

CoVLA: Vision-Language Alignment with Fewer Vision Tokens

Junyong Park, Yong-Ju Lee

Electronics and Telecommunications Research Institute (ETRI)
Daejeon, Republic of Korea
{junyong.park, yongju}@etri.re.kr

Daeyoung Kim

Korea Advanced Institute of Science and Technology (KAIST)
Daejeon, Republic of Korea
kimd@kaist.ac.kr

Abstract

Vision-Language Models project every visual patch into the language model, producing hundreds of tokens that dominate prefill cost despite uneven importance across patches. Reducing this overhead without sacrificing accuracy is challenging: existing approaches either add learned selection modules that introduce parameters and training complexity, or compress representations post-hoc on models trained with full patch coverage. We introduce CoVLA, a VLM built around the RawPool connector, which ranks patches by importance signals already present in the frozen vision encoder—CLS-to-patch attention for CLIP, feature L_2 norm for SigLIP. RawPool keeps the selected top- K patches at full resolution and supplements them with a global mean token and pooled spatial summary tokens, all projected through a shared GatedSwiGLU. Because the selection mechanism uses no learned parameters, K can be changed at inference time; a curriculum that randomizes K during training lets a single checkpoint expose a controllable efficiency frontier without retraining. On MMBench with Vicuna-7B over three seeds, CoVLA improves standard/circular accuracy by +1.8 / +3.1 points at 321 tokens, a 44% reduction from the full 576-token visual input, over the MLP connector at matched backbone and data. Against a lower-resolution MLP baseline matched at comparable LLM prefill (CLIP-224 input, 256 visual tokens), CoVLA at $K=256$ gains +3.5 / +6.4 points, showing that *where* the visual token budget is spent matters, not only how large it is. On the stronger Qwen3-4B + SigLIP-2 backbone, full-token MLP remains slightly ahead but CoVLA retains 99.4% / 98.3% of its MMBench standard/circular accuracy at 44% fewer tokens; the trade-off is task-dependent—holistic understanding benchmarks tend to tolerate aggressive reduction while detail-dense OCR-heavy tasks favor full patch coverage. These findings, together with negative results on four learned-saliency alternatives, suggest that frozen encoder signals are a strong practical basis for vision token selection.

1 Introduction

Modern Vision-Language Models follow a three-component design: a vision encoder extracts patch-level features, a connector maps them into the language model’s embedding space, and a large language model (LLM) generates text conditioned on both modalities (Liu et al., 2023; 2024b; Li et al., 2025a). In the standard pipeline the connector is a linear projector that forwards every visual patch to the LLM; CoVLA replaces this with importance-based patch selection and token construction. A typical ViT-L encoder at 336×336 resolution produces 576 patches, and multi-tile processing (Liu et al., 2024b) can push this to 2,880 or

more. The resulting vision tokens often exceed 90% of the total sequence, yet analysis of attention patterns shows that most patches contribute little to the LLM’s predictions (Chen et al., 2024a).

Prior work has revealed that frozen vision encoders already contain strong importance signals. CLS-to-patch attention in CLIP concentrates over 60% of its mass on fewer than 6% of patches (Chen et al., 2024a), and feature norms in SigLIP show a similar pattern (Yang et al., 2025). Several methods exploit these signals to prune or merge tokens at inference time (Chen et al., 2024a; Shang et al., 2025; Yang et al., 2025), but they are applied post-hoc to models trained with full patch coverage. Other approaches learn a selection mechanism (Cha et al., 2024; Ye et al., 2025), adding parameters and training complexity. Neither strategy produces a single model that operates well across a range of token budgets.

We propose CoVLA (**Controllable Vision-Language Alignment**), a VLM that replaces the standard MLP connector with the RawPool connector, a new vision connector designed in this work. RawPool uses frozen encoder importance signals to select only the top- K most informative patches, supplements them with summary tokens for spatial coverage, and maps them through a shared GatedSwiGLU projector into the LLM (details in Section 3). The key design insight is that by training with a curriculum that randomizes K across batches, a single checkpoint learns to perform well at every token budget, producing a controllable efficiency frontier without retraining.

Our experiments span three LLM scales and nine benchmarks across four task families—holistic visual reasoning (MMBench, SEED-Image, MMStar), science and diagram reasoning (ScienceQA, AI2D, MMMU), math reasoning (MathVista), and OCR/generation (TextVQA, ChartQA, DocVQA)—and yield three main findings. First, a single CoVLA checkpoint traces a smooth accuracy-efficiency curve: at 321 tokens (44% fewer than MLP), circular accuracy drops by at most 1.3pp, and further reducing to 193 tokens incurs a further 0.6pp drop. Second, the trade-off is task-dependent: CoVLA is competitive on holistic benchmarks (SEED-Image, MMBench) while MLP retains a clear edge on detail-dense tasks (ScienceQA, MathVista) where comprehensive patch coverage matters. Third, alternative attempts to learn an importance signal (via STE, CLIP text supervision, or delta residuals) failed to outperform the frozen encoder’s own signal, a practical finding that simplifies connector design.

In summary, this paper makes the following contributions:

- A curriculum training strategy over K that produces a controllable efficiency frontier from a single checkpoint—no prior token-reduction method yields a single model that operates across a range of token budgets.
- The RawPool connector, which selects patches using frozen encoder importance and maps them through a shared GatedSwiGLU projector with zero added selection FLOPs and zero added parameters compared to the standard MLP connector.
- Comprehensive evaluation across three LLM scales and nine benchmarks, revealing a task-dependent boundary between token-efficient and detail-sensitive regimes.
- Negative results on learned saliency, documenting that frozen encoder signals are hard to improve upon in practice.

2 Related Work

The dominant VLM architecture, established by LLaVA (Liu et al., 2023) and refined in subsequent work (Liu et al., 2024a;b; Li et al., 2025a), projects every visual patch into the language model through a simple MLP connector. As resolution and multi-tile strategies have scaled up, so has the number of vision tokens: LLaVA-NeXT (Liu et al., 2024b) processes up to 2,880 tokens per image. This growth is at odds with the quadratic attention cost of transformer LLMs and motivates methods that reduce the vision token count without proportional accuracy loss.

Several approaches have been proposed to address this overhead. Training-free methods such as FastV (Chen et al., 2024a) and SparseVLM (Zhang et al., 2025b) prune

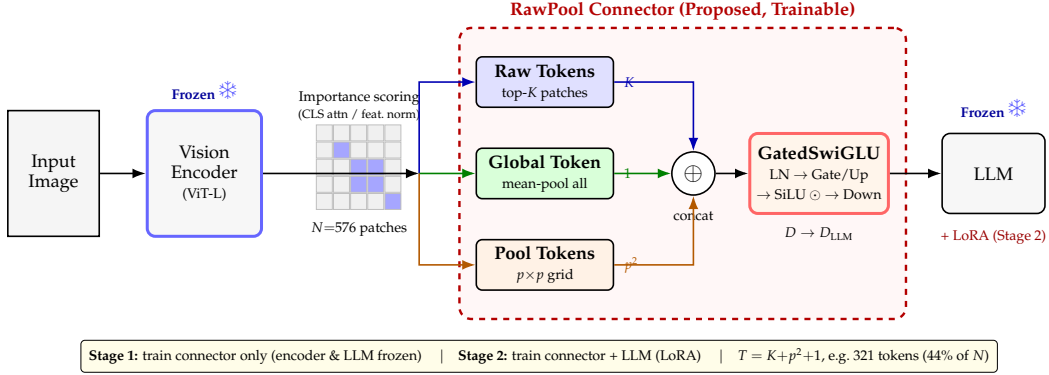


Figure 1: **CoVLA architecture**. The RawPool connector (dashed box, our proposed component) takes patch features from a frozen vision encoder and constructs three token types: K raw tokens selected by importance (blue), one global mean-pool token (green), and p^2 spatially pooled tokens (orange) for coarse spatial coverage. All tokens pass through a shared GatedSwiGLU projector into the LLM. Blue borders with $\star$ indicate frozen components; red borders indicate trainable components. In Stage 1 only the connector is trained; in Stage 2 the connector and LLM (via LoRA) are trained jointly.

low-importance patches at inference time, while PruMerge (Shang et al., 2025) and VisionZip (Yang et al., 2025) combine pruning with token merging to further compress the representation. At the extreme, LLaVA-Mini (Zhang et al., 2025a) compresses the entire image to a single token. These methods are effective but operate post-hoc on models trained with full patch coverage, limiting the model’s ability to adapt to reduced budgets. A separate line of work introduces learned selection: Honeybee (Cha et al., 2024) uses selective resampling, ATP-LLaVA (Ye et al., 2025) learns adaptive token pruning, and architectures such as Qwen-VL (Bai et al., 2023) and InternVL (Chen et al., 2024c) employ cross-attention or pixel shuffle for compression. These approaches add trainable parameters for the selection process itself.

The importance signals that guide selection need not be learned. CLS attention in CLIP concentrates mass on semantically salient patches (Chen et al., 2024a), and feature norms serve the same role for encoders without a CLS token (Yang et al., 2025). Huang et al. (2026) observed that naive pruning can destroy positional structure, motivating spatial-aware designs. Our work differs from prior methods in two respects: we train the model with a curriculum over K , so a single checkpoint produces an entire efficiency frontier rather than a single operating point, and we use frozen encoder signals directly for selection, avoiding the complexity of learned saliency modules.

3 Method

3.1 The CoVLA Architecture

CoVLA consists of three components: a frozen vision encoder, the RawPool connector (our proposed component), and a language model. The vision encoder and language model follow standard designs; the contribution of this paper is the RawPool connector and its associated training strategy. Figure 1 illustrates the full architecture.

Given an image I , a frozen vision encoder $\mathcal{V}$ produces N patch features $\mathbf{F} = \{\mathbf{f}_1, \dots, \mathbf{f}_N\} \in \mathbb{R}^{N \times D}$ and an importance vector $\mathbf{s} \in \mathbb{R}^N$. The RawPool connector transforms these into a compact sequence of $T = 1 + p^2 + K$ vision tokens in the LLM’s embedding space $\mathbb{R}^{D_{\text{LLM}}}$, where $T \ll N$ for typical operating points.

Importance scoring. The importance signal $\mathbf{s}$ is obtained directly from the frozen vision encoder, requiring no additional parameters. For encoders with a CLS token (e.g., CLIP

ViT-L/14), we extract the CLS-to-patch attention from the penultimate layer:

$$\mathbf{s}_{\text{CLS}} = \frac{1}{H} \sum_{h=1}^H \mathbf{A}_{0,1:N}^{(h)} \quad (1)$$

where $\mathbf{A}^{(h)}$ is the attention matrix of head h at layer $L-2$, index 0 is the CLS token, and indices $1, \dots, N$ are the N patch tokens. This signal is highly concentrated: the top-32 patches (5.6% of 576) capture 60.2% of the total attention mass, a $10.8\times$ concentration factor over uniform (Appendix, Table 9). For encoders without a CLS token (e.g., SigLIP-2), we use the L_2 norm of patch features:

$$\mathbf{s}_{\text{norm}} = \|\mathbf{f}_i\|_2, \quad i = 1, \dots, N \quad (2)$$

For NaFlex (Google DeepMind, 2025) variable-resolution encoders, padding tokens are masked: $\mathbf{s}_i \leftarrow \mathbf{s}_i \cdot m_i$. Importantly, both signals are read directly from tensors the encoder forward pass already computes—a slice of the attention matrix or feature norms—so importance scoring adds *no learned-network FLOPs and no parameters* relative to the standard MLP connector; the top- K and gather operations are non-parametric indexing whose cost is negligible compared with the encoder and LLM (Appendix, Table 7). Learned-saliency baselines (e.g., Honeybee (Cha et al., 2024), ATP-LLaVA (Ye et al., 2025)) instead require additional trainable heads operating over all N patches at each forward pass.

Token construction. Given $\mathbf{F}$ and $\mathbf{s}$, the RawPool connector constructs three types of tokens:

1. Raw Tokens (K tokens). The top- K patches by importance are selected without compression:

$$\mathcal{I}_K = \text{argtop-}K(\mathbf{s}), \quad \mathbf{R} = \{\mathbf{f}_i : i \in \mathcal{I}_K\} \quad (3)$$

These tokens preserve the full resolution of the most important patches.

2. Global Token (1 token). A single vector summarizing the entire image:

$$\mathbf{g} = \frac{1}{N} \sum_{i=1}^N \mathbf{f}_i \quad (4)$$

3. Pool Tokens (p^2 tokens). The patch features are reshaped into a 2D grid and spatially pooled:

$$\mathbf{P} = \text{AdaptiveAvgPool2d}(\text{reshape}(\mathbf{F}, h, w), (p, p)) \quad (5)$$

where (h, w) are the spatial grid dimensions and $p \in \{6, 8, 12\}$. Raw tokens focus on fine-grained detail in important regions, while pool tokens may provide complementary spatial coverage of the full image, though they do not affect MMBench accuracy in our ablation (Appendix, Table 10). Their role may be more significant on tasks requiring spatial understanding.

Projection. All three token types pass through a shared GatedSwiGLU projector (Shazeer, 2020), chosen over a standard two-layer MLP because gated projections improve representation quality and we found GatedSwiGLU to outperform GELU MLP in preliminary experiments:

$$\mathbf{o} = W_{\text{down}}(\text{SiLU}(W_{\text{gate}}(\text{LN}(\mathbf{x}))) \odot W_{\text{up}}(\text{LN}(\mathbf{x}))) \quad (6)$$

where $W_{\text{gate}}, W_{\text{up}} \in \mathbb{R}^{D \times D_{\text{LLM}}}$ and $W_{\text{down}} \in \mathbb{R}^{D_{\text{LLM}} \times D_{\text{LLM}}}$. Token type embeddings (global, pooled, raw) are added after projection. To summarize the relationship between components: CoVLA is the full VLM system, RawPool is the connector (importance scoring + token construction + projector), and the GatedSwiGLU projector is the learned projection layer within RawPool.

3.2 Training

Curriculum over K . To produce a single model that operates across all token budgets, we train with a randomized curriculum: a warmup phase (first 30%) uses fixed $K=128$, $p=8$, then the remaining 70% samples $K \in \{64, 128, 256, 384\}$ and $p \in \{6, 8, 12\}$ uniformly per

Table 1: **Main results and LLM scaling.** Top: CoVLA vs. MLP connector across LLM scales; only the connector differs. Vicuna-7B (both connectors) and Qwen3-4B CoVLA report mean $\pm$ std over 3 seeds; other entries are single runs. Bottom: CoVLA across four backbone combinations. Bold indicates the higher score in each pair.

LLM	Connector / Encoder	Tokens	Dataset	Standard	Circular
Vicuna-7B	MLP (all patches)	576	665K	71.2 $\pm$ 2.4	58.5 $\pm$ 5.7
	CoVLA ($K=256, p=8$)	321	665K	73.0$\pm$1.8	61.6$\pm$4.2
Qwen3-4B	MLP (all patches)	576	OV-2M	82.9	76.1
	CoVLA ($K=256, p=8$)	321	OV-2M	82.4 $\pm$ 0.4	74.8 $\pm$ 0.7
Qwen3-1.7B	MLP (all patches)	576	OV-2M	78.2	69.1
	CoVLA ($K=256, p=8$)	321	OV-2M	78.1	67.8
<i>Scaling CoVLA across backbones ($K=256, p=8, 321$ tokens)</i>					
Vicuna-7B	CLIP ViT-L/14@336	321	665K	73.0	61.6
Qwen3-1.7B	SigLIP-2-NaFlex	321	OV-2M	78.1	67.8
Qwen3-4B	SigLIP-2-NaFlex	321	OV-2M	82.4	74.8
Minstral-3-8B	SigLIP-2	321	OV-2M	80.7	73.8

batch. At inference time, K and p can be freely chosen, so a single training run produces the entire efficiency frontier. The resulting token count is $T = 1 + p^2 + K$; for $K=256, p=8$ this is 321 (a 44% reduction from MLP’s 576). Token reduction affects only the LLM; the vision encoder cost is unchanged (Appendix, Table 7).

Two-stage pipeline. We follow the standard two-stage VLM training recipe. In Stage 1 (vision-text alignment), only the RawPool connector parameters are trained while the vision encoder and LLM remain frozen. Training data consists of 558K image-caption pairs (Liu et al., 2023). This stage teaches the connector to project vision features into the LLM’s embedding space across all curriculum configurations. In Stage 2 (instruction tuning), the connector and LLM (via LoRA (Hu et al., 2022)) are jointly trained on instruction-following data. We evaluate on two dataset scales: LLaVA-665K (Liu et al., 2024a) and OneVision-2M (Li et al., 2025a). The vision encoder remains frozen throughout both stages.

4 Experiments

We evaluate CoVLA across three LLM scales and two vision encoders: Vicuna-7B (Zheng et al., 2023) with CLIP ViT-L/14@336 (576 patches, 1024-dim), Qwen3-4B (Qwen Team, 2025) with SigLIP-2-NaFlex (Google DeepMind, 2025) (576 patches, 1152-dim), and Qwen3-1.7B (Qwen Team, 2025) with the same SigLIP-2-NaFlex encoder. For each LLM, we compare CoVLA against the standard MLP connector (Liu et al., 2023), a two-layer GELU MLP that projects all N patches without selection. All baselines use identical training recipes (same data, hyperparameters, LoRA configuration) and differ only in the connector architecture.

Stage 1 trains for 1 epoch on LLaVA-558K with learning rate 2×10^{-3} and cosine decay. Stage 2 uses LoRA ($r=128, \alpha=256$) on all attention and MLP projections, with learning rate 2×10^{-4} . All experiments run on 4–8 $\times$ A100 (80GB) or A6000 (48GB) GPUs with DDP. We report results on the MMBench dev-en split (Liu et al., 2024c) using both standard accuracy (single letter scoring) and circular accuracy (correct across all four MCQ option rotations), which is more robust to position bias (Liu et al., 2024c). For the Qwen3-4B scale, we additionally evaluate on a comprehensive 9-benchmark suite: SEED-Image (Li et al., 2024), AI2D (Kembhavi et al., 2016), ScienceQA (Lu et al., 2022), MMMU (Yue et al., 2024), MMStar (Chen et al., 2024b), MathVista (Lu et al., 2024) (MCQ benchmarks via logit scoring), TextVQA (Singh et al., 2019), ChartQA (Masry et al., 2022), and DocVQA (Mathew et al., 2021) (open-ended benchmarks via generation).

Table 2: **Comparison with token reduction methods.** All use CLIP + Vicuna-7B unless noted. † = training-free, applied to LLaVA-1.5. We report MMBench circular accuracy, which is robust to position bias. ‡ = reproduced under our evaluation protocol. CoVLA (Qwen3-4B) benefits from a stronger backbone; Vicuna-7B results shown for controlled comparison.

Method	Tokens	Train	MMB ^C	SQA	SEED	GQA	POPE	TextVQA
<i>Baseline</i>								
LLaVA-1.5-7B (Liu et al., 2024a)	576	FT	64.3	66.8	58.6	62.0	85.9	58.2
<i>Training-free methods (applied to LLaVA-1.5-7B)</i>								
FastV† (Chen et al., 2024a)	192	No	61.2	67.3	57.1	52.7	64.8	52.5
VisionZip† (Yang et al., 2025)	192	No	63.0	68.9	56.4	59.3	85.3	57.3
VisionZip† (Yang et al., 2025)	64	No	60.1	69.0	52.2	55.1	77.0	55.5
SparseVLM† (Zhang et al., 2025b)	128	No	64.5	68.6	58.2	58.4	85.0	56.7
<i>Methods requiring training (LLaVA-1.5-7B architecture)</i>								
PruMerge+ (Shang et al., 2025)	~144	LoRA	64.9	68.3	—	—	84.0	57.1
TokenPacker (Li et al., 2025b)	144	FT	65.1	—	—	61.9	87.0	—
TokenPacker (Li et al., 2025b)	64	FT	64.1	—	—	61.1	86.3	—
MQT-LLaVA (Hu et al., 2024)	256	FT	64.3	67.6	—	61.6	84.4	—
MQT-LLaVA (Hu et al., 2024)	64	FT	63.5	67.0	—	60.0	83.6	—
LLaVA-Mini (Zhang et al., 2025a)	1	FT	65.6	70.4	58.5	60.9	84.4	57.0
ATP-LLaVA (Ye et al., 2025)	~144	FT	66.0	69.1	57.3	59.5	84.2	—
<i>Ours (controlled: same CLIP + Vicuna-7B backbone)</i>								
MLP (reproduced)‡	576	LoRA	65.0	67.5	65.4	60.4	89.7	48.2
CoVLA (Vicuna-7B + CLIP)‡	321	LoRA	64.3	69.1	63.0	58.9	87.1	44.6
<i>Ours (Qwen3-4B + SigLIP-2, stronger backbone)</i>								
MLP (Qwen3-4B)‡	576	LoRA	76.1	88.1	72.5	50.8	91.4	64.5
CoVLA (Qwen3-4B)‡	321	LoRA	74.8	83.7	72.4	49.3	87.9	64.0

4.1 Main Results

Table 1 presents our main comparison. With Vicuna-7B (3-seed average), CoVLA leads MLP on both standard (+1.8pp) and circular (+3.1pp) accuracy while using 44% fewer vision tokens (321 vs. 576). CoVLA also exhibits lower variance across seeds (std 1.8 vs. 2.4 on standard, 4.2 vs. 5.7 on circular), suggesting that curriculum training acts as implicit regularization. With Qwen3-4B (CoVLA averaged over 3 seeds), the full-token MLP is slightly ahead—82.9 / 76.1 vs. 82.4 / 74.8 for CoVLA at $K=256$ —so at this stronger backbone the result is an efficiency trade-off rather than an accuracy win: CoVLA retains 99.4% / 98.3% of MLP accuracy at 44% fewer tokens. At Qwen3-1.7B, MLP leads by 1.3pp on circular, suggesting smaller LLMs may benefit more from full patch coverage. The bottom panel shows that CoVLA’s connector design is portable across backbones.

4.2 Token Efficiency Analysis

Table 2 compares CoVLA against prior token reduction methods. Our reproduced MLP (65.0% circular) closely matches LLaVA-1.5’s published 64.3%, anchoring the comparison. CoVLA achieves comparable accuracy at 321 tokens with no learned selection mechanism.

Table 3 shows the efficiency curve from a single CoVLA model. The elbow is at $K=128$ (193 tokens), where circular accuracy is only 0.6pp below the peak. Below $K=32$, performance degrades sharply. At $K=256$ (321 tokens), CoVLA matches MLP on standard accuracy (74.0% vs. 73.9%) while trailing by 0.7pp on circular (64.3% vs. 65.0%), using 44% fewer tokens.

A natural question is whether CoVLA’s advantage comes only from feeding the LLM fewer tokens, in which case a uniformly lower-resolution MLP at a matched prefill budget should recover the same operating point. Table 4 shows it does not. At a comparable LLM-side budget, CoVLA at $K=256$ (321 tokens) outperforms the lower-resolution MLP (256 tokens) by **+3.5pp standard / +6.4pp circular**; even CoVLA at $K=128$ (193 tokens—fewer than the MLP control) leads by +2.9 / +5.8pp. Cutting tokens by lowering resolution degrades all

Table 3: **Token efficiency sweep** (Vicuna-7B + CLIP, single run). A single CoVLA model evaluated at different K values. The efficiency elbow at $K=128$ retains 99% of peak circular accuracy while using 40% fewer tokens than $K=256$.

K	Pool p	Tokens	Standard	Circular
256	8	321	74.0	64.3
128	8	193	73.4	63.7
64	8	129	71.7	61.5
32	8	97	64.2	51.9
16	8	81	48.9	32.2
8	8	73	43.7	26.0
MLP baseline		576	73.9	65.0

Table 4: **Inference iso-prefill control** (Vicuna-7B + CLIP, LoRA $r=128$). To test whether CoVLA’s gain is merely from feeding the LLM fewer tokens, we compare against a lower-resolution MLP (CLIP ViT-L/14 at 224×224 , 256 patches) matched to CoVLA’s prefill budget. MLP@224 is a two-seed mean. Bold = best at a comparable budget.

Method	Vision tokens	Standard	Circular
MLP @ CLIP-224	256	70.5	57.9
CoVLA ($K=128$)	193	73.4	63.7
CoVLA ($K=256$)	321	74.0	64.3
MLP @ CLIP-336 (full)	576	73.9	65.0

patches uniformly, whereas CoVLA preserves the most salient patches at full resolution: *where* the token budget is spent matters, not only how large it is.

A per-category breakdown on Qwen3-4B (Appendix, Table 11) shows CoVLA winning 11 of 20 MMBench categories at $K=256$, with the largest gains on image quality (+15.1pp) and social relation (+4.6pp).

4.3 Extended Evaluation

Table 5 extends evaluation to 6 MCQ benchmarks. CoVLA leads on SEED (+0.6pp) and MM-Star (+0.4pp), while MLP leads on detail-dense benchmarks (ScienceQA +4.3pp, MathVista +5.8pp, AI2D +1.9pp). Degradation is graceful: $K=512$ to $K=128$ ($3.5 \times$ token reduction) costs only 2.5pp on average.

A preliminary experiment with budget-aware GRPO (Shao et al., 2024) (Appendix, Table 8) suggests that RL can partially close the gap on detail-heavy benchmarks at equal token budget, though this is orthogonal to the connector contribution.

A Vicuna-7B multi-benchmark comparison (Appendix, Table 12) shows MLP leading on 4 of 5 benchmarks at this scale, though gaps are narrow (<2 pp on most) and CoVLA uses 44% fewer tokens.

Text-rich stress test (VCR). To probe whether frozen-encoder importance discards small or low-contrast text, we evaluate on Visual Caption Restoration (VCR), a text-heavy caption-restoration benchmark, on its full 5,000-sample test split (Qwen3-4B + SigLIP-2). On VCR-easy, CoVLA at $K=256$ (321 tokens) *exceeds* the MLP baseline (576 tokens) by +2.96 F1, and by +4.57 F1 at $K=512$ (iso-token) — contrary to the expectation that text-rich inputs require dense patch coverage. On VCR-hard, both connectors sit near the exact-match floor ($<1\%$ EM); MLP holds a small aggregate F1 edge (17.2 vs. 16.3 at $K=256$), but a paired 5,000-example analysis splits almost evenly (CoVLA wins 30.1%, ties 35.3%, MLP wins 34.7%), indicating a shared restoration-capability ceiling for both connectors rather than a systematic loss of text-relevant evidence under selection.

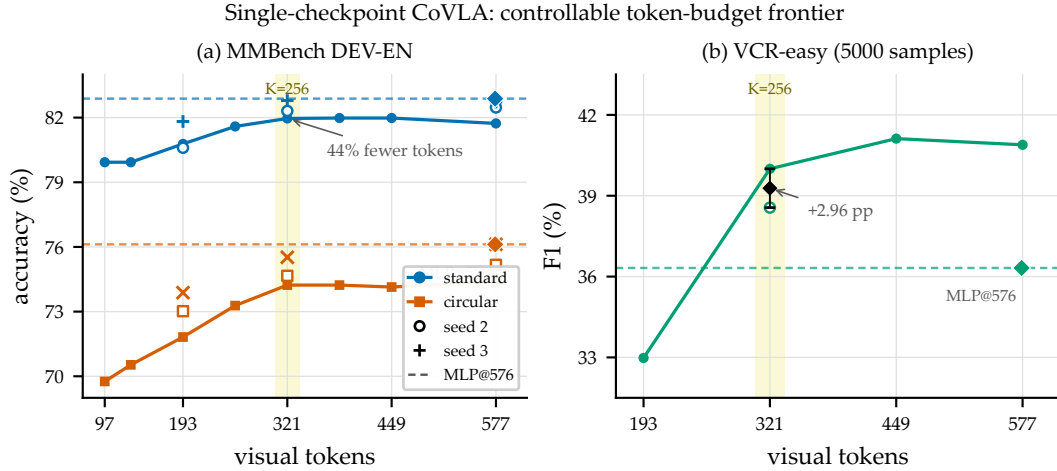


Figure 2: **Controllable token-budget frontier from a single CoVLA checkpoint** (Qwen3-4B + SigLIP-2-NaFlex, LoRA $r=128$). Each point is the same model evaluated at a different K ; seed 2 points overlaid where measured. **(a)** MMBench DEV-EN standard and circular accuracy across $K \in \{32, 64, 128, 192, 256, 320, 384, 512\}$. **(b)** VCR-easy F1 across $K \in \{128, 256, 384, 512\}$ on the full 5000-sample test split; diamond shows the two-seed mean at $K=256$ with $\pm$ SD. Dashed lines and diamonds at 576 tokens denote the MLP baseline. At $K=256$ (321 tokens, 44% fewer than MLP), CoVLA retains $\geq 99\%$ of MLP MMBench accuracy and *exceeds* MLP on VCR-easy by 2.96 pp F1.

Table 5: **Multi-benchmark comparison** (Qwen3-4B + SigLIP-2-NaFlex, OV-2M, MCQ benchmarks via logit scoring). CoVLA at three token budgets vs. MLP. Bold = best per benchmark.

Benchmark	CoVLA			MLP
	$K=512$ (577 tok)	$K=256$ (321 tok)	$K=128$ (165 tok)	576 tok
SEED-Image	73.1	72.4	70.2	72.5
AI2D	72.2	71.6	69.4	74.1
ScienceQA-I	83.8	83.7	82.3	88.1
MMMU-val	43.8	44.0	43.8	45.1
MMStar	52.4	50.5	47.7	52.0
MathVista	51.8	50.4	48.9	57.6
Average	62.9	62.1	60.4	64.9

We provide additional analyses in Appendix A: (1) qualitative visualization of patch selection at different K (Figure 3); (2) a component ablation on MMBench showing that raw tokens are the primary accuracy driver, while pool tokens provide complementary spatial coverage (Table 10); (3) CLS attention concentration justifying top- K selection (Table 9); and (4) negative results on learned importance signals.

4.4 Full Fine-Tuning

All preceding experiments use LoRA adapters on a frozen LLM. To test whether CoVLA’s token efficiency advantage persists when the LLM is fully trainable, we conduct a full fine-tuning ablation using DeepSpeed ZeRO-2 with Vicuna-7B + CLIP ViT-L/14@336 on LLaVA-665K. We evaluate CoVLA at three token budgets ($K=128$, $K=256$, $K=512$) against the MLP baseline (576 tokens).

Table 6: Full fine-tuning ablation (Vicuna-7B + CLIP, LLaVA-665K). CoVLA wins 6 of 11 metrics at $K=256$ (321 tokens) vs. MLP (576 tokens), with mixed results across benchmarks. Bold indicates the best score per benchmark.

Benchmark	MLP	CoVLA		
	576 tok	$K=128$ (193 tok)	$K=256$ (321 tok)	$K=512$ (577 tok)
<i>MMBench</i>				
Standard	73.1	73.8	74.1	73.8
Circular	64.9	63.7	64.5	63.7
<i>MCQ Benchmarks</i>				
SEED-Image	64.5	62.5	62.4	62.5
AI2D	54.5	52.9	52.8	53.2
ScienceQA-I	66.8	68.2	69.1	68.2
MMMU-val	37.8	37.5	38.2	37.7
MMStar	32.9	33.0	33.0	33.0
MathVista	24.1	25.9	25.1	25.8
<i>Open-Ended Benchmarks</i>				
TextVQA	49.4	43.0	44.0	43.2
ChartQA	15.3	14.7	15.6	14.7
DocVQA	4.9	3.6	3.7	3.6

The full fine-tuning results reveal a nuanced but consistent pattern. At $K=256$ (321 tokens, 44% fewer than MLP), CoVLA leads on MMBench standard accuracy (+1.0pp), ScienceQA (+2.3pp), MMMU (+0.4pp), MMStar (+0.1pp), MathVista (+1.0pp), and ChartQA (+0.3pp), winning 6 of 11 metrics while using substantially fewer tokens. MLP retains advantages on SEED-Image (−2.1pp), AI2D (−1.7pp), and particularly TextVQA (−5.4pp), consistent with the LoRA finding that detail-dense benchmarks favor full patch coverage. MMBench circular accuracy is nearly tied (64.5% vs. 64.9%, −0.4pp).

Notably, the token efficiency curve is flat: reducing from $K=256$ to $K=128$ (a further 40% token reduction) costs at most 1.5pp on any individual benchmark, and CoVLA at $K=128$ (193 tokens, 67% fewer than MLP) still wins on ScienceQA, MathVista, and MMBench standard. This confirms that the token-efficient behavior observed under LoRA (Section 4) extends to the full fine-tuning regime. LLM-level efficiency measurements (Appendix, Table 7) show $2.6\times$ prefill speedup and 1.4 GB peak memory savings at batch size 8 for 193 vs. 576 tokens.

5 Conclusion

We introduced CoVLA, a Vision-Language Model built around the RawPool connector, which selects the most informative visual patches using importance signals from a frozen vision encoder and projects them into the language model through a shared GatedSwiGLU layer. By training with a curriculum that randomizes the token budget K , a single CoVLA checkpoint produces a controllable efficiency frontier: at 321 tokens (44% fewer than the standard MLP connector), accuracy drops by at most 1.3pp on MMBench circular across three LLM scales, and the elbow at 193 tokens (67% fewer) retains 99% of peak accuracy.

Evaluation across nine benchmarks reveals that the trade-off is task-dependent. CoVLA is competitive on holistic understanding tasks such as SEED-Image and MMBench, while MLP retains an advantage on detail-dense tasks such as ScienceQA and MathVista, where comprehensive patch coverage matters. This pattern persists under full fine-tuning. Our negative results on learned saliency further suggest that frozen encoder features provide a strong importance signal that additional learned mechanisms did not improve upon in our experiments.

Limitations. MLP retains a clear edge on detail-dense benchmarks (ScienceQA, MathVista, TextVQA); we hypothesize that frozen encoder importance signals may inherently under-

weight small or low-contrast patches (e.g., text, fine symbols), leading to the observed drops on OCR-heavy tasks. Token reduction affects only the LLM; the vision encoder processes all patches regardless, so end-to-end speedups are modest at batch size 1. The full token sweep is shown on Vicuna-7B + CLIP (Table 3) and Qwen3-4B + SigLIP-2 (Figure 2); frontier shape on other backbones we have not evaluated may differ. All experiments use single-tile images to isolate the efficiency of the connector architecture without the confounding variables of multi-tile fusion; AnyRes compatibility, per-sample adaptive K , and broader multi-seed evaluation remain future work. Applying post-hoc training-free compression methods (FastV, SparseVLM) to the Qwen3-4B backbone for a controlled comparison is also left for future work.

Acknowledgments

This work was supported by Institute of Information & Communications Technology Planning & Evaluation (IITP) grants funded by the Korea government (MSIT) (No. 2022-0-00871, Development of AI Autonomy and Knowledge Enhancement for AI Agent Collaboration, 50%) and Electronics and Telecommunications Research Institute (ETRI) grant funded by the Korean government (26CB1200, Development and Application of Science-Specialized Multimodal Foundation Models, 50%).

LLM Usage Disclosure

Large language models were used to assist with experiment management and paper editing. All scientific claims, methodological decisions, and experimental designs were made by the authors.

References

- Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. Qwen-VL: A versatile vision-language model for understanding, localization, text reading, and beyond. *arXiv preprint arXiv:2308.12966*, 2023.
- Junbum Cha, Wooyoung Kang, Jonghwan Mun, and Byungseok Roh. Honeybee: Locality-enhanced projector for multimodal LLM. In *CVPR*, 2024.
- Liang Chen, Haozhe Zhao, Tianyu Liu, Shuai Bai, Junyang Lin, Chang Zhou, and Baobao Chang. An image is worth 1/2 tokens after layer 2: Plug-and-play inference acceleration for large vision-language models. In *ECCV*, 2024a.
- Lin Chen et al. Are we on the right way for evaluating large vision-language models? In *NeurIPS*, 2024b.
- Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, Bin Li, Ping Luo, Tong Lu, Yu Qiao, and Jifeng Dai. InternVL: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *CVPR*, 2024c.
- Google DeepMind. SigLIP 2: Multilingual vision-language encoders with improved semantic understanding, localization, and dense features. *arXiv preprint arXiv:2502.14786*, 2025.
- Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *ICLR*, 2022.
- Wenbo Hu, Zi-Yi Dou, Liunian Harold Li, et al. Matryoshka query transformer for large vision-language models. In *NeurIPS*, 2024.
- Yihong Huang, Fei Ma, Yihua Shao, Jingcai Guo, Zitong Yu, Laizhong Cui, and Qi Tian. Nüwa: Mending the spatial integrity torn by VLM token pruning. *arXiv preprint arXiv:2602.02951*, 2026.
- Aniruddha Kembhavi, Michael Salvato, Eric Kolve, Minjoon Seo, Hannaneh Hajishirzi, and Ali Farhadi. A Diagram Is Worth a Dozen Images. In *ECCV*, 2016.
- Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Yanwei Li, Ziwei Liu, and Chunyuan Li. LLaVA-OneVision: Easy visual task transfer. *TMLR*, 2025a.

- Bo Li et al. SEED-Bench: Benchmarking multimodal llms with generative comprehension. In *CVPR*, 2024.
- Wentong Li et al. TokenPacker: Efficient visual projector for multimodal LLM. *International Journal of Computer Vision*, 2025b.
- Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *NeurIPS*, 2023.
- Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. In *CVPR*, 2024a.
- Haotian Liu, Chunyuan Li, Yuheng Li, Bo Wang, and Yong Jae Lee. LLaVA-NeXT: Improved reasoning, OCR, and world knowledge. *Blog post*, 2024b.
- Yuan Liu et al. MMBench: Is your multi-modal model an all-around player? In *ECCV*, 2024c.
- Pan Lu et al. Learn to Explain: Multimodal reasoning via thought chains for ScienceQA. In *NeurIPS*, 2022.
- Pan Lu et al. MathVista: Evaluating mathematical reasoning of foundation models in visual contexts. In *ICLR*, 2024.
- Ahmed Masry et al. ChartQA: A benchmark for question answering about charts with visual and logical reasoning. In *Findings of ACL*, 2022.
- Minesh Mathew et al. DocVQA: A dataset for vqa on document images. In *WACV*, 2021.
- Qwen Team. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.
- Yuzhang Shang, Mu Cai, Bingxin Xu, Yong Jae Lee, and Yan Yan. LLaVA-PruMerge: Adaptive token reduction for efficient large multimodal models. In *ICCV*, 2025.
- Zhihong Shao et al. DeepSeekMath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.
- Noam Shazeer. GLU variants improve transformer. *arXiv preprint arXiv:2002.05202*, 2020.
- Amanpreet Singh et al. Towards VQA Models That Can Read. In *CVPR*, 2019.
- Senqiao Yang et al. VisionZip: Longer is better but not necessary in vision language models. In *CVPR*, 2025.
- Jingwei Ye et al. ATP-LLaVA: Adaptive token pruning for large vision language models. In *CVPR*, 2025.
- Xiang Yue et al. MMMU: A massive multi-discipline multimodal understanding and reasoning benchmark for expert agi. In *CVPR*, 2024.
- Shaolei Zhang et al. LLaVA-Mini: Efficient image and video large multimodal models with one vision token. In *ICLR*, 2025a.
- Yuan Zhang et al. SparseVLM: Visual token sparsification for efficient vision-language model inference. In *ICML*, 2025b.
- Lianmin Zheng et al. Judging LLM-as-a-judge with MT-Bench and chatbot arena. In *NeurIPS*, 2023.



Figure 3: **Patch selection at different K values** (CLIP ViT-L/14, MMBench sample). Left to right: original image, CLS attention importance heatmap, and top- K selected patches (bright) with unselected patches dimmed. At $K=32$ (5%), only the most salient regions are retained. At $K=128$ (22%), the main subject is well covered. At $K=256$ (44%), coverage is comprehensive. The top- K raw tokens capture fine-grained detail on important regions, while pool tokens (not shown) are intended to provide a coarse spatial summary of the entire image at aggressive token budgets.

A Additional Analysis

A.1 Qualitative Visualization of Patch Selection

A.2 LLM-Level Efficiency

Table 7: LLM-level efficiency (Qwen3-4B, A5000). Token reduction yields prefill speedup and memory savings that grow with batch size. The vision encoder cost is identical across all settings; per-token decode is memory-bandwidth-bound at this scale.

Metric	193 tok	321 tok	576 tok
LLM prefill, bs=1	38 ms	52 ms	87 ms
LLM prefill, bs=8	228 ms	352 ms	595 ms
Decode (ms/tok)	34.5	34.6	34.5
Peak memory, bs=8	8.9 GB	9.3 GB	10.3 GB

Token reduction yields a $2.3\times$ LLM prefill speedup at batch 1 and $2.6\times$ at batch 8 for 193 vs. 576 tokens. Per-token decode is memory-bandwidth-bound and insensitive to KV cache size at this model scale. End-to-end gains at batch 1 are smaller because the shared vision encoder dominates latency.

A.3 Budget-Aware Reinforcement Learning

Table 8: **Budget-aware GRPO** (Qwen3-4B + SigLIP-2-NaFlex, $K=512$, 577 tokens). After 200 RL steps with randomized K per prompt, GRPO narrows the gap with MLP on detail-heavy benchmarks. This is at equal token budget (577 vs. 576) and orthogonal to the token efficiency contribution. Bold indicates best per benchmark.

Benchmark	MLP (576 tok)	CoVLA (577 tok)	CoVLA+GRPO (577 tok)
SEED-Image	72.5	73.1	73.2
AI2D	74.1	72.2	73.3
ScienceQA-I	88.1	83.8	86.6
MMMU-val	45.1	43.8	47.8
MMStar	52.0	52.4	53.5
MathVista	57.6	51.8	61.0
TextVQA	64.5	64.0	64.2
ChartQA	29.5	26.4	28.4
DocVQA	13.0	13.3	13.5
MCQ Average	64.9	62.9	65.9

After 200 RL steps, MathVista improves by +9.2pp (51.8% $\rightarrow$ 61.0%) and the MCQ average rises to 65.9%, overtaking MLP’s 64.9%. This suggests part of the gap on detail-heavy benchmarks is behavioral rather than an inherent limitation of token selection. Note that this comparison is at equal token budget (577 vs. 576) and does not reflect the token efficiency contribution.

A.4 Importance Signal Distribution

Table 9: **CLS attention concentration** in CLIP ViT-L/14. The frozen encoder’s attention is highly non-uniform, with the top 5.6% of patches capturing over 60% of the total mass.

Top-K	Fraction of N	Attention Mass	Concentration
32	5.6%	60.2%	10.8 $\times$
128	22.2%	85.4%	3.9 $\times$
256	44.4%	95.1%	2.1 $\times$
576	100%	100%	1.0 $\times$

Table 9 quantifies how concentrated the CLS attention signal is. With only 32 patches (5.6% of 576), the model captures 60.2% of the total attention mass, a 10.8 $\times$ concentration over uniform. This empirically justifies the top- K selection strategy: a small subset of patches carries the majority of the information that the vision encoder deems important.

A.5 Negative Result: Learning Importance Fails

An important contribution of this work is documenting the failure of several attempts to learn importance signals on top of the frozen encoder. Over six months of development, we systematically attempted:

1. **Saliency head + STE**: A small MLP predicting per-patch importance, trained with the Straight-Through Estimator. Result: saliency entropy remained at 0.93 (near-uniform), failed to differentiate patches.
2. **CLIP text supervision**: Supervised the saliency head with CLIP text-to-patch similarities. Result: gradient magnitude dropped to 0.003 within 200 steps; the signal was too weak for meaningful learning.

3. **Delta residuals:** Added a learned residual channel where raw patches contribute their deviation from the pooled cell mean. Result: the model converged to a delta scale of 0.19 to 0.21 regardless of initialization, effectively ignoring the delta channel. Accuracy was identical with or without it (63.57%).
4. **MCQ rotation augmentation:** Trained with randomized answer option ordering to reduce position bias. Result: accuracy dropped from 64.3% to about 58%, as the augmentation made each training step roughly $4\times$ harder without additional data.

These failures point to a consistent practical conclusion: the frozen vision encoder already provides a strong importance signal, and our additional learned mechanisms did not improve on it in this training regime.

A.6 Component Ablation

Table 10: Component ablation on MMBench (Vicuna-7B + CLIP, LoRA $r=128$, single run). Raw tokens are the primary accuracy driver on this benchmark. Pool token resolution does not affect MMBench accuracy, though pool tokens may provide complementary spatial coverage on other tasks.

Configuration	Tokens	Std	Circ
$K=256, p=8$ (full)	321	74.0	64.3
$K=256, p=0$ (no pool)	257	74.0	64.3
$K=256, p=1$	258	74.0	64.3
$K=64, p=8$ (full)	129	71.7	61.5
$K=64, p=0$ (no pool)	65	71.7	61.5
$K=0, p=8$ (pool only)	65	42.2	23.5
MLP baseline	576	73.9	65.0

On MMBench, removing raw tokens ($K=0$) collapses performance to near-chance, confirming they are the primary accuracy driver. Removing pool tokens ($p=0$) does not affect MMBench accuracy; they may still matter for tasks requiring broader spatial coverage (e.g., document layout, spatial reasoning) that are not captured by this benchmark.

B Additional Benchmark Results

B.1 Per-Category Breakdown and Vicuna-7B Comparison

Table 11: **Per-category MMBench breakdown** (Qwen3-4B, circular %, single seed). CoVLA ($K=256$, 321 tokens) vs. MLP (576 tokens). On this seed CoVLA wins 11/20 categories; largest gains are image quality +15.1pp and social relation +4.6pp, while MLP leads on action recognition +3.7pp and image emotion +4.0pp. Over three seeds the aggregate MMBench circular score has MLP marginally ahead (Table 1); this breakdown is retained to show *where* the two connectors differ by category.

Category	CoVLA	MLP	Category	CoVLA	MLP
Identity Reason.	97.8	97.8	Physical Relation	75.0	70.8
Image Scene	96.2	97.1	OCR	71.8	74.4
Attr. Recognition	93.2	90.5	Nature Relation	64.6	62.5
Social Relation	93.0	88.4	Struct. Img-Text	64.1	61.5
Action Recog.	88.9	92.6	Image Quality	60.4	45.3
Function Reason.	89.9	86.1	Future Prediction	57.5	57.5
Image Topic	88.9	91.7	Physical Property	56.0	61.3
Celebrity Recog.	85.9	81.8	Attr. Comparison	52.3	56.8
Image Emotion	84.0	88.0	Object Local.	53.1	54.3
Image Style	83.0	86.8	Spatial Relation	28.9	22.2
Overall				75.4	74.6

CoVLA wins 11 of 20 categories, with the largest gains on image quality (+15.1pp) and social relation (+4.6pp). Both methods struggle on spatial relationship (<30%).

Table 12: **Vicuna-7B + CLIP + LLaVA-665K** controlled comparison (both LoRA $r=128$). MLP leads on 4/5 benchmarks; CoVLA wins ScienceQA (+1.6pp) at 44% fewer tokens.

Benchmark	MLP (576 tok)	CoVLA (321 tok)
SEED-Image	65.4	63.0
ScienceQA-I	67.5	69.1
GQA	60.4	58.9
POPE	89.7	87.1
TextVQA	48.2	44.6

At the Vicuna-7B scale, MLP holds a modest advantage on most benchmarks, consistent with the main-text finding that the efficiency-accuracy trade-off is backbone-sensitive. CoVLA remains competitive while using 44% fewer vision tokens.