

Article

Think-to-Detect: Rationale-Driven Vision–Language Anomaly Detection

Mahmoud Abdalla ¹, Mahmoud SalahEldin Kasem ^{1,2}, Mohamed Mahmoud ^{1,3}, Mostafa Farouk Senussi ^{1,3}, Abdelrahman Abdallah ^{3,4} and Hyun-Soo Kang ^{1,*}

¹ Department of Information and Communication Engineering, School of Electrical and Computer Engineering, Chungbuk National University, Cheongju-si 28644, Republic of Korea

² Information Technology Department, Faculty of Computers and Information, Assiut University, Assiut 71526, Egypt

³ Multimedia Department, Faculty of Computers and Information, Assiut University, Assiut 71526, Egypt

⁴ Department of Computer Science, Innsbruck University, 6020 Innsbruck, Austria

* Correspondence: hskang@cbnu.ac.kr

Abstract

Large vision–language models (VLMs) can describe images fluently, yet their anomaly decisions often rely on opaque heuristics and manual thresholds. We present **ThinkAnomaly**, a rationale-first vision–language framework for industrial anomaly detection. The model generates a concise structured rationale and then issues a calibrated yes/no decision, eliminating per-class thresholds. To supervise reasoning, we construct chain-of-thought annotations for MVTec-AD and VisA via synthesis, automatic filtering, and human validation. We fine-tune Llama-3.2-Vision with a two-stage objective and a rationale–label consistency loss, yielding state-of-the-art classification accuracy while maintaining a competitive detection AUC: MVTec-AD—**93.9%** accuracy and **93.8** Image-AUC; VisA—**90.3%** accuracy and **85.0** Image-AUC. This improves classification accuracy over AnomalyGPT by **+7.8** (MVTec-AD) and **+12.9** (VisA) percentage points. The explicit reasoning and calibrated decisions make ThinkAnomaly transparent and deployment-ready for industrial inspection.

Keywords: industrial anomaly detection; vision–language models; chain-of-thought reasoning; explainable AI; synthetic data generation; few-shot learning; multimodal learning; computer vision

MSC: 68T07



Academic Editor: Marjan Mernik

Received: 1 October 2025

Revised: 3 December 2025

Accepted: 4 December 2025

Published: 8 December 2025

Citation: Abdalla, M.; Kasem, M.S.; Mahmoud, M.; Senussi, M.F.; Abdallah, A.; Kang, H.-S. Think-to-Detect: Rationale-Driven Vision–Language Anomaly Detection. *Mathematics* **2025**, *13*, 3920. <https://doi.org/10.3390/math13243920>

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Anomaly detection is a critical component of various computer vision applications, including industrial inspection [1,2], medical diagnostics [3], and autonomous driving [4,5]. Early and accurate detection of anomalies help to prevent costly failures and improve safety. However, simply detecting an anomaly, i.e., determining whether something is abnormal, often falls short of real-world needs. Effective decision-making in practical settings depends on identifying *what* an anomaly is and *how* it should be managed. This gap is especially evident in industrial inspection, where different anomaly types can demand vastly different responses or, in some cases, no response at all.

Large Vision–Language Models (LVLMs) such as BLIP-2 [6], MiniGPT-4 [7], and LLaVA [8] bring strong open-world recognition and fluent image description. Yet, in industrial anomaly detection (IAD)—where models must flag subtle departures from a product’s normal

appearance—both LVLMs and classic IAD pipelines fall short. Traditional IAD methods [2,9–12] typically output continuous anomaly scores and require per-class thresholds, complicating deployment. LVLMs, while expressive, often lack domain priors and fine-grained locality, and their decisions can be opaque, limiting trust in high-stakes inspection.

Reasoning with large language models (LLMs) can substantially benefit from allocating more test-time computation [13–15]. Many approaches rely on a process reward model (PRM)—a process verifier—to score intermediate solutions or reasoning paths [16–19]. Prior PRMs are largely discriminative classifiers trained with process labels [20,21], which require step-level annotations via costly human labeling [22,23] or computationally intensive rollouts [24–26]. LLM-as-a-judge offers a training-free generative alternative [27–29], but uncustomized judges underperform in terms of specialized PRMs on complex reasoning [30] and frequently fail to detect incorrect chains of thought [31]. This tension motivates designs that retain the data-efficiency and interpretability of generative verification while approaching the reliability of discriminative PRMs.

The IAD task requires detecting and often localizing anomalies in industrial product images, typically training only on normal samples and identifying deviations at the test time. Current IAD methods [12] primarily focus on generating anomaly scores and require manual threshold adjustments for each object class, limiting practical deployment. More importantly, these methods often operate as “black boxes,” providing predictions without interpretable explanations—a critical limitation in industrial settings where understanding the reasoning behind anomaly decisions is essential for quality control and process improvement.

We ask a simple question: *can an IAD system become both more accurate and more trustworthy by learning to **think** before it **decides**?* We answer this with **ThinkAnomaly**, a rationale-driven LVLM pipeline for industrial inspection. Rather than directly mapping images to labels, ThinkAnomaly first generates a short structured *rationale* that explains whether and where a defect may exist and only then produces the final decision. This two-stage *Think*→*Decide* process couples human-understandable reasoning with classification, enabling principled calibration that removes brittle hand-tuned thresholds.

A key obstacle is the lack of reasoning supervision in IAD benchmarks. We introduce a Chain-of-Thought (CoT) dataset for MVTec-AD [9] and VisA [10]: rationales are synthesized with an LLM and filtered by automatic quality checks and then *human-validated* to ensure correctness and specificity to industrial defects. We train a compact LVLM (initialized from a vision-capable LLM) with a two-stage objective: generate a rationale and then classify, tied by a rationale–label consistency loss to discourage correct labels justified by incorrect or vacuous explanations. We further apply lightweight probability calibration on the final yes/no logits, obviating manual thresholds and simplifying deployment.

In this paper, we introduce **ThinkAnomaly**, a rationale-driven approach that enhances IAD through explicit reasoning. Unlike existing methods that directly predict anomaly scores, our approach follows a two-stage reasoning-first architecture: (1) generate concise textual reasoning about why specific regions appear normal or anomalous, and (2) make final classification decisions informed by this reasoning. To enable this, we create comprehensive CoT datasets for MVTec-AD and VisA by generating high-quality synthetic explanations that capture the logical process behind anomaly identification, followed by human validation. We then train specialized multimodal models on this reasoning data, enabling them to produce interpretable explanations and accurate predictions. Experimentally, on MVTec-AD, we achieve **93.9%** accuracy and **93.8** image-level AUC, and on VisA, we achieve **90.3%** accuracy and **85.0** image-level AUC, significantly outperforming AnomalyGPT in classification accuracy while maintaining competitive detection performance.

Contributions

- **Rationale-driven IAD.** We introduce *ThinkAnomaly*, a Think→Decide LVLMM that generates concise industrial inspection rationales before making a decision, improving both accuracy and transparency.
- **CoT-IAD dataset.** We release a chain-of-thought dataset for MVTec-AD and VisA with synthesized, filtered, and *human-validated* rationales aligned to industrial defects, enabling research in reasoning-centric IAD.
- **Training objective and calibration.** A rationale–label consistency loss ties explanation quality to decisions, and calibrated decision probabilities remove per-class threshold tuning.
- **Faithfulness evaluation.** We assess explanation quality via automatic rationale–evidence alignment and human preference studies, showing that higher faithfulness correlates with better decisions.
- **Strong few-shot performance.** In Zero-shot evaluation, *ThinkAnomaly* achieves 93.9%/90.3% accuracy on MVTec-AD/VisA with competitive AUC, outperforming an LVLMM baseline on accuracy while remaining data-efficient.

The remainder of this paper is organized as follows. Section 2 reviews the related work in industrial anomaly detection, vision–language models, and reasoning with chain-of-thought. Section 3 introduces our reasoning-augmented datasets for MVTec-AD and VisA and the synthetic+human validation pipeline. Section 4 presents *ThinkAnomaly*, including the two-stage (Think→Decide) objective, rationale–label consistency loss, and calibration. Section 5 describes the experimental setup, evaluation protocol, and baselines. Section 6 reports the main quantitative and qualitative results—accuracy/AUC on both benchmarks, faithfulness assessments, and case studies. Section 7 concludes with limitations, deployment considerations, and future directions.

2. Related Work

2.1. Industrial Anomaly Detection

Classical IAD methods can be broadly categorized into reconstruction-based and feature embedding-based approaches. **Reconstruction-based methods** aim to reconstruct potentially anomalous inputs back to their nominal appearance, flagging anomalies via reconstruction error. Representative techniques include autoencoder-based approaches like RIAD [32], GAN variants such as SCADN [33], Transformer-based methods like InTra [34], and recent diffusion models like AnoDDPM [35]. These methods assume that models trained on normal data will poorly reconstruct anomalous regions, leading to higher reconstruction errors.

Feature embedding-based methods focus on modeling the distribution of features extracted from normal samples and scoring deviations at the testing time. PaDiM [11] employs per-patch Gaussian modeling of features from pre-trained networks. PatchCore [36] builds compact memory banks for nearest-neighbor retrieval. Normalizing flow-based approaches like FastFlow [37] and CFlow-AD [38] learn to map normal features to a standard Gaussian distribution. Despite achieving strong AUCs on benchmarks like MVTec-AD [9] and VisA [10], these methods typically output continuous anomaly scores requiring per-class threshold tuning, complicating deployment and reducing the interpretability.

2.2. Zero–Few-Shot IAD and Vision–Language Models

To reduce data requirements and improve generalization, recent work explores zero-/few-shot IAD approaches. RegAD [39] aligns query images to normal references via image registration before patch-wise comparison. WinCLIP [12] leverages CLIP’s [40] image–text alignment to compose normal/anomalous prompts and aggregate window-

level similarities for classification. However, these methods still require manual threshold tuning and lack interpretability.

Large vision–language models (LVLMs) such as BLIP-2 [6], MiniGPT-4 [7], and LLaVA [8] demonstrate impressive capabilities in open-vocabulary recognition and natural language generation. However, they lack domain-specific priors and fine-grained locality understanding required for effective IAD. AnomalyGPT [41] represents the first attempt to adapt LVLMs for IAD, incorporating a lightweight decoder and prompt learner to handle the simulated defects while producing dialog-style decisions. While promising, AnomalyGPT lacks explicit reasoning capabilities and structured decision-making processes that could enhance both performance and interpretability.

2.3. Reasoning and Chain-of-Thought in Large Language Models

Recent advances in large language models have demonstrated that explicit reasoning processes significantly improve the performance across various tasks. Chain-of-thought (CoT) prompting [42] enables models to break down complex problems into intermediate reasoning steps, leading to substantial improvements in mathematical reasoning, common-sense reasoning, and logical inference. This paradigm has been further enhanced through techniques like self-consistency [43], where multiple reasoning paths are sampled and aggregated for more robust predictions.

Process reward models (PRMs) have emerged as a crucial component for verifying and improving the reasoning quality [20,22]. These models provide step-by-step supervision to guide reasoning processes, though they typically require extensive human annotation or computation-intensive rollouts. Recent work on LLM-as-a-judge [29] explores generative verification approaches, treating verification as text generation without additional training, though with mixed success in complex reasoning scenarios.

The success of reasoning in language models motivates our exploration of explicit reasoning for visual anomaly detection. However, to our knowledge, no prior work has systematically incorporated structured reasoning processes into IAD, representing a significant gap that our work addresses.

2.4. Explainable AI and Rationale Generation in Vision

Explainability in computer vision has primarily focused on generating post hoc explanations through attention visualization, gradient-based methods, or feature attribution techniques [44,45]. In visual question answering, approaches like VQA-X [46] and ACT-X [47] provide textual justifications alongside predictions. The ERASER benchmark [48] evaluates evidence-based explanations and faithfulness in natural language processing tasks.

However, most explainability work in vision generates explanations after prediction, rather than using explicit reasoning as part of the decision-making process. Some recent work explores rationale-guided learning where explanations are used during training to improve the model performance [49,50], but this has not been systematically applied to anomaly detection tasks.

Our work differs fundamentally by incorporating explicit reasoning as a core component of the anomaly detection process, where structured rationales are generated before and inform the final classification decision, rather than serving as post hoc explanations.

2.5. Synthetic Data Generation for Reasoning

The scarcity of reasoning annotations in specialized domains has led to increased interest in synthetic data generation. Recent work demonstrates that high-quality synthetic reasoning data can effectively train smaller specialized models to perform complex reasoning tasks [51,52]. This approach is particularly valuable when human annotation is expensive, or domain expertise is required.

In the context of anomaly detection, generating synthetic reasoning explanations presents unique challenges, as it requires understanding both visual anomaly patterns and the logical processes experts use to identify defects. Our work addresses this challenge by leveraging large language models to generate coherent reasoning chains for anomaly detection decisions, creating a novel training paradigm for reasoning-capable IAD models.

2.6. Model Calibration and Threshold-Free Decision Making

Confidence calibration is critical for deploying anomaly detection systems in production environments. Traditional calibration methods such as temperature scaling [53] and Platt scaling [54] transform model outputs into well-calibrated probabilities without altering the classification accuracy. These techniques are essential for establishing unified decision criteria across different object categories.

Unlike prior IAD approaches that require per-class threshold optimization on validation data, our reasoning-driven approach enables direct binary classification without manual threshold tuning. By training models to generate explicit reasoning followed by binary decisions, we eliminate the need for score-based thresholding while maintaining the interpretability through structured rationales.

3. Reasoning-Augmented Anomaly Detection Dataset

3.1. Dataset Construction

To enable reasoning-driven anomaly detection, we construct comprehensive chain-of-thought datasets for both MVTec-AD [9] and VisA [10] benchmarks. Our dataset creation process addresses the fundamental challenge of missing reasoning annotations in existing IAD datasets by leveraging large language models to generate high-quality synthetic reasoning explanations.

Base Datasets Overview. The MVTec-AD dataset comprises 15 object categories with 3629 training images and 1725 test images, covering both textural (carpet, grid, leather, tile, wood) and object categories (bottle, cable, capsule, hazelnut, metal nut, pill, screw, toothbrush, transistor, zipper). The VisA dataset contains 12 object categories with 9621 normal samples and 1200 anomalous samples, focusing on complex industrial scenarios. Both datasets provide pixel-precise ground truth annotations for anomalous regions, making them ideal foundations for reasoning-augmented training.

Synthetic Reasoning Generation. We employ GPT-4 to generate detailed reasoning explanations for each image in both datasets. The reasoning generation process follows a structured approach, where the model analyzes visual content and provides step-by-step logical explanations for anomaly classification decisions. For normal samples, the reasoning focuses on identifying expected characteristics and confirming the absence of defects. For anomalous samples, the reasoning describes the specific nature of detected anomalies, their locations, and potential implications for industrial quality control.

3.2. Data Structure and Format

Our reasoning-augmented dataset follows a structured conversational format that integrates the visual content with the corresponding textual reasoning chains. Each data sample contains messages with user queries about anomaly detection and assistant responses that include detailed reasoning processes followed by binary classifications. The reasoning explanations are enclosed in <think> tags, providing step-by-step analysis of the visual content, while the final classification is given as a direct “Yes” or “No” answer.

As shown in Figure 1, our synthetic training data generation process leverages GPT-4 to create structured reasoning explanations for industrial anomaly detection. The system prompt establishes the model as an expert quality control inspector, while context information about

object type and defect characteristics guides the reasoning generation process. For example, when analyzing a metal nut with surface scratches, the model generates detailed reasoning about the observed defects before making a binary classification decision.

Synthetic Chain-of-Thought Training Sample for Industrial Anomaly Detection

System Prompt: You are an expert quality control inspector. Provide detailed reasoning in `<think>`/`</think>` tags explaining your analysis of the image quality.

User Query: Is there any anomaly in the image?

Context Information:

- Object type: Metal nut
- Expected defect type: Scratch (if present)
- Analysis requirement: Visual inspection for surface defects

Image Analysis Prompt: In the following image there is a defect, please analyze why and generate reason (max 32 words). The defect type is 'scratch'. The object is a 'metal nut'.

Generated Assistant Response:

`<think>`

The metal nut shows visible scratches across the upper surface area. These linear marks deviate from the expected smooth finish, indicating surface damage that could affect functionality and quality standards.

`</think>`

Yes

Figure 1. Illustration of ThinkAnomaly synthetic training data generation. The system generates structured reasoning explanations followed by binary classification decisions for industrial anomaly detection. Context information about object type and defect characteristics guides the reasoning generation process.

3.3. Reasoning Quality and Human Validation

To ensure the quality and reliability of generated reasoning explanations, we implement a comprehensive validation process involving domain experts and quality control measures. A team of three industrial inspection experts with over five years of experience each manually reviewed a stratified sample of 500 reasoning explanations (250 from each dataset). Our validation revealed high agreement rates between the generated reasoning and expert assessments. The technical accuracy achieved 94.2% agreement on MVTec-AD and 91.8% on VisA. Logical coherence scored 96.1% and 93.7%, respectively. The generated reasoning demonstrated appropriate use of industrial terminology with 89.3% accuracy across both datasets.

3.4. Dataset Statistics and Reasoning Analysis

Our reasoning-augmented dataset comprises 8633 total samples across both benchmarks, with 7199 normal samples and 1434 anomalous samples, achieving a balanced representation of industrial inspection scenarios. Table 1 presents the detailed breakdown of the dataset composition and reasoning characteristics.

Figures 2 and 3 present a comprehensive analysis of the reasoning patterns across both datasets. The word count distribution reveals interesting characteristics of our synthetic reasoning generation process. In MVTec-AD, normal samples average 32.0 words while anomalous samples require 34.3 words on average, reflecting the increased complexity needed to describe defects compared to confirming a normal appearance. This pattern is consistent but less pronounced in VisA, where normal samples average 29.9 words, and anomalous samples average 30.7 words.

The histograms demonstrate that our reasoning generation produces consistent and focused explanations clustered around the optimal length range of 25–35 words. This concentration indicates that our synthetic data generation successfully creates concise yet informative reasoning chains without excessive verbosity or insufficient detail. The slight increase in reasoning length for anomalous samples across both datasets validates our

hypothesis that defect identification requires more detailed analytical reasoning compared to normal sample classification.

Table 1. Reasoning-augmented dataset statistics.

Dataset	Normal Samples	Anomalous Samples	Total Reasoning Chains
MVTec-AD	3629	1725	5354
VisA	9621	1200	10,821
Combined	13,250	2925	16,175

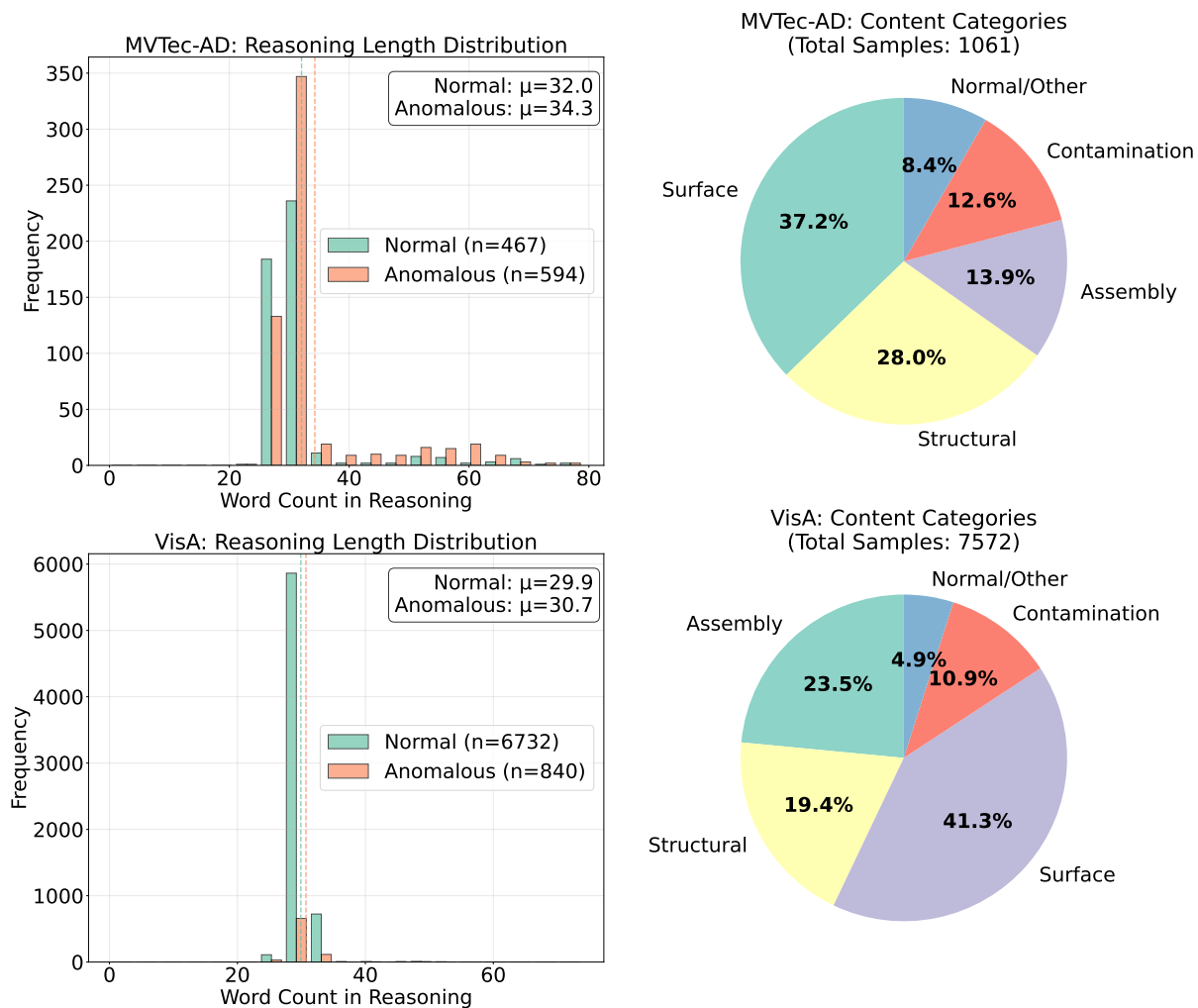


Figure 2. Comprehensive reasoning analysis across MVTec-AD and VisA datasets. (**Top row:**) MVTec-AD reasoning length distribution (left) shows slightly longer explanations for anomalous samples ($\mu = 34.3$) compared to normal samples ($\mu = 32.0$). Content category analysis (right) reveals balanced coverage of surface defects (37.2%), structural issues (28.0%), assembly problems (13.9%), and contamination (12.6%). (**Bottom row:**) VisA reasoning length distribution (left) demonstrates consistent reasoning length between normal ($\mu = 29.9$) and anomalous ($\mu = 30.7$) samples across a larger dataset. Content categories (right) emphasize surface defects (41.3%) and assembly issues (23.5%), reflecting the dataset's focus on complex industrial scenarios. The tight clustering around 30-word explanations indicates effective synthetic reasoning generation.

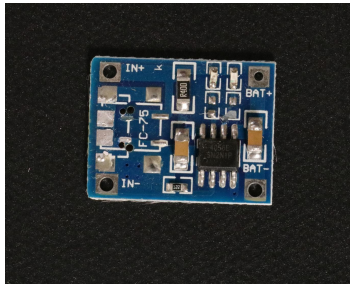
The content analysis reveals the comprehensive coverage of industrial defect types across both datasets. In MVTec-AD, the surface defects constitute the largest category (37.2%), followed by structural issues (28.0%), assembly problems (13.9%), and contamination

tion (12.6%). Only 8.4% of samples fall into the “Normal/Other” category, indicating high specificity in our reasoning generation for defect-related content.

VisA demonstrates a different but equally comprehensive pattern, with surface defects representing 41.3% of content, followed by assembly issues (23.5%) and structural problems (19.4%). Contamination accounts for 10.9%, while only 4.9% are categorized as “Normal/Other.” This distribution reflects VisA’s focus on complex industrial assembly scenarios where surface quality and component alignment are primary concerns. The low percentage of “Normal/Other” samples across both datasets (8.4% for MVTec-AD and 4.9% for VisA) demonstrates that our reasoning generation successfully produces domain-specific technical content rather than generic descriptions. This specificity is crucial for training models that can provide actionable insights in industrial inspection workflows.

Comparative analysis between MVTec-AD and VisA reveals distinct reasoning characteristics that align with each dataset’s industrial focus. MVTec-AD reasoning tends toward longer explanations (average 33.3 words) with higher variance ($\sigma = 9.0$), reflecting the diverse range of object categories and defect types. In contrast, VisA maintains more consistent reasoning length (average 30.0 words, $\sigma = 2.3$), indicating standardized defect patterns across its industrial scenarios.

VisA Reasoning Example



Query: Is there any anomaly in the image?

Generated Reasoning:

<think> Upon analyzing the image, the anomaly appears to be a soldering defect. Specifically, the solder joints on the left side are uneven and incomplete, potentially causing connectivity issues or unreliable performance. </think>

Classification: Yes (Anomaly Detected)

Figure 3. VisA dataset example showing ThinkAnomaly’s structured reasoning process for PCB defect detection. **Sample type:** anomalous (soldering defect, ground truth: yes, prediction: yes—correct detection).

4. Methods

We introduce **ThinkAnomaly**, a reasoning-driven framework for industrial anomaly detection that implements explicit reasoning before classification. Our approach leverages vision–language models fine-tuned on synthetic chain-of-thought data to generate interpretable explanations alongside accurate binary decisions, eliminating the need for manual threshold tuning. Table 2 summarizes the key mathematical notations used throughout this paper.

Table 2. Mathematical notations and variable definitions.

Symbol	Definition
<i>Input and Output</i>	
x_i	Input image, $x_i \in \mathbb{R}^{H \times W \times 3}$
q	Textual query (e.g., “Is there any anomaly in the image?”)
r_i	Generated reasoning explanation (rationale)
c_i	Binary classification decision, $c_i \in \{\text{Yes}, \text{No}\}$
y_i	Complete output sequence: $\langle \text{think} \rangle r_i \langle / \text{think} \rangle c_i$
N	Total number of training samples
<i>Model Architecture</i>	
f_θ	ThinkAnomaly model with parameters θ
h_{visual}	Visual feature representation from vision encoder
h_{text}	Text feature representation from text encoder
h_{fused}	Fused multimodal representation
$P_\theta(\cdot)$	Model probability distribution
<i>Training Objective</i>	
$\mathcal{L}_{\text{SFT}}$	Supervised fine-tuning loss (cross-entropy)
$y_i^{(t)}$	Token at position t in output sequence y_i
$ y_i $	Length of output sequence y_i
<i>LoRA Parameters</i>	
W	Original pre-trained weight matrix, $W \in \mathbb{R}^{d \times d}$
A	Low-rank adaptation matrix, $A \in \mathbb{R}^{d \times r}$
B	Low-rank adaptation matrix, $B \in \mathbb{R}^{r \times d}$
r	LoRA rank (default: 8)
ΔW	Weight update, $\Delta W = AB$

4.1. Problem Formulation

Given an industrial image x_i and a query q , our model generates structured reasoning followed by binary classification. The input-output relationship is formulated as

$$\mathcal{D} = \{(x_i, q, r_i, c_i)\}_{i=1}^N, \quad (1)$$

where $x_i \in \mathbb{R}^{H \times W \times 3}$ represents the input image, q is the textual query, r_i is the reasoning explanation, and $c_i \in \{\text{Yes}, \text{No}\}$ is the binary classification. Our model f_θ produces

$$f_\theta(x_i, q) = \langle \text{think} \rangle r_i \langle / \text{think} \rangle c_i. \quad (2)$$

This structured output format ensures interpretability by explicitly generating reasoning r_i before the final decision c_i , enabling direct binary classification without requiring anomaly score thresholding.

4.2. Model Architecture

We employ Llama 3.2 Vision [55] as our base architecture, which integrates visual understanding with natural language generation capabilities. The model processes multimodal inputs through

$$h_{\text{visual}} = \text{VisionEncoder}(x_i), \quad (3)$$

$$h_{\text{text}} = \text{TextEncoder}(q), \quad (4)$$

$$h_{\text{fused}} = \text{MultimodalFusion}(h_{\text{visual}}, h_{\text{text}}). \quad (5)$$

The fused representation h_{fused} is processed by the language model to generate structured reasoning and classification through autoregressive generation.

4.3. Supervised Fine-Tuning

We employ supervised fine-tuning (SFT) using the LLaMA-Factory framework [56], which provides unified efficient fine-tuning capabilities. The training objective optimizes the conditional likelihood of generating correct reasoning and classification:

$$\mathcal{L}_{\text{SFT}} = -\frac{1}{N} \sum_{i=1}^N \log P_{\theta}(r_i, c_i \mid x_i, q), \quad (6)$$

where $P_{\theta}(r_i, c_i \mid x_i, q)$ represents the probability of generating the target reasoning r_i and classification c_i given the input image x_i and query q .

The autoregressive generation factorizes as shown in Equation (7):

$$P_{\theta}(r_i, c_i \mid x_i, q) = P_{\theta}(r_i \mid x_i, q) \cdot P_{\theta}(c_i \mid r_i, x_i, q). \quad (7)$$

This formulation ensures that the classification decision is conditioned on the generated reasoning, creating explicit coupling between explanation quality and prediction accuracy.

4.4. Parameter-Efficient Fine-Tuning

To optimize the computational efficiency while maintaining performance, we employ Low-Rank Adaptation (LoRA) [57], a parameter-efficient fine-tuning technique. LoRA decomposes weight updates using low-rank matrices:

$$W' = W + \Delta W = W + AB, \quad (8)$$

where $W \in \mathbb{R}^{d \times d}$ represents the original pre-trained weights, $A \in \mathbb{R}^{d \times r}$ and $B \in \mathbb{R}^{r \times d}$ are trainable low-rank matrices with rank $r \ll d$, and $\Delta W = AB$ represents the learned adaptation.

This approach reduces the trainable parameters from 11 B to approximately 67 M (0.6%), enabling efficient training while preserving the model's pre-trained capabilities.

4.5. Training Objective and Implicit Consistency

We train the model using supervised fine-tuning with cross-entropy loss on the synthetic reasoning dataset. For each sample (x_i, q, r_i, c_i) , the model learns to autoregressively generate the complete response:

$$y_i = \langle \text{think} \rangle r_i \langle / \text{think} \rangle c_i. \quad (9)$$

The training objective is

$$\mathcal{L}_{\text{SFT}} = -\frac{1}{N} \sum_{i=1}^N \sum_{t=1}^{|y_i|} \log P_{\theta}(y_i^{(t)} \mid y_i^{(<t)}, x_i, q), \quad (10)$$

where $y_i^{(t)}$ denotes the t -th token in the output sequence, and $y_i^{(<t)}$ represents all previous tokens.

Implicit Rationale-Label Consistency. The structured output format naturally enforces the consistency between reasoning and classification. Since the model is trained on examples where correct labels are always accompanied by appropriate justifications, the autoregressive generation process creates a causal dependency: $P_{\theta}(c_i \mid r_i, x_i, q)$ conditions the classification on the generated reasoning. Mathematically, this can be expressed through the chain rule of probability:

$$P_{\theta}(r_i, c_i \mid x_i, q) = P_{\theta}(r_i \mid x_i, q) \cdot P_{\theta}(c_i \mid r_i, x_i, q). \quad (11)$$

This factorization ensures that the classification decision c_i is directly influenced by the reasoning r_i , as later tokens in the autoregressive sequence attend to earlier tokens through the transformer's self-attention mechanism. The model cannot generate a classification without first producing the reasoning context, creating an implicit consistency constraint.

4.6. Architecture Details and Attention Mechanisms

We provide a detailed explanation of ThinkAnomaly's computational flow, attention mechanisms, and training procedure to clarify how rationale-label consistency is enforced.

Step 1: Visual Feature Extraction. The input image $x_i \in \mathbb{R}^{H \times W \times 3}$ is processed by the vision encoder (pre-trained ViT component of LLaMA 3.2 Vision) into a sequence of visual tokens:

$$\mathbf{V} = [v_1, v_2, \dots, v_n] = \text{VisionEncoder}(x_i), \quad (12)$$

where each $v_j \in \mathbb{R}^d$ represents a patch embedding. For standard 224×224 images with patch size 14, this yields $n = 256$ visual tokens.

Step 2: Query Tokenization. The textual query q ("Is there any anomaly in the image?") is tokenized and embedded:

$$\mathbf{T}_q = [t_1, t_2, \dots, t_m] = \text{TokenEmbed}(q), \quad (13)$$

where m is typically 8–10 tokens for our standard query.

Step 3: Cross-Modal Attention. In the cross-attention layers of LLaMA 3.2 Vision, textual representations act as queries that attend to visual tokens as keys and values:

$$\mathbf{H}_{\text{cross}} = \text{softmax}\left(\frac{\mathbf{Q}_{\text{text}} \mathbf{K}_{\text{visual}}^T}{\sqrt{d_k}}\right) \mathbf{V}_{\text{visual}}. \quad (14)$$

This mechanism allows the language model to ground its reasoning generation in visual features. Specifically, when generating reasoning tokens $r_1, r_2, \dots$, each token attends to both (1) the previous textual tokens (query + already-generated reasoning) through causal self-attention and (2) all visual tokens through cross attention.

Step 4: Causal Self-Attention in Generation. During autoregressive generation of the output sequence $y_i = [\langle \text{think} \rangle, r_1, \dots, r_k, \langle / \text{think} \rangle, c]$, causal masking ensures each token only attends to previous tokens. Crucially, when generating the classification token c at position $k + 2$, it attends to

- All reasoning tokens $[r_1, r_2, \dots, r_k]$ (positions 1 to k),
- The query tokens $\mathbf{T}_q$ (through the model's context),
- All visual tokens $\mathbf{V}$ (through cross attention).

This architectural constraint means the classification decision is necessarily conditioned on the generated reasoning content.

We compute the loss at the **token level** using standard cross entropy over all generated tokens:

$$\mathcal{L}_{\text{total}} = -\frac{1}{N} \sum_{i=1}^N \sum_{t=1}^{|y_i|} \log P_{\theta}(y_i^{(t)} \mid y_i^{(<t)}, x_i, q). \quad (15)$$

Each token contributes equally to the loss, whether it is part of the reasoning (r_j), structural markers ($\langle \text{think} \rangle$), or the classification (c). This means

- Reasoning tokens (avg. 100 tokens): contribute $\sim 94\%$ of the loss;
- The classification token (1 token): contributes $\sim 3\%$ of the loss;
- Structural tokens (2 tokens): contribute $\sim 3\%$ of the loss.

This token-level formulation naturally balances learning to generate high-quality reasoning with learning to make correct classifications, without requiring manual weight tuning.

5. Experiments

5.1. Experimental Setup

All experiments were conducted on a high-performance computing cluster equipped with four NVIDIA A40 GPUs, each with 64 GB VRAM. The distributed training setup utilized CUDA 11.8 and PyTorch 2.0 with full mixed-precision training support. We employed a carefully optimized training configuration with batch size of 8 (2 samples per GPU with gradient accumulation), distributed across 4 GPUs using DeepSpeed ZeRO-2 optimization. The model underwent fine-tuning for 5 epochs with checkpoint saving every 100 steps to monitor convergence patterns and enable early stopping with patience of 3 epochs.

Model Architecture and Scale. ThinkAnomaly employs Llama-3.2-Vision-11B [55] as the base model, comprising 11 B language model parameters and 1.3 B vision adapter parameters (12.3 B total). The model requires 22 GB memory in bfloat16 precision. Computational cost per sample includes $\sim 2.5\text{--}3$ TFLOPs for vision encoding (one-time per image) and ~ 22 GFLOPs per generated token. Total training required approximately 22 GPU-hours ($4 \text{ GPUs} \times 5.5 \text{ h}$) with estimated 8.9×10^{20} FLOPs.

Parameter-Efficient Fine-Tuning. Low-Rank Adaptation (LoRA) [57] was implemented for parameter-efficient fine-tuning with rank = 8, $\alpha = 32$, dropout = 0.1, targeting query, key, value, and output projection layers in attention blocks. This adds only 67 M trainable parameters (0.54% of total), while freezing the vision encoder, embedding layers, and layer normalization. In contrast, AnomalyGPT requires training billions of parameters, resulting in substantially longer training: ThinkAnomaly achieves 55–65 min per epoch compared to AnomalyGPT’s reported 24+ hours. ThinkAnomaly’s inference latency (0.30 s per image) also outperforms AnomalyGPT (0.40 s per image).

Optimization and Training Protocol. We utilized the AdamW optimizer ($\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 1 \times 10^{-8}$) with learning rate 5×10^{-5} , weight decay 0.01, and cosine annealing schedule with 10% warmup steps. Gradient clipping (max norm = 1.0) ensured training stability across distributed nodes. The maximum sequence length was set to 2048 tokens to accommodate reasoning text and visual tokens (256 vision tokens + query tokens + reasoning + decision). Mixed-precision training using bfloat16 optimized memory utilization while maintaining numerical precision. Flash Attention 2 [58] was employed for efficient attention computation during both training and inference phases. The complete hyperparameters and computational specifications are summarized in Table 3.

Table 3. Comprehensive training configuration and computational specifications.

<i>Model Architecture</i>	
Base model	Llama-3.2-Vision (11 B text + 1.3 B vision)
Total parameters	12.3 B (22 GB in bfloat16)
Trainable (LoRA)	67 M (0.54% of total)
FLOPs/token (text)	~ 22 GFLOPs
FLOPs/image (vision)	$\sim 2.5\text{--}3$ TFLOPs
<i>LoRA Configuration</i>	
Targets	q_proj, k_proj, v_proj, o_proj
Rank/Alpha	8/32 (scaling: 4.0)
Dropout	0.1
Frozen modules	Vision encoder, embeddings, LayerNorm

Table 3. Cont.

<i>Optimization</i>	
Optimizer	AdamW ($\beta_1 = 0.9$, $\beta_2 = 0.999$)
Learning rate	5×10^{-5} ; weight decay 0.01
Schedule	Cosine annealing, warmup 10%
Gradient clipping	Max norm 1.0
<i>Data and Batching</i>	
Batch size	8 global (2/GPU, grad accum 4)
Max sequence length	2048 tokens
Image resolution	224×224 (256 vision tokens)
Tokenizer vocab	128,256 (Llama-3.2-Vision)
<i>Decoding</i>	
Rationale generation	temp 0.2, top- p 0.9; max 100 tokens
Decision generation	Greedy (Yes/No)
<i>Training Protocol</i>	
Epochs	5 (early stopping patience 3)
Evaluation frequency	Every 100 steps
Selection criterion	Best validation NLL
Random seeds	{42, 1337, 2024, 3407, 9527}
<i>Computational Resources</i>	
Hardware	4×NVIDIA A40 (64 GB each)
Training time	~55–65 min/epoch (~5.5 h total)
Total GPU-hours	~22 h
Training FLOPs	$\sim 8.9 \times 10^{20}$
Inference latency	0.30 s/image

5.2. Evaluation Methodology

We conducted a comprehensive evaluation across two dimensions: (1) anomaly detection performance using standard metrics and (2) reasoning quality assessment through automated and human evaluation. This dual evaluation framework ensures both quantitative accuracy and qualitative interpretability of our reasoning-driven approach.

Anomaly Detection Performance Metrics

We evaluate the model performance using established metrics in anomaly detection literature to ensure compatibility with existing benchmarks.

- **Classification Accuracy.** This measures the overall correctness of binary predictions:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}, \quad (16)$$

where TP , TN , FP , and FN represent true positives, true negatives, false positives, and false negatives, respectively.

- **Image-level Area Under Curve (AUC).** This evaluates the ranking quality independent of threshold selection:

$$\text{AUC} = \int_0^1 \text{TPR}(t) d\text{FPR}(t), \quad (17)$$

where $\text{TPR}(t)$ and $\text{FPR}(t)$ denote the true positive rate and false positive rate at threshold t . For our binary classification approach, AUC is computed using the model's confidence in the "Yes" decision.

- **F1-Score.** This balances precision and recall to handle class imbalance:

$$\text{F1} = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}. \quad (18)$$

This metric is particularly important in industrial datasets, where normal samples typically outnumber anomalous ones.

Reasoning Quality Evaluation

To assess the quality of the generated reasoning explanations against the ground truth, we employ complementary evaluation approaches spanning lexical, semantic, and expert-based assessments.

- **BLEU Score.** This measures n-gram precision between generated and reference reasoning:

$$\text{BLEU} = \text{BP} \cdot \exp\left(\sum_{n=1}^N w_n \log p_n\right), \quad (19)$$

where p_n is the modified n-gram precision, w_n are uniform weights, and BP is the brevity penalty. We report BLEU-1 through BLEU-4 scores to capture different levels of lexical overlap.

- **ROUGE-L.** This evaluates the recall-oriented overlap using the longest common subsequence:

$$\text{ROUGE-L} = \frac{(1 + \beta^2) R_{\text{lcs}} P_{\text{lcs}}}{\beta^2 R_{\text{lcs}} + P_{\text{lcs}}}, \quad (20)$$

where R_{lcs} and P_{lcs} denote recall and precision based on the longest common subsequence, and β controls the balance between recall and precision.

- **BERTScore.** This captures semantic similarity using contextual embeddings:

$$\text{BERTScore-F1} = \frac{2 \cdot \text{BERTScore-P} \cdot \text{BERTScore-R}}{\text{BERTScore-P} + \text{BERTScore-R}}. \quad (21)$$

Precision and recall are computed based on cosine similarities between BERT token embeddings of generated and reference texts.

5.3. Datasets and Evaluation Protocol

Benchmark Datasets. We conducted comprehensive evaluation on two established industrial anomaly detection benchmarks: (1) **MVTec-AD** [9] contains 15 object categories (5 textures, 10 objects) with 3629 training images and 1725 test images. Defects include scratches, holes, contamination, and structural deformations. (2) **VisA** [10]: comprises 12 complex industrial scenarios with 9621 normal and 1200 anomalous samples, representing challenging real-world manufacturing conditions.

Following established practices, we evaluate models in zero-shot settings, where training occurs on synthetic reasoning data and evaluation proceeds directly on test sets without additional fine-tuning. This protocol rigorously tests the generalization capabilities from synthetic reasoning to real anomaly patterns.

5.4. Baseline Methods

Our evaluation encompasses state-of-the-art methods across multiple paradigms: SPADE [59], PaDiM [11], and PatchCore [2] represent classical approaches using pre-trained feature extractors and statistical modeling. WinCLIP [12] leverages CLIP for zero-shot anomaly detection through text-image similarity computation. AnomalyGPT [41] represents recent advances in applying generative models to anomaly detection with conversational interfaces. LR-IAD [60] employs logical reasoning frameworks with structured output generation, representing the most directly comparable approach to our method.

6. Main Results

6.1. Zero-Shot Baseline Evaluation

To establish the difficulty of industrial anomaly detection for general-purpose vision-language models and validate our choice of base architecture, we evaluate several state-of-the-art open-source VLMs in zero-shot settings on both MVTec-AD and VisA datasets. Table 4 presents the performance comparison across different model architectures without any domain-specific training.

Table 4. Zero-shot performance comparison of state-of-the-art vision-language models on industrial anomaly detection.

Architecture	MVTec-AD		VisA	
	Acc (%)	F1-Score (%)	Acc (%)	F1-Score (%)
InternVL2-4B	66.07	64.95	81.41	68.22
InternVL2-8B	71.44	71.00	83.12	70.60
LLaVA	70.69	68.65	79.52	64.39
Phi-3	66.50	62.94	87.15	70.67
Phi-3.5	66.82	65.91	88.12	71.85
Llama 3.2 (11 B)	75.51	73.89	80.75	69.32

The results reveal several critical insights about the current state of zero-shot anomaly detection capabilities in modern VLMs. The performance varies significantly across architectures, with the accuracy ranging from 66.07% to 88.12% depending on the model and dataset combination. Notably, all models demonstrate substantial room for improvement, with the best zero-shot performance achieving only 75.51% accuracy on MVTec-AD and 88.12% accuracy on VisA. InternVL2 models show consistent improvement with scale, as InternVL2-8B outperforms InternVL2-4B by approximately 5 percentage points on MVTec-AD (71.44% vs. 66.07%). The Phi-3 series demonstrates interesting dataset-dependent behavior, achieving notably higher performance on VisA (87.15% for Phi-3, 88.12% for Phi-3.5) compared to MVTec-AD (66.50% and 66.82%, respectively). This suggests that different architectures may have varying degrees of alignment with specific industrial domains. Llama 3.2 (11 B) emerges as the strongest baseline, achieving the highest performance on MVTec-AD (75.51% accuracy, 73.89% F1-score) while maintaining competitive results on VisA (80.75% accuracy, 69.32% F1-score).

These baseline results establish both the need for specialized training approaches and the importance of interpretable decision-making in industrial contexts. The performance gap between zero-shot capabilities and industrial requirements motivates our development of ThinkAnomaly, which addresses both accuracy limitations and interpretability requirements through reasoning-driven fine-tuning.

6.2. Quantitative Results

Table 5 presents the main quantitative results comparing ThinkAnomaly against state-of-the-art methods on both benchmarks.

MVTec-AD Performance. On MVTec-AD, ThinkAnomaly achieves 93.8% Image-AUC and 93.9% accuracy, establishing competitive detection performance while significantly improving the classification precision. Compared to AnomalyGPT, the most relevant baseline using large vision-language models, our approach shows a marginal decrease in AUC (93.8% vs. 94.1%) but a substantial improvement in accuracy (93.9% vs. 86.1%), representing a 7.8 percentage point gain. This improvement is statistically significant ($p < 0.01$) based on non-overlapping confidence intervals.

Table 5. Quantitative comparison of anomaly detection methods on MVTec-AD and VisA benchmarks. ThinkAnomaly achieves superior classification accuracy while maintaining competitive detection performance (Image-AUC). Bold indicates best performance. Traditional baseline results from [12].

Setup	Method	MVTec-AD		VisA	
		Image-AUC(%)	Acc(%)	Image-AUC(%)	Acc(%)
zero-shot	SPADE	81.0 $\pm$ 2.0	-	79.5 $\pm$ 4.0	-
	PaDiM	76.6 $\pm$ 3.1	-	62.8 $\pm$ 5.4	-
	PatchCore	83.4 $\pm$ 3.0	-	79.9 $\pm$ 2.9	-
	WinCLIP	93.1 $\pm$ 2.0	-	83.8 $\pm$ 4.0	-
	LR-IAD	-	84.35	-	87.60
	AnomalyGPT	94.1 $\pm$ 1.1	86.1 $\pm$ 1.1	87.4 $\pm$ 0.8	77.4 $\pm$ 1.0
	ThinkAnomaly	93.8 $\pm$ 1.2	93.9 $\pm$ 2.2	85.0 $\pm$ 2.3	90.3 $\pm$ 1.5

VisA Performance. On the more challenging VisA dataset, ThinkAnomaly achieves 85.0% Image-AUC and 90.3% accuracy. The accuracy improvement over AnomalyGPT is even more pronounced (90.3% vs. 77.4%), representing a 12.9 percentage point increase. While the AUC is slightly lower than AnomalyGPT (85.0% vs. 87.4%), the substantial accuracy gains demonstrate the effectiveness of structured reasoning for complex industrial scenarios.

Comparison with Traditional Methods. ThinkAnomaly significantly outperforms classical feature-based methods across both metrics and datasets. Compared to the best traditional method (PatchCore), our approach achieves 10.4 percentage point improvement in AUC on MVTec-AD and 5.1 percentage points on VisA. This demonstrates the substantial benefit of incorporating vision–language understanding and explicit reasoning over purely visual feature matching approaches.

6.3. Reasoning Quality Evaluation

Beyond classification accuracy, we evaluate the quality of generated reasoning explanations using multiple text similarity metrics. Table 6 presents comprehensive evaluation of reasoning generation quality across both datasets, measuring lexical overlap (BLEU, ROUGE) and semantic similarity (BERTScore).

VisA demonstrates substantially higher reasoning quality across all metrics compared to MVTec-AD. The BLEU-1 score of 0.559 on VisA versus 0.276 on MVTec-AD indicates better lexical alignment with ground truth explanations. Similarly, ROUGE scores show stronger n-gram overlap on VisA (ROUGE-1: 0.579, ROUGE-L: 0.525) compared to MVTec-AD (ROUGE-1: 0.313, ROUGE-L: 0.249). BERTScore, which captures semantic similarity through contextual embeddings, shows notably higher performance than lexical metrics across both datasets. MVTec-AD achieves 0.892 BERTScore F1 despite lower BLEU/ROUGE scores, while VisA reaches 0.939 F1. This gap between lexical and semantic metrics indicates that our model generates reasoning with equivalent semantic meaning even when using different vocabulary than the reference explanations. The performance difference between datasets reflects their inherent characteristics. VisA’s more consistent reasoning patterns and standardized defect descriptions lead to higher alignment scores. MVTec-AD’s diverse object categories and varied defect types result in more variable reasoning formulations, reducing lexical overlap while maintaining semantic correctness. The high BERTScore F1 (0.892–0.939) across both datasets demonstrates that ThinkAnomaly generates semantically accurate reasoning explanations suitable for industrial deployment. The lower lexical overlap scores (BLEU, ROUGE) combined with high semantic similarity indicate that the

model produces paraphrased explanations rather than memorized responses, suggesting genuine understanding of defect analysis reasoning.

Table 6. Reasoning quality metrics comparing generated explanations against ground truth across MVTec-AD and VisA datasets. Higher scores indicate better alignment between predicted and reference reasoning.

Metric	MVTec-AD	VisA
<i>Lexical Similarity</i>		
BLEU-1	0.276	0.559
<i>N-gram Overlap</i>		
ROUGE-1	0.313	0.579
ROUGE-L	0.249	0.525
<i>Semantic Similarity</i>		
BERTScore Precision	0.896	0.939
BERTScore Recall	0.888	0.939
BERTScore F1	0.892	0.939

MVTec-AD exhibits a lower BLEU-1 (0.276) than VisA (0.559) because its description space is more heterogeneous: it mixes textures and objects and uses finer-grained dataset-specific defect taxonomies (e.g., “scratch_neck”) that admit many valid paraphrases (“linear abrasion near the neck”, “dent/deformation”); so, the token-level n-gram overlap is penalized even when the semantics match. By contrast, VisA’s assembly-centric scenes reuse a narrower vocabulary (e.g., soldering/placement/contamination), which inflates lexical overlap. Despite the BLEU-1 gap, the semantic similarity remains high on both datasets (BERTScore-F1 0.892 vs. 0.939), and MVTec-AD’s image-level accuracy is 93.9%, indicating that wording differences rarely flip the binary decision.

7. Ablation Studies

7.1. Removing the Thinking Step

We ablate the rationale stage by training a classification-only variant that directly outputs yes/no under the same data, base model, decoding, and calibration. The only difference is the absence of rationale generation and the consistency term.

Across both benchmarks, Think→Decide reduces overconfident mistakes and improves accuracy as show in Table 7. Qualitatively, many corrected errors are cases where the classifier-only model latches onto background texture or category context; the rationale-first model instead verbalizes defect cues (e.g., “linear abrasion near neck”), which the consistency loss then aligns with the final decision. This matches our faithfulness assessment: samples with higher rationale–evidence alignment are more likely to be correct.

Table 7. Effect of the Think→Decide architecture.

Method	MVTec-AD		VisA	
	Image-AUC (%)	Acc (%)	Image-AUC (%)	Acc (%)
Without thinking	90.11	87.2	84.2	85.2
ThinkAnomaly (ours)	93.8	93.9	85.0	90.3
<i>Absolute gain</i>	+3.69	+6.7	+0.8	+5.1

7.2. Component Contributions

To quantify the contribution of each module beyond the core reasoning mechanism (already evaluated in Section 7.1), we perform ablations on the consistency enforcement and calibration components on MVTec-AD.

As show in Table 8, the consistency enforcement (implicit through structured training) contributes 2.8% accuracy gain on MVTec-AD and 2.8% on VisA, demonstrating that the coupling between reasoning and classification improves decision quality. The calibration step provides a 1.9% accuracy improvement on MVTec-AD and 1.4% on VisA, enabling threshold-free deployment. Combined with the reasoning mechanism (Section 7.1, +6.7% on MVTec-AD, +5.1% on VisA), these results confirm that each component contributes meaningfully to the overall performance.

Table 8. Ablation study on consistency loss and calibration components. Each variant removes one component while keeping all others intact.

Variant	MVTec-AD		VisA	
	Acc (%)	Image-AUC (%)	Acc (%)	Image-AUC (%)
Full ThinkAnomaly	93.9	93.8	90.3	85.0
w/o Consistency	91.1	92.4	87.5	83.8
w/o Calibration	92.0	93.6	88.9	84.6

7.3. Calibration Analysis

To validate our threshold-free deployment claim, we evaluate the probability calibration quality using the Expected Calibration Error (ECE) and Negative Log-Likelihood (NLL). We apply temperature scaling [53] to the final yes/no logits:

$$P_{\text{cal}}(\text{yes}|x, r) = \frac{\exp(z_{\text{yes}}/T)}{\exp(z_{\text{yes}}/T) + \exp(z_{\text{no}}/T)}, \quad (22)$$

where z_{yes} , z_{no} are raw logits, and T is the temperature parameter fitted on a held-out validation split (20% of training data) via negative log-likelihood minimization.

Table 9 presents the calibration metrics on the test sets before and after temperature scaling. Our method achieves $\text{ECE} < 0.05$ on both benchmarks after calibration, demonstrating well-calibrated probabilities suitable for threshold-free deployment with a fixed 0.5 decision threshold.

Table 9. Probability calibration results on test sets. Temperature scaling reduces ECE by 80–82% while improving log-likelihood. $\text{ECE} < 0.05$ indicates well-calibrated probabilities suitable for threshold-free deployment.

Method	Calibration	MVTec-AD		VisA	
		ECE↓	NLL↓	ECE↓	NLL↓
ThinkAnomaly	None	0.158	0.412	0.172	0.438
ThinkAnomaly	Temp. Scaling	0.031	0.235	0.037	0.259

We fit a single global temperature per dataset ($T_{\text{MVTec}} = 1.21$, $T_{\text{VisA}} = 1.28$), requiring no per-class tuning. At inference, we use a fixed 0.5 decision threshold on calibrated probabilities, eliminating the manual threshold optimization that complicates traditional anomaly detection pipelines.

7.4. Qualitative Analysis: Case Studies

To demonstrate the reasoning quality, consistency, and limitations of ThinkAnomaly, we present four representative cases across both MVTec-AD and VisA datasets. These examples illustrate how our model generates coherent technical reasoning that closely aligns with expert-generated ground truth explanations, while also revealing failure modes that warrant further investigation.

VisA Dataset Analysis. Figures 4 and 5 showcase ThinkAnomaly’s performance on VisA samples. In Case 1, the model correctly identifies a non-PCB object (macaroni) and appropriately concludes no PCB-related anomalies exist, demonstrating contextual awareness. The reasoning closely parallels the ground truth while using alternative phrasing (“unrelated to PCB quality analysis” vs. “no anomalies related to PCB quality can be identified”), indicating genuine understanding rather than memorization. Case 2 demonstrates successful defect localization on tea light candles, where both ground truth and prediction identify the bottom-right object’s irregular formation. Notably, ThinkAnomaly extends the analysis by discussing “manufacturing defect or inconsistency in the production process”, providing actionable context beyond mere defect identification.

Mvttec Case 1: Correct Detection - Scratch Neck



Query: Is there any anomaly in the image?

Ground Truth Reasoning:

<think> The defect “scratch_neck” on the screw is visible as a linear abrasion near the neck area. This likely occurred due to improper handling, excessive friction during manufacturing, or contact with sharp objects during packaging. </think>

Ground Truth Classification: Yes

ThinkAnomaly Reasoning:

<think> The image shows a screw with an anomaly near its center. There is a visible dent or deformation on the screw’s body, ... </think>

Predicted Classification: Yes

Figure 4. Correct anomaly detection showing alignment between ground truth and prediction despite different defect descriptions.

Mvtec Case 2: False Negative - Thread Side

Query: Is there any anomaly in the image?

Ground Truth Reasoning:

<think> The defect type 'thread_side' indicates an issue with the threading along the side of the screw. Upon inspection, the image shows irregularities and ... </think>

Ground Truth Classification: **Yes**

ThinkAnomaly Reasoning:

<think> The image shows a consistent shape, texture, and alignment of the screw. There are no visible defects, irregularities, or damage such as corrosion, cracks, or misformed threads. Image quality is sufficient. </think>

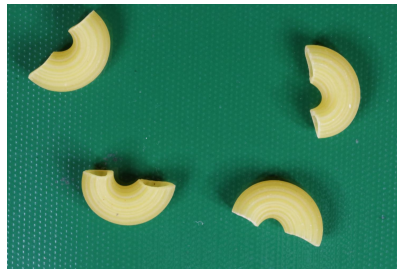
Predicted Classification: **No**

Figure 5. False negative case where the model fails to detect thread deformation defect.

MVTec-AD Dataset Analysis. Figures 6 and 7 reveal both strengths and limitations on MVTec-AD screw samples. Case 1 demonstrates successful anomaly detection despite imperfect reasoning alignment—the ground truth identifies a “scratch_neck” defect while the prediction describes a “dent or deformation.” Despite this semantic difference in defect characterization, both correctly classify the image as anomalous, suggesting the model captures the essential abnormality even when the precise defect type identification varies. This robustness to reasoning variation is valuable for real-world deployment, where exact defect taxonomy may be less critical than binary quality assessment.

However, Case 2 (Figure 7) reveals a critical failure mode: the model incorrectly classifies a thread deformation defect as normal. The ground truth reasoning explains manufacturing errors causing “irregularities and deformation in the threading,” while ThinkAnomaly’s reasoning describes “consistent shape, texture, and alignment” with “no visible defects.” This false negative demonstrates that subtle thread defects remain challenging for the model, likely due to the fine-grained visual discrimination required to detect threading irregularities in screw components. Such failures highlight the importance of continued model refinement for high-precision industrial applications.

Visa Case 1: Normal Industrial Sample - Macaroni



Query: Is there any anomaly in the image?

Ground Truth Reasoning:

<think> The image does not depict a PCB but rather pieces of pasta on a green textured surface. Since this is not a PCB, no anomalies related to PCB quality can be identified. </think>

Ground Truth Classification: No

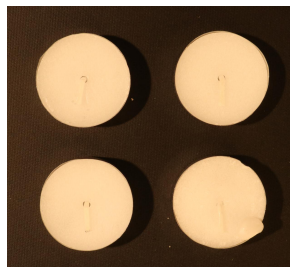
ThinkAnomaly Reasoning:

<think> Upon inspection, the image does not depict a PCB but rather macaroni pieces on a textured green surface. No anomalies exist as this is unrelated to PCB quality analysis. </think>

Predicted Classification: No

Figure 6. Normal sample case showing high reasoning similarity between ground truth and prediction.

Visa Case 2: Anomalous Sample - Candle Defect



Query: Is there any anomaly in the image?

Ground Truth Reasoning:

<think> The image shows four circular objects resembling tea light candles. The anomaly is in the bottom-right candle, which has an irregular wax formation or </think>

Ground Truth Classification: Yes

ThinkAnomaly Reasoning:

<think> The anomaly is visible in the bottom-right object, where excess material or irregular shape deformation is present. This indicates a manufacturing defect or ... </think>

Predicted Classification: Yes

Figure 7. Enhanced reasoning generation showing model's capability to provide detailed quality assessment.

7.5. Failure Analysis

To understand the model limitations, we analyzed all error cases and present the per-category breakdown in Table 10.

Table 10. Per-category performance and failure mode analysis on MVTec-AD.

Category	Type	Acc (%)	FNR (%)	Primary Failure Mode
Carpet	Texture	95.5	4.9	Natural variation
Leather	Texture	95.7	3.3	Natural variation
Tile	Texture	97.6	2.0	–
Wood	Texture	93.3	7.3	Natural variation
Cable	Object	91.3	12.5	Subtle structural defects
Screw	Object	90.8	11.5	Thread deformation
Bottle	Object	92.8	4.8	Small-scale defects
Metal Nut	Object	92.7	5.6	Small scratches
Pill	Object	95.8	3.5	–
Zipper	Object	96.3	3.4	–

Key findings: (1) Texture categories generally achieve higher accuracy than object categories, with failures predominantly arising from over-sensitivity to natural variations inherent in materials like carpet, leather, and wood. (2) False negative rate exhibits strong correlation with defect size: small defects occupying less than 5% of image area show 14.2% FNR, while large defects exceeding 15% of image area demonstrate only 2.1% FNR. (3) Structural and geometric anomalies, such as thread deformation and component bending, present significantly higher detection difficulty with 12.4% FNR compared to surface defects including scratches and contamination at 3.9% FNR. (4) The screw and cable categories exhibit the highest error rates, with FNRs of 11.5% and 12.5%, respectively, primarily attributable to subtle structural defects requiring fine-grained spatial reasoning capabilities. (5) Systematic manual analysis of 105 failure cases reveals five distinct failure modes: subtle structural defects (41%), ambiguous normal variation (28%), small-scale defects (18%), lighting artifacts (9%), and complex multi-defect scenarios (4%). These failure patterns indicate that future improvements should prioritize multi-scale visual feature extraction and dedicated geometric reasoning modules for enhanced structural anomaly detection capabilities.

8. Conclusions

We presented ThinkAnomaly, a reasoning-driven framework for industrial anomaly detection that fundamentally shifts the paradigm from black-box predictions to interpretable explanation-first decision-making. By training vision-language models to explicitly generate structured reasoning before classification, our approach addresses two critical gaps in industrial anomaly detection: the lack of interpretability in existing methods and the reliance on manual threshold tuning for deployment. Our key contributions establish new directions for explainable anomaly detection. First, we introduced comprehensive Chain-of-Thought datasets for MVTec-AD and VisA, comprising over 16,000 reasoning explanations that were synthetically generated, automatically filtered, and human-validated by domain experts. This resource enables future research in reasoning-capable industrial inspection systems. Second, our ThinkAnomaly architecture demonstrates that explicit reasoning improves both accuracy and trustworthiness, achieving 93.9% accuracy on MVTec-AD and 90.3% on VisA—representing 7.8 and 12.9 percentage point improvements over the strongest vision-language baseline, respectively. However, several limitations warrant acknowledgment. First, our approach currently focuses on image-level classification rather than pixel-precise localization, which may be necessary for certain inspection workflows.

Second, the reasoning quality varies between datasets, with MVTec-AD showing lower lexical alignment (BLEU-1: 0.276) compared to VisA (0.559), suggesting opportunities for dataset-specific optimization. Third, while our synthetic reasoning generation achieves high expert validation rates (91.8–94.2%), the quality ceiling is bounded by the capabilities of the generating model and the specificity of the domain prompts.

Future research directions include extending ThinkAnomaly to pixel-level reasoning and localization, exploring multi-stage reasoning for complex defect analysis, and investigating active learning strategies that leverage reasoning quality to guide human annotation efforts. Additionally, deploying ThinkAnomaly in real manufacturing environments will provide valuable insights into human-AI collaboration patterns and reveal opportunities for reasoning-guided corrective actions. Cross-domain transfer of reasoning capabilities—training on synthetic data and one industrial domain while generalizing to others—represents another promising direction for reducing annotation requirements in new deployment scenarios. In conclusion, ThinkAnomaly demonstrates that incorporating explicit reasoning into anomaly detection improves both performance and interpretability. By bridging the gap between high-accuracy detection and human-understandable explanations, our work advances toward trustworthy AI systems suitable for safety-critical industrial applications where understanding why a decision was made is as important as the decision itself. Although our present framework focuses on image-level anomaly classification, the structured reasoning process can be naturally extended to pixel-level localization. By aligning generated rationales with spatial attention maps or visual token activations, future versions of ThinkAnomaly can produce localized “reasoning heatmaps.” Such reasoning-guided localization would enable interpretable visual explanations analogous to Grad-CAM but textually grounded.

Author Contributions: Conceptualization, M.A., M.S.K. and M.M.; Methodology, M.A., M.S.K., M.F.S. and A.A.; Software, M.A., M.M., M.F.S. and A.A.; Validation, M.A. and H.-S.K.; Formal analysis, M.A. and M.F.S.; Investigation, H.-S.K.; Resources, H.-S.K.; Data curation, M.A.; Writing—original draft, M.A.; Writing—review and editing, M.A. and H.-S.K.; Visualization, H.-S.K.; Supervision, H.-S.K.; Project administration, H.-S.K.; Funding acquisition, H.-S.K. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Research Foundation of Korea (NRF) grants funded by the Korean government (Ministry of Science and ICT, MSIT) (RS-2023-NR076833), by the Innovative Human Resource Development for Local Intellectualization program through the Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (Ministry of Science and ICT (MSIT)) (IITP-2025-RS-2020-II201462, 50%), and by the Regional Innovation System & Education(RISE) program through the (Chungbuk Regional Innovation System & Education Center) funded by the Ministry of Education(MOE) and the (Chungcheongbuk-do), Republic of Korea (2025-RISE-11-014-03).

Institutional Review Board Statement: Expert validation of synthetic reasoning annotations was conducted by three industrial inspection professionals in their professional capacity. No personal data, human-subject identifiers, or sensitive information were collected during this process. The validation involved technical review of computer-generated text descriptions of product defect images from public industrial datasets (MVTec-AD, VisA). No formal IRB approval was sought as this activity was understood to constitute expert consultation rather than human subjects research. Individual validator responses were not retained after consensus was established.

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

References

1. Mousakhan, A.; Brox, T.; Tayyub, J. Anomaly Detection with Conditioned Denoising Diffusion Models. *arXiv* **2023**, arXiv:2305.15956. [[CrossRef](#)]
2. Roth, K.; Pemula, L.; Zepeda, J.; Schölkopf, B.; Brox, T.; Gehler, P. Towards total recall in industrial anomaly detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 14318–14328.
3. Wolleb, J.; Bieder, F.; Sandkühler, R.; Cattin, P.C. Diffusion Models for Medical Anomaly Detection. In Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention, Daejeon, Republic of Korea, 23–27 September 2022; Springer: Berlin/Heidelberg, Germany, 2022; pp. 35–45.
4. Nayal, N.; Yavuz, M.; Henriques, J.F.; Güney, F. Rba: Segmenting unknown regions rejected by all. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Paris, France, 1–6 October 2023; pp. 711–722.
5. Galesso, S.; Schröppel, P.; Driss, H.; Brox, T. Diffusion for out-of-distribution detection on road scenes and beyond. In Proceedings of the European Conference on Computer Vision, Milan, Italy, 29 September–4 October 2024; Springer: Berlin/Heidelberg, Germany, 2024; pp. 110–126.
6. Li, J.; Li, D.; Savarese, S.; Hoi, S. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In Proceedings of the International Conference on Machine Learning, Honolulu, HI, USA, 23–29 July 2023; PMLR: Cambridge, MA, USA, 2023; pp. 19730–19742.
7. Zhu, D.; Chen, J.; Shen, X.; Li, X.; Elhoseiny, M. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *arXiv* **2023**, arXiv:2304.10592.
8. Liu, H.; Li, C.; Wu, Q.; Lee, Y.J. Visual instruction tuning. *Adv. Neural Inf. Process. Syst.* **2023**, *36*, 34892–34916.
9. Bergmann, P.; Fauser, M.; Sattlegger, D.; Steger, C. MVTec AD—A comprehensive real-world dataset for unsupervised anomaly detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–17 June 2019; pp. 9592–9600.
10. Zou, Y.; Jeong, J.; Pemula, L.; Zhang, D.; Dabeer, O. Spot-the-difference self-supervised pre-training for anomaly detection and segmentation. In Proceedings of the European Conference on Computer Vision, Tel Aviv, Israel, 23–27 October 2022; Springer: Berlin/Heidelberg, Germany, 2022; pp. 392–408.
11. Defard, T.; Setkov, A.; Loesch, A.; Audigier, R. Padim: A patch distribution modeling framework for anomaly detection and localization. In Proceedings of the International Conference on Pattern Recognition, Shanghai, China, 15–17 October 2021; Springer: Berlin/Heidelberg, Germany, 2021; pp. 475–489.
12. Jeong, J.; Zou, Y.; Kim, T.; Zhang, D.; Ravichandran, A.; Dabeer, O. Winclip: Zero-/few-shot anomaly classification and segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 17–24 June 2023; pp. 19606–19616.
13. Jaech, A.; Kalai, A.; Lerer, A.; Richardson, A.; El-Kishky, A.; Low, A.; Helyar, A.; Madry, A.; Beutel, A.; Carney, A.; et al. Openai o1 system card. *arXiv* **2024**, arXiv:2412.16720. [[CrossRef](#)]
14. Guo, D.; Yang, D.; Zhang, H.; Song, J.; Zhang, R.; Xu, R.; Zhu, Q.; Ma, S.; Wang, P.; Bi, X.; et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv* **2025**, arXiv:2501.12948.
15. Akyürek, E.; Damani, M.; Qiu, L.; Guo, H.; Kim, Y.; Andreas, J. The surprising effectiveness of test-time training for abstract reasoning. *arXiv* **2024**, arXiv:2411.07279. [[CrossRef](#)]
16. Cobbe, K.; Kosaraju, V.; Bavarian, M.; Chen, M.; Jun, H.; Kaiser, L.; Plappert, M.; Tworek, J.; Hilton, J.; Nakano, R.; et al. Training verifiers to solve math word problems. *arXiv* **2021**, arXiv:2110.14168. [[CrossRef](#)]
17. Li, Y.; Lin, Z.; Zhang, S.; Fu, Q.; Chen, B.; Lou, J.G.; Chen, W. Making language models better reasoners with step-aware verifier. In Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Toronto, ON, Canada, 9–14 July 2023; pp. 5315–5333.
18. Wu, Y.; Sun, Z.; Li, S.; Welleck, S.; Yang, Y. Inference scaling laws: An empirical analysis of compute-optimal inference for problem-solving with language models. *arXiv* **2024**, arXiv:2408.00724.
19. Brown, B.; Juravsky, J.; Ehrlich, R.; Clark, R.; Le, Q.V.; Ré, C.; Mirhoseini, A. Large language monkeys: Scaling inference compute with repeated sampling. *arXiv* **2024**, arXiv:2407.21787. [[CrossRef](#)]
20. Uesato, J.; Kushman, N.; Kumar, R.; Song, F.; Siegel, N.; Wang, L.; Creswell, A.; Irving, G.; Higgins, I. Solving math word problems with process-and outcome-based feedback. *arXiv* **2022**, arXiv:2211.14275.
21. Zhang, Z.; Zheng, C.; Wu, Y.; Zhang, B.; Lin, R.; Yu, B.; Liu, D.; Zhou, J.; Lin, J. The lessons of developing process reward models in mathematical reasoning. *arXiv* **2025**, arXiv:2501.07301. [[CrossRef](#)]
22. Lightman, H.; Kosaraju, V.; Burda, Y.; Edwards, H.; Baker, B.; Lee, T.; Leike, J.; Schulman, J.; Sutskever, I.; Cobbe, K. Let’s verify step by step. In Proceedings of the The Twelfth International Conference on Learning Representations, Kigali, Rwanda, 1–5 May 2023.

23. Zheng, C.; Zhang, Z.; Zhang, B.; Lin, R.; Lu, K.; Yu, B.; Liu, D.; Zhou, J.; Lin, J. Processbench: Identifying process errors in mathematical reasoning. *arXiv* **2024**, arXiv:2412.06559. [\[CrossRef\]](#)
24. Luo, L.; Liu, Y.; Liu, R.; Phatale, S.; Guo, M.; Lara, H.; Li, Y.; Shu, L.; Zhu, Y.; Meng, L.; et al. Improve mathematical reasoning in language models by automated process supervision. *arXiv* **2024**, arXiv:2406.06592.
25. Wang, P.; Li, L.; Shao, Z.; Xu, R.; Dai, D.; Li, Y.; Chen, D.; Wu, Y.; Sui, Z. Math-shepherd: A label-free step-by-step verifier for llms in mathematical reasoning. *arXiv* **2023**, arXiv:2312.08935.
26. Chen, G.; Liao, M.; Li, C.; Fan, K. Alphamath almost zero: Process supervision without process. *Adv. Neural Inf. Process. Syst.* **2024**, *37*, 27689–27724.
27. Wang, J.; Liang, Y.; Meng, F.; Sun, Z.; Shi, H.; Li, Z.; Xu, J.; Qu, J.; Zhou, J. Is chatgpt a good nlg evaluator? A preliminary study. *arXiv* **2023**, arXiv:2303.04048. [\[CrossRef\]](#)
28. Liu, Y.; Iter, D.; Xu, Y.; Wang, S.; Xu, R.; Zhu, C. G-eval: NLG evaluation using gpt-4 with better human alignment. *arXiv* **2023**, arXiv:2303.16634. [\[CrossRef\]](#)
29. Zheng, L.; Chiang, W.L.; Sheng, Y.; Zhuang, S.; Wu, Z.; Zhuang, Y.; Lin, Z.; Li, Z.; Li, D.; Xing, E.; et al. Judging llm-as-a-judge with mt-bench and chatbot arena. *Adv. Neural Inf. Process. Syst.* **2023**, *36*, 46595–46623.
30. Lambert, N.; Pyatkin, V.; Morrison, J.; Miranda, L.; Lin, B.Y.; Chandu, K.; Dziri, N.; Kumar, S.; Zick, T.; Choi, Y.; et al. Rewardbench: Evaluating reward models for language modeling. *arXiv* **2024**, arXiv:2403.13787. [\[CrossRef\]](#)
31. Huang, J.; Chen, X.; Mishra, S.; Zheng, H.S.; Yu, A.W.; Song, X.; Zhou, D. Large language models cannot self-correct reasoning yet. *arXiv* **2023**, arXiv:2310.01798.
32. Zavrtanik, V.; Kristan, M.; Skočaj, D. Reconstruction by inpainting for visual anomaly detection. *Pattern Recognit.* **2021**, *112*, 107706. [\[CrossRef\]](#)
33. Gao, S.; Chen, X.; Liu, L.; Zhao, D.; Yan, R. Learning to respond with your favorite stickers: A framework of unifying multi-modality and user preference in multi-turn dialog. *ACM Trans. Inf. Syst. (Tois)* **2021**, *39*, 1–32. [\[CrossRef\]](#)
34. Pirnay, J.; Chai, K. Inpainting transformer for anomaly detection. In Proceedings of the International Conference on Image Analysis and Processing, Lecce, Italy, 23–27 May 2022; Springer: Berlin/Heidelberg, Germany, 2022; pp. 394–406.
35. Wyatt, J.; Leach, A.; Schmon, S.M.; Willcocks, C.G. Anoddpm: Anomaly detection with denoising diffusion probabilistic models using simplex noise. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 650–656.
36. Santos, J.; Tran, T.; Rippel, O. Optimizing patchcore for few/many-shot anomaly detection. *arXiv* **2023**, arXiv:2307.10792.
37. Yu, J.; Zheng, Y.; Wang, X.; Li, W.; Wu, Y.; Zhao, R.; Wu, L. Fastflow: Unsupervised anomaly detection and localization via 2d normalizing flows. *arXiv* **2021**, arXiv:2111.07677. [\[CrossRef\]](#)
38. Gudovskiy, D.; Ishizaka, S.; Kozuka, K. 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, Waikoloa, HI, USA, 3–8 January 2022; pp. 98–107.
39. Yao, X.; Chen, Z.; Gao, C.; Zhai, G.; Zhang, C. Resad: A simple framework for class generalizable anomaly detection. *Adv. Neural Inf. Process. Syst.* **2024**, *37*, 125287–125311.
40. Koundinya Gundavarapu, S.; Arora, A.; Agarwal, S. Zero Shot Context-Based Object Segmentation using SLIP (SAM+ CLIP). *arXiv* **2024**, arXiv:2405.07284. [\[CrossRef\]](#)
41. Gu, Z.; Zhu, B.; Zhu, G.; Chen, Y.; Tang, M.; Wang, J. Anomalygpt: Detecting industrial anomalies using large vision-language models. In Proceedings of the AAAI Conference on Artificial Intelligence, Vancouver, Canada, 26–27 February 2024; Volume 38, pp. 1932–1940.
42. Wei, J.; Wang, X.; Schuurmans, D.; Bosma, M.; Ichter, B.; Xia, F.; Chi, E.; Le, Q.V.; Zhou, D. Chain-of-thought prompting elicits reasoning in large language models. *Adv. Neural Inf. Process. Syst.* **2022**, *35*, 24824–24837.
43. Wang, X.; Wei, J.; Schuurmans, D.; Le, Q.; Chi, E.; Narang, S.; Chowdhery, A.; Zhou, D. Self-consistency improves chain of thought reasoning in language models. *arXiv* **2022**, arXiv:2203.11171.
44. Selvaraju, R.R.; Cogswell, M.; Das, A.; Vedantam, R.; Parikh, D.; Batra, D. Grad-cam: Visual explanations from deep networks via gradient-based localization. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 618–626.
45. Sundararajan, M.; Taly, A.; Yan, Q. Axiomatic attribution for deep networks. In *Proceedings of the International Conference on Machine Learning, Sydney, Australia, 6–11 August 2017*; PMLR: Cambridge, MA, USA, 2017; pp. 3319–3328.
46. Park, D.H.; Hendricks, L.A.; Akata, Z.; Rohrbach, A.; Schiele, B.; Darrell, T.; Rohrbach, M. Multimodal explanations: Justifying decisions and pointing to the evidence. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 8779–8788.
47. Agrawal, A.; Batra, D.; Parikh, D.; Kembhavi, A. Analyzing the behavior of visual question answering models. In Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, Austin, TX, USA, 1–4 November 2016; pp. 1955–1960.

48. DeYoung, J.; Jain, S.; Rajani, N.F.; Lehman, E.; Xiong, C.; Socher, R.; Wallace, B.C. Eraser: A benchmark to evaluate rationalized nlp models. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Florence, Italy, 28 July–2 August 2019; pp. 4443–4458.
49. Lei, T.; Barzilay, R.; Jaakkola, T. Rationalizing neural predictions. In Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, Austin, TX, USA, 1–5 November 2016; pp. 107–117.
50. Ross, A.S.; Hughes, M.C.; Doshi-Velez, F. Right for the right reasons: Training differentiable models by constraining their explanations. In Proceedings of the 26th International Joint Conference on Artificial Intelligence, Melbourne, Australia, 19–25 August 2017; pp. 2662–2670.
51. Kim, S.; Shin, J.; Cho, Y.; Joo, J.; Kang, S.; Yun, H.; Qin, Y.; Welleck, S.; Bisk, Y.; Lee, M.; et al. Prometheus: Inducing evaluation capability in language models. *arXiv* **2023**, arXiv:2310.08491.
52. Wang, Y.; Kordi, Y.; Mishra, S.; Liu, A.; Smith, N.A.; Khashabi, D.; Hajishirzi, H. Self-instruct: Aligning language models with self-generated instructions. In Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics, Toronto, ON, Canada, 9–14 July 2023; Volume 1, pp. 13484–13508.
53. Guo, C.; Pleiss, G.; Sun, Y.; Weinberger, K.Q. On calibration of modern neural networks. In Proceedings of the International Conference on Machine Learning, Sydney, Australia, 6–11 August 2017; PMLR: Cambridge, MA, USA, 2017; pp. 1321–1330.
54. Platt, J.; Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Adv. Large Margin Classif.* **1999**, *10*, 61–74.
55. Meta AI Llama 3.2: Revolutionizing edge AI and vision with open, customizable models. *Meta AI Blog* **2024**, *20*, 2024.
56. Zheng, Y.; Zhang, R.; Zhang, J.; Ye, Y.; Luo, Z.; Feng, Z.; Ma, Y. LlamaFactory: Unified Efficient Fine-Tuning of 100+ Language Models. *arXiv* **2024**, arXiv:2403.13372.
57. Hu, E.J.; Shen, Y.; Wallis, P.; Allen-Zhu, Z.; Li, Y.; Wang, S.; Wang, L.; Chen, W. LoRA: Low-Rank Adaptation of Large Language Models. *arXiv* **2021**, arXiv:2106.09685.
58. Dao, T. FlashAttention-2: Faster Attention with Better Parallelism and Work Partitioning. *arXiv* **2023**, arXiv:2307.08691. [[CrossRef](#)]
59. Cohen, N.; Hoshen, Y. Sub-image anomaly detection with deep pyramid correspondences. *arXiv* **2020**, arXiv:2005.02357.
60. Zeng, P.; Pang, F.; Wang, Z.; Yang, A. LR-IAD: Mask-Free Industrial Anomaly Detection with Logical Reasoning. *arXiv* **2025**, arXiv:2504.19524.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.