VLM-Guided Prompt Enrichment for Few-Shot Object Detection: CVPR 2026 Roboflow-20VL Challenge Report

Norah Alshammari
TAHAKOM
Noalshammari@tahakom.com

Dalal Alayban
TAHAKOM
Dalayaban@tahakom.com

Reema Alsugair
TAHAKOM
Ralsugair@tahakom.com

Abstract

We present our solution to the CVPR 2026 Roboflow-20VL Foundational Few-Shot Object Detection Challenge. Our approach builds upon ObjEmbed-2B as a training-free baseline and systematically improves performance through three complementary techniques: (1) VLM-enriched text prompt optimization using Qwen3-VL-235B to generate visually-grounded, per-class descriptive prompts; (2) a dynamic prompt strategy that selects per class, from a single enriched prompt, an embedding ensemble, or prompt concatenation based on few-shot validation AP; and (3) inference refinements comprising dynamic proposal count selection and Soft-NMS post-processing. Starting from a baseline AP of 20.97%, our full pipeline achieves 29.79% AP, representing an absolute improvement of 8.82 percentage points without any fine-tuning of the detection model.

1. Introduction

Recent advances in vision-language models (VLMs) have significantly expanded the capabilities of open-vocabulary object detection, enabling models to recognize arbitrary categories at inference time through rich textual descriptions. Despite this progress, adapting such models to specialized or fine-grained domains remains challenging, as object appearances, contexts, and naming conventions in these domains often diverge substantially from pre-training data distributions.

The CVPR 2025 Roboflow-20VL Few-Shot Object Detection Challenge [1] targets exactly this scenario: given only 10 annotated examples per class across 20 diverse domain-specific datasets, participants must maximize mean Average Precision (mAP) under a training-free or few-shot constraint. The benchmark, built on the Roboflow100-VL dataset [1], covers domains ranging from medical imaging to aerial reconnaissance, making semantic alignment between class labels and visual content a central bottleneck.

Our solution adopts ObjEmbed [2] as a strong training-free baseline. ObjEmbed is a multimodal large language model (MLLM)-based object embedding framework that encodes object proposals and text queries into a shared semantic space, scoring detections through both semantic similarity and predicted localization quality. While powerful, its performance is sensitive to the quality of the input text queries, using bare class names often yields suboptimal results due to semantic ambiguity and domain mismatch.

To address this, we propose a multi-stage approach: we use Qwen3-VL-235B [4] to generate rich, context-aware class descriptions by analyzing both dataset documentation and few-shot training images, then a use of a dynamic prompt strategy that selects per class, from a single enriched prompt, an embedding ensemble, or prompt concatenation based on few-shot . We further improve inference through dynamic proposal selection and Soft-NMS. Our pipeline requires no gradient updates to the detection model, making it broadly applicable and computationally efficient.

2. Method

ObjEmbed [2] is an MLLM-based object embedding model fine-tuned from Qwen3-VL-Instruct. It generates proposals using WeDetect-Uni [3], encodes each as an object embedding paired with an IoU embedding, and matches them against text query embeddings. The final score multiplies semantic similarity by predicted IoU, jointly rewarding relevance and localization accuracy. With 100 proposals per image and raw class names as queries, ObjEmbed-2B achieves 20.97% AP on the challenge benchmark, which serves as our baseline.

2.1. VLM-Enriched Prompt Generation

Bare class names often fail in domain-specific datasets where objects are visually ambiguous or underrepresented in pre-training data. To address this, we use Qwen3-VL-235B [4] in a two-stage process for each class:

Stage 1: Context extraction: We supply Qwen3-VL-235B with the dataset description, class-level annotation instructions from the Roboflow100-VL benchmark metadata, and a subset of few-shot training images to build domain-grounded context.

Stage 2: Per-class description: For each class, we send a few-shot training image with the target object's bounding box coordinates highlighted, along with the class name and documentation. The model produces multiple descriptive

prompts emphasizing shape, texture, color, typical context, and relative scale.

2.2. Dynamic Prompt Strategy Selection

Given the set of enriched prompts generated per class, we do not apply a fixed aggregation strategy. Instead, for each class we evaluate three strategies and select the one that maximizes AP on the few-shot validation split: (i) ***Single prompt***, using the best individual enriched description; (ii) ***embedding ensemble***, averaging the text embeddings of multiple prompts or (iii) ***prompt concatenation***, combining two or more descriptions into a single input string before encoding. The optimal strategy is selected independently per class based on validation AP, resulting in a heterogeneous but data-driven assignment across the benchmark.

2.3. Dynamic Proposal Count

In our baseline configuration, we set ObjEmbed to generate 100 proposals per image. While this benefits dense or small-object datasets, we observe that some datasets contain fewer, larger objects where a smaller proposal count is preferable. In such cases, reducing proposals from 100 to 50 suppresses low-quality candidates and improves precision without sacrificing recall. We evaluate both settings for each dataset and select the better-performing configuration using the few-shot validation split.

2.4. Soft-NMS Post-processing

Standard NMS can suppress valid detections when instances partially overlap. Soft-NMS [5] applies Gaussian score decay instead of hard suppression:

$$s\_i \leftarrow s\_i \cdot \exp(-IoU(M, b\_i)^2 / \sigma)$$

We search over $\sigma \in \{0.3, 0.5, 0.7, 1.0\}$ and NMS threshold $\in \{0.2, 0.4, 0.5\}$ on the training split.

3. Experiments

Dataset The RF-20VL datasets are provided by the organizers of the 2026 Foundational Few-Shot Object Detection Challenge.

Implementation Details We evaluate on the official Roboflow-20VL challenge test set. All experiments use ObjEmbed-2B with no detection model fine-tuning. Prompts are generated offline by Qwen3-VL-235B. Proposal counts are chosen from {50, 100} per dataset. Soft-NMS parameters are tuned on the few-shot training split. All experiments are conducted on 2 NVIDIA A100 GPUs.

Results VLM-enriched prompts contribute the largest gain (+7.81% AP), confirming that text query quality is the primary bottleneck for embedding-based open-vocabulary detectors in domain-specific settings. Dynamic proposals add +0.13% AP through per-dataset calibration. Soft-NMS contributes +0.88% AP by reducing false suppression of partially overlapping instances. The full pipeline yields 29.79% AP, an absolute improvement of 8.82 points over the baseline. Table 1 reports cumulative AP gains from each component of our pipeline.

Method	**AP (%)**	**Δ AP**
ObjEmbed-2B (class names only)	20.97	-
+ Enriched Prompts	28.78	+7.81
+ Dynamic Proposals	28.91	+0.13
+ Soft-NMS (Final)	**29.79**	+0.88

Table 1. Ablation on Roboflow-20VL test set.

4. Conclusion

We presented a training-free inference pipeline that significantly improves ObjEmbed-2B on the foundational few-shot object detection challenge. VLM-guided prompt enrichment using Qwen3-VL-235B, combined with dynamic per-class prompt strategy selection, dynamic proposal selection, and Soft-NMS, yields 29.79% AP, an 8.82-point gain over the baseline, with no model fine-tuning. Our results highlight text query quality as a key lever for open-vocabulary detection in domain-specific few-shot scenarios.

References

[1] P. Robicheaux et al. Roboflow100-VL: A multi-domain object detection benchmark for vision-language models. arXiv:2505.20612, 2025.

[2] Fu, Shenghao, et al. "ObjEmbed: Towards Universal Multimodal Object Embeddings." *arXiv preprint arXiv:2602.01753* (2026).

[3] Y. Fu et al. WeDetect: A Universal Object Proposal Generator. arXiv preprint, 2025.

[4] S. Bai et al. Qwen3-VL Technical Report. arXiv:2511.21631, 2025.

[5] N. Bodla et al. Soft-NMS — Improving Object Detection with One Line of Code. ICCV, 2017.