect anomalies in the image. AnomalyGPT can not only provide information about the image but also indicate the presence and location of anomaly.

Introduction

Large Language Models (LLMs) like GPT-3.5 (Ouyang et al. 2022) and LLaMA (Touvron et al. 2023) have demonstrated remarkable performance on a range of Natural Language Processing (NLP) tasks. More recently, novel methods including MiniGPT-4 (Zhu et al. 2023), BLIP-2 (Li et al. 2023), and PandaGPT (Su et al. 2023) have further extended the ability of LLMs into visual processing by aligning visual features with text features, bringing a significant revolution in the domain of Artificial General Intelligence (AGI). While LVLMs are pre-trained on amounts of data sourced from the Internet, their domain-specific knowledge is relatively limited and they lack sensitivity to local details within objects, which restricts their potentiality in IAD task.

IAD task aims to detect and localize anomalies in industrial product images. Due to the rarity and unpredictability of real-world samples, models are required to be trained only on normal samples and distinguish anomalous samples that deviate from normal samples. Current IAD methods (Jeong et al. 2023; Huang et al. 2022; You et al. 2022) typically only provide anomaly scores for test samples and require manually specification of thresholds to distinguish between normal and anomalous instances for each class of items, which is not suitable for real production environments.

As illustrated in Figure 1 and Table 1, neither existing IAD methods nor LVLMs can address IAD problem well, so we introduce AnomalyGPT, a novel IAD approach based

*These authors contributed equally.
†Corresponding author.

Methods	Few-shot learning	Anomaly score	Anomaly localization	Anomaly judgement	Multi-turn dialogue
Traditional IAD methods		✓	✓		
Few-shot IAD methods	✓	✓	✓		
LVLMs	✓				✓
AnomalyGPT (ours)	✓	✓	✓	✓	✓

Table 1: Comparison between our AnomalyGPT and existing methods across various functionalities. The "Traditional IAD methods" in the table refers to "one-class-one-model" methods such as PatchCore (Roth et al. 2022), InTra (Pirnay and Chai 2022), and PyramidFlow (Lei et al. 2023). "Few-shot IAD methods" refers to methods that can perform few-shot learning like RegAD (Huang et al. 2022), Graphcore (Xie et al. 2023), and WinCLIP (Wang et al. 2023). "LVLMs" represents general large vision-language models like MiniGPT-4 (Zhu et al. 2023), LLaVA (Liu et al. 2023), and PandaGPT (Su et al. 2023). "Anomaly score" in the table represents just providing scores for anomaly detection, while "Anomaly judgement" indicates directly assessing the presence of anomaly.

on LVLM. AnomalyGPT can detect the presence and location of anomalies without the need for manual threshold settings. Moreover, our method can provide information about the image and allows for interactive engagement, enabling users to ask follow-up questions based on their needs and the provided answers. AnomalyGPT can also perform in-context learning with a small number of normal samples, enabling swift adaptation to previously unseen objects.

Specifically, we focus on fine-tuning the LVLM using synthesized anomalous visual-textual data, integrating IAD knowledge into the model. However, direct training with IAD data presents numerous challenges. The first is data scarcity. Methods like LLaVA (Liu et al. 2023) and PandaGPT (Su et al. 2023) are pre-trained on 160k images with corresponding multi-turn dialogues. However, existing IAD datasets (Bergmann et al. 2019; Zou et al. 2022) contain only a few thousand samples, rendering direct fine-tuning easy to overfitting and catastrophic forgetting. To address this, we use prompt embeddings to fine-tune the LVLM instead of parameter fine-tuning. Additional prompt embeddings are added after image inputs, introducing supplementary IAD knowledge into the LVLM. The second challenge relates to fine-grained semantic. We propose a lightweight, visual-textual feature-matching-based decoder to generate pixel-level anomaly localization results. The decoder's outputs are introduced to the LVLM along with the original test images through prompt embeddings, which allows the LVLM to utilize both the raw image and the decoder's outputs to make anomaly determinations, improving the accuracy of its judgments.

Experimentally, we conduct extensive experiments on the MVTec-AD (Bergmann et al. 2019) and VisA (Zou et al. 2022) datasets. With unsupervised training on the MVTec-AD dataset, we achieve an accuracy of 93.3%, an image-level AUC of 97.4%, and a pixel-level AUC of 93.1%. When one-shot transferred to the VisA dataset, we achieve an accuracy of 77.4%, an image-level AUC of 87.4%, and a pixel-level AUC of 96.2%. Conversely, after unsupervised training on the VisA dataset, one-shot transferred to the MVTec-AD dataset result in an accuracy of 86.1%, an image-level AUC of 94.1%, and a pixel-level AUC of 95.3%.

Our contributions are summarized as follows:

- We present the pioneering utilization of LVLM for addressing IAD task. Our method not only detects and locates anomaly without manually threshold adjustments but also supports multi-round dialogues. To the best of our knowledge, we are the first to successfully apply LVLM to the domain of industrial anomaly detection.
- The lightweight, visual-textual feature-matching-based decoder in our work addresses the limitation of the LLM's weaker discernment of fine-grained semantic and alleviates the constraint of LLM's restricted ability to solely generate text outputs.
- We employ prompt embeddings for fine-tuning and train our model concurrently with the data utilized during LVLM pre-training, thus preserving the LVLM's inherent capabilities and enabling multi-turn dialogues.
- Our method retains robust transferability and is capable of engaging in in-context few-shot learning on new datasets, yielding outstanding performance.

Related Work

Industrial Anomaly Detection Existing IAD methods can be categorized into reconstruction-based and feature embedding-based approaches. Reconstruction-based methods primarily aim to reconstruct anomalous samples to their corresponding normal counterparts and detect anomalies by calculating the reconstruction error. RIAD (Zavrtanik, Kristan, and Skočaj 2021), SCADN (Yan et al. 2021), InTra (Pirnay and Chai 2022) and AnoDDPM (Wyatt et al. 2022) employ different reconstruction network architectures, ranging from autoencoder and Generative Adversarial Network (GAN) to Transformer and diffusion model.

Feature embedding-based methods focus on modeling the feature embeddings of normal samples. Approaches such as PatchSVDD (Yi and Yoon 2020) aim to find a hypersphere that tightly encapsulates normal samples. Cflow-AD (Gudovskiy, Ishizaka, and Kozuka 2022) and PyramidFlow (Lei et al. 2023) use normalizing flows to project normal samples onto a Gaussian distribution. PatchCore (Roth et al. 2022) and CFA (Lee, Lee, and Song 2022) establish a memory bank of patch embeddings from normal samples and detect anomalies by measuring the distance between a test sample embedding and its nearest normal embedding.

These methods typically follow the "one-class-one-model" learning paradigm, requiring plentiful normal sam-

ples for each object class to learn its distribution, making them impractical for novel object categories and less suitable for dynamic production environments. In contrast, our method facilitates in-context learning for novel object categories, enabling inference with only few normal samples.

Zero-/Few-Shot Industrial Anomaly Detection Recent efforts have focused on methods utilizing minimal normal samples to accomplish IAD task. PatchCore (Roth et al. 2022) constructs a memory bank using only a few normal samples, resulting in a noticeable performance decline. RegAD (Huang et al. 2022) trained an image registration network to align test images with normal samples, followed by similarity computation for corresponding patches. WinCLIP (Jeong et al. 2023) leveraged CLIP (Radford et al. 2021) to compute similarity between images and textual descriptions representing normal and anomalous semantics, distinguishing anomalies based on their relative scores.

However, these methods can only provide anomaly scores for test samples during inference. To distinguish normal samples from anomalous ones, it's necessary to experimentally determine the optimal threshold on a test set, which contradicts the original intent of IAD task that only utilize normal data. For instance, while PatchCore (Roth et al. 2022) achieves an image-level AUC of 99.3% on MVTec-AD in unsupervised setting, its accuracy drops to 79.76% when using a unified threshold for inference. Our method, in contrast, enables the LVLM to directly assess test samples for the presence of anomalies and pinpoint their locations, demonstrating enhanced practicality.

Large Vision-Language Models LLMs, traditionally successful in NLP, are now explored for visual tasks. BLIP-2 (Li et al. 2023) leverages Q-Former to input visual features from Vision Transformer (Dosovitskiy et al. 2020) into the Flan-T5 (Chung et al. 2022) model. MiniGPT-4 (Zhu et al. 2023) connects the image segment of BLIP-2 and the Vicuna (Chiang et al. 2023) model with a linear layer, performing a two-stage fine-tuning process using extensive image-text data. PandaGPT (Su et al. 2023) establishes a connection between ImageBind (Girdhar et al. 2023) and the Vicuna (Chiang et al. 2023) model via a linear layer, allowing for multi-modal input. These approaches showcase the potential of LLM-based polymathic models.

However, as mentioned earlier, these models are trained on general data and lack domain-specific expertise. In this paper, through the utilization of simulated anomaly data, image decoder and prompt embeddings, AnomalyGPT is introduced as an novel approach that achieves IAD task without the need for manually specified thresholds, while also enabling few-shot in-context learning. Table 1 illustrates a comparison between AnomalyGPT and existing methods across various functionalities.

Method

AnomalyGPT is a novel conversational IAD vision-language model, primarily designed for detecting anomalies in images of industrial artifacts and pinpointing their positions. We leverage a pre-trained image encoder and a LLM to align IAD images and their corresponding textual descriptions via simulated anomaly data. We introduce a decoder module and a prompt learner module to enhance IAD performance and achieve pixel-level localization output. Employing prompt tuning and alternate training with pre-training data preserves the LLM's transferability and prevents catastrophic forgetting. Our method exhibits robust few-shot transfer capability, enabling anomaly detection and localization for previously unseen items with merely one normal sample provided.

Model Architecture

Figure 2 illustrates the comprehensive architecture of AnomalyGPT. Given a query image $x \in \mathbb{R}^{H\times W\times C}$, the final features $F_{img} \in \mathbb{R}^{C_1}$ extracted by the image encoder are passed through the linear layer to obtain the image embedding $E_{img} \in \mathbb{R}^{C_{emb}}$, which is then fed into the LLM. In unsupervised setting, the patch-level features extracted by intermediate layers of image encoder are fed into the decoder together with text features to generate pixel-level anomaly localization results. In few-shot setting, the patch-level features from normal samples are stored in memory banks and the localization result can be obtained by calculating the distance between query patches and their most similar counterparts in the memory bank. The localization results is subsequently transformed into prompt embeddings through the prompt learner, serving as a part of LLM input. The LLM leverages image input, prompt embeddings, and user-provided textual input to detect anomalies and identify their locations, thus generating responses for the user.

Decoder and Prompt Learner

Decoder To achieve pixel-level anomaly localization, we employ a lightweight feature-matching-based image decoder that supports both unsupervised IAD and few-shot IAD. The design of the decoder is primarily inspired by PatchCore (Roth et al. 2022), WinCLIP (Jeong et al. 2023), and APRIL-GAN (Chen, Han, and Zhang 2023).

As illustrated in the upper part of Figure 2, we partition the image encoder into 4 stages and obtain the intermediate patch-level features extracted by every stage $F^i_{patch} \in \mathbb{R}^{H_i\times W_i\times C_i}$, where i indicates the i-th stage. Following the idea from WinCLIP (Jeong et al. 2023), a natural approach is to compute the similarity between F^i_{patch} and the text features $F_{text} \in \mathbb{R}^{2\times C_{text}}$ respectively representing normality and abnormality, such as *A photo of a normal bottle* and *A photo of an abnormal capsule*. However, since these intermediate features have not undergone the final image-text alignment, they cannot be directly compared with text features. To address this, we introduce additional linear layers to project these intermediate features to $\tilde{F}^i_{patch} \in \mathbb{R}^{H_i\times W_i\times C_{text}}$, and align them with text features representing normal and abnormal semantics. The localization result $M \in \mathbb{R}^{H\times W}$ can be obtained by Eq. (1):

$$M = Upsample\left(\sum_{i=1}^{4} softmax(\tilde{F}^i_{patch}F^T_{text})\right). \quad (1)$$

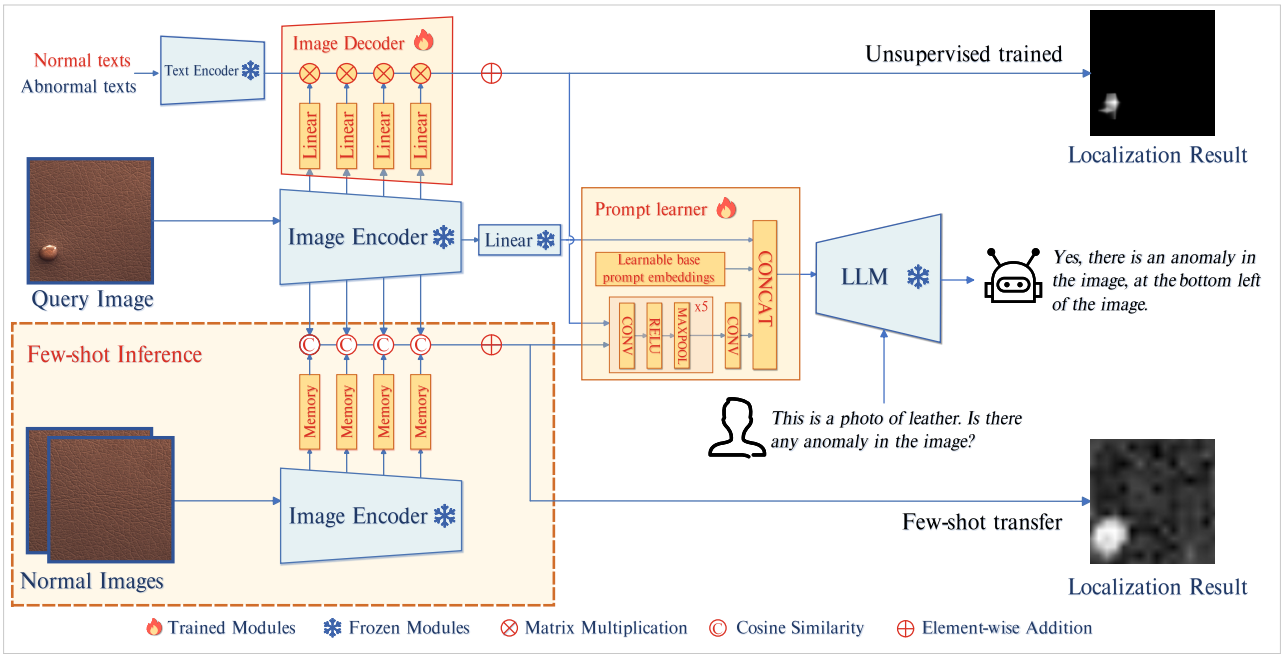

Figure 2: The architecture of AnomalyGPT. The query image is passed to the frozen image encoder and the patch-level features extracted from intermediate layers are fed into image decoder to compute their similarity with normal and abnormal texts to obtain localization result. The final features extracted by the image encoder are fed to a linear layer and then passed to the prompt learner along with the localization result. The prompt learner converts them into prompt embeddings suitable for input into the LLM together with user text inputs. In few-shot setting, the patch-level features from normal samples are stored in memory banks and the localization result can be obtained by calculating the distance between query patches and their most similar counterparts in the memory bank.

For few-shot IAD, as illustrated in the lower part of Figure 2, we utilize the same image encoder to extract intermediate patch-level features from normal samples and store them in memory banks $B^i \in \mathbb{R}^{N \times C_i}$, where i indicates the i-th stage. For patch-level features $F^i_{patch} \in \mathbb{R}^{H_i \times W_i \times C_i}$, we calculate the distance between each patch and its most similar counterpart in the memory bank, and the localization result $M \in \mathbb{R}^{H \times W}$ can be obtained by Eq. (2):

$$M = Upsample\left(\sum_{i=1}^{4}\left(1 - max(F^i_{patch} \cdot {B^i}^T)\right)\right). \quad (2)$$

Prompt Learner To leverage fine-grained semantic from images and maintain semantic consistency between LLM and decoder outputs, we introduce a prompt learner that transforms the localization result into prompt embeddings. Additionally, learnable base prompt embeddings, unrelated to decoder outputs, are incorporated into the prompt learner to provide extra information for the IAD task. Finally, these embeddings, along with the original image information, are fed into the LLM.

As illustrated in Figure 2, the prompt learner consists of the learnable base prompt embeddings $E_{base} \in \mathbb{R}^{n_1 \times C_{emb}}$ and a convolutional neural network. The network converts the localization result $M \in R^{H \times W}$ into n_2 prompt embeddings $E_{dec} \in \mathbb{R}^{n_2 \times C_{emb}}$. E_{base} and E_{dec} form a set of $n_1 + n_2$ prompt embeddings $E_{prompt} \in \mathbb{R}^{(n_1+n_2) \times C_{emb}}$ that are combined with the image embedding into the LLM.

Data for Image-Text Alignment

Anomaly Simulation We primarily adopt the approach proposed by NSA (Schlüter et al. 2022) to simulate anomalous data. The NSA (Schlüter et al. 2022) method builds upon the Cut-paste (Li et al. 2021) technique by incorporating the Poisson image editing (Pérez, Gangnet, and Blake 2003) method to alleviate the discontinuity introduced by pasting image segments. Cut-paste (Li et al. 2021) is a common technique in IAD domain for generating simulated anomaly images. This method involves randomly cropping a block region from an image and then pasting it onto a random location in another image, thus creating a simulated anomalous portion. Simulated anomaly samples can significantly enhance the performance of IAD models, but this procedure often results in noticeable discontinuities, as illustrated in Figure 3. The Poisson editing method (Pérez, Gangnet, and Blake 2003) has been developed to seamlessly clone an object from one image into another image by solving the Poisson partial differential equations.

Question and Answer Content To conduct prompt tuning on the LVLM, we generate corresponding textual queries based on the simulated anomalous images. Specifically, each query consists of two components. The first part involves a

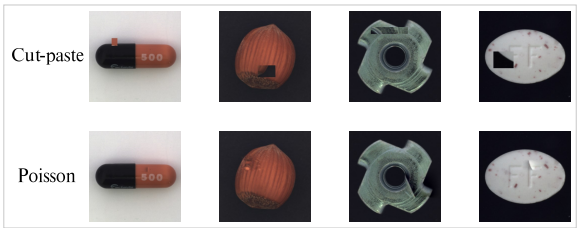

Figure 3: Illustration of the comparison between cut-paste and poisson image editing. The results of cut-paste exhibit evident discontinuities and the results of poisson image editing are more natural.

description of the input image, providing information about the objects present in the image and their expected attributes, such as *This is a photo of leather, which should be brown and without any damage, flaw, defect, scratch, hole or broken part.* The second part queries the presence of anomalies within the object, namely *Is there any anomaly in the image?* The LVLM firstly responds to whether anomalies are present. If anomalies are detected, the model continues to specify the number and location of the anomalous areas, such as *Yes, there is an anomaly in the image, at the bottom left of the image.* or *No, there are no anomalies in the image.* We divide the image into a grid of 3×3 distinct regions to facilitate the LVLM in verbally indicating the positions of anomalies, as shown in Figure 4. The descriptive content about the image furnishes the LVLM with foundational knowledge of the input image, aiding in the model's better comprehension of the image contents. However, during practical applications, users may opt to omit this descriptive input, and the model is still capable of performing IAD task based solely on the provided image input.

Prompts fed to the LLM typically follow the format:

Human: <Img>E_{img}*</Img>*E_{prompt}*[Image Description]Is there any anomaly in the image?###Assistant:*

$E_{img} \in \mathbb{R}^{C_{emb}}$ represents the image embedding being processed through the image encoder and linear layer, $E_{prompt} \in \mathbb{R}^{(n_1+n_2)\times C_{emb}}$ refers to the prompt embeddings generated by the prompt learner, and *[Image Description]* corresponds to the textual description of the image.

Loss Functions

To train the decoder and prompt learner, we primarily employed three loss functions: cross-entropy loss, focal loss (Lin et al. 2017), and dice loss (Milletari, Navab, and Ahmadi 2016). The latter two are primarily utilized to enhance the pixel-level localization accuracy of the decoder.

Cross-Entropy Loss Cross-entropy loss is commonly employed for training language models, which quantifies the disparity between the text sequence generated by the model and the target text sequence. The formula is as follows:

$$L_{ce} = -\sum_{i=1}^{n} y_i log(p_i), \tag{3}$$

where n is the number of tokens, y_i is the true label for token i and p_i is the predicted probability for token i.

Figure 4: Illustration of the 3×3 grid of image, which is used to let LLM verbally indicate the abnormal position.

Focal Loss Focal loss (Lin et al. 2017) is commonly used in object detection and semantic segmentation to address the issue of class imbalance, which introduces an adjustable parameter γ to modify the weight distribution of cross-entropy loss, emphasizing samples that are difficult to classify. In IAD task, where most regions in anomaly images are still normal, employing focal loss can mitigate the problem of class imbalance. Focal loss can be calculated by Eq. (4):

$$L_{focal} = -\frac{1}{n}\sum_{i=1}^{n}(1-p_i)^{\gamma} log(p_i), \tag{4}$$

where $n = H \times W$ represents the total number of pixels, p_i is the predicted probability of the positive classes and γ is a tunable parameter for adjusting the weight of hard-to-classify samples. In our implementation, we set γ to 2.

Dice Loss Dice loss (Milletari, Navab, and Ahmadi 2016) is a commonly employed loss function in semantic segmentation tasks. It is based on the dice coefficient and can be calculated by Eq. (5):

$$L_{dice} = -\frac{\sum_{i=1}^{n} y_i \hat{y}_i}{\sum_{i=1}^{n} y_i^2 + \sum_{i=1}^{n} \hat{y}_i^2}, \tag{5}$$

where $n = H \times W$, y_i is the output of decoder and $\hat{y}_i$ is the ground truth value.

Finally, the overall loss function is defined as:

$$L = \alpha L_{ce} + \beta L_{focal} + \delta L_{dice}, \tag{6}$$

where α, β, δ are coefficients to balance the three loss functions, which are set to 1 by default in our experiments.

Experiments

Datasets We conduct experiments primarily on the MVTec-AD (Bergmann et al. 2019) and VisA (Zou et al. 2022) datasets. The MVTec-AD dataset comprises 3629 training images and 1725 testing images across 15 different categories, making it one of the most popular datasets for IAD. The training images only consist of normal images, while the testing images contain both normal and anomalous images. The image resolutions vary from 700×700 to 1024×1024. VisA, a newly introduced IAD dataset, contains 9621 normal images and 1200 anomalous images across 12 categories, with resolutions approximately around 1500×1000. Consistent with previous IAD methods, we only use the normal data from these datasets for training.

Setup	Method	MVTec-AD			VisA		
		Image-AUC	Pixel-AUC	Accuracy	Image-AUC	Pixel-AUC	Accuracy
1-shot	SPADE	81.0 ± 2.0	91.2 ± 0.4	-	79.5 ± 4.0	95.6 ± 0.4	-
	PaDiM	76.6 ± 3.1	89.3 ± 0.9	-	62.8 ± 5.4	89.9 ± 0.8	-
	PatchCore	83.4 ± 3.0	92.0 ± 1.0	-	79.9 ± 2.9	95.4 ± 0.6	-
	WinCLIP	93.1 ± 2.0	95.2 ± 0.5	-	83.8 ± 4.0	**96.4 ± 0.4**	-
	AnomalyGPT (ours)	**94.1 ± 1.1**	**95.3 ± 0.1**	**86.1 ± 1.1**	**87.4 ± 0.8**	96.2 ± 0.1	**77.4 ± 1.0**
2-shot	SPADE	82.9 ± 2.6	92.0 ± 0.3	-	80.7 ± 5.0	96.2 ± 0.4	-
	PaDiM	78.9 ± 3.1	91.3 ± 0.7	-	67.4 ± 5.1	92.0 ± 0.7	-
	PatchCore	86.3 ± 3.3	93.3 ± 0.6	-	81.6 ± 4.0	96.1 ± 0.5	-
	WinCLIP	94.4 ± 1.3	**96.0 ± 0.3**	-	84.6 ± 2.4	**96.8 ± 0.3**	-
	AnomalyGPT (ours)	**95.5 ± 0.8**	95.6 ± 0.2	**84.8 ± 0.8**	**88.6 ± 0.7**	96.4 ± 0.1	**77.5 ± 0.3**
4-shot	SPADE	84.8 ± 2.5	92.7 ± 0.3	-	81.7 ± 3.4	96.6 ± 0.3	-
	PaDiM	80.4 ± 2.5	92.6 ± 0.7	-	72.8 ± 2.9	93.2 ± 0.5	-
	PatchCore	88.8 ± 2.6	94.3 ± 0.5	-	85.3 ± 2.1	96.8 ± 0.3	-
	WinCLIP	95.2 ± 1.3	96.2 ± 0.3	-	87.3 ± 1.8	**97.2 ± 0.2**	-
	AnomalyGPT (ours)	**96.3 ± 0.3**	**96.2 ± 0.1**	**85.0 ± 0.3**	**90.6 ± 0.7**	96.7 ± 0.1	**77.7 ± 0.4**

Table 2: Few-shot IAD results on MVTec-AD and VisA datasets. Results are listed as the average of 5 runs and the best-performing method is in bold. The results for SPADE, PaDiM, PatchCore and WinCLIP are reported from (Jeong et al. 2023).

Method	Image-AUC	Pixel-AUC	Accuracy
PaDiM (Unified)	84.2	89.5	-
JNLD (Unified)	91.3	88.6	-
UniAD	96.5	**96.8**	-
AnomalyGPT (ours)	**97.4**	93.1	**93.3**

Table 3: Unsupervised anomaly detection results on MVTec-AD dataset. The best-performing method is in bold and the results for PaDiM and JNLD are reported from (Zhao 2023).

Evaluation Metrics Following existing IAD methods, we employ the Area Under the Receiver Operating Characteristic (AUC) as our evaluation metric, with image-level and pixel-level AUC used to assess anomaly detection and anomaly localization performance, respectively. However, our approach uniquely allows for determining the presence of anomalies without the need for manually-set thresholds. Therefore, we also utilize the image-level accuracy to evaluate the performance of our method.

Implementation Details We utilize ImageBind-Huge (Girdhar et al. 2023) as the image encoder and Vicuna-7B (Chiang et al. 2023) as the inferential LLM, connected through a linear layer. We initialize our model using pre-trained parameters from PandaGPT (Su et al. 2023). We set the image resolution at 224×224 and feed the outputs from the 8th, 16th, 24th, and 32nd layers of ImageBind-Huge's image encoder to the image decoder. Training is conducted on two RTX-3090 GPUs over 50 epochs, with a learning rate of 1e-3 and a batch size of 16. Linear warm-up and a one-cycle cosine learning rate decay strategy are applied. We perform alternating training using both the pre-training data of PandaGPT (Su et al. 2023) and our anomaly image-text data. Only the decoder and prompt learner undergo parameter updates, while the remaining parameters are all kept frozen.

Quantitative Results

Few-Shot Industrial Anomaly Detection We compare our work with prior few-shot IAD methods, selecting SPADE (Cohen and Hoshen 2020), PaDiM (Defard et al. 2021), PatchCore (Roth et al. 2022), and WinCLIP (Jeong et al. 2023) as the baselines. The results are presented in Table 2. Across both datasets, our method notably outperforms previous approaches in terms of image-level AUC and achieves competitive pixel-level AUC and good accuracy.

Unsupervised Industrial Anomaly Detection In the setting of unsupervised training with a large number of normal samples, given that our method trains a single model on samples from all classes within a dataset, we selected UniAD (You et al. 2022), which is trained under the same setup, as a baseline for comparison. Additionally, we compare our model with PaDiM (Defard et al. 2021) and JNLD (Zhao 2022) using the same unified setting. The results on MVTec-AD dataset are presented in Table 3.

Qualitative Examples

Figure 5 illustrates the performance of our AnomalyGPT in unsupervised anomaly detection, and Figure 6 showcases the results in the 1-shot in-context learning. Our model is capable of indicating the presence of anomalies, pinpointing their locations, and providing pixel-level localization results. Users can engage in multi-turn dialogues related to image content. In the 1-shot in-context learning setting, due to the absence of training, the model's localization performance is slightly lower than the unsupervised setting.

Decoder	Prompt learner	LLM	LoRA	MVTec-AD (unsupervised)			VisA (1-shot)		
				Image-AUC	Pixel-AUC	Accuracy	Image-AUC	Pixel-AUC	Accuracy
		✓		-	-	72.2	-	-	56.5
	✓	✓		-	-	73.4	-	-	56.6
		✓	✓	-	-	79.8	-	-	63.4
✓		✓		97.1	90.9	72.2	85.8	96.2	56.5
✓		✓	✓	97.1	90.9	84.2	85.8	96.2	64.7
✓	✓	✓	✓	96.0	88.1	83.9	85.8	**96.5**	72.7
✓				97.1	90.9	90.3	85.8	96.2	75.4
✓	✓	✓		**97.4**	**93.1**	**93.3**	**87.4**	96.2	**77.4**

Table 4: Results of ablation studies. The ✓ in "Decoder" and "Prompt learner" columns indicate module inclusion. The ✓ in "LLM" column denotes whether use LLM for inference and the ✓ in "LoRA" column denotes whether use LoRA to fine-tune the LLM. In settings without LLM, the maximum anomaly score from normal samples is used as the classification threshold. In settings without decoder, due to the sole textual output from the LLM, we cannot compute image-level and pixel-level AUC.

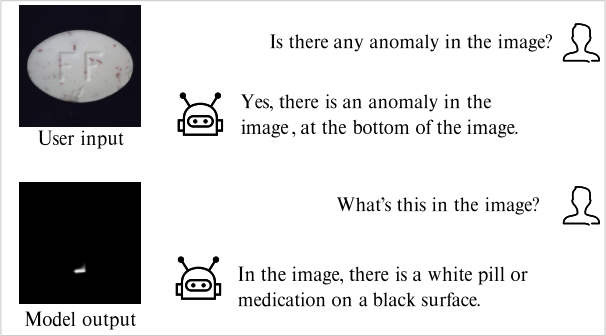

Figure 5: Qualitative example of AnomalyGPT in the unsupervised setting. AnomalyGPT is capable of detecting anomaly, pinpointing its location, providing pixel-level localization results and answering questions about the image.

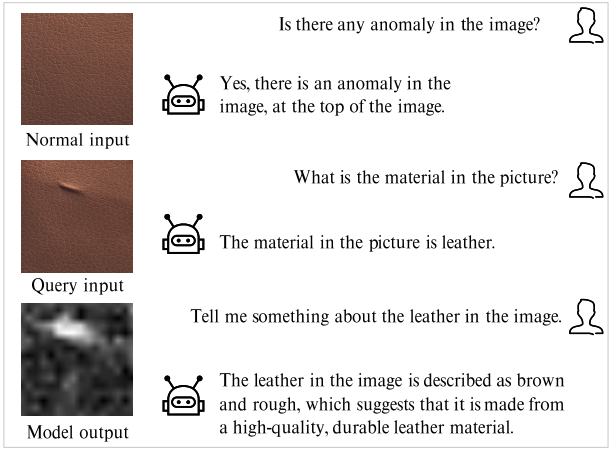

Figure 6: Qualitative example of AnomalyGPT in the one-normal-shot setting. The localization performance is slightly lower compared to the unsupervised setting due to the absence of parameter training.

Ablation Studies

To prove the efficacy of each proposed module, extensive ablation experiments are conducted on both the MVTec-AD and VisA datasets. We primarily focus on four aspects: the decoder, prompt learner, the usage of LLM for inference, and the utilization of LoRA to fine-tune the LLM. The principal results are presented in Table 4. Unsupervised training and testing are carried out on the MVTec-AD dataset, while the one-shot performance is evaluated on the visa dataset. It can be observed that the decoder demonstrates impressive pixel-level anomaly localization performance. Compared to manually-set thresholds, the LLM exhibits superior inference accuracy and provides additional functionality. Furthermore, prompt tuning outperforms LoRA in terms of accuracy and transferability.

Conclusion

We introduce AnomalyGPT, a novel conversational IAD vision-language model, leveraging the powerful capabilities of LVLM. AnomalyGPT can determine whether an image contains anomalies and pinpoint their locations without the need for manually specified thresholds. Furthermore, AnomalyGPT enables multi-turn dialogues focused on anomaly detection and demonstrates remarkable performance in few-shot in-context learning. The effectiveness of AnomalyGPT is validated on two common datasets. Our work delves into the potential application of large visual language models in anomaly detection, offering fresh ideas and possibilities for the field of industrial anomaly detection.

Acknowledgements

This work was supported in part by the National Key R&D Program of China (No.2022ZD0160601), National Natural Science Foundation of China (No.62276260, 62076235, 61976210, 62006230), Beijing Municipal Science and Technology Project (Z231100007423004), sponsored by Zhejiang Lab (No.2021KH0AB07).

References

Bergmann, P.; Fauser, M.; Sattlegger, D.; and Steger, C. 2019. MVTec AD–A comprehensive real-world dataset for unsupervised anomaly detection. In *Proceedings of*

the IEEE/CVF conference on computer vision and pattern recognition, 9592–9600.

Chen, X.; Han, Y.; and Zhang, J. 2023. 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. *arXiv preprint arXiv:2305.17382*.

Chiang, W.-L.; Li, Z.; Lin, Z.; Sheng, Y.; Wu, Z.; Zhang, H.; Zheng, L.; Zhuang, S.; Zhuang, Y.; Gonzalez, J. E.; et al. 2023. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality. *See https://vicuna. lmsys. org (accessed 14 April 2023)*.

Chung, H. W.; Hou, L.; Longpre, S.; Zoph, B.; Tay, Y.; Fedus, W.; Li, E.; Wang, X.; Dehghani, M.; Brahma, S.; et al. 2022. Scaling instruction-finetuned language models. *arXiv preprint arXiv:2210.11416*.

Cohen, N.; and Hoshen, Y. 2020. Sub-image anomaly detection with deep pyramid correspondences. *arXiv preprint arXiv:2005.02357*.

Defard, T.; Setkov, A.; Loesch, A.; and Audigier, R. 2021. Padim: a patch distribution modeling framework for anomaly detection and localization. In *International Conference on Pattern Recognition*, 475–489. Springer.

Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

Girdhar, R.; El-Nouby, A.; Liu, Z.; Singh, M.; Alwala, K. V.; Joulin, A.; and Misra, I. 2023. Imagebind: One embedding space to bind them all. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 15180–15190.

Gudovskiy, D.; Ishizaka, S.; and Kozuka, K. 2022. Cflow-ad: Real-time unsupervised anomaly detection with localization via conditional normalizing flows. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 98–107.

Huang, C.; Guan, H.; Jiang, A.; Zhang, Y.; Spratling, M.; and Wang, Y.-F. 2022. Registration based few-shot anomaly detection. In *European Conference on Computer Vision*, 303–319. Springer.

Jeong, J.; Zou, Y.; Kim, T.; Zhang, D.; Ravichandran, A.; and Dabeer, O. 2023. Winclip: Zero-/few-shot anomaly classification and segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 19606–19616.

Lee, S.; Lee, S.; and Song, B. C. 2022. Cfa: Coupled-hypersphere-based feature adaptation for target-oriented anomaly localization. *IEEE Access*, 10: 78446–78454.

Lei, J.; Hu, X.; Wang, Y.; and Liu, D. 2023. PyramidFlow: High-Resolution Defect Contrastive Localization using Pyramid Normalizing Flow. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 14143–14152.

Li, C.-L.; Sohn, K.; Yoon, J.; and Pfister, T. 2021. Cutpaste: Self-supervised learning for anomaly detection and localization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 9664–9674.

Li, J.; Li, D.; Savarese, S.; and Hoi, S. 2023. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. *arXiv preprint arXiv:2301.12597*.

Lin, T.-Y.; Goyal, P.; Girshick, R.; He, K.; and Dollár, P. 2017. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, 2980–2988.

Liu, H.; Li, C.; Wu, Q.; and Lee, Y. J. 2023. Visual instruction tuning. *arXiv preprint arXiv:2304.08485*.

Milletari, F.; Navab, N.; and Ahmadi, S.-A. 2016. V-net: Fully convolutional neural networks for volumetric medical image segmentation. In *2016 fourth international conference on 3D vision (3DV)*, 565–571. Ieee.

Ouyang, L.; Wu, J.; Jiang, X.; Almeida, D.; Wainwright, C.; Mishkin, P.; Zhang, C.; Agarwal, S.; Slama, K.; Ray, A.; et al. 2022. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35: 27730–27744.

Pérez, P.; Gangnet, M.; and Blake, A. 2003. Poisson image editing. In *ACM SIGGRAPH 2003 Papers*, 313–318.

Pirnay, J.; and Chai, K. 2022. Inpainting transformer for anomaly detection. In *International Conference on Image Analysis and Processing*, 394–406. Springer.

Radford, A.; Kim, J. W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; et al. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, 8748–8763. PMLR.

Roth, K.; Pemula, L.; Zepeda, J.; Schölkopf, B.; Brox, T.; and Gehler, P. 2022. Towards total recall in industrial anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 14318–14328.

Schlüter, H. M.; Tan, J.; Hou, B.; and Kainz, B. 2022. Natural synthetic anomalies for self-supervised anomaly detection and localization. In *European Conference on Computer Vision*, 474–489. Springer.

Su, Y.; Lan, T.; Li, H.; Xu, J.; Wang, Y.; and Cai, D. 2023. Pandagpt: One model to instruction-follow them all. *arXiv preprint arXiv:2305.16355*.

Touvron, H.; Lavril, T.; Izacard, G.; Martinet, X.; Lachaux, M.-A.; Lacroix, T.; Rozière, B.; Goyal, N.; Hambro, E.; Azhar, F.; et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Wang, W.; Chen, Z.; Chen, X.; Wu, J.; Zhu, X.; Zeng, G.; Luo, P.; Lu, T.; Zhou, J.; Qiao, Y.; et al. 2023. Visionllm: Large language model is also an open-ended decoder for vision-centric tasks. *arXiv preprint arXiv:2305.11175*.

Wyatt, J.; Leach, A.; Schmon, S. M.; and Willcocks, C. G. 2022. Anoddpm: Anomaly detection with denoising diffusion probabilistic models using simplex noise. In *Proceed-*

ings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 650–656.

Xie, G.; Wang, J.; Liu, J.; Zheng, F.; and Jin, Y. 2023. Pushing the limits of fewshot anomaly detection in industry vision: Graphcore. *arXiv preprint arXiv:2301.12082*.

Yan, X.; Zhang, H.; Xu, X.; Hu, X.; and Heng, P.-A. 2021. Learning semantic context from normal samples for unsupervised anomaly detection. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, 3110–3118.

Yi, J.; and Yoon, S. 2020. Patch svdd: Patch-level svdd for anomaly detection and segmentation. In *Proceedings of the Asian conference on computer vision*.

You, Z.; Cui, L.; Shen, Y.; Yang, K.; Lu, X.; Zheng, Y.; and Le, X. 2022. A unified model for multi-class anomaly detection. *Advances in Neural Information Processing Systems*, 35: 4571–4584.

Zavrtanik, V.; Kristan, M.; and Skočaj, D. 2021. Reconstruction by inpainting for visual anomaly detection. *Pattern Recognition*, 112: 107706.

Zhao, Y. 2022. Just noticeable learning for unsupervised anomaly localization and detection. In *2022 IEEE International Conference on Multimedia and Expo (ICME)*, 01–06. IEEE.

Zhao, Y. 2023. OmniAL: A unified CNN framework for unsupervised anomaly localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 3924–3933.

Zhu, D.; Chen, J.; Shen, X.; Li, X.; and Elhoseiny, M. 2023. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *arXiv preprint arXiv:2304.10592*.

Zou, Y.; Jeong, J.; Pemula, L.; Zhang, D.; and Dabeer, O. 2022. Spot-the-difference self-supervised pre-training for anomaly detection and segmentation. In *European Conference on Computer Vision*, 392–408. Springer.