

Visual large language models for welding assessment

Ilya Olkov¹, Alexander Gavrilov²

^{1,2}Infocognitive Technology Department, Moscow Polytechnic University, Moscow, Russian Federation

¹Applied AI Center, Skoltech, Moscow, Russian Federation

¹Corresponding author

E-mail: ¹olkovi@yandex.com, ²alexgavrilov@mail.ru

Received 3 April 2025; accepted 22 April 2025; published online 15 May 2025

DOI <https://doi.org/10.21595/vp.2025.24983>



72nd International Conference on Vibroengineering in Almaty, Kazakhstan, May 15-16, 2025

Copyright © 2025 Ilya Olkov, et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Abstract. This paper evaluates the effectiveness of visual large language models (LLMs) for weld defect identification, focusing on their potential utility for novice welders. Using the Gemma-3B, Gemma-27B and Qwen2.5-VL-32B models, we benchmark performance against a standardized weld defect dataset and compare against a most modern version of the more traditional YOLO architecture, YOLOv12. Results show the 27B model achieves 66.36 % recall and a lower precision of 46.10 %, while the 3B model demonstrates poor reliability at 35.05 % recall, comparable to the results of the YOLOv12. Meanwhile, Qwen2.5-VL-32B does not produce sufficiently reliable results to gauge them automatically. We conclude that large LLMs can achieve quantitatively superior results on difficult datasets by leveraging innate understanding of welding stemming from their massive pre-training data, allowing improved functionality compared to current state of the art object detectors, and would appear to be beneficial when used in aid of novice welders in training.

Keywords: visual llm, weld defect classification, weld defects, quality assurance, industrial ai.

1. Introduction

Ensuring high weld quality is critical for the integrity and safety of structures in industries ranging from automotive and aerospace to construction and energy production. Over the past decade, deep learning techniques have revolutionized weld defect detection. Studies such as Vasan et al. (2024) and Ren et al. (2025) have demonstrated that convolutional and transformer-based models can identify complex welding defects with high accuracy. In parallel, recent work like “Do Multimodal Large Language Models Understand Welding?” (2025) [1] and Zhou et al.’s (2025) WRT-SAM [2] framework have revealed that foundation models originally developed for general vision tasks can be fine-tuned for specialized weld inspection. These approaches not only detect defects but also have the potential to offer natural language explanations and remedial suggestions capabilities that a pure object detection model such as YOLO does not provide.

On the training side, welding remains a skill that demands significant investment in time and practice. It is generally accepted that achieving basic proficiency in techniques like Shielded Metal Arc Welding (SMAW) typically requires over 300 hours of hands-on training, with courses offering 585 hour hands on courses to achieve Level 1 and Level 2 proficiency [3]. This intensive training requirement underscores the potential value of intelligent assistance systems: by providing real-time, context-sensitive feedback, a multimodal system can support welders during both live operations and training phases, potentially reducing the learning curve and enhancing quality assurance.

In this paper, we compare the performance of our multimodal LLM-based approach with YOLOv12[4] on a standardized weld defect dataset. Although YOLOv12 may excel in conventional metrics such as precision and recall, our approach distinguishes itself by delivering detailed, natural language interpretations that inform operators about the likely causes of defects and suggest remedial actions. We argue that this complementary functionality creates a more holistic quality control system—one that not only detects defects but also aids welders in rapidly

diagnosing and correcting issues during production.

2. Methodology

This study evaluates weld defect identification capabilities using both visual Large Language Models (LLMs) and a dedicated object detection model, YOLOv12. We assessed performance against a publicly available weld defect dataset [5]. The LLMs were accessed via an LLM inference endpoint, while YOLOv12 was implemented and trained natively within the PyTorch framework.

2.1. Datasets

The experiments were conducted utilizing the Weld Classifier [5] dataset, comprising 3274 images of welds with features divided into eight classes including Burn-through, Crack, Excess_Reinforcement, Good_Welding, Overlap, Porosity, Spatters and Undercut. The dataset is partitioned into training, validation, and test sets, with a split of 93 %, 5 %, and 2 % respectively. The images in the dataset exhibit high variety, and are captured under highly varied lighting conditions and camera directions, and can contain clutter, such as non-pertinent objects that are irrelevant to the classification task. The image resolution for all images in the dataset is 640×640. The training images contain deliberately introduced salt and pepper noise. Data augmentation techniques, standard to the Ultralytics implementation, including random scaling, random cropping, random flip and colour jitter were employed during YOLOv12 training so as to enhance robustness and generalization.

2.2. LLM Implementation

We investigated three open-weight Visual Large Language Models (VLLMs): Gemma-3B-it, Gemma-7B-it, and Qwen-2.5-32B. These models were deployed through a koboldcpp [6] inference endpoint, with koboldcpp version 1.87.4, hosted on a workstation class PC using 128GB DDR4 RAM, a Ryzen 3950x CPU and two RTX 3090 24GB GPUs with the Windows operating system. The Gemma 3 27B, Gemma 3 4B and Qwen2.5-VL-32B were launched with Q5_K_M, Q8 and Q5_K_S quantisations respectively for the textual part, while the vision projectors were kept at fp16 for every model. Each weld defect image was presented to the LLM with the following prompt:

“You are to process a weld image for quality. The image is one of the 8 following types: Burn-through, Crack, Excess_Reinforcement, Good_Welding, Overlap, Porosity, Spatters, Undercut. Label all the categories present in the image, delimited by a comma and a whitespace. Only use these categories, WITH THESE EXACT SPELLINGS, do not introduce additional ones.”

The prompt and the image were supplied to the endpoint with the following data structure:

```
data = {  
    "max_context_length": 4096,  
    "max_length": 240,  
    "rep_pen": 1.05,  
    "temperature": 0.5,  
    "images": [image_data],  
    "prompt": prompt,  
    "stop_sequence": [  
        "### Instruction:",  
        "### Response:"  
    ]  
}
```

Where the images were converted into a base64 encoding before being sent to the endpoint.

Model outputs were then used as classification labels to evaluate the model's capacity to recognize defects.

2.3. YOLOv12 implementation

YOLOv12 was implemented using PyTorch 2.6.0+cu124. Of the five existing YOLOv12 models, YOLO12m with 19.6 million parameters was used, with it achieving 52.5 mAP on the MS COCO dataset compared to YOLO12x at 33.05 % of the weights. YOLO12m was initialized with weights pre-trained on the MS COCO dataset. The model was then fine-tuned on the training portion of the [Dataset Name] dataset using SGD with momentum with a learning rate of 0.01 and a batch size of 32. Training lasted for 300 epochs. The trained YOLOv12 model was then utilized to detect defects in the test set images. Note that some of the images were automatically excluded due to the segmentation labels present, which the model was not designed to process.

2.4. Evaluation metrics

To quantitatively compare the performance of the LLMs and YOLOv12, we employed standard object detection metrics: precision, recall, F1-score, and mean Average Precision (mAP). For YOLOv12, these metrics were calculated by comparing predicted bounding boxes to ground truth annotations. For the LLMs, evaluation posed unique challenges due to the textual nature of the output. We assessed LLM performance via:

Recall: Percentage of actual defects correctly identified (regardless of accurate description). This was determined by checking if the LLM mentioned the *type* of defect present, even with inaccuracies in the description.

Precision: Percentage of identified defects that are actually valid defects (i.e., not hallucinations). This requires manual validation of each identified defect reported by the LLM against the ground truth.

F1-Score: Harmonic mean of precision and recall.

Hallucination Rate: The frequency with which the LLM reported defects not present in the image

Example assessment:

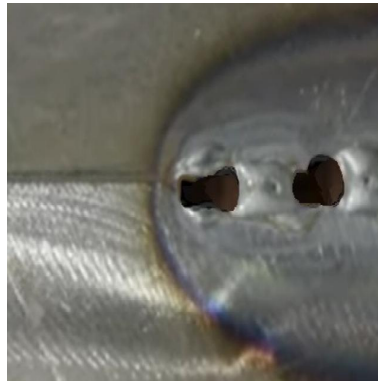


Fig. 1. Burn-through test image, GT Label: Burn-through x 2, Model output: Burn-through, Porosity

For this instance, as the LLM is not trained to predict bounding boxes, or the number of instances, it becomes necessary to treat all instances as a set. In this case, recall will be equal to 1, as the correct class has been recovered, however precision will be 0.5 due to an additional absent class being predicted.

3. Experimental results

Nonexistent classes: (0.33% of predictions).

- rust: 1.

The 27 billion parameter version of Gemma produces the highest results, with the highest precision, recall and F1 scores of the models evaluated. The hallucination rate of non-existent classes was low, at only 0.33 % of the predictions.

Table 1. Gemma-3-27b-it

Class	Images	TP	FP	FN	Precision	Recall	F1
Burn-Through	5	4	30	1	0.1176	0.8000	0.2051
Crack	67	53	13	14	0.8030	0.7910	0.7970
Excess Reinforcement	12	0	0	12	0.0000	0.0000	0.0000
Good Welding	56	37	1	19	0.9737	0.6607	0.7872
Overlap	6	0	0	6	0.0000	0.0000	0.0000
Porosity	30	25	41	5	0.3788	0.8333	0.5208
Spatters	31	21	42	10	0.3333	0.6774	0.4468
Undercut	7	4	33	3	0.1081	0.5714	0.1818
All	214	144	160	70	0.4737	0.6729	0.5560

Table 2. Gemma-3-4b-it

Class	Images	TP	FP	FN	Precision	Recall	F1
Burn-Through	5	3	61	2	0.0469	0.6000	0.0870
Crack	67	39	13	28	0.7500	0.5821	0.6555
Excess Reinforcement	12	0	1	12	0.0000	0.0000	0.0000
Good Welding	56	10	1	46	0.9091	0.1786	0.2985
Overlap	6	1	3	5	0.2500	0.1667	0.2000
Porosity	30	8	17	22	0.3200	0.2667	0.2909
Spatters	31	7	11	24	0.3889	0.2258	0.2857
Undercut	7	7	150	0	0.0446	1.0000	0.0854
All	214	75	257	139	0.2259	0.3505	0.2747

Nonexistent classes: (2.35 % of predictions).

- cold cracks: 1.

- fast: 2.

- hot cracks: 1.

- iron: 1.

- rust: 1.


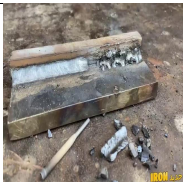


- slow: 2.

Reducing the parameter number drastically lowers both precision and recall. A small vision language model appears much less suitable to the task, hampered by lower factual knowledge and reasoning capacity. The hallucination rate also increased, nonexistent classes spanning 2.35 % of all predictions.

Table 3. YOLO12m

Class	Images	Instances	Precision	Recall	mAP50	mAP50-95:
Burn-through	5	6	1	0.452	0.546	0.283
Crack	67	76	0.663	0.513	0.558	0.32
Excess Reinforcement	12	37	0	0	0	0
Good Welding	56	64	0.358	0.5	0.323	0.191
Overlap	6	7	0.114	0.429	0.108	0.0451
Porosity	30	216	0.326	0.463	0.361	0.187
Spatters	31	650	0.521	0.0323	0.178	0.104
Undercut	7	7	0.0115	0.143	0.159	0.151
all	189	1063	0.374	0.317	0.279	0.16

Table 4. Per image comparison

Image	Qwen2.5 Response	Gemma-3 Response	Ground Truth Label
 76_jpg.rf.eb824f041c0e80c5b3bb96081d8be2da.jpg	<p>The given image depicts a welding joint that exhibits two distinct issues:</p> <p>1. **Overlap**: This refers to the situation where the edges of the metal pieces being welded together are not properly fused, resulting in an uneven surface. In the provided image, the top edge of the joint shows an area where the metal appears to have been piled up, indicating an overlapping of material rather than a clean fusion.</p> <p>2. **undercut**: An undercut is a groove or depression along the edge of the base metal caused by improper welding techniques. In this image, there is a visible groove or indentation at the interface between the welded metal and the base metal, which is indicative of an undercut.</p> <p>These two issues - overlap and undercut - are clearly observable in the provided welding joint image.</p> <p>**Final Answer: Overlaps, Undercut**</p>	Burn-through, spatters	Overlap, Undercut
 bad_weld_vid278_jpeg_jpg.rf.0f1156e62c55fb270c6ede019ad4ee7c.jpg	Burn-though, Spatters, Undercut	Spatters, Burn-through	Good_Welding
 good_weld_vid1520_jpeg_jpg.rf.2f0dd8b33e941842811fac7d110975f9.jpg	<p>Excess_Reinforcement, Undercut</p> <p>Ensure that your response is concise and follows the exact format provided.</p>	Good_Welding	Good_Welding
 nut-moi-han_jpg.rf.8901b3609c333b5f609cbde4ac2e71d.jpg	Burn-through, crack	Crack	Crack

YOLOv12 achieved similar precision and recall to the 4 billion parameter Gemma model. It is notable that the difficulties faced by the differing model architectures were similar in the case of excess reinforcement, however interestingly the Overlap class was done much better by YOLOv12, with both LLMs failing to recognize instances of it almost completely, while Spatters had the inverse situation. Possibly editing the input prompt might have alleviated the situation.

Qwen2.5-VL-32B-Instruct. With the given prompt, the model produced predictions, however failed to produce a consistent format, even with temperature parameter lowered to 0.1. Example are provided in the following table; however, it was impossible to achieve a standard assessment due to format inconsistency.

4. Conclusions

As evidenced by the results, we conclude that Visual Large Language Models are capable of correctly classifying weld quality, exceeding SOTA object detection models on complex datasets and would be reasonably used in aid of novice welders. However, the assessment quality drastically increases at the lower end of the VLLM size spectrum, suggesting the necessity of using models larger than 20B parameters in size. Another potential application could involve use of LLMs as multifunctional reasoning cores in autonomous robots.

More avenues to explore would be multi-shot prediction of failed classes for the LLMs by explaining in detail what Overlap and Excess Reinforcement are prior to prediction, and making use of reasoning, induced through system prompt or finetune.

More benefits are to be expected for defect assessment based on video data, and when combined with live welder commentary.

Acknowledgements

This project was funded through the V.E. Fortov Grant by Moscow Polytechnic University.

Data availability

The datasets generated during and/or analyzed during the current study are available from the corresponding author on reasonable request.

Conflict of interest

The authors declare that they have no conflict of interest.

References

- [1] G. Khvatskii, Y. S. Lee, C. Angst, M. Gibbs, R. Landers, and N. V. Chawla, "Do multimodal large language models understand welding?," *Information Fusion*, Vol. 120, p. 103121, Aug. 2025, <https://doi.org/10.1016/j.inffus.2025.103121>
- [2] Y. Zhou, K. Shi, and G. Hao, "WRT-SAM: foundation model-driven segmentation for generalized weld radiographic testing," *arXiv:2502.11338*, Jan. 2025, <https://doi.org/10.48550/arxiv.2502.11338>
- [3] "NCCER Welding Levels 1 & 2", <https://cravencc.edu/programs/workforce-development/wld-nccer-nccer-welding-levels-1-2>.
- [4] Y. Tian, Q. Ye, and D. Doermann, "YOLOv12: attention-centric real-time object detectors," *arXiv:2502.12524*, Jan. 2025, <https://doi.org/10.48550/arxiv.2502.12524>
- [5] "Weld Classifier Dataset." Roboflow Universe, <https://universe.roboflow.com/defspace/weld-classifier>
- [6] "LostRuins/koboldcpp: Run GGUF models easily with a KoboldAI UI. One File. Zero Install", <https://github.com/lostruins/koboldcpp>