Results in Engineering 28 (2025) 108171

Contents lists available at ScienceDirect

Results in Engineering

journal homepage: www.sciencedirect.com/journal/results-in-engineering

Research paper

BlueprintSymVL: A discriminative benchmark for VLM symbol recognition in engineering blueprints

Vasil Shteriyanov [a,b,*], Rimma Dzhusupova [a,b], Jan Bosch [b,c], Helena Holmström Olsson [d]

[a] McDermott, Engineering, Pr. Beatrixlaan 35, The Hague, 2595 AK, the Netherlands
[b] Eindhoven University of Technology, Mathematics and Computer Science, Groene Loper 3, Eindhoven, 5612 AE, the Netherlands
[c] Chalmers University of Technology, Computer Science and Engineering, Chalmersgatan 4, Gothenburg, 412 96, Sweden
[d] Malmö University, Computer Science and Media Technology, Nordenskiöldsgatan 1, Malmö, 211 19, Sweden

A R T I C L E I N F O

Dataset link: Benchmark Dataset for VLM Symbol Recognition in Engineering Blueprints (Original data)

Keywords:
Vision Language Models (VLMs)
Visual in-context learning
Benchmark
Symbol recognition
Engineering blueprints

A B S T R A C T

The application of Vision Language Models (VLMs) to industrial automation, specifically engineering blueprint analysis, is severely hampered by the absence of domain-specific evaluation tools. Existing benchmarks fail to replicate the critical visual challenges of this domain, such as high symbol density, occlusion, and visual similarity. Furthermore, they assume reliable pre-trained knowledge or standardized symbology, which rarely hold in real-world industrial settings. To address these critical gaps, we introduce BlueprintSymVL, the first benchmark explicitly designed to evaluate VLM symbol recognition in engineering blueprints. BlueprintSymVL is engineered as a strong discriminator, with test cases that systematically introduce challenges to differentiate model capabilities. A key innovation is our robust evaluation method, centered on a one-shot visual in-context querying strategy. At query time, the model is provided with a visual exemplar of a symbol. This approach eliminates reliance on unreliable pre-existing knowledge and is paired with a strict evaluation criterion demanding correctness on both symbol counts and their labels, setting a rigorous standard for quality assurance in high-stakes applications. We conducted a comprehensive benchmark of four leading VLMs (GPT-4o, Gemini 2.5 Pro, InternVL 2.5 78B, and Qwen 2.5 VL 72B). Our analysis provides the first baseline on their readiness, revealing that BlueprintSymVL is highly discriminative. We pinpoint specific failure modes, including a notable degradation in cluttered environments, confusion when faced with visually similar distractors, and a concerning propensity to hallucinate symbols. These insights demonstrate that current VLMs are not yet suitable for autonomous deployment in blueprint analysis and are best integrated into human-in-the-loop workflows.

1. Introduction

The use of Vision Language Models (VLMs) is increasingly being explored for industrial automation. Due to their ability to process both visual and textual modalities, these multimodal models hold significant promise for enhancing efficiency in industrial sectors like engineering and manufacturing. A critical application in this domain is the automated extraction of data from engineering blueprints. These documents serve as the definitive source of information for engineering projects, containing essential data about design, maintenance, and operations. Thus, it is envisioned that VLMs could accelerate engineering workflows and assist in quality control [1].

However, the practical use of VLMs in the engineering industry requires overcoming significant domain-specific challenges. While foundation models are designed for general-purpose applicability, high-stakes industrial domains demand a higher standard of validation that goes beyond generalist capabilities. The goal is not just broad competence, but industrial-grade reliability in critical tasks. The visual domain of engineering blueprints is fundamentally different from the images on which these models are typically trained. Industrial engineering blueprints often feature many symbols appearing in close proximity,

* Corresponding author at: Eindhoven University of Technology, Mathematics and Computer Science, Groene Loper 3, Eindhoven, 5612 AE, the Netherlands.
E-mail addresses: v.shteriyanov1@tue.nl (V. Shteriyanov), rdzhusupova@mcdermott.com (R. Dzhusupova), j.bosch1@tue.nl (J. Bosch), helena.holmstrom.olsson@mau.se (H.H. Olsson).

https://doi.org/10.1016/j.rineng.2025.108171
Received 29 July 2025; Received in revised form 21 October 2025; Accepted 9 November 2025

Available online 13 November 2025

occlusions in the form of revision markups, as well as numerous distinct symbols that are visually similar [2]. The challenges of visual density and noise are well-established barriers to accurate object detection [3–5]. However, in the engineering industry, mistakes can lead to failures, safety incidents, and significant financial consequences [6]. There is little to no room for perceptual error.

This study is directly grounded in these challenges encountered at McDermott, a global provider of engineering and construction solutions to the energy industry. The lack of symbol standardization across engineering projects [7] creates a significant operational bottleneck, making the traditional approach of training a specialized computer vision model for each project prohibitively expensive and slow. This industrial context motivated our research to investigate a more scalable paradigm, using the generalist nature of VLMs to recognize symbols from a single visual example, on the fly. This potential shift from project-specific training to flexible, one-shot detection makes the need for a targeted, reliable evaluation methodology particularly acute.

This industrial need highlights a critical gap that existing VLM benchmarks are not suited for evaluating VLM reliability for this task. They typically fall into two categories: generalist benchmarks that focus on commonsense reasoning, or specialized benchmarks that target high-level reasoning using simplified or non-representative visual inputs. Neither is designed to systematically test a model's performance under the specific combination of density, occlusion, and visual similarity inherent to real-world engineering blueprints [2,5,8]. Compounding this issue, existing benchmark evaluation strategies often assume either reliable pre-trained knowledge of the visual domain or document standardization. These assumptions break down in blueprint analysis, where symbols are often non-standardized and project-specific.

These evaluation gaps motivate our research question: **How can state-of-the-art Vision Language Models be systematically evaluated for symbol recognition in engineering blueprints, and what is the impact of the domain-specific challenges of visual density, occlusion, and similarity on their reliability?**

To address this question, we introduce BlueprintSymVL,[1] a domain-specific benchmark designed to systematically evaluate VLM capabilities for symbol recognition in engineering blueprints [9]. The benchmark defines targeted test cases to evaluate VLM performance, focusing on: symbol recognition when no challenges are present, recognition in dense layouts, recognition among visually similar symbols, recognition in occluded regions, and the rejection of irrelevant inputs (negative testing). Uniquely, BlueprintSymVL employs a one-shot visual in-context querying strategy, in which a model is given a single visual exemplar of the target symbol at query time. This strategy avoids the pitfalls of relying on pre-trained knowledge or standardized documentation and to our knowledge, has not been used in prior VLM benchmarks.

The contributions of this paper are as follows:

- A Novel, Domain-Specific Benchmark (BlueprintSymVL): To our knowledge, we designed BlueprintSymVL as the first targeted benchmark to evaluate VLM symbol recognition in engineering blueprints that feature the challenges of clutter, occlusion, and visual similarity. The benchmark was explicitly designed as a strong discriminator: it establishes a solvable baseline on simple tasks, where leading models perform well, while systematically introducing challenges that expose critical failure modes and differentiate the capabilities of the models.
- A Robust Symbol Recognition Evaluation Method for VLMs: We introduce a novel evaluation method designed to assess VLM reliability for high-stakes industrial tasks. This method innovatively combines two key components: (1) a one-shot visual in-context querying setup, where models are guided by a visual example, eliminating reliance on unreliable prior knowledge and (2) a strict evaluation criterion that demands correctness on both symbol counts and their specific labels. Together, these ensure genuine symbol identification, establishing a rigorous standard for quality assurance in critical AI applications.
- A Comprehensive Benchmark of State-of-the-Art VLMs: We conducted a detailed benchmark of four leading VLMs using BlueprintSymVL: GPT-4o, Gemini 2.5 Pro, InternVL 2.5 78B, and Qwen 2.5 VL 72B. This benchmark provides the first baseline on the current readiness of these models for engineering blueprint symbol recognition.
- Actionable Insights into VLM Failure Modes: Our analysis pinpoints specific, critical failure modes in current VLMs, including a notable degradation in cluttered environments, confusion when faced with visually similar distractors, and a concerning propensity to hallucinate or misidentify symbols in negative test cases.

The remainder of this paper is organized as follows. We begin by surveying related work in Section 2. In Section 3, we present our core methodology, detailing the design of the BlueprintSymVL benchmark, our novel evaluation strategy, and the specifics of the experimental setup. Section 4 is dedicated to presenting and analyzing the results of our comprehensive model benchmarks. We then discuss our findings and their broader context in Section 5, before concluding in Section 6.

2. Related work

The evaluation of Vision Language Models (VLMs) is crucial for understanding their capabilities and limitations. However, we argue that existing general-purpose and expert-level benchmarks are insufficient for validating VLMs for deployment in the specific context of engineering blueprint analysis. This is due to fundamental differences in both the visual domain and the nature of the required reference information. To provide a systematic overview of existing research and situate the contribution of our work, we categorize and detail prominent benchmarks in Table 1.

Initially, benchmarks, such as VQA [10] and GQA [11], focused on measuring if a model could connect textual queries to visual content. The task these benchmarks target is open-ended visual question answering, where the goal is to answer questions about the image content. These benchmarks were built upon large-scale datasets of real-world images, such as COCO [12]. While foundational, their visual domain of natural scenes is fundamentally misaligned with the symbolic and abstract nature of engineering drawings.

To evaluate the advanced capabilities of modern VLMs, recent benchmarks have shifted focus from simple perception to complex reasoning. MMBench [13] is a benchmark that uses a set of challenging multiple-choice questions to evaluate the reasoning and perception capabilities of VLMs. The questions are built on a hierarchical skill tree pinpointing specific areas of failure. MMBench contains a diverse set of images, including real-world photographs, as well as diagrams and graphs. Similarly, SEED-Bench [14] focuses on compositional understanding, assessing a model's ability using a multiple-choice question format to comprehend spatial and temporal relationships in both images and videos. The visual content in SEED-Bench is focused on real-world scenes depicting interactions and events. Although these benchmarks are excellent for measuring breadth of knowledge, they lack the specificity to probe the critical failure modes of engineering blueprints. A model could score highly on MMBench but achieve low recognition accuracy on a dense blueprint because such specific, high-risk scenarios are not isolated and stress-tested.

Specialized benchmarks have also emerged to target specific skills and flaws. For instance, MathVista [15] tests visual mathematical reasoning. To address reliability, POPE uses questions about object presence to effectively measure model hallucinations. Benchmarks like ChartQA [16] and AI2D [17] evaluate models' ability to interpret chart and graph data. This involves extracting quantitative values, identify-

[1] https://doi.org/10.5281/zenodo.17250377.

Table 1
Comparison of Prominent VLM Benchmarks.

Category	Benchmark(s)	Primary Focus & Purpose	Task & Visual Domain
Foundational VQA	VQA, GQA	Establish basic vision-language grounding.	Open-ended Q&A on natural, real-world images (e.g., COCO).
Generalist Evaluation	MMBench, SEED-Bench	Evaluate VLM compositional and spatial-temporal reasoning, as well as perceptual skills.	Multiple-choice Q&A on diverse images, diagrams, and videos.
Specialized Skill & Flaw Probing	MathVista, POPE, ChartQA, AI2D, Image2Struct	Test specific capabilities: mathematical reasoning, hallucination detection, data chart interpretation, and structured data reconstruction.	Varied tasks on specialized visuals (e.g., math problems, charts, webpages).
Expert-Level Reasoning	MMMU, DesignQA	Evaluate specialized knowledge and multi-step reasoning in professional and academic contexts.	Answering college-level exam questions or verifying design compliance using 3D CAD models.
Symbol Recognition in Blueprints	**BlueprintSymVL (Our Work)**	**Symbol recognition under challenging conditions common in industrial blueprints.**	**Visual search and identification of 2D symbols in cluttered, occluded, and visually similar layouts.**

ing trends, and understanding relationships between visual elements. Meanwhile, Image2Struct [18] focuses on a different challenge: reconstructing structured data, such as generating the HTML or LaTeX code from an image of a webpage or formula. Together, these benchmarks probe specific strengths and weaknesses. The existence of these benchmarks confirms that specialized domains require specialized evaluation tools. However, none of them addresses the unique visual syntax and semantics of engineering blueprints.

Recently, the focus has shifted to testing VLMs on tasks requiring deep, specialized knowledge. The MMMU benchmark [19] is designed to evaluate multimodal models on tasks requiring college-level reasoning. MMMU consists of multimodal questions from college exams, quizzes, and textbooks, covering six major disciplines. Another example is DesignQA [20], a benchmark designed to test VLMs in comprehending technical documentation. It evaluates a model's ability to connect a 3D CAD model of vehicles to a comprehensive, 140-page textual rulebook provided as context, to perform rule extraction, comprehension, and compliance, including component identification. Nevertheless, the benchmark design simplifies critical real-world challenges. Perceptual difficulties in DesignQA are intentionally minimized through the use of highlighted components or constrained, close-up views, which avoid common engineering obstacles like clutter, occlusion, and high visual similarity. More fundamentally, this methodology relies on the existence of a standardized rulebook. However, this assumption does not hold for many real-world engineering drawings, where symbology is inconsistent and a universal standard for all visual elements is often unavailable. The task in DesignQA is to ground visual data in a provided text, while the task for many blueprint analyses is to find visual patterns in the absence of a reliable textual guide.

This disconnect from industrial realities highlights a broader need to consider the paradigms used for symbol recognition. Beyond evaluating VLMs in isolation, it is important to situate them relative to traditional methods. Historically, industrial pipelines have employed techniques like template matching or, more recently, specialized deep learning object detectors [21–24]. While these detectors can achieve high accuracy for a fixed set of symbols, they require extensive, project-specific training data to handle variations in symbology, a process that is both costly and inflexible. Template matching, meanwhile, has been shown to be suboptimal for P&IDs, as it struggles with the subtle visual variations between different symbol classes across drawings [25]. The core limitation of these approaches is their reliance on either brittle matching or costly retraining for each new project.

Our work is motivated by a different paradigm: visual in-context learning, which has emerged as a powerful capability of modern VLMs. This approach involves guiding models with on-the-fly examples. The concept of providing visual exemplars has proven effective for various tasks, such as in-context segmentation [26], and various visual prompting techniques have been proposed to improve VLM performance [27]. However, many of these methods depend on an intermediate recognition step, which uses models like SAM or CLIP to generate masks or regions of interest before the VLM performs its final analysis [28–31]. Such multistage pipelines can be brittle in specialized domains like engineering drawings, where intermediate models may fail. While techniques for in-context learning exist, a systematic benchmark to evaluate the end-to-end reliability of VLMs for industrial symbol recognition has been missing.

As our review shows, a critical evaluation gap exists, defined by three key factors: the visual domain, the evaluation strategy, and the limitations of alternative solution paradigms. No existing benchmark systematically tests VLM performance under the perceptual challenges specific to blueprints, such as cluttered layouts, occlusions, and high visual similarity between symbols. The importance of these factors is emphasized by previous research focusing on object recognition [3,4]. For instance, our earlier study [5] demonstrated that occlusions are a primary cause of performance degradation for traditional object detection models on blueprints.

Equally important, existing evaluation methods in benchmarks are misaligned with the practical realities of this domain. Most benchmarks adopt a zero-shot evaluation approach, where models are expected to recognize visual concepts based solely on their pre-trained knowledge. Other benchmarks, such as DesignQA, circumvent this by grounding model outputs in structured documentation, such as standardized rulebooks. However, these assumptions fail in blueprint domains where symbols are often non-standardized, project-specific, and visually ambiguous, and where no reliable symbolic reference can be assumed.

The limitations of existing benchmarks and the inflexibility of traditional methods became evident while exploring a more flexible approach to industrial symbol detection: leveraging generalist VLMs to identify symbols from a visual example, thereby eliminating the need for costly retraining with each new project's symbol representations. However, this promising workflow cannot be validated using current tools. This disconnect between industrial demands and academic evaluation directly motivated our work.

To address these research gaps, we developed the BlueprintSymVL benchmark. It combines a curated set of blueprint-specific visual challenges with a novel one-shot visual in-context querying evaluation strategy. In our setup, models are shown a single visual exemplar of the target symbol, and are required to identify all matching instances in a new blueprint region. This setup avoids assumptions of prior symbol knowledge and of standardization. Instead, it evaluates a model's ability to generalize visually from context in the way a human analyst might when given a drawing legend. To our knowledge, no prior benchmark evaluates VLMs in this way, making BlueprintSymVL the first to provide a realistic and discriminative assessment of VLM readiness for high-stakes industrial visual tasks.

3. The BlueprintSymVL benchmark

This research was executed at McDermott, a global provider of engineering and construction solutions to the energy industry. The methodology utilized was Action Research [32], which is well-suited for investigating and solving practical problems within an organizational setting. As the researchers are members of the company's AI team, this approach allowed for direct engagement with engineering domain experts. This process revealed the critical industrial challenge our work addresses: the high cost and inflexibility of retraining custom computer vision models for each project's unique set of visual symbols. Our research, therefore, investigates if generalist VLMs can offer a more scalable solution by detecting symbols based on a single visual example.

This investigation led our team to develop BlueprintSymVL[2] [9], a diagnostic benchmark framework designed to evaluate VLM performance on engineering symbol recognition. It is formally defined by three core components:

- A Curated Dataset: A set of blueprint regions selected with domain expertise to represent distinct, real-world challenges. The scenarios include: (1) regions without challenges, (2) dense regions with visual clutter, (3) regions with visually similar distractors, (4) negative cases where the target is absent, and (5) occluded versions of regions to test robustness.
- A Standardized Evaluation Strategy: The benchmark employs a one-shot, in-context visual querying strategy. A model is provided with a single, clear visual example of the target symbol and prompted to identify all matching instances in a given query region, forcing it to learn from context rather than pre-trained knowledge.
- A Strict Performance Criterion: Performance is measured by requiring correctness on both the total symbol count and the specific text labels associated with each instance. This strict criterion ensures genuine recognition.

3.1. BlueprintSymVL dataset

To ensure the industrial relevance and validity of our benchmark, the selection of all test case regions was conducted in close collaboration with senior engineering experts at McDermott. These experts reviewed and validated that the chosen scenarios for symbol density, visual similarity, and occlusion were highly representative of the most common and challenging situations encountered in real-world projects. The following sections explain the benchmark data collection and subsequent design.

3.1.1. Data collection and pre-processing

The benchmark data is derived from a public, synthetic Piping and Instrumentation Diagram (P&ID) dataset [7], which contains 500 annotated blueprints. This source was selected for its rich set of visually similar symbols, a common challenge in blueprint analysis.

Our pre-processing workflow involved two main steps. First, each drawing is split into 16 regions with a size of 860x860 pixels. An example region is shown in Fig. 1. This approach is a common strategy for handling high-resolution documents [33] in computer vision tasks. Second, the original annotations, which recorded the position of each symbol and its type via an index identifier, are transformed into human-readable semantic labels. Our annotation files contain the actual names of specific symbols in a region. These symbols are: "gate valve", "butterfly valve", "check valve", "reducer", "spacer", "ball valve", "globe valve", "needle valve", "open spectacle blind", and "closed spectacle blind". The dataset is particularly suitable for our goals due to its inclusion of challenging, visually similar symbol groups, such as the various valve types, as well as pairs like ("reducer", "check valve") and ("spacer", "spectacle blind"). The symbols are shown in Fig. 2.

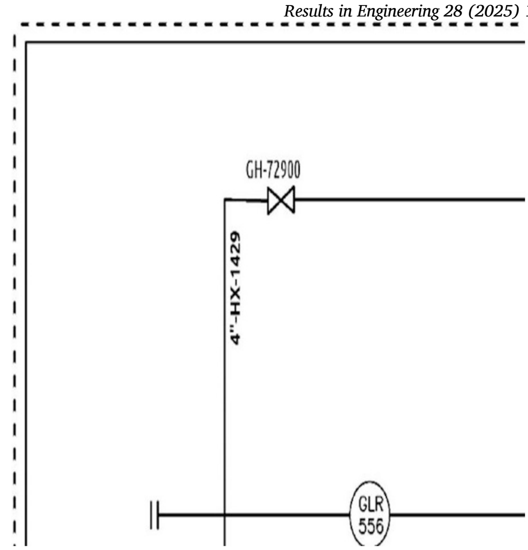

Fig. 1. An example blueprint region.

3.1.2. BlueprintSymVL design

BlueprintSymVL is designed for targeted analysis. It focuses on five symbol classes of interest: "gate valve", "butterfly valve", "check valve", "reducer", "spacer". For each of these classes, we sampled regions to represent the four evaluation scenarios mentioned previously with the following specific criteria:

- Baseline: A region containing the target symbol with few or no other symbols present. No similar symbols to the target are present.
- Dense Region: We define a region as dense if it contains at least 10 symbols, based on our industrial experience, given its 860x860 size.
- Similar Symbols: A region containing 2 or 3 symbols in total, including the target and at least one visually similar distractor.
- Negative Case: A region that does not contain the target symbol, used to evaluate the model's ability to avoid false positives. The regions may contain similar symbols.

For each combination of class and scenario, we selected 5 representative regions. An example region for each of these cases for the "butterfly valve" is shown in Fig. 3. The choice of 5 regions per scenario is due to experiment design and dataset limitations. For several scenario-symbol combinations, the dataset itself yielded no more than five truly representative regions. Nevertheless, this selection allowed us to balance between capturing representative diversity and the significant manual effort required to identify and verify these cases across the entire source dataset. This curation process resulted in a set of 100 unique blueprint regions (5 classes $\times$ 4 scenarios $\times$ 5 regions), designed to be representative of specific, recurring challenges in engineering drawing analysis.

To evaluate model robustness against common visual obstructions, we generated an occluded counterpart for each of the 100 curated regions. This design choice is directly motivated by our previous findings [5], which empirically demonstrated that occlusions are a significant factor reducing the performance of automated symbol detectors in engineering drawings. Occlusions were synthetically introduced using shapes that resemble revision clouds, which are graphical elements used in real-world engineering drawings to mark design changes. These occlusions partially or fully cover the symbols. An example of a clean region and its occluded variant is shown in Fig. 4. This augmented set allows for a direct comparison of VLM performance on clean versus oc-

[2] https://doi.org/10.5281/zenodo.17250377.

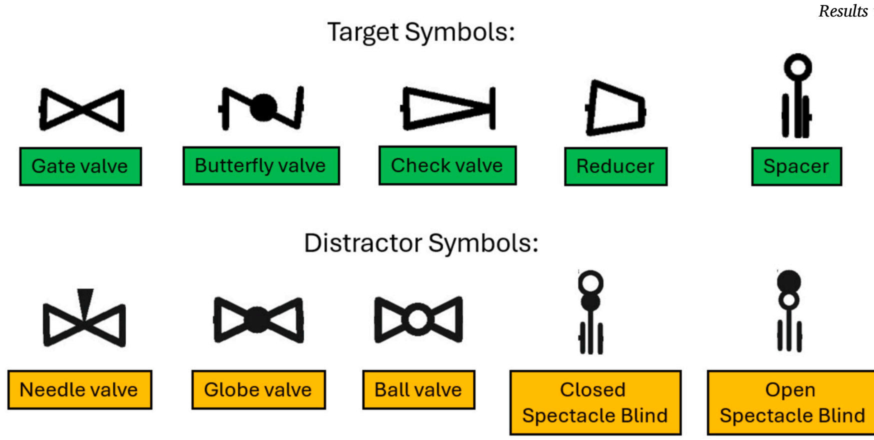

Fig. 2. Benchmark symbols.

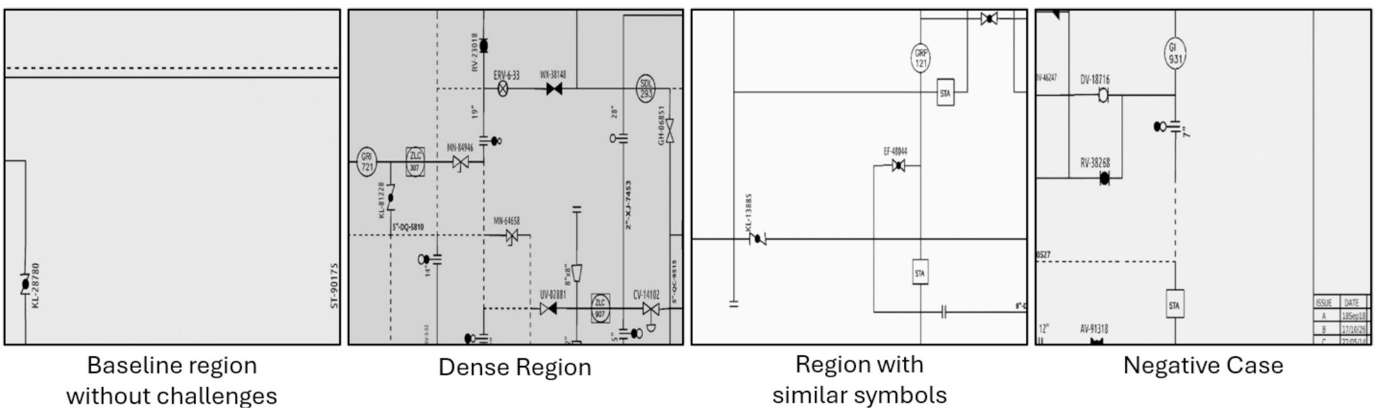

Fig. 3. Example regions for the “butterfly valve” symbol.

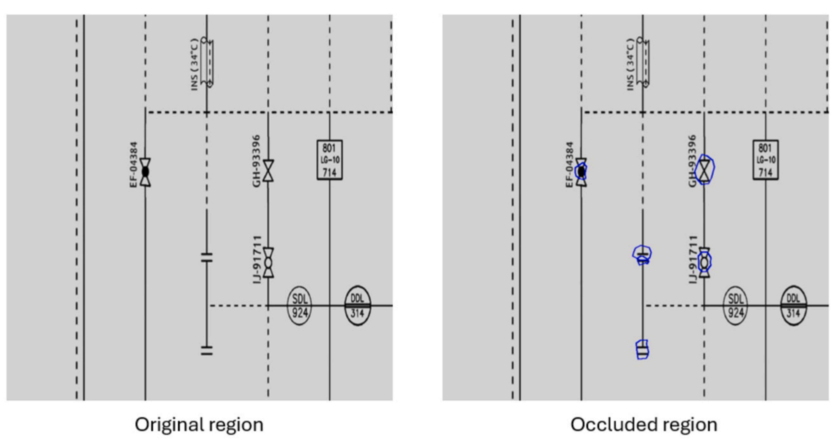

Fig. 4. Example of a clean region and its occluded variant.

cluded symbols, thereby testing the model’s resilience to the type of visual degradation frequently encountered in real engineering drawings.

Thus, the BlueprintSymVL benchmark comprises 200 regions: 100 original and 100 corresponding occluded regions. To provide the necessary components for evaluation, the benchmark includes structured annotation files for each of the 200 regions, which consist of a query file and a ground-truth answer file.

Each region is paired with a query file that defines the primary task. This file contains a single question following the format: *“How many X symbols are in this image?”*, where X is the target symbol class for that region (e.g., spacer). Additionally, we create a ground truth answer file, which enables a more rigorous evaluation than simple counting. The ground truth file for each region contains two components:

- Symbol Count: The correct answer to the query (e.g., 3).
- Label Identifiers: A list of the specific text labels associated with each individual target symbol instance. These labels, such as a symbol’s size (e.g., “4″”) or a unique alphanumeric tag (e.g., “GH-29744”), are extracted directly from the text located near the symbol in the original drawing.

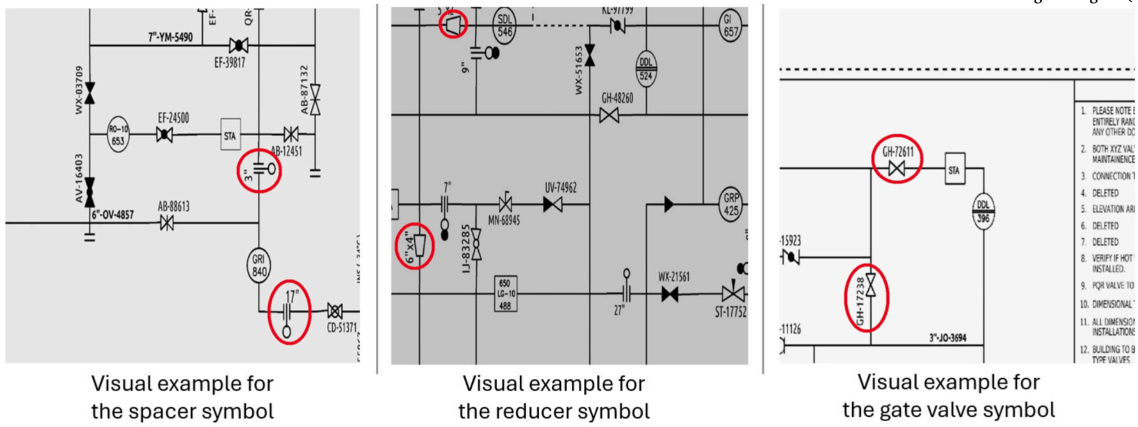

Fig. 5. The visual examples for the spacer, reducer, and gate valve symbols.

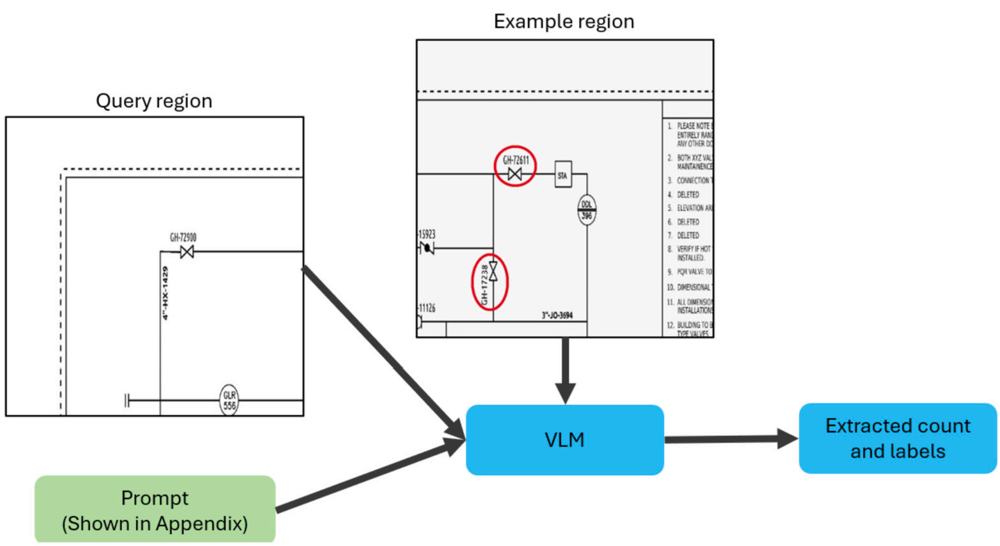

Fig. 6. Visualization of the querying strategy.

The ground truth labels serve as the basis for our evaluation protocol, ensuring that a high score reflects genuine recognition ability, not a lucky guess.

In addition to these files, the benchmark provides a set of 5 dedicated exemplar regions, one for each of the 5 target symbol classes. To provide a clear and robust visual definition, each of these exemplar regions was intentionally selected to contain two instances of the target symbol, all of which are clearly highlighted with red circles. The visual examples for the spacer, reducer, and gate valve symbols are shown in Fig. 5.

These example regions serve as the definitive visual reference for the evaluation protocol, ensuring a consistent definition of the target symbol is provided to the VLMs. In total, the final benchmark provides all the necessary data for a precise and replicable evaluation of VLM performance on blueprint symbol recognition.

3.2. Evaluation

The following sections explain the evaluation querying strategy and the evaluation metrics.

3.2.1. One-shot visual querying via in-context example

BlueprintSymVL's evaluation framework is structured as a one-shot in-context visual querying task, which is shown in Fig. 6. Unlike other benchmarks employing a zero-shot strategy, this approach is designed to provide a clear visual definition of the target at query time instead of relying on a model's pre-existing knowledge of such symbols, which can be inconsistent or missing. This makes it particularly advantageous in the engineering domain, where the visual representations of symbols may vary. It stands in contrast to evaluations in domains that rely on standardized definitions.

For each task, the VLM is presented with two image regions:

- An example region: The region for the target symbol class, where instances are highlighted with red circles. This serves to unambiguously define the target for the model. The highlighting technique has been shown to be effective in zero-shot localization tasks [27].
- A query region: The actual benchmark region for which the model must perform the task.

This evaluation setup tests the model's ability to generalize from a single in-context visual example, which is a capability underexplored in current VLM benchmarks.

The VLM is then guided by a detailed prompt template, which is consistent across all test cases. The query from the region's question file (e.g., "How many X symbols are in this image?") is inserted into this prompt template. Thus, this template instructs the model to count all instances of the symbol of interest X. Additionally, the template prompt instructs the VLM to extract the text labels located adjacent to each identified symbol. The prompt also requires a structured response that contains the final count and a list of associated text labels. This consistent template structure ensures that the instructions and output format

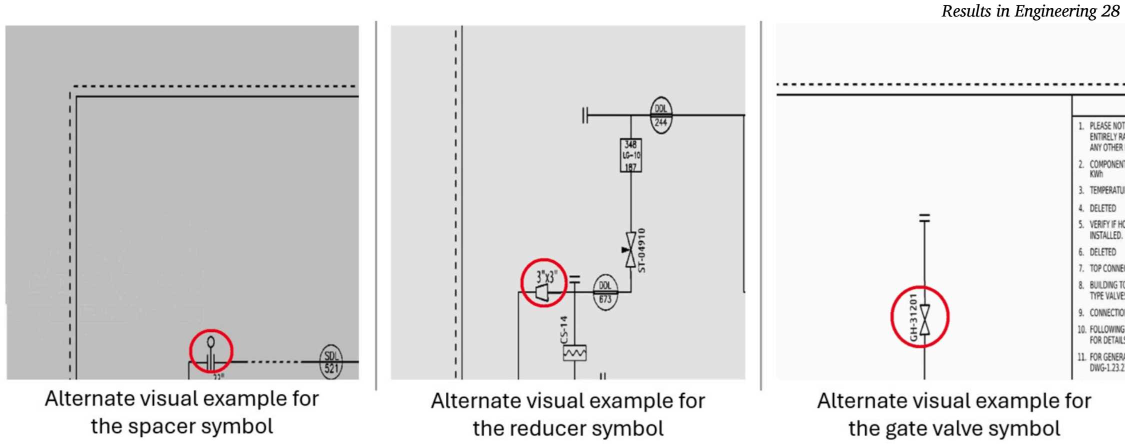

Fig. 7. The alternate visual examples for the spacer, reducer, and gate valve symbols.

requirements remain identical for every evaluation run. The complete prompt template used in our experiments is provided in the Appendix.

3.2.2. Evaluation metrics

Central to our evaluation is the use of fuzzy string matching for comparing the VLM's extracted labels against the ground truth. This technique calculates a normalized similarity score between two strings, accounting for minor character-level differences. We employ this method because VLMs can make trivial character recognition errors (e.g., mistaking "O" for "0" or "1" for "I") that do not reflect a failure to identify the correct symbol. We set the similarity threshold at 85% based on empirical analysis. This value proved effective at accommodating minor errors while being sufficiently strict to prevent incorrect matches between distinct labels.

Based on this matching logic, we evaluate performance using three metrics:

- Exact Match Rate (EMR): This custom-defined metric measures the percentage of benchmark regions where the VLM's output is entirely correct. A response is considered correct if the VLM reports both the correct final count and a complete set of labels that all meet the 85% fuzzy matching threshold against the ground truth. This is crucial for practical engineering applications where errors are not tolerable.
- Precision and Recall: To measure symbol recognition performance, we calculate precision and recall from the set of extracted labels. A True Positive (TP) is a predicted label that fuzzy-matches a ground truth label.
- False Positives in Negative Regions: To measure VLM's reliability, we count the number of False Positive (FP) labels generated for the "Negative Case" scenarios. Since these regions contain no target symbols, any extracted label is an error.

These metrics provide a complementary view of model performance. While Precision and Recall measure aggregate accuracy across the entire dataset, EMR assesses per-region reliability by measuring the percentage of regions where the model's output is flawless. The distinction is critical, as the gap between these scores is a key diagnostic indicator. A model can exhibit high overall Precision and Recall, suggesting strong general capabilities, yet a low EMR reveals that it rarely recognizes any single region perfectly. This highlights a critical lack of robustness, as real-world applications demand not just general accuracy but consistent, error-free performance.

3.2.3. Experiment setup

To demonstrate the utility of the BlueprintSymVL benchmark, we evaluated a selection of four state-of-the-art VLMs. This selection was chosen to represent the current landscape of VLM development and includes both leading proprietary and open-source models. The proprietary models evaluated are GPT-4o [34] and Gemini 2.5 Pro [35]. Complementing these, the open-source models selected are InternVL 2.5 78B [36] and Qwen 2.5 VL 72B [37].

To enable a direct and fair comparison of each model's intrinsic capabilities, we established a standardized inference configuration, a common practice in foundation model benchmarking [13]. To do so, we standardized key inference parameters across all models. We set the image detail parameter to "high" to provide the models with the maximum visual information from the blueprint regions. Furthermore, the models' temperature parameter was set to 0. This setting minimizes randomness in the models' outputs, promoting deterministic and consistent response behavior, which is essential for a controlled and repeatable evaluation. This experimental design allows for a direct comparison of these four models on the specific challenges curated in our benchmark. Furthermore, this study intentionally evaluates the models using visual in-context learning exclusively, without any domain-specific fine-tuning. This methodological choice is essential for evaluating the inherent, out-of-the-box visual in-context learning capabilities of these models. This evaluation directly tests our core hypothesis: whether generalist VLMs can offer a more agile alternative to the costly, project-specific retraining required for traditional object detectors.

3.2.4. Ablation study design

To validate the robustness of our evaluation protocol and address the importance of prompt design and visual example selection, we designed an ablation study to be conducted on the top-performing model identified in our main benchmark analysis. The study compares our standard setup against two alternative conditions:

- Alternate Visual Example: This condition tests the protocol's sensitivity to the specific visual example provided. Our original examples contain at least two annotated instances of the target symbol. In contrast, the alternate examples were selected to contain only a single annotated instance. This modification allows us to test not only the robustness to the example's appearance but also whether the number of annotated instances within the example image affects model performance. The alternate visual examples for the spacer, reducer, and gate valve symbols are shown in Fig. 7.
- Symbol Crop Visual Example: This condition tests the impact of the visual prompt's format. The context-rich exemplar image was replaced with a tight, decontextualized crop of only the target symbol, as shown in Fig. 2.

4. Results

We evaluated the four VLMs on the BlueprintSymVL benchmark. The results validate the benchmark as an effective diagnostic tool, as

Table 2
Overall Results.

	GPT-4o			Gemini 2.5 Pro			InternVL 2.5 78B			Qwen 2.5 VL 72B		
Metric	EMR %	R %	P %	EMR %	R %	P %	EMR %	R %	P %	EMR %	R %	P %
Total Overall	30	63.02	37.58	50.5	74.48	53.16	4.5	26.69	15.32	40.5	71.35	43.49

EMR: Exact Match Rate; R: Recall; P: Precision.

Table 3
Performance on Baseline Regions, Dense Regions, Regions with similar symbols.

	GPT-4o			Gemini 2.5 Pro			InternVL 2.5 78B			Qwen 2.5 VL 72B		
Scenario Type	EMR %	R %	P %	EMR %	R %	P %	EMR %	R %	P %	EMR %	R %	P %
Baseline	72	86	71.67	86	88	91.67	16	42	25.61	86	90	90
Dense Regions	26	51.14	41.67	32	65.91	60.42	2	15.91	12.84	10	55.68	35.77
Similar Symbols	18	61.11	36.67	38	75.93	53.25	0	40.74	24.44	28	79.63	47.78

EMR: Exact Match Rate; R: Recall; P: Precision.

it clearly differentiates the capabilities of these state-of-the-art models. More importantly, the findings reveal systemic weaknesses in current VLMs when confronted with the domain-specific challenges of blueprint analysis that motivated this work.

4.1. Overall results

Table 2 summarizes the aggregate performance of each model. A clear performance hierarchy emerged: Gemini 2.5 Pro stands out as the top performer with the highest EMR of 50.5% and the best Recall (74.48%) and Precision (53.16%). Qwen 2.5 VL 72B is a strong second with an EMR of 40.5%, followed by GPT-4o at 30% EMR. InternVL 2.5 78B struggled significantly across all metrics, achieving an EMR of only 4.5%, indicating it is not yet optimized for this type of specialized task. This wide performance spread demonstrates that BlueprintSymVL is a highly effective model discriminator.

A critical pattern observed across all models is the large disparity between Recall and Precision. For instance, Qwen 2.5 VL 72B's 71.35% Recall is paired with a much lower 43.49% Precision. This indicates a general tendency for current VLMs to produce a high number of false positives, which is a key weakness the benchmark successfully highlights. This finding directly relates to the high-stakes nature of the domain, where low precision is particularly problematic, as it could lead to critical errors in downstream processes.

4.2. Performance across scenarios

The results for simple regions, dense regions, and regions with similar symbols are shown in Table 3. In the simple regions scenario, Gemini 2.5 Pro and Qwen 2.5 VL 72B both excelled, achieving 86% EMR, with Gemini demonstrating exceptionally high precision (91.67%). GPT-4o was competitive at 72% EMR. However, all models' performance dropped significantly in more complex scenarios. In Dense Regions, the EMR of Gemini 2.5 Pro dropped to 32% and GPT-4o's fell to 26%. Qwen's performance collapsed to just 10% EMR, suggesting it is particularly vulnerable to clutter. The Similar Symbols scenario also degraded performance for all models. Here, Gemini led with 38% EMR, while Qwen showed the highest Recall (79.63%) but lower EMR (28%). InternVL achieved only 16% EMR on simple regions and fell to 2% and 0% in the Dense Regions and Similar Symbols scenarios, respectively.

To measure model reliability, we tracked the false positives (FPs) generated in Negative Case scenarios where no target symbol was present. The results are shown in Table 4. Qwen 2.5 VL 72B proved most reliable, generating only 38 FPs. Gemini 2.5 Pro was next with 48 FPs. In contrast, GPT-4o produced 64 FPs, showing it is less reliable in this scenario. InternVL was the least reliable, generating 91 FPs, which aligns with its low overall precision and EMR scores. This metric is crucial for practical applications, as it quantifies a model's trustworthiness when a target is absent. The high FP count for some models underscores a dangerous behavior in a high-stakes setting.

Table 4
Performance on Regions without target symbols (Negative Case).

	GPT-4o	Gemini 2.5 Pro	InternVL 2.5 78B	Qwen 2.5 VL 72B
FP	64	48	91	38

FP: False Positive.

4.3. Per-symbol analysis

To understand if model performance is uniform or dependent on the target's visual characteristics, we analyzed the results on a per-symbol basis. This is critical for assessing the risks in a domain with non-standardized symbology. The results in Table 5 reveal a significant performance variance across different symbols, even for the best-performing models. Gemini 2.5 Pro demonstrated strong performance on specific symbols, achieving a 67.5% EMR on check and 65% on reducer symbols. Qwen also performed well on reducer (47.5% EMR), while GPT-4o's performance was more modest across all types, peaking at 35% EMR for butterfly valve symbols. Performance on gate valve symbols was a challenge for all top models, with EMRs ranging from 30-37.5%. This disparity demonstrates that the visual complexity of certain symbols presents a common challenge for current VLMs. InternVL 2.5 78B achieved low scores on all symbols and notably failed completely on reducer and spacer tasks (0% EMR), providing a clear signal for targeted model improvement. Finally, all models generally exhibited higher recall rates compared to precision rates for each symbol.

4.4. Robustness to occlusions

This experiment directly tests model performance against occlusion, a common challenge in real-world blueprints due to revision markups. As shown in Table 6, the introduction of occlusions consistently and significantly degraded performance for all models. Gemini 2.5 Pro's EMR saw the largest drop, from 63% on regular images to 38% on occluded ones. GPT-4o's EMR fell from 36% to 24%, and Qwen 2.5 VL 72B's dropped from 43% to 38%. Despite its significant drop, Gemini and Qwen maintained the highest EMR on the occluded set, demonstrating better, albeit still compromised, performance under occluded conditions. The consistent performance degradation across all leading models confirms that occlusions remain a significant challenge and validates this aspect of the benchmark for testing model robustness.

Table 5
Per-Symbol Results.

	GPT-4o			Gemini 2.5 Pro			InternVL 2.5 78B			Qwen 2.5 VL 72B		
Symbol Type	EMR %	R %	P %	EMR %	R %	P %	EMR %	R %	P %	EMR %	R %	P %
butterfly	35	65.62	35.59	35	65.62	40.38	5	46.88	20.83	40	65.62	39.62
check	30	72.22	38.81	67.5	86.11	77.5	12.5	63.89	32.86	40	94.44	40.96
gate	30	68.52	47.44	37.5	68.52	49.33	5	18.52	13.7	32.5	59.26	44.44
reducer	32.5	55.56	32.79	65	77.78	68.29	0	5.56	2.6	47.5	72.22	49.06
spacer	22.5	50	29.82	47.5	76.47	42.62	0	20.59	8.75	42.5	70.59	44.44

EMR: Exact Match Rate; R: Recall; P: Precision.

Table 6
Robustness to Occlusions Results.

	GPT-4o			Gemini 2.5 Pro			InternVL 2.5 78B			Qwen 2.5 VL 72B		
Main Type	EMR %	R %	P %	EMR %	R %	P %	EMR %	R %	P %	EMR %	R %	P %
Regular	36	67.71	42.48	63	77.08	58.73	6	29.17	15.22	43	76.04	46.5
Occluded	24	58.33	33.14	38	71.88	48.25	3	30.21	15.43	38	66.67	40.51

EMR: Exact Match Rate; R: Recall; P: Precision.

Table 7
Ablation study on exemplar selection using Gemini 2.5 Pro.

	EMR %	R %	P %
Original Visual Example	50.5	74.48	53.16
Alternate Visual Example	50.5	73.44	48.96
Symbol Crop Visual Example	36.5	66.67	35.07

EMR: Exact Match Rate; R: Recall; P: Precision.

4.5. Ablation study

The results of our ablation study, designed to test the stability and design choices of our evaluation protocol, are presented in Table 7.

The findings provide two critical insights. First, the identical Exact Match Rate (50.5%) for the Alternate Visual Example is a powerful validation of our protocol's robustness. It demonstrates that the model's performance is not dependent on a specific exemplar. This result is particularly noteworthy given that the alternate exemplar contained only a single annotated symbol compared to the two in our original setup. The stable performance suggests that the model effectively learns the visual concept from a single annotated instance, and providing additional instances within the same example image offers limited benefit for this task. Second, the significant degradation in performance across all metrics for the Symbol Crop Visual Example confirms the importance of visual context in the prompt design. Removing the symbol from its surrounding environment made it a less effective guide for the model, leading to lower reliability. This finding empirically validates our design choice to use context-rich highlighted regions, which better enable the model's pattern recognition capabilities.

4.6. Summary

Our evaluation on the BlueprintSymVL benchmark provides a multifaceted view of current VLM capabilities for symbol recognition in engineering blueprints, highlighting their potential benefits, critical weaknesses, and distinct performance profiles.

Strengths and Potential: The results reveal that leading VLMs possess foundational capabilities for this task. Under ideal conditions, such as identifying symbols in simple, uncluttered regions, models like Gemini 2.5 Pro and Qwen 2.5 VL 72B demonstrated high effectiveness, achieving 86% EMR. This indicates a strong potential for automating symbol recognition in straightforward scenarios. Furthermore, the generally high recall rates across models suggest an aptitude for locating potential candidate symbols, which is a crucial first step in any analysis pipeline.

Systemic Weaknesses: Despite this potential, the benchmark uncovered systemic weaknesses that currently limit the reliability of VLMs for this domain. The most critical issue is a universal lack of precision, leading to a high rate of false positives. This problem is exacerbated in more challenging, yet common, situations. Performance degraded severely when models were confronted with visual clutter in dense regions, visually similar symbols, or occlusions from revision markups. These weaknesses are particularly concerning in a high-stakes domain where accuracy is paramount.

Similarities: All evaluated models exhibited a significant gap between their recall and precision scores, pointing to a shared tendency to generate false positives. Likewise, all models proved vulnerable to complexity, with performance dropping consistently in the Dense Regions, Similar Symbols, and Occlusion scenarios.

Differences: A clear performance ranking emerged: Gemini 2.5 Pro stood out as the top performer, distinguished by its superior precision and overall EMR. Qwen 2.5 VL 72B was a strong second, proving the most reliable in negative cases (fewest false positives) but showing a unique vulnerability to visual clutter. GPT-4o was a competitive third, while InternVL 2.5 78B struggled across all metrics, indicating it is not yet optimized for this domain.

Collectively, these findings validate BlueprintSymVL as an effective diagnostic tool. Our ablation studies further confirmed the robustness of our evaluation protocol, providing a stable foundation for these conclusions. While today's VLMs show promise, they require significant improvements in precision and robustness to complexity before they can be considered truly reliable for engineering blueprint analysis.

5. Discussion

Our results confirm BlueprintSymVL as a powerful and discriminative benchmark, establishing a clear performance hierarchy with Gemini 2.5 Pro as the top performer. However, the benchmark's primary value lies not just in ranking models, but in its ability to pinpoint distinct, model-specific failure modes. For example, it highlighted Qwen 2.5 VL 72B's sharp collapse in cluttered environments and GPT-4o's higher tendency to generate false positives in negative test cases. It also quantified the universal challenge of occlusions, causing performance degradation across all models. However, it revealed different degrees of vulnerability, with Gemini 2.5 Pro seeing the largest absolute drop in EMR, while Qwen 2.5 VL 72B's performance remained more stable. By identifying these distinct, model-specific failure modes, BlueprintSymVL proves its value as an essential diagnostic tool.

Beyond these model-specific diagnostics, the results provide strong empirical evidence of a systemic performance collapse when models confront industrial complexity. This is observable in the stark degradation of performance as challenges were introduced. Focusing on the three top-performing models, their average Exact Match Rate (EMR) from a high of 81.3% in simple baseline scenarios plummeted to just 22.7% in dense regions and to 28.0% when facing visually similar distractors. These figures represent absolute drops of 58.6 and 53.3 percentage points, respectively, corresponding to relative performance collapses of over 72% and 65%. This breakdown in reliability is mirrored in their precision scores. From an average of 84.4% in the baseline, precision fell to 46.0% in dense regions and 45.9% when faced with similar symbols. This is a nearly identical drop to less than half their initial accuracy. These are not minor fluctuations, but rather large-scale effects demonstrating consistent failure modes in complex settings.

Based on these findings, we suggest that while VLMs hold promise, they are not yet ready for autonomous use in blueprint analysis. The best-performing model still produced an incorrect output in over half of the test cases, an error rate unacceptable in an industry where mistakes have severe safety and financial consequences. This unreliability stems from a large gap between high recall and low precision, resulting in a high number of false positives across all VLMs. Consequently, we recommend that practitioners treat current VLMs as assistive tools within a human-in-the-loop workflow, rather than as standalone decision-makers.

While these performance limitations are significant, they must be contextualized against the drawbacks of traditional industrial approaches. A specialized object detector fine-tuned on a large, project-specific dataset would likely achieve superior precision and recall for a known set of symbols. However, this accuracy comes at the cost of significant operational inflexibility. Each new project with different visual standards requires a costly and time-consuming cycle of data collection, annotation, and model retraining. The one-shot VLM paradigm evaluated by BlueprintSymVL, in contrast, represents a fundamental shift towards operational agility. The primary practical gain lies not in its current accuracy, but in its potential to dramatically reduce the cost and time of adaptation. Instead of a full retraining pipeline, a VLM can be adapted to a new project's symbology with a single visual example, which is a task an engineer can complete in seconds. This potential directly addresses the critical operational bottleneck of project-specific model development that motivated our research.

This focus on evaluating a new, more flexible paradigm explains the motivation behind our work, which addresses the disconnect between how VLMs are typically evaluated and the rigorous validation required for real-world engineering applications. Existing benchmarks fail to represent the domain's unique visual challenges, such as high symbol density, occlusion, and similarity. Furthermore, their evaluation strategies are misaligned with industrial practice. These approaches typically rely on either unreliable pre-existing knowledge or standardized documentation, neither of which can be assumed in the diverse and non-standardized domain of engineering blueprints. In contrast, BlueprintSymVL is designed to overcome these limitations. It introduces carefully curated scenarios that test these domain-specific visual challenges, enabling evaluation where other benchmarks fail. Crucially, it employs a novel one-shot visual in-context querying strategy, in which the model is given a single visual exemplar at query time and must generalize from that context to identify matching instances in a new region. The robustness and specific design choices of this strategy were validated through a dedicated ablation study. This study provided two key insights. First, the protocol is robust to the specific exemplar image chosen. When using an alternate visual example, Gemini 2.5 Pro's EMR remained stable at 50.5%, confirming that performance is unaffected by the number of annotated instances within the exemplar. Second, the study confirmed the importance of visual context in the prompt design. When the exemplar symbol was cropped from its surrounding context, EMR dropped sharply to 36.5% and precision nearly halved (from 53.2% to 35.1%), demonstrating that decontextualized prompts significantly impair model performance. These findings affirm the effectiveness of our design choices. To our knowledge, this evaluation approach has not been explored in prior VLM benchmarks. Unlike traditional setups, our method provides a visually grounded, context-driven task that better reflects how human analysts interpret unfamiliar symbols in practical settings. This makes BlueprintSymVL uniquely capable of exposing model limitations that other benchmarks fail to reveal.

While the BlueprintSymVL benchmark specifically addresses symbol recognition in engineering blueprints, its underlying methodological framework and evaluation strategy offer a highly generalizable template for assessing VLMs across diverse domain-specific industries. Our approach emphasizes a diagnostic assessment crucial for real-world applications where objects are often non-standardized. This generalizability stems from several key aspects. Our evaluation strategy is designed to mirror how a human expert adapts to non-standard symbols. By using a one-shot, in-context task, we test a VLM's ability to learn a novel visual concept from a single example, rather than relying on pre-trained knowledge of a fixed symbol library. This approach is essential for industrial settings where standards are inconsistent or project-specific. The benchmark's scenarios, such as visual clutter and symbol similarity, are drawn from real-world challenges identified by domain experts, ensuring the test is not just novel but also practical. Finally, the multi-faceted evaluation provides diagnostic insights into a model's true reliability versus mere aggregate accuracy. This comprehensive framework offers a path for industries seeking to deploy robust VLMs in their unique, high-stakes visual domains.

5.1. Limitations and future work

While this study validates BlueprintSymVL as an effective diagnostic tool, its limitations define key directions for future work. These limitations arise from the study's industrial context, the benchmark's data source and design, and the evaluation methodology.

First, the research is framed by a specific industrial challenge identified within McDermott. Our Action Research methodology ensured the benchmark addresses a critical real-world problem: the high cost of retraining custom computer vision models for each project. Consequently, the selection of the five symbol classes and the focus on certain visual challenges (e.g., clutter, similarity) were guided by the priorities of this specific context. While this grounding ensures practical relevance, the challenges may differ in other engineering disciplines or companies. Future work should involve validating and expanding the benchmark in collaboration with experts from other domains to ensure broader applicability.

Second, a notable constraint is the benchmark's current scale. With 200 manually curated regions, its size is modest compared to large-scale generalist benchmarks. This is attributable to both dataset limitations encountered during the curation and a deliberate choice to prioritize the selection of high-quality, representative, and challenging scenarios rather than aiming for sheer data volume. The benchmark also focuses on five symbol classes selected from P&IDs. Although these classes were selected to include visually similar groups, they represent only a small portion of the broader symbology used across engineering disciplines such as electrical, mechanical, and civil engineering. While this scope has been sufficient to reveal meaningful differences among state-of-the-art models, expanding both the number of regions and the diversity of symbols will be essential to improve the statistical robustness of future evaluations.

Finally, our study explores a single querying evaluation strategy. The one-shot visual in-context querying evaluation approach was chosen for its practical advantages, but other prompting strategies could be explored. Alternative approaches, such as prompting with both visual examples and common textual indicators, could yield different insights. The structure of the prompt and the choice of visual examples [38,39] also introduce variables that may significantly influence model performance.

Based on the results of the benchmark and the limitations, there are several promising directions for future work:

- Expanding the Benchmark: An immediate next step is to expand BlueprintSymVL to include a larger number of regions and symbol classes from other engineering domains.
- Enhancing Fine-Grained Visual Discrimination: A challenge revealed by our benchmark is the models' difficulty in distinguishing between visually similar symbols. Future work could investigate novel techniques, such as contrastive learning, aimed specifically at enhancing a VLM's core ability to perceive minor visual differences in objects. An improvement in this general capability would not only boost performance on blueprints but also benefit a wide range of other industrial and scientific domains where fine-grained visual analysis is critical.
- Improving In-Context Visual Learning for Robust Recognition: While leading VLMs demonstrate a capability for in-context visual learning, our benchmark reveals that this skill is often unreliable for object recognition. A critical research direction is to transform this inconsistent property into a robust capability. This could be achieved through targeted training that strengthens in-context visual reasoning, with a specific focus on the failure modes identified by our benchmark. Such training should emphasize recognition in dense and cluttered conditions and include negative test cases to improve reliability and reduce the generation of false positives, thereby making generalization from a single exemplar more dependable in real-world scenarios.
- Investigating Domain-Adaptive Fine-Tuning: As a complementary approach, research could also explore a one-time, domain-adaptive fine-tuning process. This would involve fine-tuning a base VLM on a large dataset of engineering drawings to create a domain-aware foundation model. This approach, which would still leverage the agile visual in-context learning paradigm for project-specific tasks, would imbue the model with a strong understanding of the conventions of engineering drawings, serving as a specialized baseline.
- Comparing Evaluation Strategies: Research is needed to compare different querying strategies (few-shot, one-shot, zero-shot) to determine optimal methods for interacting with VLMs in this domain.
- Human-in-the-Loop Systems: Research could explore how VLMs can best serve as assistants to engineers, perhaps by identifying and counting symbols with a high confidence score while flagging ambiguous or low-confidence detections for human review.

6. Conclusion

The practical deployment of VLMs in high-stakes industrial domains like engineering requires rigorous, domain-specific evaluation. We identified two limitations in the existing VLM evaluation for the domain of blueprint analysis. First, current benchmarks fail to systematically test for the unique perceptual challenges of engineering blueprints, such as high symbol density, occlusion, and visual similarity. Second, their evaluation strategies are misaligned with industrial realities, often depending on either a model's unreliable pre-trained knowledge of non-standardized symbols or the unrealistic assumption of a standardized rulebook.

To address these limitations, we introduced BlueprintSymVL, a benchmark that combines curated, challenging test scenarios with a novel one-shot visual in-context evaluation strategy. By providing a clear visual exemplar at query time, our approach avoids the pitfalls of unreliable pre-trained knowledge. It also eliminates the need for standardized documentation. To our knowledge, this type of evaluation has not been adopted in any existing VLM benchmark. This makes our framework uniquely suited for assessing how well VLMs can operate in practical, industrial settings where standardization is lacking.

We applied this benchmark to four leading VLMs: GPT-4o, Gemini 2.5 Pro, InternVL 2.5 78B, and Qwen 2.5 VL 72B. The results demonstrate that the benchmark is an effective diagnostic tool. Our findings reveal a clear performance hierarchy, with Gemini 2.5 Pro performing best overall. However, all models, including the top performer, exhibited a significant drop in performance when moving from simple baseline tasks to more complex scenarios involving clutter, visual distractors, occlusions, and negative testing. A critical failure mode identified across all models was a large disparity between high recall and low precision, indicating a strong tendency to generate false positives. These results indicate that current state-of-the-art VLMs are not yet reliable enough for autonomous use in engineering blueprint analysis.

Based on our findings, we recommend that practitioners view these models as assistive tools within a human-in-the-loop framework rather than as standalone solutions. Future studies should focus on expanding the benchmark's scope, investigating domain-specific fine-tuning to improve precision, and designing effective human-AI collaborative systems. Thus, our benchmark is an essential tool for guiding the development of VLMs toward the industrial-grade reliability required for real-world engineering applications.

CRediT authorship contribution statement

Vasil Shteriyanov: Writing – review & editing, Writing – original draft, Visualization, Software, Methodology, Investigation, Conceptualization, Data curation, Validation. **Rimma Dzhusupova:** Writing – review & editing, Supervision, Resources, Project administration, Investigation, Conceptualization. **Jan Bosch:** Writing – review & editing, Supervision. **Helena Holmström Olsson:** Writing – review & editing, Supervision.

Declaration of generative AI and AI-assisted technologies in the writing process

During the preparation of this work, the authors used the GPT-4o model in order to improve the readability of the text. After using this tool/service, the authors reviewed and edited the content as needed and take full responsibility for the content of the publication.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Vasil Shteriyanov reports a relationship with McDermott International, Inc. that includes: employment and travel reimbursement. Rimma Dzhusupova reports a relationship with McDermott International, Inc. that includes: employment and travel reimbursement. If there are other authors, they declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This work is supported by McDermott Inc. and Software Center (Gothenburg, Sweden) and conducted in collaboration with Eindhoven University of Technology, The Netherlands.

Appendix A. Prompt

The first image is an example of a P&ID drawing where the target symbols are in red circles. The second image is a P&ID drawing without red circles. Your task is to answer the following question for the second image, using symbols in the red circle in the first image as the visual references for the target symbol: {original_question}

Please state the final count clearly at the end of your response. Additionally, below the final count, provide the label text(s) which appears next to the target symbol(s) and do not use bullets.

Example output format:

Final count: 2
Labels:
ZX-00000
ZY-00500

Data availability

The dataset presented in this study is openly available in Zenodo at https://doi.org/10.5281/zenodo.17250377

Benchmark Dataset for VLM Symbol Recognition in Engineering Blueprints (Original data) (Zenodo)

References

[1] R. Dzhusupova, J. Bosch, H.H. Olsson, Choosing the right path for ai integration in engineering companies: a strategic guide, J. Syst. Softw. 210 (2024) 111945.
[2] R. Rahul, S. Paliwal, M. Sharma, L. Vig, Automatic information extraction from piping and instrumentation diagrams, arXiv preprint, arXiv:1901.11383, 2019.
[3] M. Ahmed, K. Hashmi, A. Pagani, M. Liwicki, D. Stricker, M. Afzal, Survey and Performance Analysis of Object Detection in Challenging Environments, 2021.
[4] K. Saleh, S. Szénási, Z. Vámossy, Occlusion handling in generic object detection: a review, in: 2021 IEEE 19th World Symposium on Applied Machine Intelligence and Informatics (SAMI), IEEE, 2021, pp. 000477–000484.
[5] V. Shteriyanov, R. Dzhusupova, J. Bosch, H.H. Olsson, Unraveling the impact of density and noise on symbol recognition in engineering drawings, in: 2024 IEEE 12th International Conference on Intelligent Systems (IS), IEEE, 2024, pp. 1–7.
[6] U. Vashishtha, V. Bawaniya, V.K. Paul, Design errors in building construction projects, Int. Res. J. Eng. Technol. 7 (7) (2020) 3850.
[7] S. Paliwal, A. Jain, M. Sharma, L. Vig, Digitize-PID: Automatic Digitization of Piping and Instrumentation Diagrams, Springer International Publishing, 2021, pp. 168–180.
[8] L. Ding, S. Liu, Markup in engineering design: a discourse, Future Internet 2 (1) (2010) 74–95.
[9] V. Shteriyanov, Benchmark dataset for vlm symbol recognition in engineering blueprints, https://doi.org/10.5281/zenodo.17250377, Oct. 2025.
[10] S. Antol, A. Agrawal, J. Lu, M. Mitchell, D. Batra, C.L. Zitnick, D. Parikh, Vqa: visual question answering, in: Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 2425–2433.
[11] D.A. Hudson, C.D. Manning, Gqa: a new dataset for real-world visual reasoning and compositional question answering, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 6700–6709.
[12] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, C.L. Zitnick, Microsoft coco: common objects in context, in: Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6–12, 2014, Proceedings, Part v 13, Springer, 2014, pp. 740–755.
[13] Y. Liu, H. Duan, Y. Zhang, B. Li, S. Zhang, W. Zhao, Y. Yuan, J. Wang, C. He, Z. Liu, et al., Mmbench: is your multi-modal model an all-around player?, in: European Conference on Computer Vision, Springer, 2024, pp. 216–233.
[14] B. Li, Y. Ge, Y. Ge, G. Wang, R. Wang, R. Zhang, Y. Shan, Seed-bench: benchmarking multimodal large language models, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2024, pp. 13299–13308.
[15] P. Lu, H. Bansal, T. Xia, J. Liu, C. Li, H. Hajishirzi, H. Cheng, K.-W. Chang, M. Galley, J. Gao, Mathvista: evaluating mathematical reasoning of foundation models in visual contexts, arXiv preprint, arXiv:2310.02255, 2023.
[16] A. Masry, D.X. Long, J.Q. Tan, S. Joty, E. Hoque, Chartqa: a benchmark for question answering about charts with visual and logical reasoning, arXiv preprint, arXiv:2203.10244, 2022.
[17] A. Kembhavi, M. Salvato, E. Kolve, M. Seo, H. Hajishirzi, A. Farhadi, A diagram is worth a dozen images, in: Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, the Netherlands, October 11–14, 2016, Proceedings, Part IV 14, Springer, 2016, pp. 235–251.
[18] J.S. Roberts, T. Lee, C.H. Wong, M. Yasunaga, Y. Mai, P. Liang, Image2struct: benchmarking structure extraction for vision-language models, Adv. Neural Inf. Process. Syst. 37 (2024) 115058–115097.
[19] X. Yue, Y. Ni, K. Zhang, T. Zheng, R. Liu, G. Zhang, S. Stevens, D. Jiang, W. Ren, Y. Sun, et al., Mmmu: a massive multi-discipline multimodal understanding and reasoning benchmark for expert agi, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2024, pp. 9556–9567.
[20] A.C. Doris, D. Grandi, R. Tomich, M.F. Alam, M. Ataei, H. Cheong, F. Ahmed, Designqa: a multimodal benchmark for evaluating large language models' understanding of engineering documentation, J. Comput. Inf. Sci. Eng. 25 (2) (2025) 021009.
[21] E. Elyan, L. Jamieson, A. Ali-Gombe, Deep learning for symbols detection and classification in engineering drawings, Neural Netw. 129 (2020) 91–102.
[22] H. Kim, W. Lee, M. Kim, Y. Moon, T. Lee, M. Cho, D. Mun, Deep-learning-based recognition of symbols and texts at an industrially applicable level from images of high-density piping and instrumentation diagrams, Expert Syst. Appl. 183 (2021) 115337.
[23] V. Shteriyanov, R. Dzhusupova, J. Bosch, H.H. Olsson, From text to meaning: semantic interpretation of non-standardized metadata in piping and instrumentation diagrams, Comput. Chem. Eng. 204 (2026) 109436.
[24] V. Shteriyanov, R. Dzhusupova, J. Bosch, H.H. Olsson, Automating the expansion of instrument typicals in piping and instrumentation diagrams (p&ids), in: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 39, 2025, pp. 28885–28891.
[25] S. Mani, M.A. Haddad, D. Constantini, W. Douhard, Q. Li, L. Poirier, Automatic digitization of engineering diagrams using deep learning and graph search, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, 2020, pp. 176–177.
[26] X. Wang, X. Zhang, Y. Cao, W. Wang, C. Shen, T. Huang, Seggpt: towards segmenting everything in context, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2023, pp. 1130–1140.
[27] A. Shtedritski, C. Rupprecht, A. Vedaldi, What does clip know about a red circle? Visual prompt engineering for vlms, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2023, pp. 11987–11997.
[28] L. Yang, Y. Wang, X. Li, X. Wang, J. Yang, Fine-grained visual prompting, Adv. Neural Inf. Process. Syst. 36 (2023) 24993–25006.
[29] R. Yu, W. Yu, X. Wang, Attention prompting on image for large vision-language models, in: European Conference on Computer Vision, Springer, 2024, pp. 251–268.
[30] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A.C. Berg, W.-Y. Lo, et al., Segment anything, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2023, pp. 4015–4026.
[31] A. Radford, J.W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, et al., Learning transferable visual models from natural language supervision, in: International Conference on Machine Learning, PmLR, 2021, pp. 8748–8763.
[32] S. Easterbrook, J. Singer, M.-A. Storey, D. Damian, Selecting empirical methods for software engineering research, in: Guide to Advanced Empirical Software Engineering, 2008, pp. 285–311.
[33] F. Ozge Unel, B.O. Ozkalayci, C. Cigla, The power of tiling for small object detection, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, 2019, pp. 582–591.
[34] A. Hurst, A. Lerer, A.P. Goucher, A. Perelman, A. Ramesh, A. Clark, A. Ostrow, A. Welihinda, A. Hayes, A. Radford, et al., Gpt-4o system card, arXiv preprint, arXiv:2410.21276, 2024.
[35] G. Team, R. Anil, S. Borgeaud, J.-B. Alayrac, J. Yu, R. Soricut, J. Schalkwyk, A.M. Dai, A. Hauth, K. Millican, et al., Gemini: a family of highly capable multimodal models, arXiv preprint, arXiv:2312.11805, 2023.
[36] Z. Chen, W. Wang, Y. Cao, Y. Liu, Z. Gao, E. Cui, J. Zhu, S. Ye, H. Tian, Z. Liu, et al., Expanding performance boundaries of open-source multimodal models with model, data, and test-time scaling, arXiv preprint, arXiv:2412.05271, 2024.
[37] S. Bai, K. Chen, X. Liu, J. Wang, W. Ge, S. Song, K. Dang, P. Wang, S. Wang, J. Tang, et al., Qwen2. 5-vl technical report, arXiv preprint, arXiv:2502.13923, 2025.
[38] Y. Zhang, K. Zhou, Z. Liu, What makes good examples for visual in-context learning?, Adv. Neural Inf. Process. Syst. 36 (2023) 17773–17794.
[39] Y. Sun, Q. Chen, J. Wang, J. Wang, Z. Li, Exploring effective factors for improving visual in-context learning, IEEE Trans. Image Process. (2025).