LLaVA-Mini with Dynamic Number of Compression Tokens

Technical Report by AI Research Team

July 9, 2025

Abstract

Large vision-language models (VLMs) such as LLaVA achieve strong performance on multimodal benchmarks, but struggle to efficiently allocate visual attention across diverse image types. This is particularly apparent in cases involving complex documents or diagrams, where fine-grained details are critical. Current lightweight variants like LLaVA-Mini aggressively compress image information into a single visual token, degrading quality on such inputs.

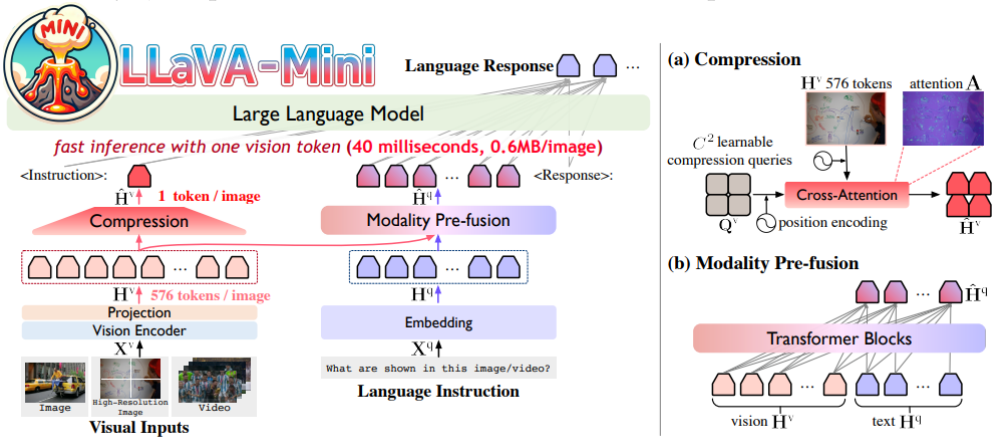
We propose a dynamic token compression module that adapts the number of visual tokens based on image complexity. Our method introduces a *DynamicResampler* layer with a learnable masking mechanism, producing either 1 or up to 256 tokens depending on the input. To support benchmarking, we adapted the lmms-eval library to run LLaVA-Mini on multimodal datasets, including DocVQA. Results demonstrate that performance on complex image tasks significantly improves with our approach. Our contribution lays the groundwork for adaptive compute allocation in vision-language systems.

1 Motivation

Vision–language models (VLMs) such as LLaVA rely on a powerful image encoder—typically CLIP's Vision Transformer (ViT)—to convert each input image into a set of patch embeddings. ViT splits the image into a 16×16 grid of patches, yielding 256 separate token embeddings per image. These visual tokens are then projected via a multilayer perceptron into the same

embedding space as the accompanying text, allowing the language model to attend jointly to visual and textual information.

While this dense representation preserves rich image details, passing 256 tokens through the language model imposes a substantial computational burden. To address this, the authors of LLaVA-Mini introduce a learned compression network that aggregates all 256 visual embeddings into a single "summary" token (see pi. Remarkably, this one-token variant matches full-scale LLaVA's performance on benchmarks such as VQAv2, GQA, VizWiz and ScienceQA, despite its drastic reduction in visual input size.



However, collapsing an image down to a single token only works well when the image content is simple—for example, scenes easily described in a few words ("a cat", "a mountain"). It fails on more complex visual inputs that require spatial reasoning or detailed layout understanding, such as document layouts, headings and multi-column text, diagrams annotated with labels or arrows, multi-object scenes where relationships between elements matter.

On these tasks, full LLaVA's 256-token representation substantially outperforms the one-token LLaVA-Mini (see Table below for a side-by-side comparison).

Model	DocVQA	ChartQA
LLaVA-Mini	0.31	0.18
LLaVA-NeXT	0.74	0.54

Table 1: Performance comparison on document-level benchmarks.

This gap highlights the need for an adaptive compression strategy: rather

than always reducing every image to a single token, the model should decide—based on image complexity—whether to use the lightweight, one-token compressor or retain a richer multi-token encoding. Such a dynamic approach promises to combine the efficiency of extreme compression on simple inputs with the expressivity of a fuller representation on complex ones.

2 Methodology

Our approach consists of two components: a binary image complexity classifier and the DynamicResampler module.

2.1 Classifier

We train a small binary neural network to predict whether an input image is *simple* or *document-like*. The input to this classifier is the global image embedding from CLIP, namely the `[CLS]` token output of the Vision Transformer. This token is designed to aggregate information from the entire image. We feed the CLIP `[CLS]` vector (of dimension 1024 for CLIP ViT-L/14) into a multilayer perceptron (MLP) with one hidden layer and a sigmoid output to produce the probability of the image being document-like. The classifier is trained on labeled examples of simple versus document images.

2.2 DynamicResampler

The DynamicResampler module uses the classifier's output to select how many visual tokens to use. We maintain two sets of learned query embeddings: Q_1 containing a single query, and Q_{256} containing 256 queries. Each query has the same feature dimension as the CLIP token embeddings. We add a 2D sinusoidal positional encoding to all queries and all CLIP patch tokens to preserve spatial information.

At inference time, if the classifier indicates "simple", we use Q_1 (1 query) to compress the image; if "document-like", we use Q_{256} (256 queries). We perform cross-attention where the chosen queries attend to all CLIP patch tokens (as keys/values). If Q_1 is used, this produces one output token; if Q_{256}, it produces 256 output tokens. These compressed tokens are then fed into the language model along with the input text.

3 Evaluation

3.1 Implementation

We implemented DynamicResampler both in LLaVA-Mini and in LLaVA-NeXT. The training procedure proceeded in two phases:

- **Phase 1: LLaVA-Mini pretraining** We pretrained the LLaVA-Mini backbone separately on three different models to assess the impact of the underlying language model size:

Backbone	Parameter Count	Description
LLaMA-7B	7B	Baseline LLaVA-Mini pretraining
Qwen-3 0.6B	0.6B	Lightweight variant
Qwen-2.5 0.5B	0.5B	Ultra-light variant

- **Phase 2: LLaVA-NeXT adaptation** After observing stability issues with LLaVA-Mini, we moved to the LLaVA-NeXT codebase using Qwen-2.5 0.5B as the backbone. All components except the CLIP ViT-L/14 vision encoder (which remains frozen) — namely the classifier, query tokens, and attention layers — are trained end-to-end during instruction tuning.

- **Phase 3: Evaluation on document benchmarks** We evaluated both versions of the model on the DocVQA dataset to assess improvements in document understanding:

  The dynamic token compression approach in LLaVA-NeXT significantly improves document-level performance.

3.2 Datasets

- *Classifier training:* we used the following datasets:
  - Positive class (document-style images): `docvqa`, `textvqa`, `chartqa`
  - Negative class (general images): `vqav2`, `st_vqa`, `localized_narratives`
- *Pretraining:* `lmms-lab/LLaVA-ReCap-558K`

4 Results

Model	DocVQA	ChartQA	MMMU (val)	POPE
LLaVA Qwen 2.5 0.5B (1 token)	0.1088	0.1192	0.3244	0.7905
LLaVA Qwen 2.5 0.5B (64 tokens)	0.0922	0.1112	0.2722	0.6838

Table 2: Impact of the number of visual tokens on performance. Using a single vision token improves performance on simpler datasets (e.g., POPE), while 64-token variants underperform due to lack of compression.

5 Future Work

Possible directions for future work include:

- Exploring a larger set of compression token counts (e.g. 1, 4, 16, 64, 256) and learning when to use each in a multi-way selection.
- Incorporating richer features for complexity classification, such as preliminary OCR output or intermediate vision-layer embeddings.
- Jointly training the classifier and language model in an end-to-end fashion to directly optimize the resampling strategy for downstream tasks.
- Extending the DynamicResampler to video understanding settings by dynamically allocating visual tokens across frames for tasks such as video question answering.

6 Conclusion

In this report, we presented *DynamicResampler*, a method for dynamically adjusting the number of visual compression tokens in LLaVA models. By using a learned binary classifier on CLIP features to distinguish simple images from document-like images, our approach allocates either one or 256 tokens as needed. This adaptive compression aims to improve performance on complex images while remaining efficient on simpler inputs. While final results are pending, we expect DynamicResampler to yield better performance on document-centric tasks. More broadly, adaptively adjusting visual token counts offers a promising approach for future vision-language models.

References

[1] Shaolei Zhang, Qingkai Fang, Zhe Yang, and Yang Feng. LLaVA-Mini: Efficient Image and Video Large Multimodal Models with One Vision Token. *arXiv preprint arXiv:2501.03895*, 2025.

[2] Kaichen Zhang, Bo Li, Peiyuan Zhang, Fanyi Pu, Joshua Adrian Cahyono, Kairui Hu, Shuai Liu, Yuanhan Zhang, Jingkang Yang, Chunyuan Li, and Ziwei Liu. LMMs-Eval: Reality Check on the Evaluation of Large Multimodal Models. *arXiv preprint arXiv:2407.12772*, 2025.

[3] Wenbo Hu, Zi-Yi Dou, Liunian Harold Li, Amita Kamath, Nanyun Peng, and Kai-Wei Chang. Matryoshka Query Transformer for Large Vision-Language Models. *arXiv preprint arXiv:2405.19315*, 2024.