Teaching Vision-Language Models to validate Technical Drawings and CAD Models

Alexander Valverde[1], Danny Xie-Li[2], Zhichao Wang[1], and Marcos V. Conde[3]

[1] Novineer, USA
[2] Instituto Tecnológico de Costa Rica, San José
[3] University of Würzburg, Germany

Abstract. In regulated manufacturing the 2D engineering drawing is the controlled document, and confirming that it still matches its 3D CAD model is a manual check repeated at every change order, quote, and legacy digitization. Hosted multimodal APIs are often ruled out by cost and by the confidentiality of proprietary or patient-linked geometry. We ask whether a compact open-weight vision-language model can perform this check locally. Framing drawing-to-model verification as a binary task, we evaluate Qwen3-VL-2B, Qwen3-VL-8B, and Gemma-3-4B zero-shot on drawing–render pairs from Omni-CAD, where the two smaller models answer Yes to nearly every pair. Our fine-tuning with LoRA on a single GPU raises them to 92% accuracy and 0.976 AUC; adapting the vision encoder, not only the language model, is what makes them balanced and calibrated. Applied unchanged to a new benchmark of 199 upper-limb prosthesis meshes, the adapters reach 75% accuracy with errors confined to a few design families. Code and adapters are available at `https://github.com/alexvalgui/validating_cad_draws`.

Keywords: Vision-language models (VLM) · Engineering drawings · CAD verification · Parameter-efficient fine-tuning · Medical prosthetics

1 Introduction

Medical devices are the backbone of Costa Rica's export economy: sales grew 22% annually from 2020 to 2024, reaching US$8.7 billion, or about 48% of goods exports. Costa Rica ranks tenth among exporters worldwide and first per capita, hosting over a hundred multinationals and a local supplier base in tooling, metalworking, and precision assembly [11, 27]. Every component in this chain is described twice, as a 3D CAD model and as a 2D engineering drawing, and in a regulated industry the latter is the controlled document: ISO 13485 requires documented control of design outputs [16] and FDA regulations require blueprints to be kept in the design history file [37]. Confirming that both still describe the same part is an obligation at every change order, supplier quote, and legacy digitization effort [26], and it is still done by hand. Automating

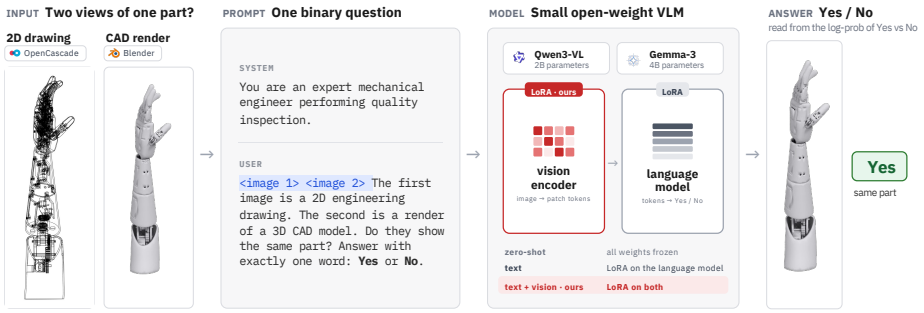


Fig. 1: Overview. A 3D model is rendered as a hidden-line drawing and a shaded image, and a small open-weight VLM is asked whether they show the same part. The answer is read from the log-probabilities of *Yes* and *No*. Zero-shot models answer Yes to almost every pair. LoRA on the language model, or on both the language model and the vision encoder (ours), turns them into calibrated verifiers.

it with a hosted multimodal API is rarely an option: the geometry is proprietary and, in prosthetics, may be linked to individual patients, so files cannot leave the company. Vision-language models (VLMs) are the natural candidate, but engineering drawings are almost absent from their training data. The CAD literature has concentrated on 3D shape [19, 30, 40, 42, 44], the few works touching 2D views use them as input for generation rather than checking [28], and perception benchmarks show that VLMs struggle with exactly what this task demands: multi-image reasoning [25, 38], fine visual distinctions [10, 36], and a prior toward answering Yes [22]. Since prosthetic designs are closed and no public test set exists, the practical route is transfer: tune a compact open-weight model with parameter-efficient methods [7, 15, 24] on generic mechanical parts and ask whether what it learns carries over to an unseen domain.

This paper tests that route. We frame drawing-to-model verification as a binary task, evaluate Qwen3-VL-2B, Qwen3-VL-8B, and Gemma-3-4B without adaptation on drawing–render pairs from Omni-CAD, fine-tune the two smaller backbones with LoRA on a single GPU, and apply the resulting adapters unchanged to an out-of-distribution medical benchmark. Our contributions are:

- **A medical prosthetics benchmark.** 199 upper-limb prosthesis meshes from GrabCAD and NIH3D, in 88 design families and eight categories, with aligned drawings and renders; the first public drawing-to-model verification set in the medical domain.
- **A parameter-efficient fine-tuning recipe.** One epoch of LoRA with hard negatives turns Qwen3-VL-2B and Gemma-3-4B, which answer Yes to nearly every pair zero-shot, into verifiers at 92% accuracy and 0.976 AUC; adapting the vision encoder, not only the language head, is what makes them balanced and calibrated.

– **Out-of-distribution transfer.** Adapters trained on generic mechanical parts reach 75% accuracy on the prosthetics benchmark with no further training, with errors confined to a few design families.
– **A fully local pipeline.** One workstation GPU and adapters of a few tens of megabytes, so proprietary or patient-linked geometry never leaves the company.

2 Related Work

2.1 Learning based CAD: generation, representations, and benchmarks

Many computer vision and graphic projects have appeared recently that focused on CAD generation and understanding. DeepCAD [40] represents a solid as a sequence of parametric operations and trains a Transformer to produce new ones, and SkexGen [43] extends the idea with disentangled codebooks over sketch and extrude operations. Later systems condition the output on other modalities. CAD-MLLM [42] accepts text, images, and point clouds. Text2CAD [19] produces construction sequences from natural language prompts.

Text-to-CadQuery [41] and CAD-Recode [30] emit executable CadQuery code. Img2CAD [44] recovers parametric geometry from single photographs, and Drawing2CAD [28], the closest to the setting considered here, translates 2D vector drawings into modeling operations. A parallel line learns discriminative encodings of solids directly from boundary representations, as in UV-Net [17] and BRepNet [21], supporting classification, segmentation, and retrieval. All of this rests on large repositories: ABC [20] with one million B-rep solids, SketchGraphs [31] with fifteen million constrained sketches, the Fusion 360 Gallery of human designs [39], and the synthetic Zero-to-CAD collection of roughly one million executable programs [2]. Where multimodal models meet engineering artifacts, evaluation has so far meant question answering over documentation, as in DesignQA [8]. None of this work checks whether a drawing and a CAD model describe the same part. Yet that is the check behind every change order, every quote against a received drawing, and every legacy digitization effort.

2.2 VLM - CAD Verification

Verification places two images before the model, a drawing and a render, and VLMs are weak in exactly this regime. NLVR2 [34] introduced true or false judgments over photograph pairs, and rendered views have long connected 3D shapes to 2D recognition [33]. Open models trained on single images degrade once a question spans several [38], picking the right image among lettered candidates defeats even proprietary systems [25], and visual correspondence across views stays hard because language offers little help [10]. Models also answer from priors instead of pixels, a problem as old as balanced VQA [12] and still visible as a Yes preference on binary probes [22], as failures on simple visual distinctions [29,36],

and as a matching skew toward particular option letters in multiple choice [45]. A skewed answer distribution does not prove the evidence is missing, though. String probabilities are distorted by surface form competition [14], while logits on true or false items are often well calibrated when read directly [18]. Whether a model fails to see or fails to decode is therefore an open question for any binary visual task, and one that calibration analysis of the answer logits can settle.

2.3 Efficient adaptation of open VLMs

Instruction tuned models such as LLaVA [23] showed that a vision encoder attached to an open language backbone can follow visual instructions after modest training. Adapting such models cheaply has since become standard practice. LoRA [15] trains low rank updates on top of frozen weights. QLoRA [7] adds quantization so that adaptation fits on a single GPU. DoRA [24] splits each pretrained weight into magnitude and direction and applies the low rank update to the direction alone, narrowing the gap to full fine tuning at no extra inference cost, with gains reported on vision language backbones. Open checkpoints have grown alongside these methods, and the Qwen VL line now spans dense models from 2B to 32B parameters [3, 4]. Together, these methods make task specific adaptation feasible on a single GPU. Whether that is enough for fine grained geometric verification has not been examined.

3 Background

Mechanical parts are designed as 3D CAD models, usually stored as boundary representations (B-reps) that define a solid through a network of faces, edges and vertices with their underlying surfaces and curves [13, 32]. The models are parametric, built from a history of operations on constrained sketches that engineers edit to regenerate the geometry [5], with freeform surfaces handled by splines [9]. The 3D model drives simulation, machining and every downstream document, but it is not an image; we therefore render it from standard viewpoints so that an off-the-shelf VLM can process the geometry visually.

The document that regulators, customers and suppliers treat as the source of truth is the 2D engineering drawing derived from that model. It breaks the part into orthographic views, with solid lines for visible and dashed lines for hidden edges, annotated with dimensions and tolerances under strict drafting standards [1]. The same geometry is thus represented twice, and keeping the two aligned is an obligation rather than good practice: ISO 13485 mandates documented control of design outputs [16] and FDA quality system regulations require the drawings in the design history file [37]. Almost every engineering change circles back to the question of whether drawing and model still describe the same part, a question that concerns suppliers quoting jobs, engineers drafting inspection plans and anyone digitizing legacy paper documentation, and that is still answered by a trained expert comparing both documents by hand despite steady advances in automated drawing analysis [26]. This work asks whether a small open-weight VLM can take over that comparison reliably and at low cost.

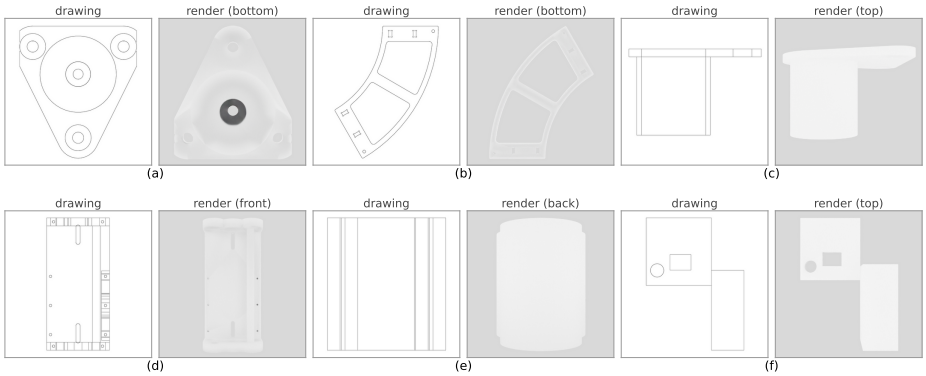


Fig. 2: Drawing–render pairs from the Omni-CAD subset, one axis-aligned view per part. Farthest-point sampling over geometric features favours rare shapes such as the curved frame (b) and the thin-walled enclosure (d) alongside simple prismatic parts (c, e, f).

4 Methodology

This section describes the datasets, 2D image generation procedure, and vision-language models used in the experiments. Paired engineering drawings and shaded renders are generated from six canonical viewpoints for each 3D model.

4.1 Datasets

We use two datasets in this work. The first is a subset of Omni-CAD, a large collection of generic mechanical parts, which we use for training and in-distribution evaluation. The second is a small set of prosthetic device models that we collected ourselves, used only to test how well the model transfers to a domain it has not seen during training. Both are described below, followed by the procedure we use to turn each 3D model into the 2D images given to the model. To support further work, we release the rendered images and hidden-line drawings of our Omni-CAD subset, the medical prosthetics benchmark, the code, and the trained adapter weights under a fully permissive license for unrestricted use; the underlying Omni-CAD data is released by its authors under the MIT license.

Omni-CAD Many 3D datasets contain degenerate models that are not manifold or watertight, which causes problems for later processing. We use Omni-CAD [42], a multimodal CAD collection of about 450K models with descriptions, point clouds, and command sequences, of which we only take the B-rep files.

From this corpus we select 10,000 models. Each STEP file is read with OpenCascade and described by a small set of geometric features: number of faces and edges, the fraction of faces of each surface type (plane, cylinder, cone, sphere, torus, B-spline, other), the fraction of curved edges, sphericity, and two

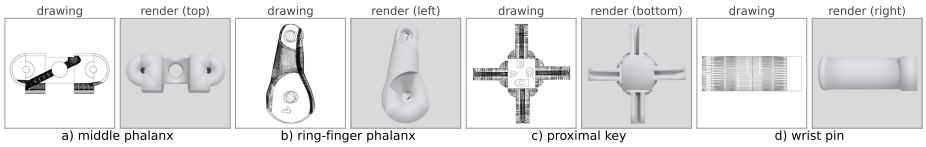


Fig. 3: Drawing–render pairs from the medical prosthetics benchmark. The parts are mesh-based, so their drawings come from polygonal HLR and carry triangulation artefacts absent from the B-rep drawings of Figure 2.

bounding-box aspect ratios. Files that cannot be read or have degenerate geometry are discarded. To drop trivial models, we fit a Gaussian kernel density estimate (bandwidth factor 0.15) to $\log_{10}$ of the face count, search for density minima below the median, and cut everything below the first one, with a hard floor of six faces. We then run farthest-point sampling on the standardized features, started deterministically from the point farthest from the feature mean, to pick the final subset. Because this method prefers points in sparse regions of the feature space, rare shapes such as freeform and thin-walled parts appear more often than in the original corpus. These features serve only to choose the subset; the hard negatives of Section 5.1 are mined with a separate image embedding.

Split. Models for which all six drawing–render views could be produced are shuffled with a fixed seed and split *by part*, never by view: 5% of the parts (470) form a held-out set of $470 \times 6 \times 2 = 5{,}640$ balanced pairs on which every number in Section 5 is reported, and the remaining 8,930 are used for training. No part contributes views to both sides.

Medical Prosthetics Benchmark Public 3D datasets of prosthetic devices are very scarce. The geometry of commercial prostheses is proprietary and is not released by the companies that develop them, and there is no existing benchmark for this domain. We built our own by searching open repositories and checking each model by hand for relevance, geometric integrity, and duplicates. The final set has 199 meshes of upper-limb prostheses from GrabCAD and the NIH 3D Print Exchange (NIH3D) [6], and is used only for evaluation. The NIH3D models come as full assemblies with each component available separately, so we can test on single parts as well as complete devices. We group the parts into eight functional categories by name: fingers and phalanges, pins, tensioners, palms, gauntlets and arms, locking mechanisms, forearm inserts, enclosures and covers, and straps, as listed in Table 4. The remaining meshes are complete assemblies.

Image Generation For every model in both datasets we produce two kinds of 2D images from the same six axis-aligned views (front, back, left, right, top, bottom). Both share the same view directions and image axes, so a drawing and a render of a given view are aligned.

Engineering drawings. We compute orthographic projections with hidden line removal using OpenCascade's exact HLR algorithm on the B-rep, and its polygonal version for meshes, where feature edges are detected with a 30° dihedral-angle threshold. Visible sharp edges and silhouettes go to one layer and hidden edges to a second, dashed layer; smooth tangent edges are omitted, as is standard in technical drawings. Each view is saved as a DXF and rasterized to a 1024×1024 black-on-white PNG, fit to the drawing extents with an 8% margin. We also add the overall width and height and one diameter callout per distinct hole radius, using a fixed dimension style, to produce a dimensioned variant of each drawing for ablation.

Shaded renders. We render each mesh in Blender at 512×512 pixels with a perspective camera at a 45° field of view, placed so the part fills 90% of the frame. Each model gets a uniform matte grey material and is lit by a fixed three-light key/fill/rim setup defined in the camera frame and rotated with each view, so a given surface orientation is shaded identically across views. A weak ambient light keeps any region from going fully black.

4.2 Vision-Language Models

We evaluate three open-weight models—Qwen3-VL-2B, Qwen3-VL-8B [3], and Gemma-3-4B [35]—selected because all three can be fine-tuned on cost-effective GPUs such as an NVIDIA L40S (approximately $2/hour). Our target users are small and medium engineering and manufacturing companies in Costa Rica, for which a monthly commercial multimodal API is often not an option and sending proprietary drawings to an external service raises confidentiality concerns; a model that runs locally, is fine-tuned once on the company's own data and then costs nothing per query fits this setting far better than a hosted frontier model.

Qwen3-VL. The two Qwen3-VL models share one architecture at different scales. Images are encoded by a vision transformer at native resolution, so the number of visual tokens grows with image size rather than being fixed by a square resize, which matters for line drawings, where thin edges and small dimension text are lost under heavy downsampling. The 2B model runs on a laptop GPU or a CPU with quantization; the 8B model is the largest of the family that fits comfortably in 24 GB of GPU memory. Both are Apache 2.0, allowing unrestricted commercial use.

Gemma-3. Gemma-3-4B is Google's smallest multimodal Gemma 3 model. A frozen 400M-parameter SigLIP encoder resizes every image to 896×896 pixels and compresses it into a fixed 256 visual tokens, with an optional pan-and-scan crop for non-square inputs. Image cost is therefore constant regardless of resolution, which makes inference cheap and predictable but risks losing fine detail; including it lets us compare a fixed-token encoder against a dynamic-resolution one at similar parameter count. Gemma 3 is released under Google's Gemma Terms of Use, which permit commercial use with some restrictions.

System: You are an expert mechanical engineer performing quality inspection.

User: `<image 1> <image 2>` The first image is a 2D engineering drawing. The second image is a render of a 3D CAD model. Do the drawing and the model show the same part? Answer with exactly one word: Yes or No.

Fig. 4: Prompt used for zero-shot evaluation, LoRA training and fine-tuned evaluation; only the two images change. The answer is not sampled but read from the next-token log-probabilities of *Yes* and *No* (Section 5.1).

5 Experiments

We ask whether a small, locally deployable VLM can tell if a 2D engineering drawing and a render of a 3D CAD model show the same part, the check behind verifying a client's model against its drawing, catching a revised model that no longer matches an old drawing, or retrieving the right model for a drawing from an archive. Current VLMs see far more photographs than line drawings and rarely compare two representations of one object.

We answer three questions: how well off-the-shelf VLMs do with no adaptation (Qwen3-VL-2B, Qwen3-VL-8B and Gemma-3-4B zero-shot on the 5,640 held-out pairs of Section 4.1); how much light fine-tuning helps (LoRA adapters trained on Omni-CAD pairs with hard negatives, so the model must look at geometry and not only overall shape); and whether the result transfers out of distribution (the medical prosthetics benchmark, whose parts differ considerably from industrial models but share features such as sharp edges, pins and joints). As an ablation we test whether adding dimensions to the drawings changes the result, since real drawings always carry them. All experiments use the same six-view images, the prompt of Figure 4 and Yes/No scoring, so results are comparable across models and settings; the only exception is the prompt ablation of Section 5.4.

5.1 Fine-tuning

Each input is a drawing, a render and a fixed question asking whether they show the same part, answered *Yes* or *No*. At inference we do not sample text but read the next-token log-probabilities of the two answers, aggregated over tokenizations, giving a score that can be thresholded or calibrated. Prompt and question are identical for zero-shot evaluation, training and fine-tuned evaluation.

Each positive pairs a drawing with the same-view render of its model and receives one negative: with probability 0.7 a hard negative, sampled uniformly from the three most similar *other* parts among the top ten OpenCLIP ViT-B/32 (LAION-2B) cosine neighbours of the positive render within the same split, otherwise a random other part (Figure 5). This yields 107,160 training pairs (53,580 positives, 70.4% of negatives hard), about 6,700 steps at batch size 16. Neighbours are searched over all six views, so about a third of hard negatives share the drawing's view (33% in training, 30% held-out). Neighbours

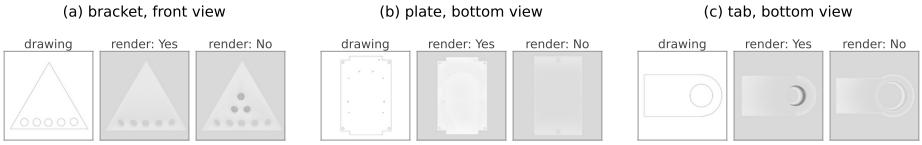


Fig. 5: Hard negatives from the held-out set: drawing, matching render and CLIP nearest-neighbour render in the same view. The pairs differ only locally: hole pattern (a), tabs and hole layout (b), boss and bore size (c).

are mined before splitting and the held-out set holds only 5% of parts, so most held-out neighbours fall in the training split and are discarded: the realised hard fraction on the held-out set is 23% (658 of 2,820 negatives), those negatives are on average less similar than in training, and the *Hard* column of Table 1 rests on these 658 pairs. Drawings carry no dimension annotations and we never train on dimensioned ones; Section 5.3 evaluates the same adapters on them without retraining. Images keep their native aspect ratio, with the visual token budget capped for predictable memory.

We fine-tune with LoRA [15] in two scopes: *text*, which adapts only the language model, and *text + vision*, which also adapts the vision encoder. The latter is our main model. Table 6 gives the modules adapted in each scope and the remaining hyperparameters, identical across scopes and backbones. The training loss is applied only to the answer tokens; prompt and image tokens are masked out. Every 250 steps the loss is measured on a fixed 500-pair subset of the held-out set and the checkpoint with the lowest such loss is kept; the accuracy-optimal thresholds τ^* and Acc^* of Tables 1 and 7 are likewise chosen on the full held-out set and are in-sample quantities. Only the adapter weights are saved, a few tens of megabytes, so one base model can serve several adapters. All runs use a single NVIDIA L40S (48 GB). Qwen3-VL-8B is evaluated zero-shot only, since training it would roughly double the cost of the 2B run and the 2B adapters already outperform it.

5.2 Main results

Table 1 gives the main result on undimensioned drawings, the harder and more portable case. Zero-shot, the two smaller models are unusable: both answer Yes to essentially every pair, so forced-choice accuracy sits at chance and ECE at 0.42 to 0.50. Qwen3-VL-8B is the only model that says No without training and reaches 74.2%, but its negative accuracy splits sharply, 92% on random negatives against 54% on hard ones, so most of what it gets right is telling apart clearly different parts.

Fine-tuning moves both small models past the zero-shot 8B model on accuracy, AUC and best-threshold accuracy. With text-only adapters, Gemma-3-4B reaches 84.9% accuracy and 0.942 AUC. Qwen3-VL-2B has the higher AUC (0.943) and best-threshold accuracy (89.6%) but a lower forced-choice accuracy of 76.6%, because its adapter is conservative: it answers Yes on only 32% of

Table 1: Verification on the Omni-CAD held-out set (470 parts, $n = 5{,}640$ balanced pairs; 658 hard and 2,162 random negatives). *Acc* uses $p(\text{Yes}) \geq 0.5$, Acc^* the best threshold chosen on this set; *Yes%* is the fraction answered Yes.

						Neg. acc. ↑	
Model	Acc ↑	AUC ↑	Acc* ↑	ECE ↓	Yes%	Hard	Random
Zero-shot							
Gemma-3-4B	50.0	0.506	51.9	0.500	99.8	0.2	0.1
Qwen3-VL-2B	50.0	0.665	62.8	0.424	100.0	0.0	0.0
Qwen3-VL-8B	74.2	0.822	75.2	0.181	41.1	53.5	92.2
Fine-tuned (text)							
Gemma-3-4B	84.9	0.942	89.0	0.115	46.7	55.2	98.2
Qwen3-VL-2B	76.6	0.943	89.6	0.198	32.3	**77.4**	**99.4**
Fine-tuned (text + vision)							
Gemma-3-4B	90.3	0.973	92.8	0.090	47.2	73.3	99.1
Qwen3-VL-2B	**92.0**	**0.976**	**93.1**	**0.070**	51.0	64.6	99.1

pairs, which gives it the best hard-negative accuracy of any model (77.4%) at the cost of positives. Table 7 shows the same thing from the probability side; its mean probability on positives is 0.534 and the accuracy-optimal threshold is 0.085, so a user who takes $p(\text{Yes}) \geq 0.5$ at face value loses thirteen points that a threshold tuned on a few dozen labelled pairs would recover. Gemma's adapter lands closer to a balanced operating point without this step. For both models random negatives are solved at 98 to 99%; what remains is the hard-negative case, the one that matters in practice.

Adapting the vision encoder as well as the language model resolves most of this. Qwen3-VL-2B reaches 92.0% accuracy and 0.976 AUC, and the operating point moves to a Yes rate of 51.0% with ECE 0.070, so the forced-choice and best-threshold accuracies now differ by one point rather than thirteen. Hard-negative accuracy falls from 77.4% to 64.6%, but this reflects the loss of the text-only adapter's global bias toward No rather than a genuine regression: the text-only model buys negative accuracy by rejecting positives as well, which is what its 32% Yes rate and 0.198 ECE record. Figures 7 and 8 (Appendix A) show held-out hard negatives that both the zero-shot model and the text-only adapter accept with high confidence and that only the text + vision adapter rejects.

5.3 Analysis

Dimension annotations at test time. The adapters of Table 1 never see dimension text during training. We re-evaluate them unchanged on the dimensioned drawings of the same held-out pairs, with the same prompt and only the drawing image differing (Table 2); this asks whether the annotations are usable without retraining, not whether they help in principle. The text adapters move in opposite directions, Gemma-3-4B toward Yes (Yes rate 53.2%, accuracy +4.1) and Qwen3-VL-2B toward No (Yes rate 23.6%, accuracy −6.2), while the

Table 2: Undimensioned adapters evaluated on the dimensioned version of the same held-out set ($n = 5{,}640$). Δ is relative to Table 1.

Model	Acc ↑	ΔAcc	AUC ↑	ΔAUC	ECE ↓	Yes%
Fine-tuned (text)						
Gemma-3-4B	89.0	+4.1	0.951	+0.009	0.077	53.2
Qwen3-VL-2B	70.4	−6.2	0.924	−0.019	0.261	23.6
Fine-tuned (text + vision)						
Gemma-3-4B	93.2	+2.9	0.981	+0.008	0.063	50.8
Qwen3-VL-2B	93.4	+1.4	0.978	+0.002	0.056	52.9

text + vision adapters both improve and stay near a balanced Yes rate. AUC moves by at most 0.02 in every case, so the operating point changes rather than the ranking. Figure 9 (Appendix A) shows dimensioned hard negatives that only the text + vision adapters reject for either backbone. Adapting the vision encoder thus lets both backbones absorb unseen annotations, whereas language-only adapters react in a model-specific way; with one adapter per scope and backbone this is suggestive rather than conclusive, and it argues for training in the dimensioned setting rather than only testing in it.

Zero-shot calibration. Table 7 (appendix) explains why the small models fail without training. Gemma-3-4B assigns $p(\text{Yes}) = 0.997$ to positives and 0.999 to negatives alike, and Qwen3-VL-2B 0.94 and 0.91; the class means differ by 0.03 or less, so no threshold separates them. Qwen3-VL-2B still has usable ranking signal (AUC 0.665, 62.8% at the optimal threshold), Gemma-3-4B does not. Fine-tuning opens the gap between class means to between 0.46 and 0.70 depending on model and adapter scope, with the text + vision adapter giving the widest separation for both backbones; for Qwen3-VL-2B it also moves the optimal threshold closest to 0.5. The zero-shot 8B model sits in between: its means are separated (0.641 against 0.184) but its optimal threshold is 0.223, so it too is miscalibrated in the direction of saying No.

Hard versus random negatives. In every row of Table 1, random negatives are near ceiling after fine-tuning and hard negatives carry the error: the gap is 39 points zero-shot for the 8B model and stays at 43 (Gemma text), 35 (Qwen text + vision) and 22 (Qwen text, only because it rejects positives as readily as negatives). Hard negatives are CLIP nearest neighbours of the positive render, sharing silhouette and proportions and differing in local detail such as holes or fillets (Figure 5; further examples with model predictions in Figures 7–9 of Appendix A). As the held-out hard negatives are the weaker ones that survived the split filter (Section 5.1), the *Hard* column is an upper bound for the case that matters in practice, a wrong model that is a close relative of the right one, which the medical benchmark probes in Section 5.4.

Table 3: Transfer to the medical benchmark (199 parts, 88 design families, $n = 2{,}284$ balanced pairs). Fine-tuned rows use the adapters of Table 1 with no further training. Columns as in Table 1.

Model	Acc ↑	AUC ↑	Acc* ↑	ECE ↓	Yes%
Zero-shot					
Gemma-3-4B	50.0	0.504	51.2	0.500	99.7
Qwen3-VL-2B	50.0	0.628	59.6	0.439	100.0
Qwen3-VL-8B	63.3	0.691	63.4	0.290	67.2
Fine-tuned (text)					
Gemma-3-4B	68.6	0.763	70.2	0.127	53.9
Qwen3-VL-2B	72.8	0.793	73.1	**0.099**	58.8
Fine-tuned (text + vision)					
Gemma-3-4B	72.2	0.806	73.7	0.123	57.1
Qwen3-VL-2B	**75.0**	**0.815**	**75.5**	0.123	63.4

Table 4: Accuracy % of the main (text + vision) adapters on the medical benchmark by functional category, same-family negatives excluded.

Category	Parts	Qwen3-VL-2B	Gemma-3-4B
Straps & small hardware	13	90	81
Enclosures & covers	15	89	85
Forearm inserts	17	88	96
Tensioners	30	88	45
Palms, gauntlets & arms	30	86	80
Fingers & phalanges	39	85	74
Pins	31	80	70
Locking mechanisms	21	29	78

5.4 Transfer to Medical Prosthetics

We evaluate the zero-shot and fine-tuned models on the medical prosthetics benchmark of Section 4.1. Table 3 reports the Omni-CAD adapters on the prosthetics benchmark with no further training. All numbers drop, as expected for organic, thin, mesh-based parts rather than prismatic B-rep solids, but the AUC ordering is unchanged and the text + vision Qwen3-VL-2B (75.0%, AUC 0.815) still beats zero-shot Qwen3-VL-8B by almost twelve points. Vision adds 2 to 4 points over the text adapters, less than on Omni-CAD. Calibration transfers better than accuracy for the text adapters (ECE 0.099 and 0.127): the models are uncertain where they are wrong, which is what lets a deployment flag doubtful pairs for a human. The text + vision adapters keep less of this (ECE 0.123) and answer Yes more often (57 to 63%).

Table 4 breaks the accuracy of the main text + vision adapters down by functional category, with same-family pairs excluded. Qwen3-VL-2B is consistent across categories at 80 to 90%, with one exception, locking mechanisms at 29%. Gemma-3-4B has its own single failure at tensioners (45%) and is otherwise

System: You are an expert mechanical engineer performing quality inspection of 3D-printed upper-limb prosthetic components.

Fig. 6: Prosthetics-specific system message of the prompt ablation; the user turn is that of Figure 4, and only the underlined clause is added.

Table 5: Medical benchmark with a prosthetics-specific system message (n = 2,284). Δ is relative to the generic prompt of Table 3; Hard and Random are negative accuracies as in Table 1.

							Neg. acc. ↑	
Model	Acc ↑	ΔAcc	AUC ↑	ΔAUC	ECE ↓	Yes%	Hard	Random
Zero-shot								
Gemma-3-4B	50.0	+0.0	0.506	+0.002	0.500	100.0	0.1	0.0
Qwen3-VL-2B	50.0	+0.0	0.610	−0.018	0.438	100.0	0.0	0.0
Fine-tuned (text)								
Gemma-3-4B	68.7	+0.1	0.763	+0.000	0.133	52.9	52.0	98.0
Qwen3-VL-2B	72.6	−0.2	0.792	−0.001	**0.098**	60.0	47.0	98.6
Fine-tuned (text + vision)								
Gemma-3-4B	72.3	+0.1	0.806	+0.000	0.118	59.5	47.5	98.3
Qwen3-VL-2B	**74.9**	−0.1	**0.813**	−0.002	0.118	64.1	44.7	98.0

70 to 96%, and does better than Qwen on forearm inserts. Each model scores well on the other's failure: Gemma-3-4B reaches 78% on locking mechanisms and Qwen3-VL-2B 88% on tensioners. Both categories are parametric sweeps of 20 to 30 size-scaled copies of one design, so a single wrong judgement is repeated across the family and moves a category by 40 points or more; the same effect inflates Gemma-3-4B's 96% on forearm inserts. Per-part numbers therefore overstate the number of independent mistakes, and since the two models fail on disjoint families, an ensemble of the two adapters would likely remove both failures.

Prompt sensitivity. The adapters were trained and evaluated with the generic prompt of Figure 4. To test whether a domain hint changes the transfer result, we re-run the medical benchmark with only the system message replaced by the prosthetics-specific one of Figure 6, keeping question and images unchanged (Table 5). Nothing moves: accuracy changes by at most 0.2 points, AUC by 0.02 and ECE by 0.01 for every model and scope, and the zero-shot models still answer Yes to every pair. The verdict is set by the two images, not by the wording of the instruction, which is what a deployment wants from a verifier. The same runs also give the negative split on this benchmark: after fine-tuning, random negatives are solved at 98% for all four adapters while hard negatives sit at 45 to 52%, so the medical errors are, as on Omni-CAD, confined to close relatives of the right part.

6 Discussion

Small VLMs can learn a visual domain absent from their pretraining: one epoch of LoRA on one GPU turned 2B and 4B models into reliable verifiers of engineering drawings, which suggests the harder goal of describing or generating a drawing from a render is within reach with the same pairs as supervision. Adapting the vision encoder matters more than the accuracy gain suggests: text-only adapters learn a decision rule over features never shaped by drawings and react to dimension text in a model-specific way, whereas adapted encoders make both models balanced and let them absorb unseen annotations without retraining, and since the remaining errors involve local features such as holes and fillets, they are a natural starting point for probing which regions drive a decision or for training on single-feature negatives.

Adapters trained on generic mechanical parts transferred to organic, mesh-based prosthetic parts, with errors concentrated in whole design families, so a little in-domain data should close most of the gap. The whole recipe runs locally, so a company can fine-tune on its own library without proprietary or patient-linked geometry leaving the site, which enables checking produced or catalogued parts against their drawings, retrieving parts across parametric families, and verifying patient-specific components against a reference model. Future work should add a separate test split, denser hard negatives, more seeds and epochs, training on dimensioned drawings, and multi-view inputs to reduce the hard-negative errors that remain.

7 Conclusion

Compact vision-language models can be turned into reliable drawing-to-model verifiers with a cheap recipe: a single GPU, one epoch of LoRA, and adapters small enough to swap per task on one base model. The main lesson is that adapting the vision encoder, not only the language model, is what makes the models understand drawings rather than guess from them, and what lets them absorb annotations and domains they never trained on. The transfer to prosthetic parts shows the method works outside its training domain, that its failures are easy to locate, and that the whole pipeline can run locally on proprietary or patient-specific data. We see this as a first step toward VLMs that read, and eventually generate, engineering drawings for specific domains.

References

1. American Society of Mechanical Engineers: Asme y14.5-2018: Dimensioning and tolerancing (2018)
2. Ataei, M., Askari, F., Malekshan, K.R., Jayaraman, P.K.: Zero-to-cad: Agentic synthesis of interpretable cad programs at million-scale without real data. arXiv preprint arXiv:2604.24479 (2026)
3. Bai, S., Cai, Y., Chen, R., et al.: Qwen3-vl technical report. arXiv preprint arXiv:2511.21631 (2025)
4. Bai, S., Chen, K., Liu, X., et al.: Qwen2.5-vl technical report. arXiv preprint arXiv:2502.13923 (2025)
5. Camba, J.D., Contero, M., Company, P.: Parametric cad modeling: An analysis of strategies for design reusability. Computer-Aided Design **74**, 18–31 (2016)
6. Coakley, M.F., Hurt, D.E., Weber, N., Mtingwa, M., Fincher, E.C., Alekseyev, V., Chen, D.T., Yun, A., Gizaw, M., Swan, J., Yoo, T.S., Yuyen, Y.: The NIH 3D Print Exchange: A Public Resource for Bioscientific and Biomedical 3D Prints. 3D Printing and Additive Manufacturing **1**(3), 137–140 (2014). https://doi.org/10.1089/3dp.2014.1503, https://journals.sagepub.com/doi/abs/10.1089/3dp.2014.1503
7. Dettmers, T., Pagnoni, A., Holtzman, A., Zettlemoyer, L.: Qlora: Efficient finetuning of quantized llms. In: Advances in Neural Information Processing Systems (NeurIPS) (2023)
8. Doris, A.C., Grandi, D., Tomich, R., Alam, M.F., Ataei, M., Cheong, H., Ahmed, F.: Designqa: A multimodal benchmark for evaluating large language models' understanding of engineering documentation. Journal of Computing and Information Science in Engineering **25**(2) (2025)
9. Farin, G.: Curves and Surfaces for CAGD: A Practical Guide. Morgan Kaufmann, 5th edn. (2002)
10. Fu, X., Hu, Y., Li, B., Feng, Y., Wang, H., Lin, X., Roth, D., Smith, N.A., Ma, W.C., Krishna, R.: Blink: Multimodal large language models can see but not perceive. In: European Conference on Computer Vision (ECCV). pp. 148–166 (2024)
11. Gereffi, G., Frederick, S., Bamber, P.: Diverse paths of upgrading in high-tech manufacturing: Costa Rica in the electronics and medical devices global value chains. Transnational Corporations **26**(1), 1–29 (2019)
12. Goyal, Y., Khot, T., Summers-Stay, D., Batra, D., Parikh, D.: Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2017)
13. Hoffmann, C.M.: Geometric and Solid Modeling: An Introduction. Morgan Kaufmann (1989)
14. Holtzman, A., West, P., Shwartz, V., Choi, Y., Zettlemoyer, L.: Surface form competition: Why the highest probability answer isn't always right. In: Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP). pp. 7038–7051 (2021)
15. Hu, E.J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., Chen, W.: Lora: Low-rank adaptation of large language models. In: International Conference on Learning Representations (ICLR) (2022)
16. International Organization for Standardization: Iso 13485:2016: Medical devices — quality management systems — requirements for regulatory purposes (2016)

17. Jayaraman, P.K., Sanghi, A., Lambourne, J.G., Willis, K.D., Davies, T., Shayani, H., Morris, N.: Uv-net: Learning from boundary representations. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 11703–11712 (2021)
18. Kadavath, S., Conerly, T., Askell, A., Henighan, T., Drain, D., Perez, E., Schiefer, N., Hatfield-Dodds, Z., et al.: Language models (mostly) know what they know. arXiv preprint arXiv:2207.05221 (2022)
19. Khan, M.S., Sinha, S., Sheikh, T.U., Stricker, D., Ali, S.A., Afzal, M.Z.: Text2cad: Generating sequential cad designs from beginner-to-expert level text prompts. In: Advances in Neural Information Processing Systems (NeurIPS). vol. 37, pp. 7552–7579 (2024)
20. Koch, S., Matveev, A., Jiang, Z., Williams, F., Artemov, A., Burnaev, E., Alexa, M., Zorin, D., Panozzo, D.: Abc: A big cad model dataset for geometric deep learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 9601–9611 (2019)
21. Lambourne, J.G., Willis, K.D., Jayaraman, P.K., Sanghi, A., Meltzer, P., Shayani, H.: Brepnet: A topological message passing system for solid models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 12773–12782 (2021)
22. Li, Y., Du, Y., Zhou, K., Wang, J., Zhao, W.X., Wen, J.R.: Evaluating object hallucination in large vision-language models. In: Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP) (2023)
23. Liu, H., Li, C., Wu, Q., Lee, Y.J.: Visual instruction tuning. In: Advances in Neural Information Processing Systems (NeurIPS) (2023)
24. Liu, S.Y., Wang, C.Y., Yin, H., Molchanov, P., Wang, Y.C.F., Cheng, K.T., Chen, M.H.: Dora: Weight-decomposed low-rank adaptation. In: International Conference on Machine Learning (ICML) (2024)
25. Meng, F., Wang, J., Li, C., Lu, Q., Tian, H., Liao, J., Zhu, X., Dai, J., Qiao, Y., Luo, P., Zhang, K., Shao, W.: Mmiu: Multimodal multi-image understanding for evaluating large vision-language models. arXiv preprint arXiv:2408.02718 (2024)
26. Moreno-García, C.F., Elyan, E., Jayne, C.: New trends on digitisation of complex engineering drawings. Neural Computing and Applications **31**(6), 1695–1712 (2019)
27. Promotora del Comercio Exterior de Costa Rica (PROCOMER): The world's health is made in Costa Rica: Medical devices sector study. Tech. rep., PROCOMER, San José, Costa Rica (2025), `https://procomer.com`
28. Qin, F., Lu, S., Hou, J., Wang, C., Fang, M., Liu, L.: Drawing2cad: Sequence-to-sequence learning for cad generation from vector drawings. In: Proceedings of the 33rd ACM International Conference on Multimedia (MM) (2025)
29. Rahmanzadehgervi, P., Bolton, L., Taesiri, M.R., Nguyen, A.T.: Vision language models are blind. In: Proceedings of the Asian Conference on Computer Vision (ACCV). pp. 18–34 (2024)
30. Rukhovich, D., Dupont, E., Mallis, D., Cherenkova, K., Kacem, A., Aouada, D.: Cad-recode: Reverse engineering cad code from point clouds. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 9801–9811 (2025)
31. Seff, A., Ovadia, Y., Zhou, W., Adams, R.P.: Sketchgraphs: A large-scale dataset for modeling relational geometry in computer-aided design. In: ICML 2020 Workshop on Object-Oriented Learning (2020)
32. Stroud, I.: Boundary Representation Modelling Techniques. Springer (2006)

33. Su, H., Maji, S., Kalogerakis, E., Learned-Miller, E.: Multi-view convolutional neural networks for 3d shape recognition. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV). pp. 945–953 (2015)
34. Suhr, A., Zhou, S., Zhang, A., Zhang, I., Bai, H., Artzi, Y.: A corpus for reasoning about natural language grounded in photographs. In: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL). pp. 6418–6428 (2019)
35. Team, G.: Gemma 3 technical report (2025), https://arxiv.org/abs/2503.19786
36. Tong, S., Liu, Z., Zhai, Y., Ma, Y., LeCun, Y., Xie, S.: Eyes wide shut? exploring the visual shortcomings of multimodal llms. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 9568–9578 (2024)
37. U.S. Food and Drug Administration: Quality system regulation, 21 cfr part 820 (1996)
38. Wang, F., Fu, X., Huang, J.Y., Li, Z., Liu, Q., Liu, X., Ma, M.D., Xu, N., Zhou, W., Zhang, K., et al.: Muirbench: A comprehensive benchmark for robust multi-image understanding. In: International Conference on Learning Representations (ICLR) (2025)
39. Willis, K.D., Pu, Y., Luo, J., Chu, H., Du, T., Lambourne, J.G., Solar-Lezama, A., Matusik, W.: Fusion 360 gallery: A dataset and environment for programmatic cad construction from human design sequences. ACM Transactions on Graphics (TOG) **40**(4) (2021)
40. Wu, R., Xiao, C., Zheng, C.: Deepcad: A deep generative network for computer-aided design models. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 6772–6782 (2021)
41. Xie, H., Ju, F.: Text-to-cadquery: A new paradigm for cad generation with scalable large model capabilities. arXiv preprint arXiv:2505.06507 (2025)
42. Xu, J., Wang, C., Zhao, Z., Liu, W., Ma, Y., Gao, S.: Cad-mllm: Unifying multimodality-conditioned cad generation with mllm. arXiv preprint arXiv:2411.04954 (2024)
43. Xu, X., Willis, K.D., Lambourne, J.G., Cheng, C.Y., Jayaraman, P.K., Furukawa, Y.: Skexgen: Autoregressive generation of cad construction sequences with disentangled codebooks. In: Proceedings of the 39th International Conference on Machine Learning (ICML). pp. 24698–24724 (2022)
44. You, Y., Uy, M.A., Han, J., Thomas, R., Zhang, H., You, S., Guibas, L.J.: Img2cad: Reverse engineering 3d cad models from images through vlm-assisted conditional factorization. arXiv preprint arXiv:2408.01437 (2024)
45. Zheng, C., Zhou, H., Meng, F., Zhou, J., Huang, M.: Large language models are not robust multiple choice selectors. In: International Conference on Learning Representations (ICLR) (2024)

Supplementary Material

A Qualitative hard-negative examples

Figures 7 and 8 show four hard negatives each — drawing/render pairs of visually similar but distinct parts (ground truth: *not* a match) — for which both the zero-shot baseline and the text-only fine-tune predict a match with high confidence, while the vision + text fine-tune correctly rejects the pair.



Fig. 7: Qwen3-VL: four hard negatives incorrectly matched by the baseline and text-only fine-tune, correctly rejected only after vision + text fine-tuning.

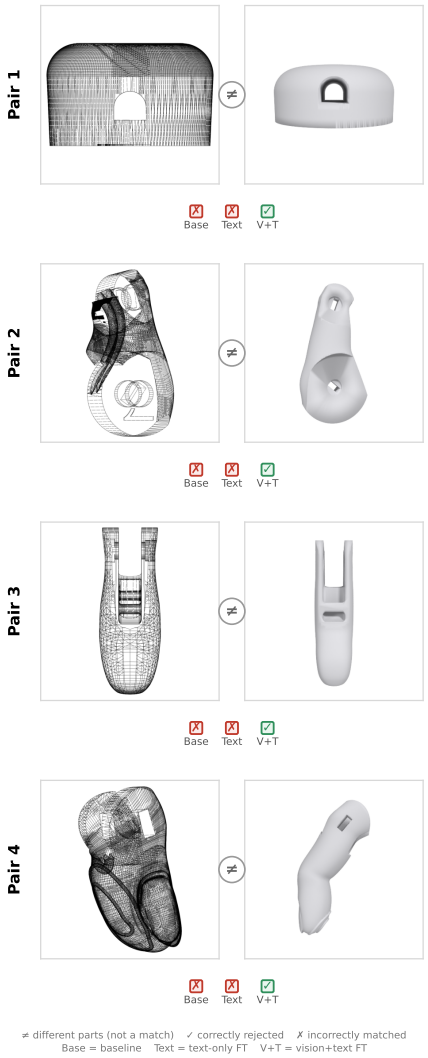


Fig. 8: Gemma-3-4B: four hard negatives incorrectly matched by the baseline and text-only fine-tune, correctly rejected only after vision + text fine-tuning.

Figure 9 repeats this analysis on the *dimensioned*-drawing variant of the held-out set (Omni-CAD, dimension annotations overlaid on the 2D view), for both backbones side by side.



Fig. 9: Dimensioned-drawing hard negatives, Qwen3-VL (left) and Gemma-3-4B (right): four pairs each, incorrectly matched by both the baseline and the text-only fine-tune, correctly rejected only after vision + text fine-tuning.

B Fine-tuning details

Table 6 lists the hyperparameters shared by all four adapters (two scopes, two backbones); only the set of adapted modules differs between the *text* and *text + vision* scopes.

Table 6: Fine-tuning hyperparameters, shared across both fine-tuned backbones.

Parameter	Value
LoRA rank / α / dropout	16 / 32 / 0.05
Adapted modules, *text*	q, k, v, o, gate, up, down (language model)
Adapted modules, *text + vision*	the above, plus attention and MLP projections of the vision encoder
Precision	bf16
Epochs	1
Learning rate	10^{-4}, cosine decay
Warmup	3%
Weight decay	0
Effective batch size	16 (1×16 accumulation)
Visual token budget	256–1024 patches of 28×28 px
Training pairs	107,160 (8,930 parts $\times$ 6 views $\times$ 2)
Negatives per positive	1; hard with prob. 0.7, else random (Section 5.1)
Hard-negative mining	OpenCLIP ViT-B/32 on renders, top-10 cosine, sample from top-3
Validation	500-pair subset of the held-out set, every 250 steps
Seed	13

C Calibration detail

Table 7: Calibration detail on the Omni-CAD held-out set. $\bar{p}_+$ and $\bar{p}_-$ are the mean p(Yes) on positive and negative pairs; τ^* is the accuracy-optimal threshold.

Model	$\bar{p}_+$	$\bar{p}_-$	τ^*
Zero-shot			
Gemma-3-4B	0.997	0.999	1.000
Qwen3-VL-2B	0.937	0.911	0.940
Qwen3-VL-8B	0.641	0.184	0.223
Fine-tuned (text)			
Gemma-3-4B	0.661	0.109	0.245
Qwen3-VL-2B	0.534	0.071	0.085
Fine-tuned (text + vision)			
Gemma-3-4B	0.752	0.068	0.119
Qwen3-VL-2B	0.781	0.078	0.269

p(Yes) is the normalised next-token probability of *Yes* against *No*, summed over tokenizations, and ECE uses 15 equal-width bins. Table 7 gives the class-mean scores and accuracy-optimal thresholds behind Section 5.3: the gap $\bar{p}_+ - \bar{p}_-$ is what a threshold can exploit, and $|\tau^* - 0.5|$ is what a user loses by taking $p(\text{Yes}) \geq 0.5$ at face value.