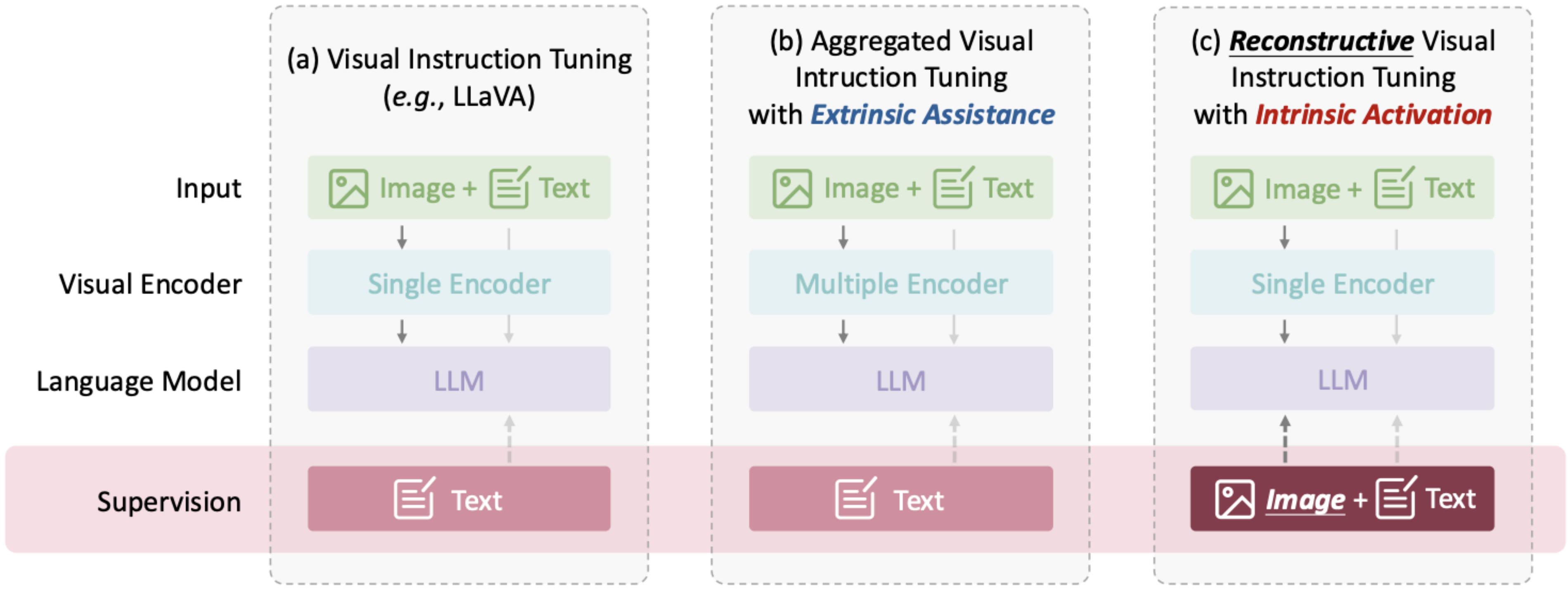
Reconstructive Visual Instruction Tuning

Haochen Wang[1,2], Anlin Zheng[2], Yucheng Zhao[4], Tiancai Wang[4] , Ge Zheng[5]
Xiangyu Zhang[4,5], and Zhaoxiang Zhang[1,2]
[1]CASIA, [2]UCAS, [3]HKU, [4]MEGVII Technology, [5]StepFun

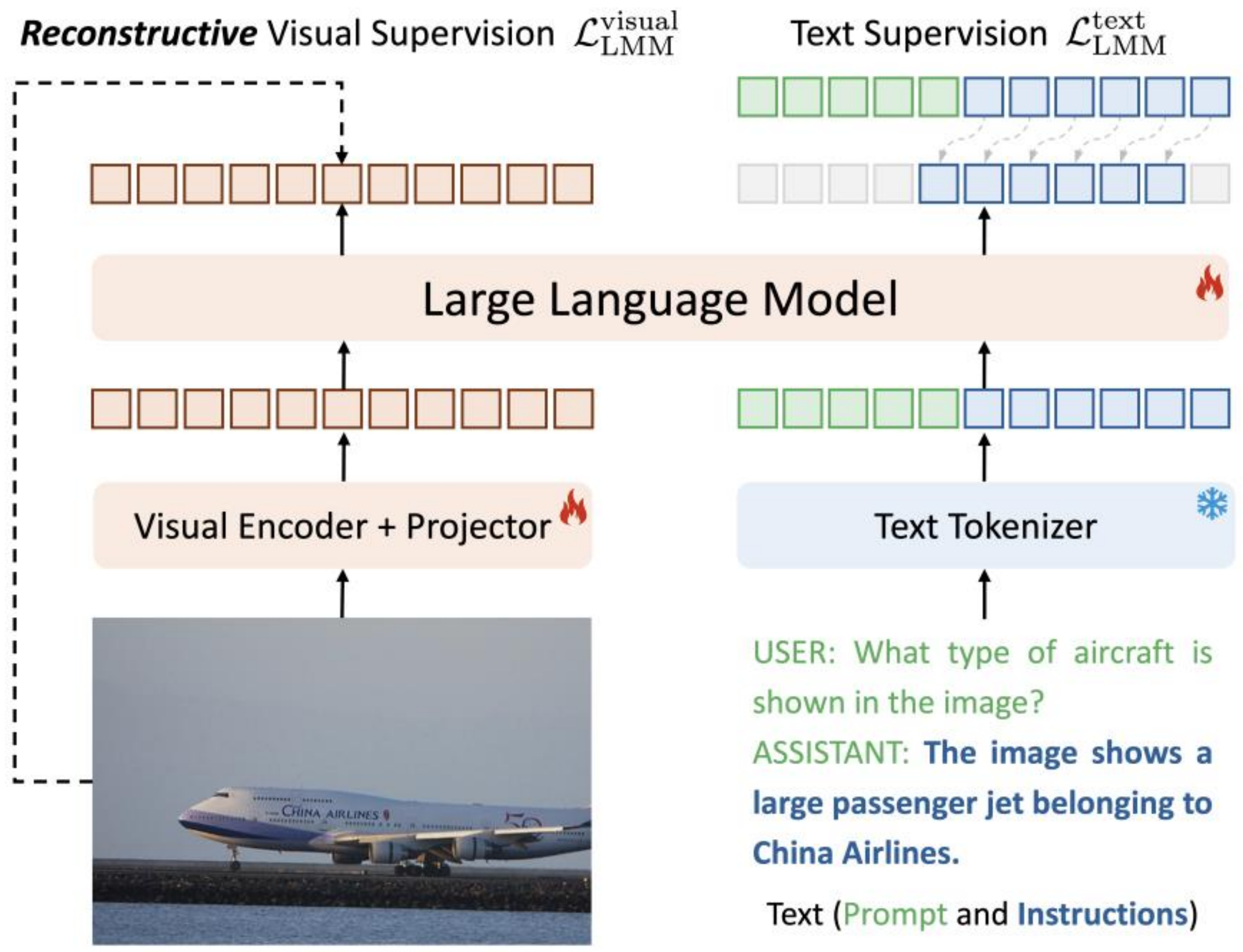
https://haochen-wang409.github.io/ross/

Motivation


(a) Visual Instruction Tuning (*e.g.*, LLaVA)
(b) Aggregated Visual Intruction Tuning with *Extrinsic Assistance*
(c) *Reconstructive* Visual Instruction Tuning with *Intrinsic Activation*
Input
Visual Encoder
Language Model
Supervision
Image + Text
Single Encoder
Multiple Encoder
LLM
Text
Image + Text

- Typical visual instruction tuning approaches, e.g., LLaVA, follow a **LLM-centric** design that solely leverage text supervision.
- Aggregated visual instruction tuning alternatives, e.g., Cambrian-1 and EAGLE, leverage extrinsic assistance via **combining several visual experts**, requiring a careful selection of visual experts.
- Our **Ross** designs extra **vision-centric reconstructive supervision** as intrinsic activation. In this way, LMMs are required to preserve every detail of input images, thereby enhancing multimodal comprehension capabilities and reducing hallucinations.
- *With a single SigLIP as the visual encoder, **Ross**-7B achieves 57.3 on HallusionBench and 54.7 on MMVP.*

Method

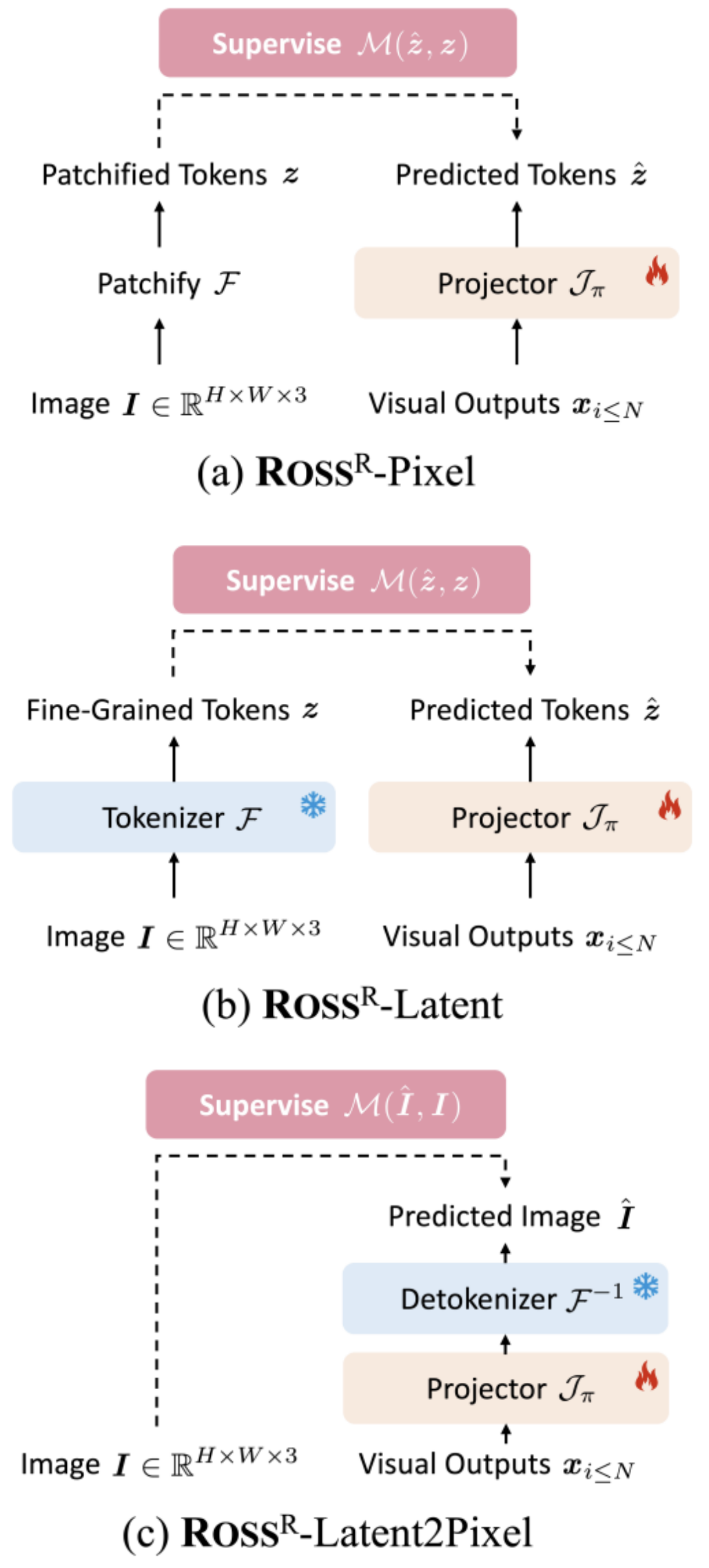

Reconstructive Visual Supervision $\mathcal{L}_{\text{LMM}}^{\text{visual}}$
Text Supervision $\mathcal{L}_{\text{LMM}}^{\text{text}}$
Large Language Model
Visual Encoder + Projector
Text Tokenizer
CHINA AIRLINES

USER: What type of aircraft is shown in the image?
ASSISTANT: **The image shows a large passenger jet belonging to China Airlines.**

Text (Prompt and **Instructions**)

High-level idea:

Supervising visual outputs using original images.

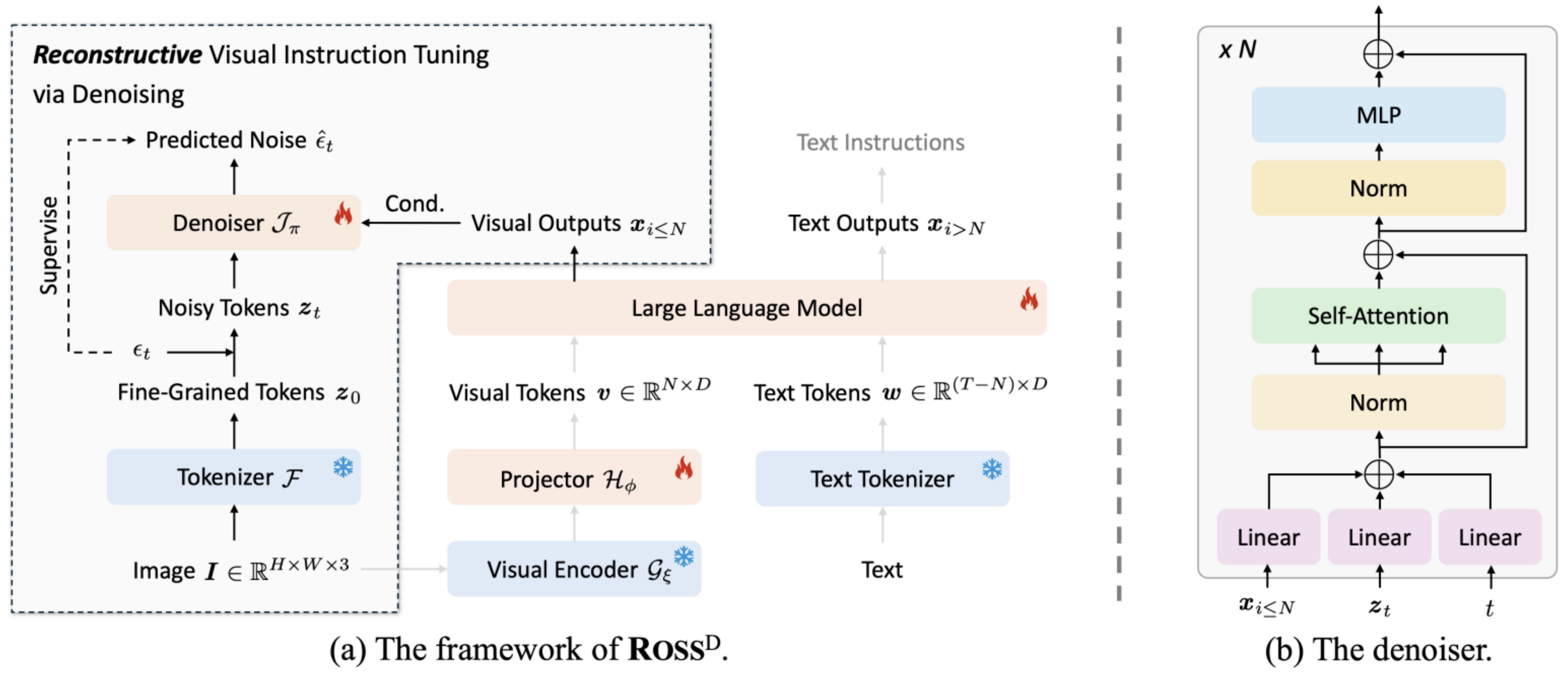

Supervise $\mathcal{M}(\hat{z}, z)$
Patchified Tokens z
Predicted Tokens $\hat{z}$
Patchify $\mathcal{F}$
Projector $\mathcal{J}_\pi$
Image $I \in \mathbb{R}^{H\times W\times 3}$
Visual Outputs $x_{i\le N}$
Supervise $\mathcal{M}(\hat{z}, z)$
Fine-Grained Tokens z
Predicted Tokens $\hat{z}$
Tokenizer $\mathcal{F}$
Projector $\mathcal{J}_\pi$
Image $I \in \mathbb{R}^{H\times W\times 3}$
Visual Outputs $x_{i\le N}$
Supervise $\mathcal{M}(\hat{I}, I)$
Predicted Image $\hat{I}$
Detokenizer $\mathcal{F}^{-1}$
Projector $\mathcal{J}_\pi$
Image $I \in \mathbb{R}^{H\times W\times 3}$
Visual Outputs $x_{i\le N}$

(a) **Ross**[R]-Pixel

(b) **Ross**[R]-Latent

(c) **Ross**[R]-Latent2Pixel

Vanilla regression performs bad.


Reconstructive Visual Instruction Tuning via Denoising
Predicted Noise $\hat{\epsilon}_t$
Supervise
Denoiser $\mathcal{J}_\pi$
Cond.
Visual Outputs $x_{i\le N}$
Noisy Tokens z_t
ϵ_t
Fine-Grained Tokens z_0
Tokenizer $\mathcal{F}$
Image $I \in \mathbb{R}^{H\times W\times 3}$
Text Instructions
Text Outputs $x_{i>N}$
Large Language Model
Visual Tokens $v \in \mathbb{R}^{N\times D}$
Text Tokens $w \in \mathbb{R}^{(T-N)\times D}$
Projector $\mathcal{H}_\phi$
Text Tokenizer
Visual Encoder $\mathcal{G}_\xi$
Text
× N
MLP
Norm
Self-Attention
Norm
Linear
Linear
Linear
$x_{i\le N}$
z_t
t

(a) The framework of **Ross**[D].

(b) The denoiser.

Final pipeline:

Using a small denoiser to project visual outputs back to pixel space.

Experiments

Model	POPE	Hallu.	MMB$^{\text{EN}}$	MMB$^{\text{CN}}$	SEED$^{\text{I}}$	MMMU	MMVP	GQA	AI2D
GPT-4V-1106 (OpenAI, 2023a)	75.4	65.8‡	75.8	75.1‡	71.6	53.8	50.0	36.8	78.2
Gemini-1.5 Pro (Team et al., 2023)	–	–	73.6	–	70.7	47.9	–	–	–
MM-1-8B (McKinzie et al., 2024)	86.6	–	72.3	–	69.9	37.0	–	72.6	–
Mini-Gemini-8B (Li et al., 2024f)	–	–	72.7	–	73.2	37.3	18.7	64.5	73.5
DeepSeek-VL-7B (Lu et al., 2024)	85.8‡	44.1‡	73.2	72.8	70.4	36.6	–	–	64.9‡
Cambrian-1-8B (Tong et al., 2024a)	87.4‡	48.7‡	75.9	68.9‡	**74.7**	42.7	51.3	64.6	73.0
Ross-7B	**88.3**	**57.1**	**79.1**	**77.1**	73.6	**46.6**	**56.7**	**65.5**	**79.3**
Base LLM: Vicuna-7B-v1.5									
LLaVA-v1.5-7B‡ (Liu et al., 2024a)	86.2	47.5	65.5	58.5	66.0	34.4	20.0	62.0	55.4
LLaVA-v1.6-7B‡ (Liu et al., 2024b)	86.5	35.8	67.4	60.1	**70.2**	35.8	37.3	**64.2**	67.1
Ross-7B$_{\text{vicuna}}$	**88.2**	**55.2**	**67.7**	**61.3**	67.6	**36.9**	**39.3**	63.7	**69.3**
Base LLM: Vicuna-13B-v1.5									
LLaVA-v1.5-13B‡ (Liu et al., 2024a)	82.5	44.9	68.8	63.6	68.2	36.6	32.0	63.3	60.8
LLaVA-v1.6-13B‡ (Liu et al., 2024b)	86.2	36.7	70.0	64.1	71.9	36.2	35.3	**65.4**	72.4
Mini-Gemini-13B (Li et al., 2024f)	–	–	68.6	–	73.2	37.3	19.3	63.7	70.1
Cambrian-1-13B (Tong et al., 2024a)	85.7‡	54.0‡	**75.7**	65.9‡	**74.4**	40.0	41.3	64.3	73.6
Ross-13B$_{\text{vicuna}}$	**88.7**	**56.4**	73.6	**67.4**	71.1	**41.3**	**44.7**	65.2	**73.8**

Comparison with state-of-the-art alternatives.

Language Model	$\mathcal{L}_{\text{LMM}}^{\text{visual}}$	POPE	Hallu.	MMVP	ChartQA	OCRBench	MMB$^{\text{EN}}$
Visual Encoder: CLIP-ViT-L/14@336							
Vicuna-7B-v1.5	–	86.3	52.5	28.0	32.9	339	67.0
	✓	**87.2** ↑ 0.9	**55.8** ↑ 3.3	**36.0** ↑ 8.0	**39.8** ↑ 6.9	**350** ↑ 11	**67.6** ↑ 0.6
Qwen2-7B-Instruct	–	87.9	55.0	29.3	34.0	363	73.8
	✓	**88.4** ↑ 0.5	**56.7** ↑ 1.7	**42.0** ↑ 12.7	**37.1** ↑ 3.1	**381** ↑ 18	**75.2** ↑ 1.4
Visual Encoder: SigLIP-ViT-SO400M/14@384							
Vicuna-7B-v1.5	–	86.0	50.4	27.3	36.2	354	64.5
	✓	**86.8** ↑ 0.8	**53.2** ↑ 2.8	**38.0** ↑ 10.7	**41.6** ↑ 5.4	**365** ↑ 11	**65.7** ↑ 1.2
Qwen2-7B-Instruct	–	88.5	57.3	40.7	44.4	432	76.3
	✓	**88.7** ↑ 0.2	**58.2** ↑ 0.9	**49.3** ↑ 8.6	**46.3** ↑ 1.9	**448** ↑ 16	**76.9** ↑ 0.6

Effectiveness of Ross with different visual encoders and LLMs.

Reconstruction results after finetuning the denoiser on ImageNet-1K.