

LogicQA: Logical Anomaly Detection with Vision Language Model Generated Questions

Yejin Kwon*, Daeun Moon*, Youngje Oh, Hyunsoo Yoon†

Department of Industrial Engineering, Yonsei University

Seoul, South Korea

{beckykwon, dani0403, yj89.oh, hs.yoon}@yonsei.ac.kr

Abstract

Anomaly Detection (AD) focuses on detecting samples that differ from the standard pattern, making it a vital tool in process control. Logical anomalies may appear visually normal yet violate predefined constraints on object presence, arrangement, or quantity, depending on reasoning and explainability. We introduce LogicQA, a framework that enhances AD by providing industrial operators with explanations for logical anomalies. LogicQA compiles automatically generated questions into a checklist and collects responses to identify violations of logical constraints. LogicQA is training-free, annotation-free, and operates in a few-shot setting. We achieve state-of-the-art (SOTA) Logical AD performance on the public benchmark, MVTec LOCO AD, with an AUROC of 87.6% and an F_1 -max of 87.0% along with the explanations of anomalies. Also, our approach has shown outstanding performance on semiconductor SEM corporate data, further validating its effectiveness in industrial applications.

1 Introduction

Anomaly detection (AD) is crucial for quality control and process optimization in industrial manufacturing. Anomalies are categorized into structural anomalies, referring to localized defects such as deformation or contamination (Bergmann et al., 2022; Zoghiami et al., 2024), and logical anomalies, which assess adherence to predefined constraints, including object presence, quantity, and arrangement (Batzner et al., 2024; Kim et al., 2024b). Unlike structural anomalies, logical anomalies demand clear explanations, as lack of reasoning may lead to misinterpretation. This necessitates an approach that not only detects but also explains logical anomalies (Zhang et al., 2024a).

Data-driven AD plays a critical role in high-quality production and minimizing downtime in industrial control systems. However, simply detecting anomalies without explanation is insufficient

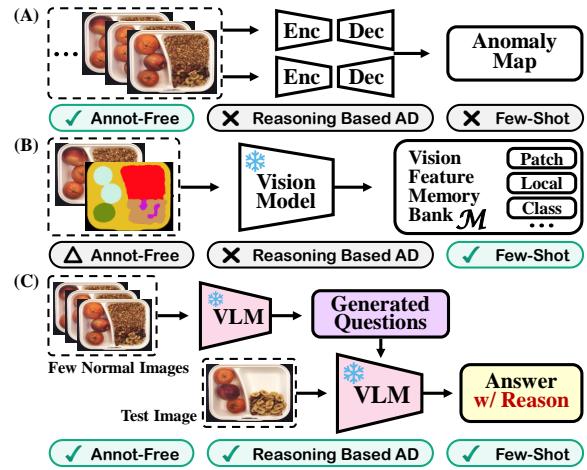


Figure 1: **Overview of Logical AD:** (A) Models trained from scratch (e.g., AutoEncoder) perform logical AD but require a large number of images. (B) Models leveraging memory-based AD methods (e.g., PatchCore) use pre-trained vision models to extract visual features from normal images, enabling few-shot AD. (C) Our method, LogicQA, utilizes a pre-trained VLM to generate anomaly-relevant questions and analyze test images, using the answers to identify and explain abnormalities.

(Wang et al., 2018). Modern industrial systems demand explainability to clarify the reasons behind anomalies (Li et al., 2023b; Gramelt et al., 2024). Understanding root causes enables security experts to take targeted actions, preventing severe malfunctions and unplanned stoppage (Xu et al., 2024).

Existing AD scores, estimating the probability of an image being anomalous, offer limited interpretability regarding the cause of anomalies (Sipple and Youssef, 2022). As shown in Figure 1(A) and (B), most approaches rely on anomaly maps derived from pixel-wise anomaly scores (Tien et al., 2023; Hsieh and Lai, 2024; Liu et al., 2023b). These heatmaps highlight abnormal regions but fail to explain why an anomaly has occurred. **LogicQA** (Logical Question Answering) (Figure 1(C)) addresses this limitation by leveraging a Vision-

Language Model (VLM) to generate anomaly-relevant questions and provide natural language explanations, enhancing human interpretability.

LogicQA introduces a few-shot logical AD framework leveraging a pre-trained VLM. Unlike conventional methods requiring class-specific models, LogicQA eliminates the need for training and manual annotations, allowing universal applicability across different classes. With just few normal images, LogicQA efficiently detects anomalies, making it scalable and practical for industrial fields.

We validate **LogicQA** on the MVTec LOCO AD dataset (Bergmann et al., 2022) and real-world semiconductor SEM dataset. This evaluation demonstrates its effectiveness in AD, particularly in semiconductor defect detection, and highlights its potential for broader industrial AD applications.

Our key contributions are as follows: (1) We achieve SOTA performance in few-shot logical AD by proposing LogicQA, using a VLM to generate anomaly-relevant questions and detect anomalies through question answering. (2) We enhance explainability in logical AD by generating natural language reasoning, helping engineers understand why logical anomalies occur. (3) We introduce a training-free and annotation-free approach, eliminating class-specific training and human-generated prompts, enabling efficient AD with few normal images for industrial uses. (4) We validate LogicQA on both public benchmark and real-world semiconductor SEM data, demonstrating its effectiveness across diverse AD settings.

2 Related Work

Logical AD Approaches Since the release of the MVTec LOCO AD dataset (Bergmann et al., 2022), various unsupervised AD approaches have been developed. Reconstruction-based methods (Bergmann et al., 2022; An and Cho, 2015) rely on AutoEncoders trained with large amounts of normal images, limiting their applicability in few-shot scenarios. As PatchCore (Roth et al., 2022) was introduced, vision memory bank-based methods (Kim et al., 2024b; Liu et al., 2023a) leverage pre-trained vision models and feature banks to improve efficiency. However, these methods require costly computational resources for fine-tuning. In contrast, LogicQA enables logical AD without fine-tuning, making it more scalable and adaptable to real-world applications.

VLMs for Logical AD Recent advancements in VLMs have enabled more interpretable AD by integrating vision and natural language reasoning (Achiam et al., 2023; Liu et al., 2024a). LogicAD (Jin et al., 2025) employs a pre-trained VLM as a text feature extractor, generating explanations via logical reasoning. However, it relies on class-specific Guided Chain-of-Thought (CoT) prompts, requiring precise and laborious prompt engineering for each anomaly category. Similarly, LogiCode (Zhang et al., 2024a) applies Large Language Models (LLMs) to generate Python-based logical constraints, achieving strong detection performance but relying on detailed manual annotations, restricting practical industrial scalability. Our LogicQA overcomes these limitations by eliminating the need for pre-defined prompts and manual annotations, making it a more efficient and adaptable solution for industrial AD.

3 LogicQA

Logical AD differs from structural AD in that it assesses whether an image adheres to predefined logical constraints rather than identifying localized defects. Since logical anomalies often appear visually normal, detecting violations requires an interpretable framework to explain the underlying reasoning.

3.1 Framework Overview

LogicQA (Logical Question Answering) is a novel framework for logical AD that ensures interpretability by generating anomaly-relevant questions and reasoning. Unlike prior methods dependent on manual annotations or class-specific prompts, LogicQA leverages a pre-trained VLM, eliminating the need for annotations and human-generated prompts. This enables scalable deployment in industrial applications without task-specific fine-tuning.

Our proposed LogicQA consists of four stages: (1) *Describing the normal images*, (2) *Summarizing the normal image context*, (3) *Generating the main questions*, and (4) *Testing*, as shown in Figure 2. All detailed prompts and examples are listed in the Appendix A.

3.2 Describing the Normal Images

To ensure effective logical AD, LogicQA begins by analyzing the characteristics of normal images using a pretrained VLM. A single normal image, along with a predefined normality definition, is fed

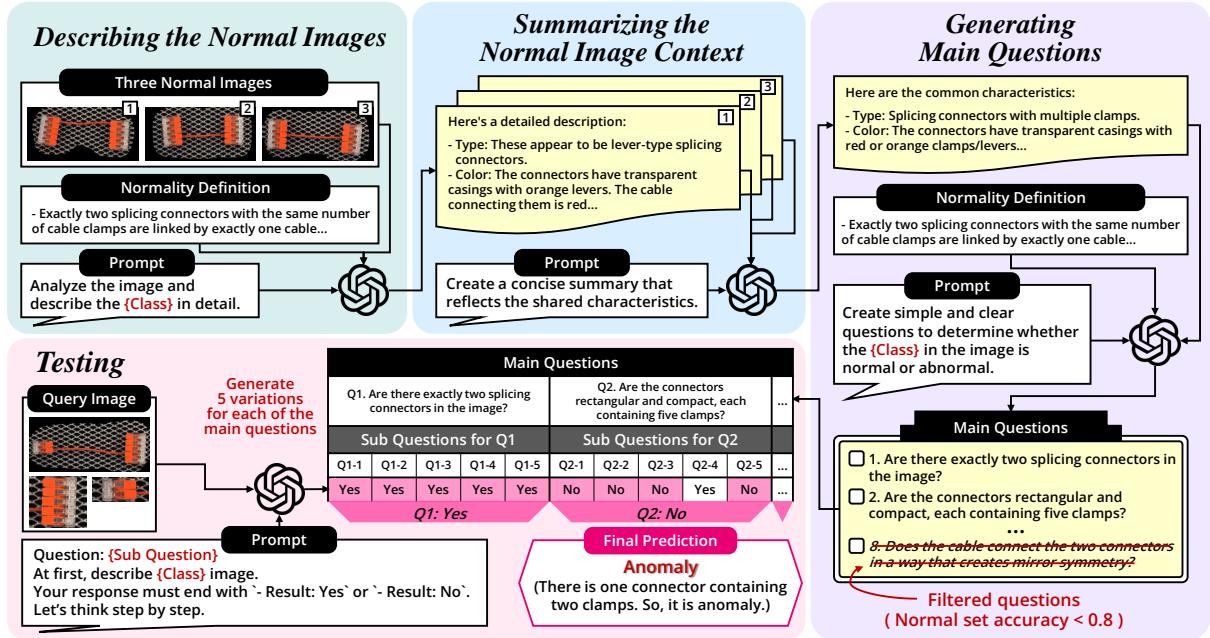


Figure 2: **Pipeline of LogicQA.** (1) **Describing the Normal Images** – The VLM generates textual descriptions of three normal images based on a predefined normality definition. (2) **Summarizing the Normal Image Context** – Shared features are extracted to define the core traits of normality. (3) **Generating Main Questions** – The VLM formulates key questions to assess whether an image is normal or anomalous. (4) **Testing** – The VLM generates sub-questions as variations of the main questions. Using a voting mechanism on the VLM’s responses, we determine whether the image satisfies the main questions. If it fails to satisfy even one, it is classified as anomalous.

to the model, prompting it to generate a detailed textual description (Jin et al., 2025). The normality definition, adopted from Bergmann et al. (2022) (Appendix C.2), establishes logical constraints that define expected object attributes and configurations in the dataset.

The descriptions capture location, quantity, and appearance of key elements, ensuring that the model focuses on relevant structural and contextual features rather than background noise. This process enhances AD robustness by aligning the model’s attention with critical aspects of normality. To further refine the understanding of normality, three distinct normal images are processed separately, with each description contributing to a consolidated representation of the dataset’s normality definition. This enables the model to generalize beyond individual examples, preserving essential normal properties.

3.3 Summarizing the Normal Image Context

The summarization step refines the extracted normality by feeding previously generated descriptions into the VLM and distilling shared attributes into a coherent representation of common features. This process ensures that AD remains robust against variations within normal images by focus-

ing on the most consistent and core characteristics.

By using diverse normal images, the model learns robust normality patterns, ensuring AD remains effective across different instances. This prevents overfitting to specific examples and allows model to focus on meaningful logical constraints.

3.4 Generating Main Questions

The question generation step refines generalized normality criteria into a checklist, prompting the VLM to generate key multiple questions to detect whether a target image is an anomaly. This method decomposes anomaly detection into multiple focused questions instead of relying on a single query. Recent studies (Ko et al., 2024; Yang et al., 2024) show that task deconstruction methods improve reliability. Hence, our method makes judgements by integrating multiple main questions (Main-Qs).

We provide the former summary and normality definition as input when prompting the VLM to extract key questions. The normality definition is reintroduced to help the VLM extract more relevant normality criteria. The resulting questions serve as candidate Main-Qs. Since only a few normal image descriptions are available, the initial set of questions may not fully generalize across all cases.

To improve robustness, we evaluate their consistency by applying them to a diverse set of normal images. As questions with low accuracy (below 80%) are indicative of bias toward the few-shot samples, they were excluded to ensure that the final set of questions remains broadly applicable without dataset-specific bias.

3.5 Testing

In the testing step, the goal is to judge whether the query image is anomalous and to analyze the cause of the anomaly. Recent VLMs are not always reliable and may generate incorrect answers or suffer from hallucinations (Mashrur et al., 2024; Zhang et al., 2024b). To mitigate this, we augment each Main-Q with five semantically equivalent sub-questions (Sub-Qs) (Zhou et al., 2022). The final decision is made through majority voting on the Sub-Qs’ responses.

By leveraging multiple outputs instead of a single response, our method effectively reduces reasoning errors. If any Main-Q receives a ‘No’ response, it means that the image violates at least one normal constraint and is classified as an anomaly. Additionally, the specific Main-Qs receiving ‘No’ provide a clear rationale for the anomaly’s cause.

To enhance interpretability, our approach follows a step-by-step (Kojima et al., 2022) reasoning process rather than a direct anomaly prediction. This aligns with the CoT approach (Wei et al., 2022), which strengthens VLM’s logical reasoning and maintains contextual consistency, thereby improving judgment reliability.

Unlike traditional AD methods that require class-specific prompts, LogicQA eliminates such dependencies, enabling flexible and intuitive modifications by adjusting only the question and class name (Portillo Wightman et al., 2023). This makes it highly applicable for industrial use, as it does not require predefined class-specific guided prompts or CoT reasoning like Jin et al. (2025), allowing for seamless adoption in real-world settings.

4 Dataset

We evaluated our method using the MVTec LOCO AD dataset and an industrial semiconductor SEM dataset collected from real-world manufacturing processes. Both datasets contain normal and logical anomaly samples. (The overview and sample images of the two datasets are included in the Appendix C and E.)

MVTec LOCO AD Dataset MVTec LOCO AD Dataset, (Bergmann et al. (2022)), consists of five object categories (breakfast box, juice bottle, pushpins, screw bag, splicing connectors) from industrial scenarios, with objects selected as close as possible to real-world applications. Each category has several types of logical anomaly.

The VLM struggles with cases in the MVTec LOCO AD dataset where images contain large background areas, leading to long input contexts (Liu et al., 2024c), or where they contain uniform objects (Campbell et al., 2024). To address this, we applied two pre-processing steps, as depicted in Figure 3. First, **Back Patch Masking (BPM)** (Lee et al., 2023) was used to isolate the target object from the background, producing an object-centered image. Second, **Language Segment-Anything model (Lang-SAM)**, combined with GroundingDINO (Liu et al., 2024d) and SAM2 (Ravi et al., 2024), was used to segment uniform objects individually, mitigating the VLM’s limitations in multi-object recognition. Details and effects of BPM and Lang-SAM are in the Appendix F .

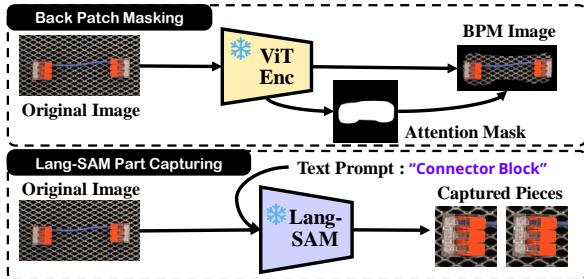


Figure 3: **Input Image Pre-Processing:** BPM applies an attention mask to the original image, masking the background, preserving objects. Lang-SAM identifies objects relevant to the given prompt and returns them as bounding boxes.

Semiconductor SEM Dataset Scanning Electron Microscopy (SEM) operates by applying a high voltage to direct an electron beam onto the surface of a sample, then secondary electrons generate a wafer image. The SEM has around 1 nm resolution to get precise wafer surface patterns. This corporate dataset reflects critical inspection stages in semiconductor manufacturing, directly affecting chip quality and production yields. The dataset has two defect types: spot and bridge. Spot defects appear as circular blemishes that degrade chip performance, while bridge defects take the form of elongated connections linking separate conductive lines (Kim et al., 2020).

MVTec LOCO AD (only Logical Anomaly)	LogicQA (Ours)	LogicAD Jin et al. (2025)	WinCLIP Jeong et al. (2023)	PatchCore Roth et al. (2022)	GCAD Bergmann et al. (2022)	AST Rudolph et al. (2023)			
Category	AUROC	F_1 -max	AUROC	F_1 -max	AUROC	F_1 -max	AUROC	AUROC	AUROC
Breakfast Box	87.6	91.6	93.1	82.7	57.6	63.3	74.8	87.0	80.0
Juice Bottle	88.2	89.6	81.6	83.2	75.1	58.2	93.9	100.0	91.6
Pushpins	98.4	97.6	98.1	98.5	54.9	57.3	63.6	97.5	65.1
Screw Bag	71.5	64.5	83.8	77.9	69.5	58.8	57.8	56.0	80.1
Splicing Connectors	92.4	91.5	73.4	76.1	64.5	59.9	79.2	89.7	81.8
Average	87.6 (1.6↑)	87.0 (3.3↑)	86.0	83.7	64.3	59.5	74.0	86.0	79.7

Table 1: **Logical AD performance on MVTec LOCO AD dataset.** AUROC and F_1 -max in % for detecting logical anomalies of all categories of MVTec LOCO AD Dataset. We report the mean over 3 runs for our method. Among models using the few-shot approach, the best results are highlighted in bold. The values highlighted in red indicate increased score compared to LogicAD. Our LogicQA demonstrates outstanding performance while incorporating a few-shot approach, explainability, and the use of auto-generated prompts.

5 Experiments and Results

5.1 Experimental Setting

We implement our experiments by leveraging three SOTA VLMs (GPT-4o (Achiam et al., 2023), Gemini-1.5 Flash (Team et al., 2024), and InternVL-2.5 38B (Chen et al., 2024)). Comprehensive details on model configurations and deployment settings are outlined in the Appendix B. All experiments are training-free and few-shot (three normal images per test image). Our assessments are based on the MVTec LOCO AD dataset and Semiconductor SEM dataset. We conducted the experiments three times for each category and calculated the average score, as indicated in Table 1.

5.2 Evaluation Metrics

Our approach uses a VLM for Vision Question-Answering (Sinha et al., 2025). If any of the responses to Main-Qs are “No”, the model predicts “Anomaly”. It is threshold-free, providing binary predictions and reasoning but not an anomaly score. So, we propose using the VLM’s log probabilities to compute an anomaly score. Kadavath et al. (2022); Kim et al. (2024a); Lee et al. (2021) have shown that low token prediction probabilities (*Log probs*) can indicate a lack of knowledge in LLMs and lead to uncertain performance on downstream tasks. We consider the VLM’s log-probability of answers to Sub-Qs as indicators of accuracy, reliability, and confidence of answer. We define key formulations:

A Sub-Q function q_{ij} outputs “Yes(0)” or “No(1)” for an input image x , where $i \in [1, m]$ represents the number of Main-Qs, and $j \in [1, 5]$ indexes the five Sub-Qs per Main-Q. Each Main-Q,

$Q_i(x)$ is defined as:

$$Q_i(x) = \begin{cases} 0, & \text{if } \sum_{j=1}^5 q_{ij}(x) < \sum_{j=1}^5 (1 - q_{ij}(x)) \\ 1, & \text{otherwise.} \end{cases}$$

A final function $F(x)$ determines whether the input is a normal image or an anomaly, defined as:

$$F(x) = \begin{cases} \text{"Normal",} & \text{if } \sum_i Q_i(x) = 0, \\ \text{"Anomaly",} & \text{otherwise.} \end{cases}$$

For each Main-Q, we take the highest log-probability among the Sub-Qs whose answers match the voted result, then apply the exponential function to all selected values. And, we get anomaly score for test image below:

$$s_i = \max_j \{\log p(q_{ij}(x)) \mid q_{ij}(x) = Q_i(x)\}$$

$$S = \{e^{s_i} \mid i = 1, \dots, m\}$$

$$\text{Anomaly Score} = \begin{cases} 1 - \text{Median}(S), & \text{if } F(x) = \text{Normal} \\ \text{Median}(S), & \text{if } F(x) = \text{Anomaly} \end{cases}$$

The *logp* function computes the log probability generated during the processing of the input. By calculating the anomaly score as above, we use F_1 -max and Area Under the Reciever Operating Characteristic (AUROC) to evaluate our method, **LogicQA**, as same as existing approaches.

5.3 Result

MVTec LOCO AD Result The performance of Logical AD tested on the MVTec LOCO AD dataset for each method is shown in Table 1, presented in terms of AUROC and F_1 -max scores. For a comprehensive comparison, the table also indicates which shot approach was chosen and whether explainability is incorporated. LogicQA consistently outperforms the existing few-shot VLM-based SOTA method (Jin et al., 2025) across all

metrics, achieving a 1.6% increase in AUROC and a 3.3% improvement in F_1 -max score. Notably, in the *splicing connectors* class, both the AUROC and F_1 -max metrics showed remarkable improvements, with AUROC increasing by 19% and F_1 -max improving by 15.4%. Even compared to full-shot methods (Liu et al., 2025b; Rudolph et al., 2023), our LogicQA outperforms in almost all classes. (Frameworks utilizing in-house annotations are in Appendix 5).

LogicQA not only employs a few-shot approach and an auto-generated question mechanism for prediction but also provides natural language explanations for anomaly causes while achieving remarkable performance compared to other models.

Semiconductor SEM Result As shown in Table 2, LogicQA (GPT-4o) outperforms PatchCore (Roth et al., 2022), a representative few-shot AD method, on the semiconductor SEM dataset, yielding an 11.1% increase in AUROC and a 14.6% improvement in F_1 -max. Also, LogicQA (GPT-4o) excels in detecting both “Bridge” and “Spot” anomalies, achieving the best scores. LogicQA significantly outperforms PatchCore even using the smaller open-source model InternVL-2.5 8B (Chen et al., 2024). This suggests applicability in real-world industrial settings, where deploying large proprietary models may not be feasible. Additionally, LogicQA shows excellent performance in Table 2 even though it did not include the process of filtering Main-Q using a few normal images.

SEM	LogicQA		PatchCore	
	GPT-4o	InternVL-2.5 8B	Roth et al. (2022)	
AUROC	89.7	90.4	80.7	83.0
Bridge	90.8	94.3	89.7	75.4
Spot				79.2
Average	90.3 (11.1 ↑)	92.4 (14.6 ↑)	85.2	77.8

Table 2: **Logical AD performance on Semiconductor SEM dataset.** Our LogicQA outperforms PatchCore regarding metrics and AD explainability. All experiments were conducted with the same three normal images.

5.4 Ablation Studies

Does LogicQA provide the correct reasoning? The MVTec LOCO AD dataset does not provide specific reasons for why each anomaly image is classified as anomalous. Therefore, we conducted a human evaluation to compare the reasons behind the model’s anomaly detection with human perception. Two annotators were provided with the dataset and Main-Qs for each class and asked to

answer accordingly. Their responses were then compared with the model’s answers. Annotator1 showed 98% agreement for normal images and 85% for anomalous ones, while Annotator2 showed 98% and 86%, respectively, demonstrating high correspondence. Notably, the strong agreement for anomalous images indicates that LogicQA not only detects anomalies but also explains their critical causes, demonstrating its ability as a comprehensive anomaly explainability model.

Can other VLMs work well with LogicQA? To verify the applicability of our LogicQA in other VLMs with fewer parameters, we conducted tests using Gemini-1.5 Flash (Team et al., 2024) and InternVL-2.5 38B (Chen et al., 2024). The experimental results, presented in Table 3 with recorded F_1 -max scores, show that both models maintained stable performance, with some classes even achieving higher scores. This suggests that LogicQA can be effectively applied across various VLMs.

VLMs	GPT-4o	Gemini-1.5 Flash	InternVL-2.5 38B
Breakfast Box	91.6	83.3	88.2
Juice Bottle	89.6	78.0	73.7
Pushpins	97.6	98.9	93.7
Screw Bag	64.5	91.7	62.6
Splicing Connectors	91.5	46.8	69.9
Average	87.0	79.7	77.6

Table 3: **LogicQA performance with other VLMs on the MVTec LOCO AD dataset.**

6 Conclusion

In this paper, we propose LogicQA, an explainable logical AD framework leveraging a Vision-Language Model (VLM) to detect anomalies and provide natural language explanations. LogicQA requires only a few normal images to define normal characteristics, significantly reducing the dependency on large labeled datasets. By eliminating class-specific fine-tuning and manually generated prompts, LogicQA facilitates efficient and scalable deployment in industrial environments. We evaluated LogicQA on the public benchmark, MVTec LOCO AD Dataset, where it outperformed existing explainable AD models. We further validated robustness of LogicQA on a real-world manufacturing dataset, Semiconductor SEM Dataset. These results confirm LogicQA as an effective, reliable, and practical solution for diverse industrial applications.

Limitations

Our framework is designed for easy application in industrial settings and delivers strong performance, though some limitations remain. Since our approach relies on VLMs, its performance inherently depends on the VLMs' visual recognition capabilities. Currently, VLMs exhibit imperfect accuracy (Wang et al., 2023; Li et al., 2023a) necessitating specific image preprocessing steps. However, as the technology evolves, this step may become less necessary (Jiang et al., 2025; Liu et al., 2025a). Additionally, generating a well-generalized Main-Qs set requires diverse images. Fortunately, normal images are relatively easy to obtain in industrial environments (Choi et al., 2021; Liu et al., 2024b), which helps mitigate this challenge. Also, the evaluation result on the Semiconductor SEM dataset confirms our model demonstrated strong anomaly detection performance even without the Main-Q filtering process.

Ethics Statement

This research uses GPT-4o and Gemini-1.5-Flash as baseline models. As with any large language model, their outputs may include unintended biases or harmful content depending on user inputs. To ensure ethical deployment, we apply engineering measures to mitigate these risks and enhance model reliability. Since both models are proprietary, with undisclosed training details and weights, assessing potential biases and risks remains challenging. Additionally, handling sensitive data with these models requires caution due to possible unintended exposure. When necessary, we recommend using open-source alternatives for greater transparency and control. AI-assisted tools were utilized solely for grammar correction and linguistic refinement during manuscript preparation. However, the originality, intellectual contributions, and core ideas of this paper are entirely the authors' own. We are committed to responsible AI use, continuous monitoring, and improving fairness and safety in real-world applications.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, and 1 others. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Jinwon An and Sungzoon Cho. 2015. Variational autoencoder based anomaly detection using reconstruction probability.

Kilian Batzner, Lars Heckler, and Rebecca König. 2024. Efficientad: Accurate visual anomaly detection at millisecond-level latencies. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 128–138.

Paul Bergmann, Kilian Batzner, Michael Fauser, David Sattlegger, and Carsten Steger. 2022. Beyond dents and scratches: Logical constraints in unsupervised anomaly detection and localization. *International Journal of Computer Vision*, 130(4):947–969.

Declan Iain Campbell, Sunayana Rane, Tyler Giallanza, C. Nicolò De Sabbata, Kia Ghods, Amogh Joshi, Alexander Ku, Steven M Frankland, Thomas L. Griffiths, Jonathan D. Cohen, and Taylor Whittington Webb. 2024. Understanding the limits of vision language models through the lens of the binding problem. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Zhe Chen, Weiyun Wang, Yue Cao, Yangzhou Liu, Zhangwei Gao, Erfei Cui, Jinguo Zhu, Shenglong Ye, Hao Tian, Zhaoyang Liu, and 1 others. 2024. Expanding performance boundaries of open-source multimodal models with model, data, and test-time scaling. *arXiv preprint arXiv:2412.05271*.

Kukjin Choi, Jihun Yi, Changhwa Park, and Sungroh Yoon. 2021. Deep learning for anomaly detection in time-series data: Review, analysis, and guidelines. *IEEE Access*, 9:120043–120065.

Daniel Gramelt, Timon Höfer, and Ute Schmid. 2024. Interactive explainable anomaly detection for industrial settings. *arXiv preprint arXiv:2410.12817*.

Yu-Hsuan Hsieh and Shang-Hong Lai. 2024. Csad: Unsupervised component segmentation for logical anomaly detection. *arXiv preprint arXiv:2408.15628*.

Jongheon Jeong, Yang Zou, Taewan Kim, Dongqing Zhang, Avinash Ravichandran, and Onkar Dabeer. 2023. Winclip: Zero-/few-shot anomaly classification and segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19606–19616.

Nicholas Jiang, Anish Kachinthaya, Suzanne Petryk, and Yossi Gandelsman. 2025. Interpreting and editing vision-language representations to mitigate hallucinations. In *The Thirteenth International Conference on Learning Representations*.

Er Jin, Qihui Feng, Yongli Mou, Stefan Decker, Gerhard Lakemeyer, Oliver Simons, and Johannes Stegmaier. 2025. Logicad: Explainable anomaly detection via vlm-based text feature extraction. *arXiv preprint arXiv:2501.01767*.

- Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, and 1 others. 2022. Language models (mostly) know what they know. *arXiv preprint arXiv:2207.05221*.
- Minsu Kim, Hoon Jo, Moonsoo Ra, and Whoi-Yul Kim. 2020. Weakly-supervised defect segmentation on periodic textures using cyclegan. *IEEE Access*, 8:176202–176216.
- Sangryul Kim, Donghee Han, and Sehyun Kim. 2024a. ProbGate at EHRSQL 2024: Enhancing SQL query generation accuracy through probabilistic threshold filtering and error handling. In *Proceedings of the 6th Clinical Natural Language Processing Workshop*, pages 687–696, Mexico City, Mexico. Association for Computational Linguistics.
- Soopil Kim, Sion An, Philip Chikontwe, Myeongkyun Kang, Ehsan Adeli, Kilian M Pohl, and Sang Hyun Park. 2024b. Few shot part segmentation reveals compositional logic for industrial anomaly detection. In *Proceedings of the AAAI conference on artificial intelligence*, volume 38, pages 8591–8599.
- Miyoung Ko, Sue Hyun Park, Joonsuk Park, and Minjoon Seo. 2024. Hierarchical deconstruction of LLM reasoning: A graph-based framework for analyzing knowledge utilization. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 4995–5027, Miami, Florida, USA. Association for Computational Linguistics.
- Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large language models are zero-shot reasoners. *Advances in neural information processing systems*, 35:22199–22213.
- Nayeon Lee, Yejin Bang, Andrea Madotto, and Pascale Fung. 2021. Towards few-shot fact-checking via perplexity. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1971–1981, Online. Association for Computational Linguistics.
- Yujin Lee, Harin Lim, Seoyoon Jang, and Hyunsoo Yoon. 2023. Uniformaly: Towards task-agnostic unified framework for visual anomaly detection. *arXiv preprint arXiv:2307.12540*.
- Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Xin Zhao, and Ji-Rong Wen. 2023a. Evaluating object hallucination in large vision-language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 292–305, Singapore. Association for Computational Linguistics.
- Zhong Li, Yuxuan Zhu, and Matthijs Van Leeuwen. 2023b. A survey on explainable anomaly detection. *ACM Transactions on Knowledge Discovery from Data*, 18(1):1–54.
- Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. 2024a. Improved baselines with visual instruction tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26296–26306.
- Jiaqi Liu, Guoyang Xie, Jinbao Wang, Shangnian Li, Chengjie Wang, Feng Zheng, and Yaochu Jin. 2024b. Deep industrial image anomaly detection: A survey. *Machine Intelligence Research*, 21(1):104–135.
- Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. 2024c. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173.
- Sheng Liu, Haotian Ye, and James Zou. 2025a. Reducing hallucinations in large vision-language models via latent space steering. In *The Thirteenth International Conference on Learning Representations*.
- Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Qing Jiang, Chunyuan Li, Jianwei Yang, Hang Su, and 1 others. 2024d. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. In *European Conference on Computer Vision*, pages 38–55. Springer.
- Tongkun Liu, Bing Li, Xiao Du, Bingke Jiang, Xiao Jin, Liuyi Jin, and Zhuo Zhao. 2023a. Component-aware anomaly detection framework for adjustable and logical industrial visual inspection. *Advanced Engineering Informatics*, 58:102161.
- Wenrui Liu, Hong Chang, Bingpeng Ma, Shiguang Shan, and Xilin Chen. 2023b. Diversity-measurable anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12147–12156.
- Zehao Liu, Mengzhou Gao, and Pengfei Jiao. 2025b. Gcad: Anomaly detection in multivariate time series from the perspective of granger causality. *arXiv preprint arXiv:2501.13493*.
- Akib Mashrur, Wei Luo, Nayyar A Zaidi, and Antonio Robles-Kelly. 2024. Robust visual question answering via semantic cross modal augmentation. *Computer Vision and Image Understanding*, 238:103862.
- Gwenyth Portillo Wightman, Alexandra Delucia, and Mark Dredze. 2023. Strength in numbers: Estimating confidence of large language models by prompt agreement. In *Proceedings of the 3rd Workshop on Trustworthy Natural Language Processing (TrustNLP 2023)*, pages 326–362, Toronto, Canada. Association for Computational Linguistics.
- Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, and 1 others. 2024. Sam 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*.

- Karsten Roth, Latha Pemula, Joaquin Zepeda, Bernhard Schölkopf, Thomas Brox, and Peter Gehler. 2022. Towards total recall in industrial anomaly detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14318–14328.
- Marco Rudolph, Tom Wehrbein, Bodo Rosenhahn, and Bastian Wandt. 2023. Asymmetric student-teacher networks for industrial anomaly detection. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 2592–2602.
- Neelabh Sinha, Vinija Jain, and Aman Chadha. 2025. Guiding vision-language model selection for visual question-answering across tasks, domains, and knowledge types. In *Proceedings of the First Workshop of Evaluation of Multi-Modal Generation*, pages 76–94, Abu Dhabi, UAE. Association for Computational Linguistics.
- John Sipple and Abdou Youssef. 2022. A general-purpose method for applying explainable ai for anomaly detection. In *International Symposium on Methodologies for Intelligent Systems*, pages 162–174. Springer.
- Gemini Team, Petko Georgiev, Ving Ian Lei, Ryan Burnell, Libin Bai, Anmol Gulati, Garrett Tanzer, Damien Vincent, Zhufeng Pan, Shibo Wang, and 1 others. 2024. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *arXiv preprint arXiv:2403.05530*.
- Tran Dinh Tien, Anh Tuan Nguyen, Nguyen Hoang Tran, Ta Duc Huy, Soan TM Duong, Chanh D Tr Nguyen, and Steven QH Truong. 2023. Revisiting reverse distillation for anomaly detection. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 24511–24520. IEEE.
- Jianwu Wang, Chen Liu, Meiling Zhu, Pei Guo, and Yapeng Hu. 2018. Sensor data based system-level anomaly prediction for smart manufacturing. In *2018 IEEE International Congress on Big Data (BigData Congress)*, pages 158–165. IEEE.
- Junyang Wang, Yiyang Zhou, Guohai Xu, Pengcheng Shi, Chenlin Zhao, Haiyang Xu, Qinghao Ye, Ming Yan, Ji Zhang, Jihua Zhu, and 1 others. 2023. Evaluation and analysis of hallucination in large vision-language models. *arXiv preprint arXiv:2308.15126*.
- Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed H. Chi, Quoc V Le, and Denny Zhou. 2022. Chain of thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*.
- Lijuan Xu, Ziyu Han, Zhen Wang, and Dawei Zhao. 2024. Finding component relationships: A deep-learning-based anomaly detection interpreter. *IEEE Transactions on Computational Social Systems*, 11(3):4149–4162.
- Qian Yang, Weixiang Yan, and Aishwarya Agrawal. 2024. Decompose and compare consistency: Measuring VLMs’ answer reliability via task-decomposition consistency comparison. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 3613–3627, Miami, Florida, USA. Association for Computational Linguistics.
- Yiheng Zhang, Yunkang Cao, Xiaohao Xu, and Weiming Shen. 2024a. Logicode: an llm-driven framework for logical anomaly detection. *IEEE Transactions on Automation Science and Engineering*.
- Yuhui Zhang, Alyssa Unell, Xiaohan Wang, Dhruba Ghosh, Yuchang Su, Ludwig Schmidt, and Serena Yeung-Levy. 2024b. Why are visually-grounded language models bad at image classification? In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.
- Yongchao Zhou, Andrei Ioan Muresanu, Ziwen Han, Keiran Paster, Silviu Pitis, Harris Chan, and Jimmy Ba. 2022. Large language models are human-level prompt engineers. In *The Eleventh International Conference on Learning Representations*.
- Firas Zoghlami, Dena Bazazian, Giovanni L. Masala, Mario Gianni, and Asiya Khan. 2024. Viglad: Vision graph neural networks for logical anomaly detection. *IEEE Access*, 12:173304–173315.

A LogicQA - Prompts

Prompt - Describing the Normal Images

This is a {Class}. Analyze the image and describe the {Class} in detail, including type, color, size (length, width), material, composition, quantity, relative location.

< Normal Constraints for a {Class} >
{Normal Definition}

{Image Prompt (Image Input)}

Example :

This is a breakfast box. Analyze the image and describe the breakfast box in detail, including type, color, size (length, width), material, composition, quantity, relative location..

<Normal Constraints for breakfast box>

- The breakfast box always contain exactly two tangerines and one nectarine that are always located on the left-hand side of the box.
- The ratio and relative position of the cereals and the mix of banana chips and almonds on the right-hand side are fixed.

Prompt - Summarizing the Normal Image Context

[Normal {Class} Description 1]
{Description 1}

[Normal {Class} Description 2]
{Description 2}

[Normal {Class} Description 3]
{Description 3}

Combine the three descriptions into one by extracting only the "common" features.
Create a concise summary that reflects the shared characteristics while removing any redundant or unique details.

Example :
[Normal Breakfast Box Description 1]
The breakfast box is divided into two sections. ...

[Normal Breakfast Box Description 2]
The breakfast box in the image contains the following items: ...

[Normal Breakfast Box Description 3]
The breakfast box in the image has two side. ...

Combine the three descriptions into one by extracting only the "common" features.
Create a concise summary that reflects the shared characteristics while removing any redundant or unique details.

Prompt - Generating Main Questions

[Description of {Class}]
{Summary Description }

[Normal Constraints for {Class}]
{Normal Definition}

Using the [Normal Constraints for {Class}] and [Description of {Class}], create several but essential , simple and important questions to determine whether the {Class} in the image is normal or abnormal. Ensure the questions are only based on visible characteristics, excluding any aspects that cannot be determined from the image. Also, simplify any difficult terms into easy-to-understand questions.

(Q1) : ...
(Q2) : ...

Example :

[Description of breakfast box]
The breakfast box is divided into two sections: ...

[Normal Constraints for breakfast box]

- The breakfast box always contain exactly two tangerines and one nectarine that are always located on the left-hand side of the box.
- The ratio and relative position of the cereals and the mix of banana chips and almonds on the right-hand side are fixed.

Using the [Normal Constraints for Breakfast Box] and [Description of Breakfast Box], create several but essential , simple and important questions to determine whether the Breakfast Box in the image is normal or abnormal. Ensure the questions are only based on visible characteristics, excluding any aspects that cannot be determined from the image. Also, simplify any difficult terms into easy-to-understand questions.

(Q1): ...
(Q1): ...

Prompt - Generating 5 variations Sub-Questions

Generate five variations of the following question while keeping the semantic meaning.

Input : {Question}

Output1:
Output2:
Output3:
Output4:
Output5:

Generate five variations of the following question while keeping the semantic meaning.

Input : Is there one nectarine visible on the left-hand side of the breakfast box?

Output 1:
Output 2:
Output 3:
Output 4:
Output 5:

Prompt - Testing

Question : {Question}

At first, describe {Class} image and then answer the question.

Your response must end with ‘- Result: Yes’ or ‘- Result: No’.

Let’s think step by step.

{Test Image Prompt (Test Image Input)}

Question : Can you see a single nectarine on the left side of the breakfast box?

At first, describe breakfast box image and then answer the question.

Your response must end with ‘- Result: Yes’ or ‘- Result: No’.

Let’s think step by step.

B VLM Implementation Details

B.1 VLMs

In our study, we uses three VLMs: GPT-4o ([Achiam et al., 2023](#)) , Gemini-1.5 Flash ([Team et al., 2024](#)) , and InternVL2.5(38B, 8B) ([Chen et al., 2024](#)). The GPT-4o model was accessed and inferred through the OpenAI API. For the GPT-4o model, we fixed *temperature* to 1.0 and other hyper-parameters to default. Regarding the Gemini-1.5 models, *temperature* is 1, *top_p* is 0.95, and *top_k* is 40. For Open-Source InternVL-2.5 from OpenGVLab, we set *temperature* to 0.2, *top_p* to 0.7, *repetition_penalty* to 1.1, *do_sample* to True, and *max_new_tokens* to 512. **All these settings are the same across all experiments and across datasets.**

B.2 Local Experimental Setup

We utilized the open-source InternVL-2.5, leveraging up to three NVIDIA A100 GPUs due to its substantial computational requirements.

B.3 Lang-SAM Prompt

When using Lang-SAM to the two classes (Pushpins, Splicing Connectors), a text prompt was needed to accurately capture the independent entities. It is as follows.

- Splicing Connectors: Connector Block

- Pushpins: The individual black compartments within the transparent plastic storage box

B.4 Data Security Option

To ensure the confidentiality and security of the **Semiconductor SEM dataset** provided by global company, we took stringent precautions when utilizing GPT-4o for our research. **Specifically, all data-sharing functionalities were disabled to strictly prevent unintended exposure or transmission of data outside the controlled research environment.** By implementing these safeguards, we ensured that no proprietary or sensitive information was inadvertently shared with external servers or third-party entities. This approach aligns with best practices for handling proprietary industrial datasets while leveraging advanced AI models for research and analysis.

C MVTec LOCO AD Dataset

C.1 MVTec LOCO AD Dataset Overview

This is a statistical outline of the public MVTec Logical Constraints Anomaly Detection (LOCO) AD Dataset. It consists of five categories (Breakfast Box, Screw Bag, Pushpins, Splicing Connectors, Juice Bottle). We conducted a few-shot experiment by randomly selecting three photos from the train-normal set.

Category	Train-Normal Images	Test-Normal Images	Test-Logical Anomaly Images	Detect types
Breakfast Box	351	102	83	22
Screw Bag	360	122	137	20
Pushpins	372	138	91	8
Splicing Connectors	354	119	108	21
Juice Bottle	335	94	142	18
Total	1772	575	561	89

Table 4: Overview of the MVTec LOCO AD dataset

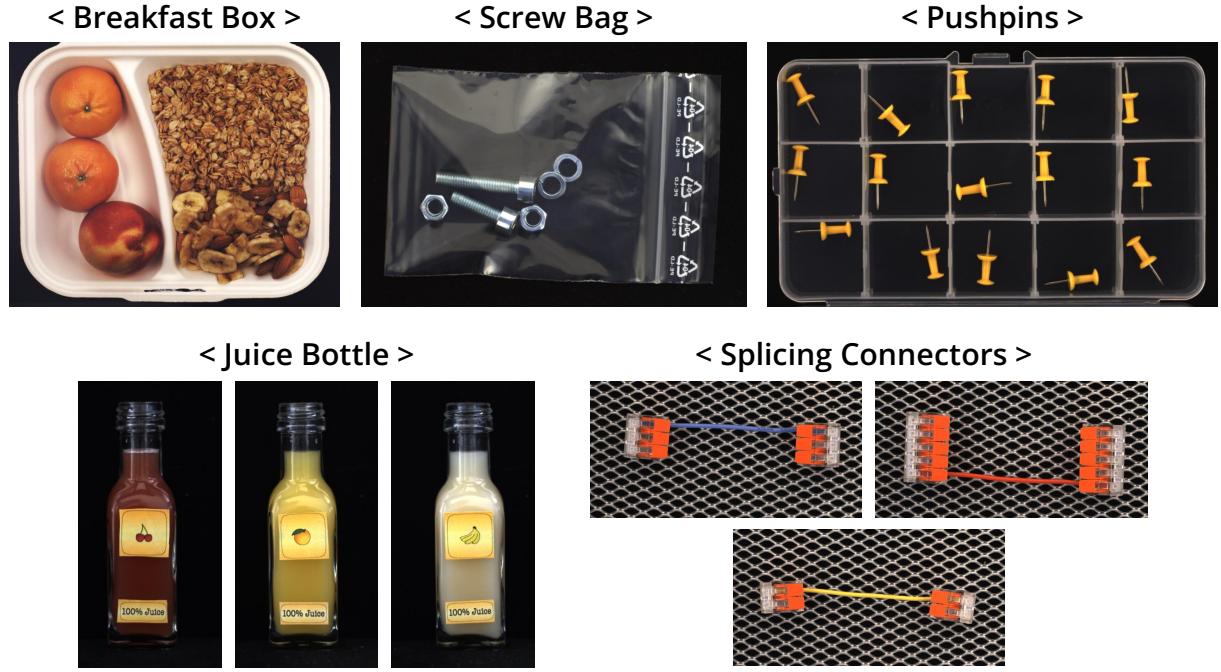


Figure 4: MVTec LOCO AD Dataset Normal sample images

C.2 MVTec LOCO AD Dataset- Normality Definition for each class

Below is a summary of the normality definitions for each class. For *Splicing Connectors* and *Juice Bottle*, the normality definitions partially change depending on the color of each cable and the fruit of the juice. The changed parts are expressed in red.

Breakfast Box

- The breakfast box always contain exactly two tangerines and one nectarine that are always located on the left-hand side of the box.
- The ratio and relative position of the cereals and the mix of banana chips and almonds on the right-hand side are fixed.

Screw Bag

- A screw bag contains exactly two washers, two nuts, one long screw, and one short screw.
- All bolts (screws) are longer than 3 times the diameter of the washer.

Pushpins

- Each compartment of the box of pushpins contains exactly one pushpin.

Splicing Connectors

- Exactly two splicing connectors with the same number of cable clamps are linked by exactly one cable.
- In addition, the number of clamps has a one-to-one correspondence to the {color} of the cable.
- The cable must be connected to the same position on both connectors to maintain mirror symmetry.
- The cable length is roughly longer than the length of the splicing connector terminal block.

Juice Bottle

- The juice bottle is filled with {fruit} juice and carries exactly two labels.
- The first label is attached to the center of the bottle, with the {fruit} icon positioned exactly at the center of the label, clearly indicating the type of {fruit} juice.
- The second is attached to the lower part of the bottle with the text “100% Juice” written on it.
- The fill level is the same for each bottle.
- The bottle is filled with at least 90% of its capacity with juice, but not 100%.

C.3 Main-Questions for each class

Breakfast Box

- Q1 : Are there exactly two tangerines visible on the left-hand side of the breakfast box?
- Q2 : Is there one nectarine visible on the left-hand side of the breakfast box?
- Q3 : Does the right-hand side of the breakfast box have cereals in the upper portion?
- Q4 : Is there a mix of banana chips and almonds in the lower portion of the right-hand side of the breakfast box?
- Q5 : Are the fruits (tangerines and nectarine) only on the left-hand side, and are the cereals with banana chips and almonds only on the right-hand side?

Screw Bag

- Q1 : Are there exactly two tangerines visible on the left-hand side of the breakfast box?
- Q2 : Is there one nectarine visible on the left-hand side of the breakfast box?
- Q3 : Does the right-hand side of the breakfast box have cereals in the upper portion?
- Q4 : Is there a mix of banana chips and almonds in the lower portion of the right-hand side of the breakfast box?
- Q5 : Are the fruits (tangerines and nectarine) only on the left-hand side, and are the cereals with banana chips and almonds only on the right-hand side?

Pushpins

- Q1 : Is there exactly one pushpin visible in the compartment?
- Q2 : Is the pushpin yellow in color?
- Q3 : Is the compartment transparent, allowing the pushpin to be visible?
- Q4 : Is the pushpin visible against a contrasting background?

Splicing Connectors - Blue

- Q1 : Are there exactly two splicing connectors visible in the image?
- Q2 : Do both connectors have the same number of wire clamps?
- Q3 : Is there only one blue cable connecting the two splicing connectors?
- Q4 : Do the connectors have transparent bodies with orange levers?
- Q5 : Do both connectors have three orange levers, indicating three cable clamps?
- Q6 : Are the connectors made from clear plastic with metal contacts inside?
- Q7 : Are the orange levers made of plastic?
- Q8 : Is the blue cable connected to the same position on both connectors?
- Q9 : Is the pushpin visible against a contrasting background?
- Q10 : Does the blue cable appear longer than the length of one of the splicing connectors?

Splicing Connectors - Red

- Q1 : Are there exactly two splicing connectors in the image?
- Q2 : Do both connectors have transparent casings with red or orange clamps/levers?
- Q3 : Are the connectors rectangular and compact, each containing five clamps?
- Q4 : Is there a single red cable connecting the two splicing connectors?
- Q5 : Is the red cable slightly longer than the length of the splicing connector terminal block?
- Q6 : Are the connectors positioned parallel to each other?
- Q7 : Are the splicing connectors transparent with orange levers?
- Q8 : Does the cable connect to the same clamp position on both connectors, maintaining mirror symmetry?
- Q9 : Are the connectors made of plastic with transparent casings?

Splicing Connectors - Yellow

- Q1 : Are there exactly two splicing connectors visible in the image?
- Q2 : Do both splicing connectors have the same number of levers?
- Q3 : Is the cable connecting the two splicing connectors yellow in color?
- Q4 : Does each connector have two levers, indicating two clamps?
- Q5 : Is the cable entering the same position on both connectors, maintaining symmetry?
- Q6 : Is the length of the yellow cable longer than the terminal block of each splicing connector?
- Q7 : Are the splicing connectors transparent with orange levers?
- Q8 : Are the connectors positioned symmetrically on either side of the yellow cable?
- Q9 : Is there exactly one yellow cable connecting the two splicing connectors?

Juice Bottle - Orange

- Q1 : Is the juice bottle filled with orange juice up to at least 90% of its capacity, but not completely full?
- Q2 : Are there exactly two labels on the juice bottle?
- Q3 : Is the center label positioned in the middle of the bottle with an orange icon clearly visible?
- Q4 : Does the center label have a light orange background?
- Q5 : Is the lower label attached to the lower part of the bottle?
- Q6 : Does the lower label display the text 100% Juice in bold, likely black, font?
- Q7 : Are the labels vertically aligned, with the center label above the lower label, creating a balanced appearance?

Juice Bottle - Cherry

- Q1 : Is the bottle made of clear glass, allowing the color of the cherry juice to be visible?
- Q2 : Does the bottle have a central label with a cherry icon precisely placed in the middle?
- Q3 : Is there a central label on the bottle with a cherry icon clearly indicating the type of juice?
- Q4 : Is there a lower label on the bottle with the text 100% Juice written on it?
- Q5 : Is the fill level of the juice in the bottle at least 90% of its capacity, with a small gap at the top indicating it is not completely full?
- Q6 : Is there a central label on the bottle with a cherry icon positioned exactly at the center of the label?
- Q7 : Is the color of the juice a deep reddish-brown, consistent with cherry juice?

Juice Bottle - Banana

- Q1 : Is the bottle made of clear glass, allowing you to see the banana juice inside?
- Q2 : Does the juice inside the bottle appear as a creamy, light yellow color, typical of banana juice?
- Q3 : Is the bottle slender and of a standard size typically used for single-serve juice bottles?
- Q4 : Is there a central label on the bottle with a banana icon located exactly at the center of the label?
- Q5 : Is there a lower label on the bottle that reads ??00% Juice?
- Q6 : Does the juice fill level reach at least 90% of the bottle's capacity, with a small gap at the top?
- Q7 : Are there exactly two labels on the bottle, one in the center and one lower down?

C.4 Sub-Questions for each class

An example of a sub-question configuration for the breakfast box class is given. The Sub-Questions can be created by applying an augmentation prompt (generating 5 variations Sub-Questions) to the Main-Questions.

Breakfast Box

Q1 Sub-Questions

- Can you see exactly two tangerines on the left side of the breakfast box?
- Is the left-hand side of the breakfast box showing precisely two tangerines?
- Do you observe exactly two tangerines on the left of the breakfast box?
- Are precisely two tangerines visible on the left side of the breakfast box?
- Does the left-hand side of the breakfast box contain exactly two tangerines?

Q2 Sub-Questions

- Can you see a single nectarine on the left side of the breakfast box?
- Is there a nectarine present on the left-hand side of the breakfast box?
- Do you spot one nectarine on the left area of the breakfast box?
- Is a nectarine visible on the left side within the breakfast box?
- Is there one nectarine that can be seen on the left part of the breakfast box?

Q3 Sub-Questions

- Are there cereals located in the upper part of the right side of the breakfast box?
- Is the upper portion of the right side of the breakfast box filled with cereals?
- Can cereals be found in the top section on the right-hand side of the breakfast box?
- Does the upper section of the right side of the breakfast box contain cereals?
- Is the top of the right-hand side of the breakfast box occupied by cereals?

Q4 Sub-Questions

- Does the lower section on the right side of the breakfast box contain a combination of banana chips and almonds?
- Can you find a blend of banana chips and almonds in the bottom part of the right-hand side of the breakfast box?
- Are banana chips and almonds mixed together in the lower right section of the breakfast box?
- Is there a combination of banana chips and almonds located in the bottom right area of the breakfast box?
- Are banana chips and almonds present together in the lower portion on the right side of the breakfast box?

Q5 Sub-Questions

- Are tangerines and nectarines exclusively on the left, and are cereals with banana chips and almonds exclusively on the right?
- Is it true that the fruits, such as tangerines and nectarines, are solely placed on the left while cereals with almonds and banana chips are only on the right?
- Are the tangerines and nectarines located only on the left side, and are the cereals containing banana chips and almonds solely on the right side?
- Are fruits like tangerines and nectarines restricted to the left-hand side, while cereals with banana chips and almonds are found only on the right?
- Is the placement such that tangerines and nectarines are just on the left, and cereals with almonds and banana chips appear only on the right?

C.5 Logical AD performance on MVTec LOCO AD dataset.

MVTec LOCO AD (only Logical Anomaly)	LogicQA (Ours)		LogicAD Jin et al. (2025)		WinCLIP Jeong et al. (2023)		PatchCore Roth et al. (2022)	GCAD Bergmann et al. (2022)	AST Rudolph et al. (2023)	LogiCode Zhang et al. (2024a)	PSAD Kim et al. (2024b)
Category	AUROC	F_1 -max	AUROC	F_1 -max	AUROC	F_1 -max	AUROC	AUROC	AUROC	AUROC	AUROC
Breakfast Box	87.6	91.6	93.1	82.7	57.6	63.3	74.8	87.0	80.0	98.8	100.0
Juice Bottle	88.2	89.6	81.6	83.2	75.1	58.2	93.9	100.0	91.6	99.4	99.1
Pushpins	98.4	97.6	98.1	98.5	54.9	57.3	63.6	97.5	65.1	98.8	100.0
Screw Bag	71.5	64.5	83.8	77.9	69.5	58.8	57.8	56.0	80.1	98.2	99.3
Splicing Connectors	92.4	91.5	73.4	76.1	64.5	59.9	79.2	89.7	81.8	98.9	91.9
Average	87.6	87.0	86.0	83.7	64.3	59.5	74.0	86.0	79.7	98.8	98.1

Table 5: (**Extension Ver.**) **Logical AD performance on MVTec LOCO AD dataset.** AUROC and F_1 -max in % for detecting logical anomalies of all categories of MVTec LOCO AD Dataset.

D Can an Anomaly Score be effectively derived from the Token Prediction Probability?

We propose using VLM’s Log Probabilities to compute an anomaly score. We assume that low token prediction probabilities (*log_probs*) lead to uncertain performance and incorrect answers, as in typical LLM studies. Therefore, we conducted additional experiments to verify whether this assumption is correct in our VLM task.

We extracted 50 normal images for each class and generated answers for each Main-Question. The VLM’s answer must be "Yes" for all normal images. Therefore, if it is "No", the answer generated by VLM is incorrect. We visualized each answer and the average token prediction probability at that time by class.

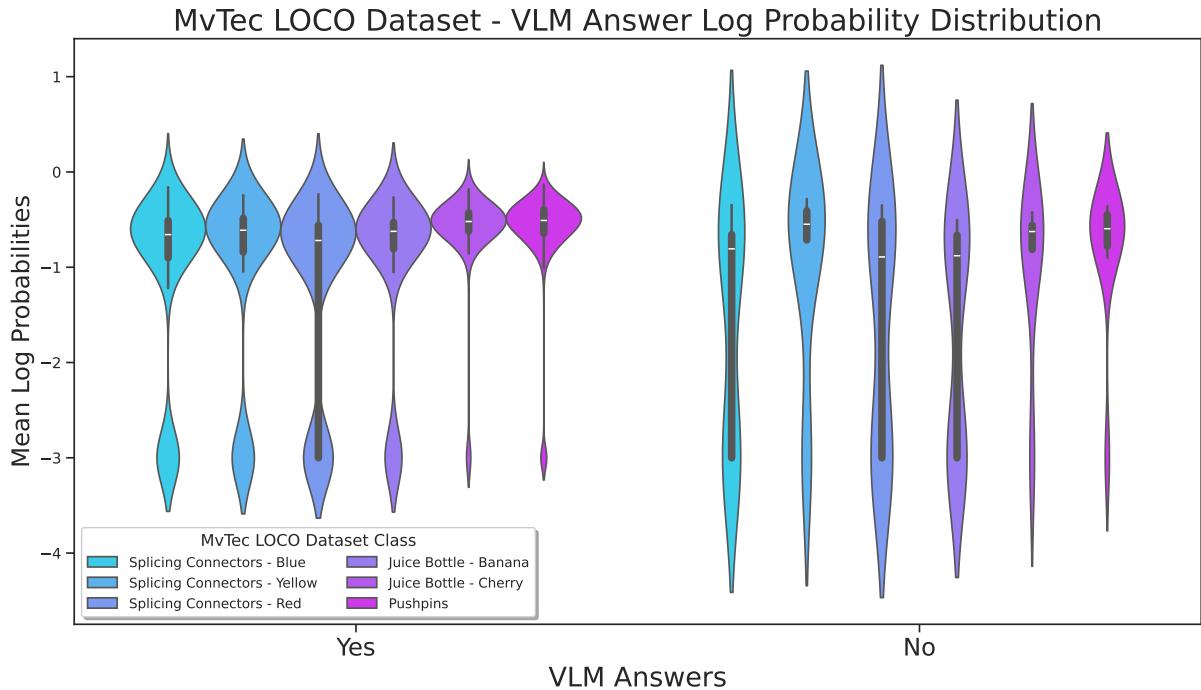


Figure 5: **Log-Probability Distribution of VLM answers**

As you can see from the figure 5, when generating the wrong answer "No" in some classes, the distribution of *log_probs* is generated relatively widely. When VLM generating "Yes", there is a clear section where the *log_probs* remains high, whereas in the case of "No", the *log_probs* come out quite diversely. Since our assumption is quite consistent with the actual data, it suggests that **as a result of verifying with actual data, it was confirmed that using the token prediction probability as the reliability of the answer and using it as the Anomaly Score is valid.**

E Semiconductor SEM Dataset

This is an overview of the Semiconductor SEM Dataset. Scanning Electron Microscopy (SEM) operates by applying a high voltage to direct an electron beam onto the surface of a sample, then detecting secondary electrons that react to this beam to generate an image. The equipment used in our experiments achieves a resolution of approximately 1 nm, making it highly effective for observing the minute patterns on wafer surfaces.

Semiconductor fabrication involves hundreds to thousands of processing steps, comprising dozens of layers. Furthermore, each layer has a distinct pattern to form integrated circuits. This indicates a wide variety of both normal and abnormal (defective) patterns, implying that a generalized anomaly detection model would require an enormously large memory bank.

There are two defect types for anomaly dataset, Spot Defect and Bridge Defect. These two types of anomaly sets share the same Normal dataset. Bridge defects occur when separate conductive lines or elements accidentally fuse, potentially causing short circuits. In contrast, spot defects appear as small, localized flaws on the wafer surface that can degrade overall device performance.

The data was provided by a global semiconductor company, and the actual data cannot be disclosed for security reasons. The sample examples below are images similar to the actual images found in the paper (Kim et al., 2020) and attached.

Type	Train-Normal Images	Test-Normal Images	Test-Logical Anomaly Images
Spot Defect			290
Bridge Defect	342	169	123
Total	342	169	413

Table 6: Overview of the Semiconductor SEM dataset

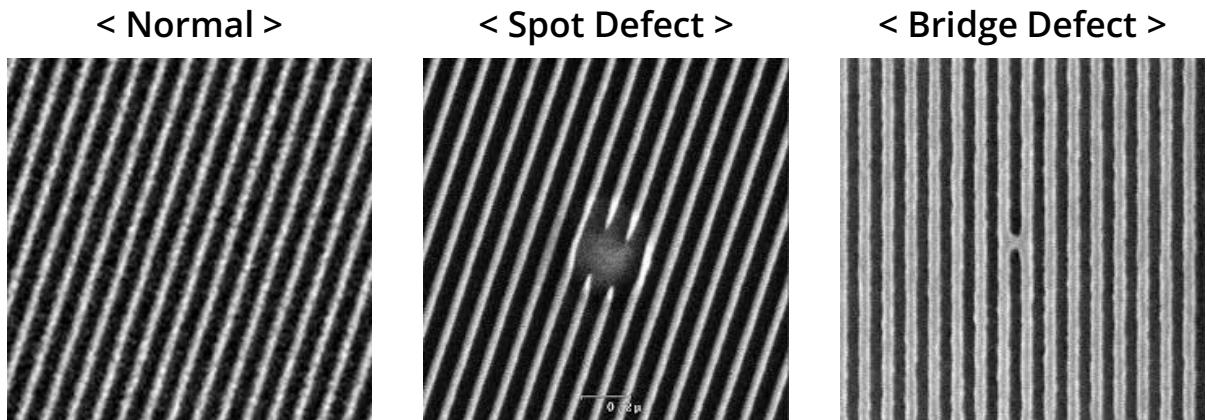


Figure 6: Semiconductor SEM Dataset sample images

E.1 Semiconductor SEM Dataset- Normality Definition

SEM wafer

- There should be no Particles, Hot Spots, or Defects.

E.2 Main-Questions

SEM wafer

- Q1 : Are there no visible particles or dust on the wafer surface?
- Q2 : Are the etched patterns consistent and evenly spaced across the image?
- Q3 : Is the surface free of bright or dark spots that look out of place?
- Q4 : Do the etched lines appear smooth and uniform without breaks or distortions?
- Q5 : Does the wafer surface look clean without any unexpected irregularities?

E.3 Sub-Questions

SEM wafer

Q1 Sub-Questions

- Is the wafer surface completely free of visible particles or dust?
- Are there any visible particles or dust present on the wafer surface?
- Can you confirm that no visible particles or dust are on the wafer surface?
- Is the wafer surface entirely clean without any visible dust or particles?
- Do you see any visible dust or particles on the wafer surface?

Q2 Sub-Questions

- Are the etched patterns uniform and evenly distributed throughout the image?
- Do the etched patterns appear consistent and evenly spaced across the entire image?
- Are the etched designs evenly spaced and consistent throughout the image?
- Is there uniformity in the etched patterns, with even spacing across the image?
- Do the etched patterns maintain consistency and equal spacing across the image?

Q3 Sub-Questions

- Does the surface have any unusual bright or dark spots?
- Are there any bright or dark spots on the surface that seem out of place?
- Is the surface completely uniform, without any irregular bright or dark spots?
- Do you notice any unexpected bright or dark spots on the surface?
- Is the surface free from any abnormal bright or dark spots?

Q4 Sub-Questions

- Are the etched lines consistently smooth and uniform, without any interruptions or distortions?
- Do the etched lines maintain a smooth and even appearance, free from breaks or irregularities?
- Are the etched lines free from distortions and interruptions, appearing smooth and uniform?
- Do the etched lines exhibit a continuous, smooth, and uniform pattern without any breaks?
- Are the etched lines well-defined, smooth, and uniform, without any visible distortions or gaps?

Q5 Sub-Questions

- Is the wafer surface free of any unexpected irregularities and appears clean?
- Does the wafer surface appear smooth and without any unwanted defects?
- Is the wafer surface visibly clean and devoid of any unexpected anomalies?
- Can you confirm that the wafer surface is clean and free from irregularities?
- Does the wafer surface exhibit a clean appearance without any noticeable defects?

F Details and Effect of BPM & Lang-SAM

The MVTec LOCO AD Dataset required image preprocessing based on class-specific features. In the Splicing Connectors class, the background consists of wire entanglement, while in the Screw Bag class, a large portion of the image is occupied by empty space within the bag. To address this, we applied **Back Patch Masking (BPM)** to these two classes. BPM isolates the foreground target from the background, enabling target-centric detection. Also, Pushpins class is uniformly placed in each compartment, and Splicing Connectors class consists of multiple identical terminals within each connector block. Since both classes exhibit the uniform objects issue that makes hallucination problem in VLM, we processed images using **Lang-SAM**.

We conducted an experiment to verify whether BPM is actually effective in improving the response accuracy of VLM. We composed a subset of 50 normal images, entered the Main-Question for each class, and checked the answer. A normal image must answer "Yes" to the Main-Questions. If it answered a "No", VLM generated an wrong answer. We calculated the correct answer rate (accuracy) for each Main-Question for a total of 50 normal images. As you can see in the figure 7 below, **the accuracy of the answer increases when BPM is processed compared to when it is not.**

We also experimented to verify whether Lang-SAM is effective for VLM performance. We conducted an experiment with the same settings as the previous BPM additional experiment. As shown in figure 7, we found that **Lang-SAM was significantly effective in improving the accuracy of VLM answers in both classes (Pushpins and Splicing Connectors).**

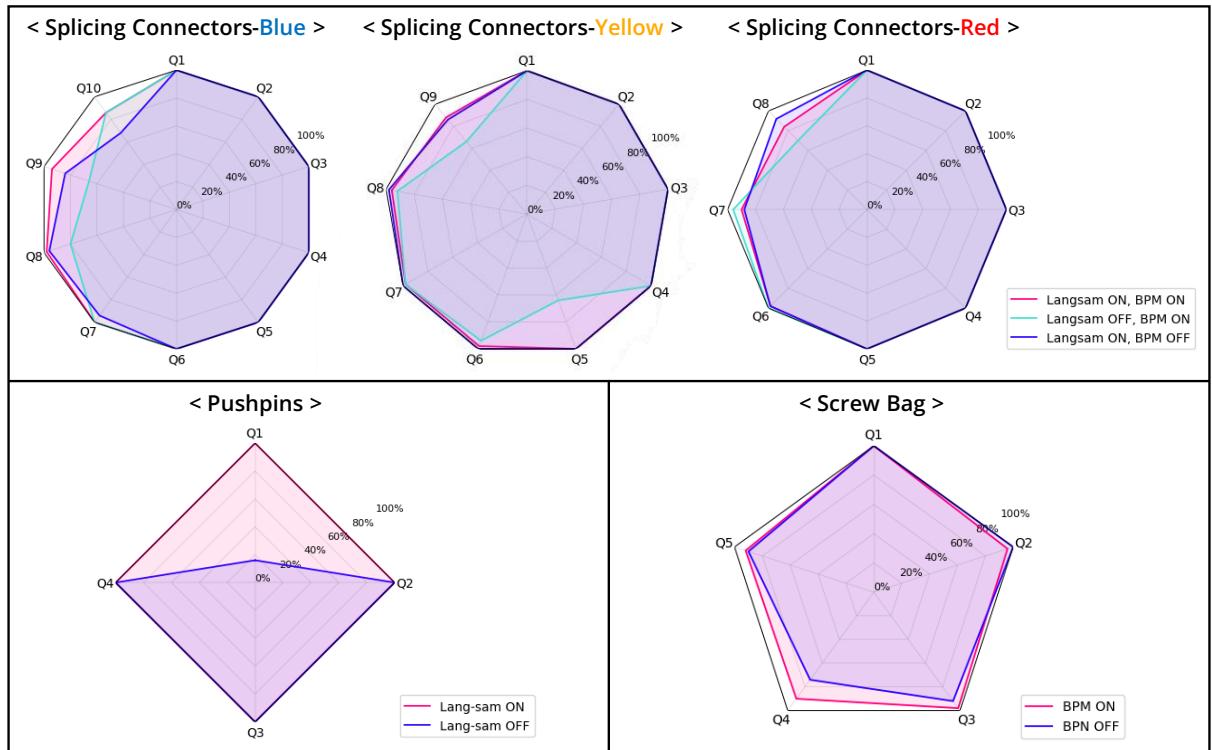


Figure 7: **BPM and Lang-SAM Effect for each class**