IVTP: Instruction-guided Visual Token Pruning for Large Vision-Language Models

Performance Speed Cost Prune

Quick View References:

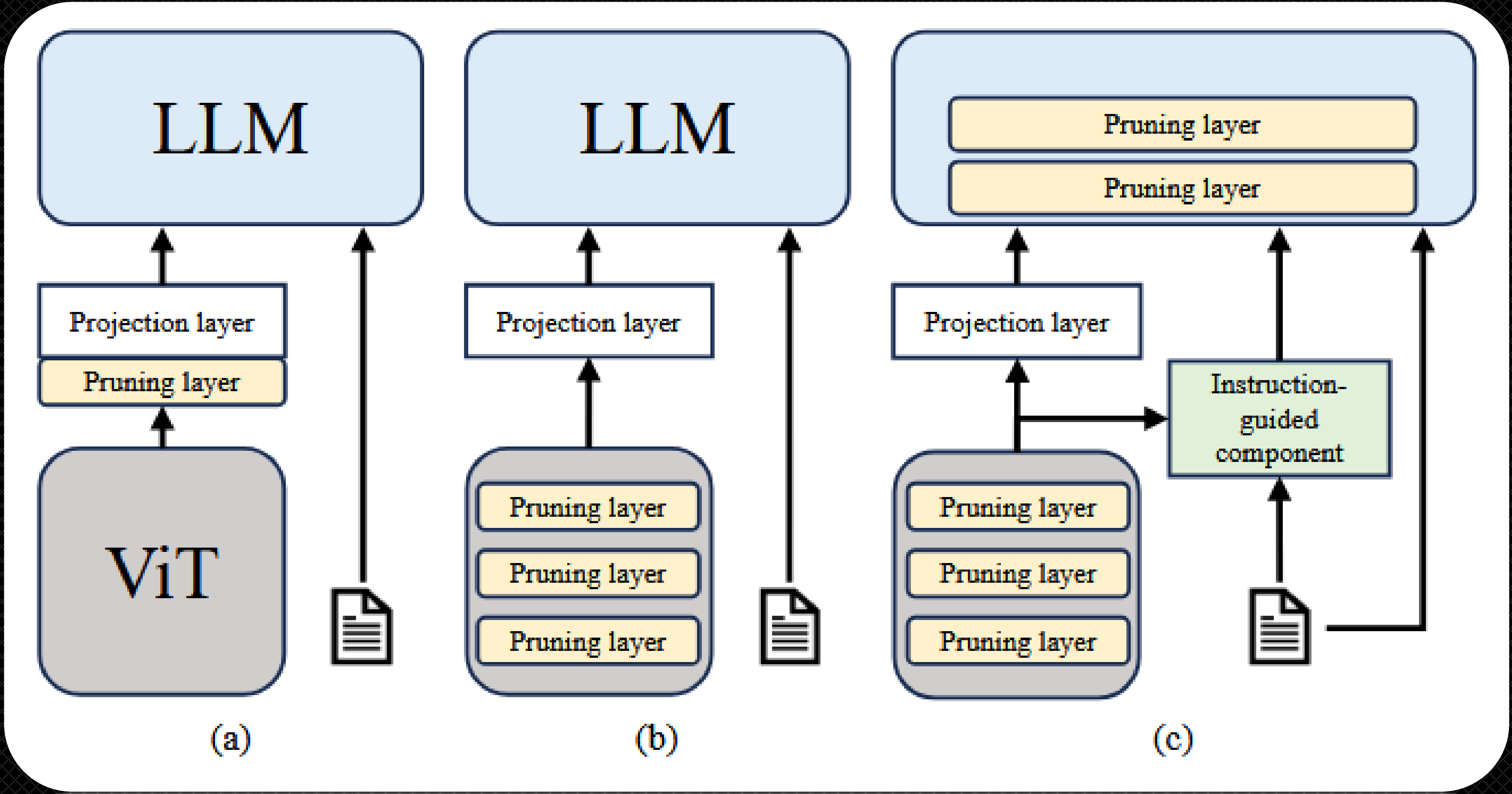
Fig. 1 Fig. 2 Fig. 3 Fig. 4 Fig. 5

Eq. 1 Eq. 2 Eq. 3 Eq. 4 Eq. 5 Eq. 6 Eq. 7 Eq. 8

Table 1 Table 2 Table 3 Table 4 Table 5 Table 6 Table 7 Table 8 Table 9

Figures

Fig. 1 – Different Token Pruning Ideas



IVTP

Fig. 2 – Instruction-guided Visual Token Pruning

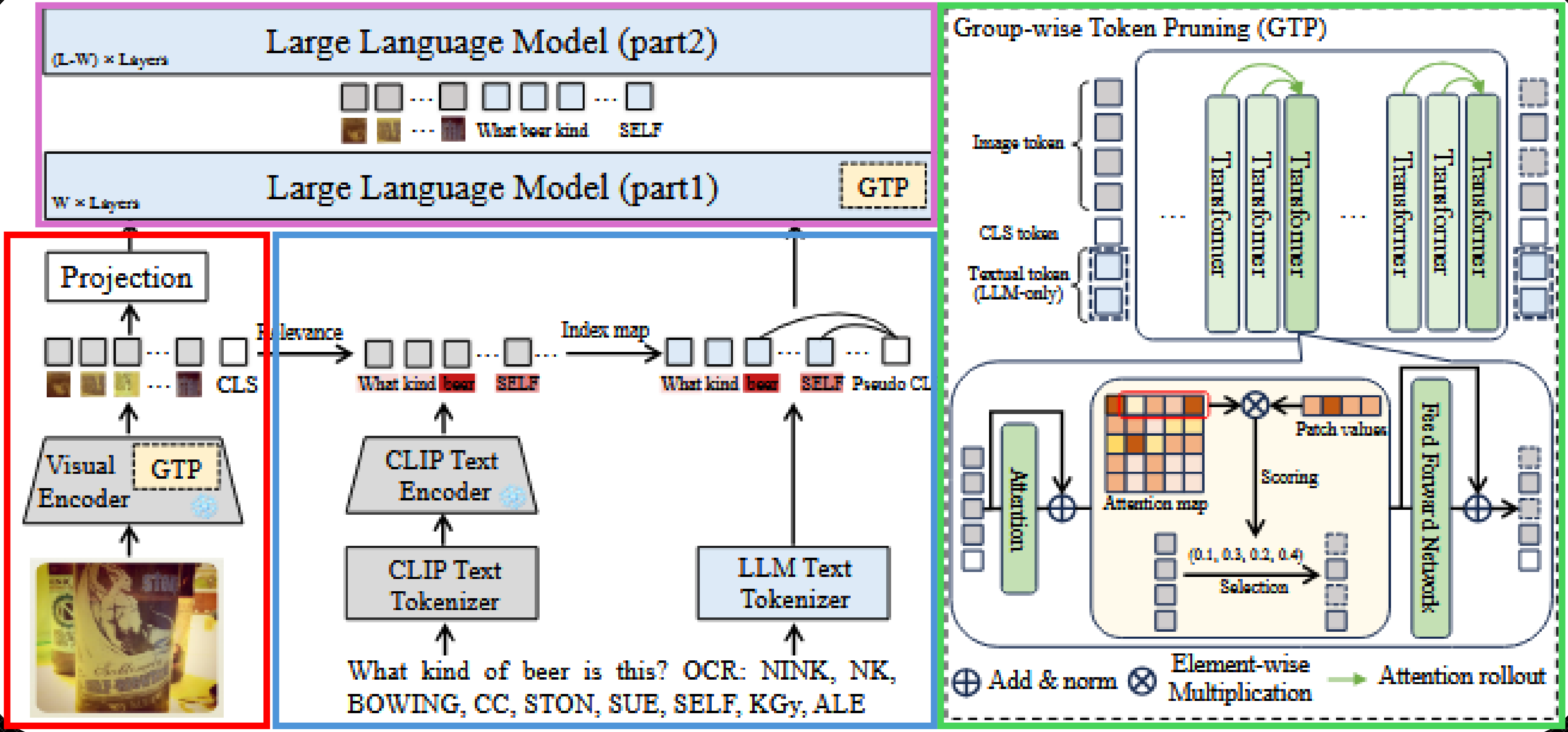
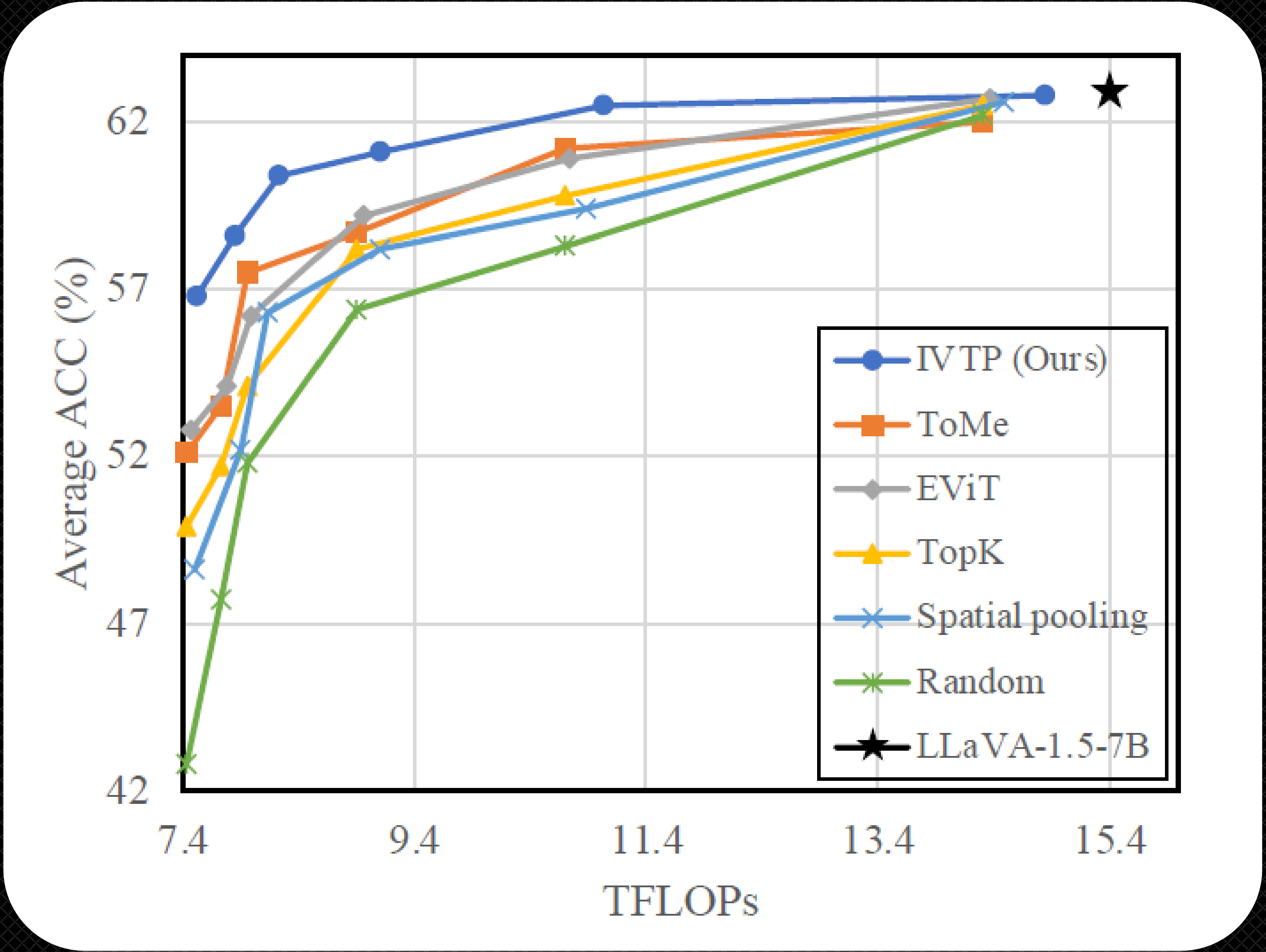
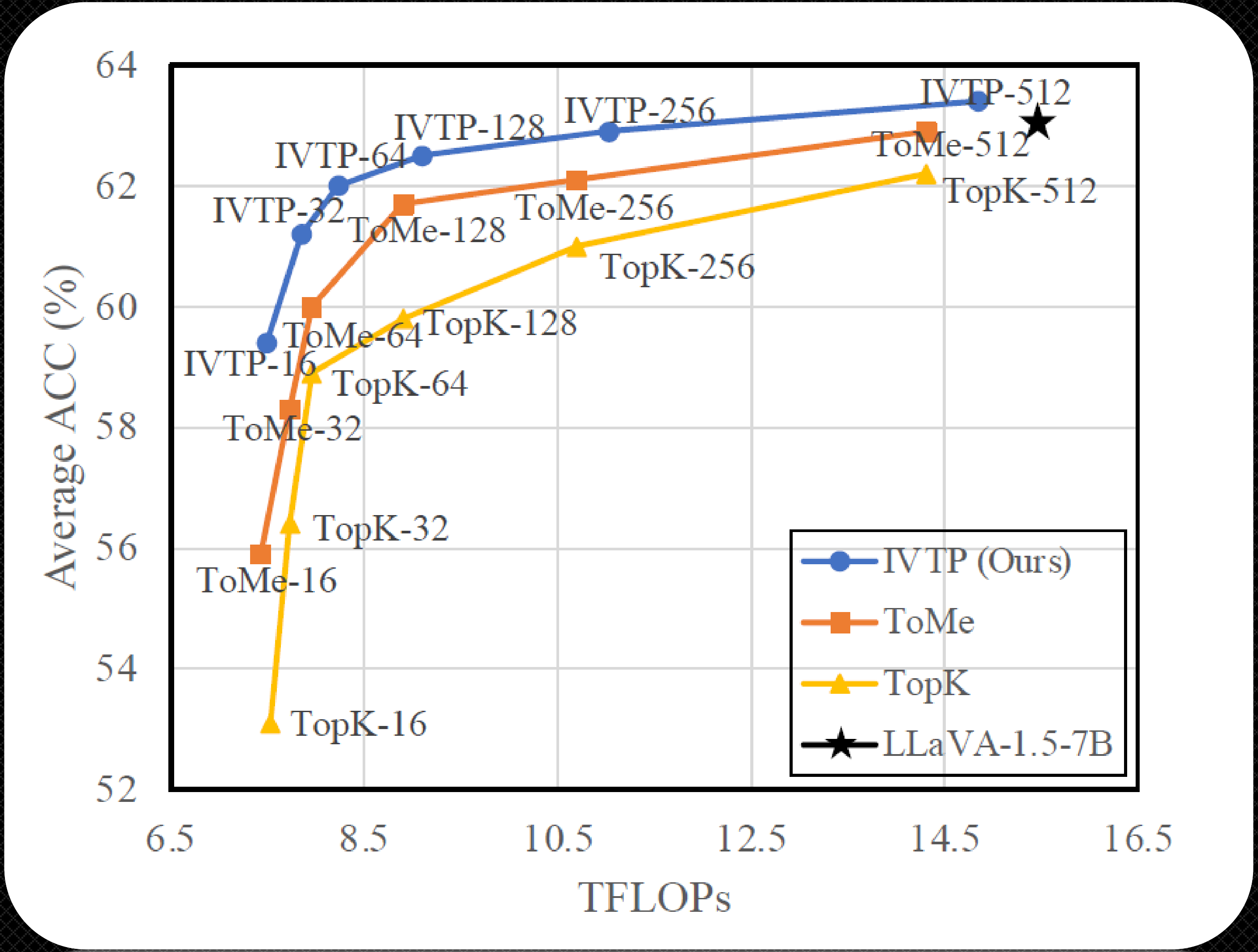


Fig. 3 – Different Token Pruning Methods



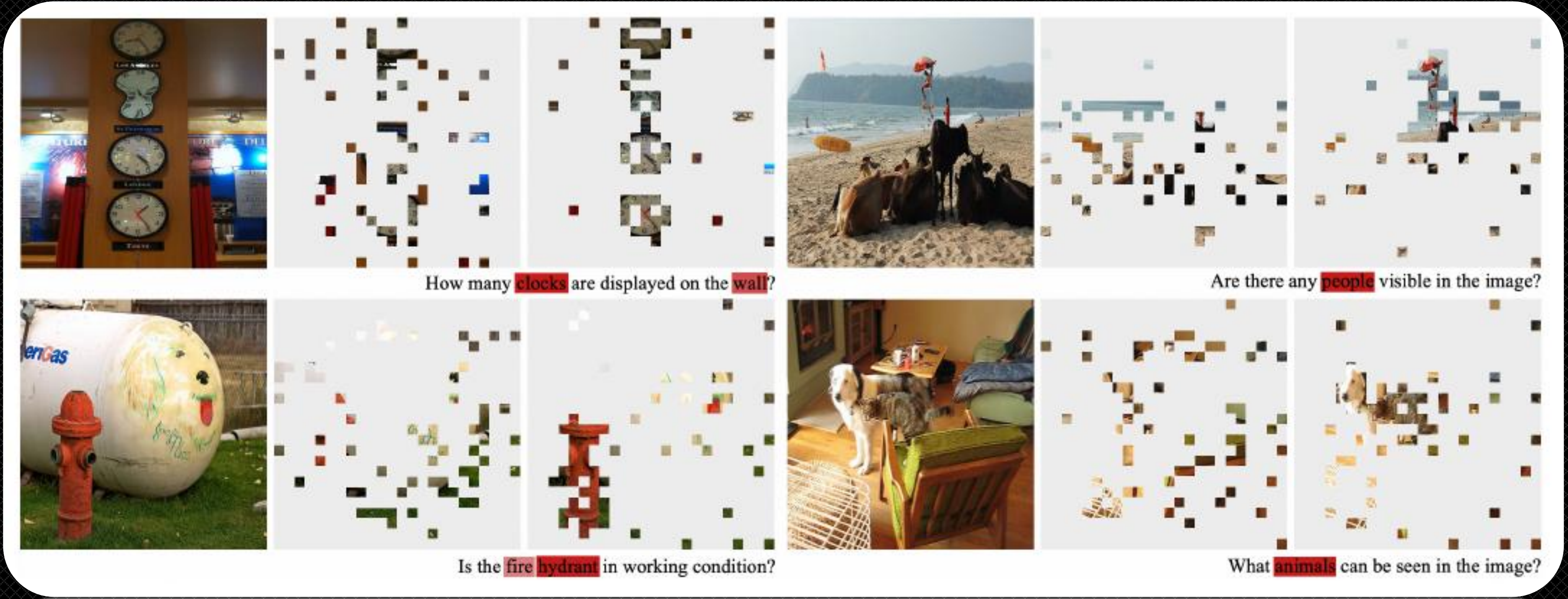
IVTP

Fig. 4 – Performance based on Tokens Pruned



IVTP

Fig. 5 – Visualization



IVTP

Equations

Eq. 1 – Response Generation

$$p(X) = p(x_1^L|x^V) \times p(x_2^L|x^V, x_1^L) \times p(x_3^L|x^V, x_1^L, x_2^L) \dots$$

Total # of language tokens in output

Product sign

Probability of the full sequence (X)

$$p(\mathbf{X}) = \prod_{i=1}^{M} p(x_i^L | x_1^V, \dots, x_F^V, x_1^L, \dots, x_{i-1}^L; \Theta),$$

The i'th language token (Word it's predicting.)

All visual tokens from image.

All previous language tokens (before current one being predicted.)

The parameters / learned weights of the model

Eq. 2 – Attention Weights

T = Transpose

Every Query to Key Comparison

$$QK^T \quad \rightarrow \quad [tokens \times d] \times [d \times tokens] = [tokens \times tokens]$$

Convert to probabilities sum to 1

Queries

Keys

Dot product

(shows how much one token attends to another)

Attention Weight Map

$$A = \text{Softmax}\left(\frac{QK^T}{\sqrt{d}} + A_{\text{mask}}\right)$$

Keep Values Stable (Normalize)

Blocks Illegal Connections

ViT = None

LLM = Can’t look at future words

Eq. 3 - Attention Rollout

Combines Attention Maps of several layers.
Outputs → Rolled Out Attention Map

Layer 1: Token A looks at Token B (A→B) INDIRECT
Layer 2: Token B looks at CLS (B→CLS) DIRECT

Rollout shows direct / indirect relationship.

Attention Map from current layer l

Identity Matrix
(Token also pays attention to itself.)
Prevents info from vanishing when combining layers.

Rolled-Out Attention Map

$$\tilde{A}^{l} = \begin{cases} A^{l_1}, & \text{if } l = l_1, \\ (A^{l} + I)\,\tilde{A}^{l-1}, & \text{if } l_1 < l \leq l_2 \end{cases}$$

Multiply Attention from Current Layer by Cumulative Attention from All Previous Layers

Start & End Layers

IVTP

Eq. 4 - Importance Score

Feature strength

Rolled-Out Attention Weight from:
CLS Token to i-th Token

How Much CLS attends to that patch.

Magnitude of Patch Vector

V_*i* → Feature Vector for Patch *i*

How strong or informative the patch is.

Importance Score for Token *i*

$$S_i = \frac{\tilde{A}_{\text{cls},i}\,\|V_i\|}{\sum_j \tilde{A}_{\text{cls},j}\,\|V_j\|}$$

Normalization Term (Ensures all Importance Scores sum to 1)

Eq. 5 – CLIP’s Text Encoder

CLIP’s Text Encoder

Output Matrix
(Each token embedding has d dimensions)

$$\{\bar{x}_1^L, \cdots, \bar{x}_{S'}^L\} = \mathcal{T}(T) \in \mathbb{R}^{S' \times d},$$

Vector for i-th text token (after encoded)

Num of text tokens

Eq. 6 – Cosine Similarity

CLS token from visual encoder (represents image)

The i-th text token from CLIPS text encoder

Total num of text tokens

Cosine Similarity Score

$$c_i = \frac{x_{\mathrm{cls}}^{\mathrm{V}} \bar{x}_i^{L}}{\left\| x_{\mathrm{cls}} \right\| \left\| \bar{x}_i^{L} \right\|}, \quad \text{for } i \in 1, \cdots, S'.$$

Magnitude (length) Image CLS Token

Magnitude (length) i-th Text token

Eq. 7 – Align CLIP Tokens to LLM Tokens

Sum
Terms k=j
to k=j+K

Takes all cosine similarity scores from CLIP tokens (Sums)

Averaging term divides sum to get mean

The i-th token in the LLM's tokenizer

Smoothed Similarity Score (for i-th LLM token)

$$C_i = \frac{1}{K+1} \sum_{k=j}^{j+K} c_k, \text{ with } T_{\text{LLM}}(i) \subseteq \{T_{\text{CLIP}}(j), \cdots, T_{\text{CLIP}}(j+K)\},$$

Window size
(num of neighboring tokens to avg)

Cosine similarity scores
Formula 6

Range of CLIP tokens that correspond to same word or phrase

Eq. 8 – Pseudo CLS Token Construction

Visual CLS Token (From Image Encoder)

Num of tokens in LLM input (length of the text)

threshold (important or relevant decider)

psuedo CLS token

$$x_{\text{cls}}^{T} = \begin{cases} x_{\text{cls}}^{V}, & \text{if } \sum I_{\{C_i \geq \tau\}} = 0, \\ \frac{1}{\sum_{i=1}^{S} I_{\{C_i \geq \tau\}}} \sum_{i=1}^{S} x_i^{L} I_{\{C_i \geq \tau\}}, & \text{otherwise.} \end{cases}$$

If no text tokens are relevant

Averages the embeddings of only the most relevant words (above thredshold)

Visual relevance score for token i
Formula 7

i-th token embedding from LLM's text encoder

Tables

Table 1 – Pruning Method Benchmarks (Vicuna-7B)

Method	VQA^{v2} [11]	GQA [14]	VisWiz [13]	SQA^{I} [28]	VQA^{T} [33]	POPE [23]	MME [9]	MMB [27]	MMB^{CN} [27]	SEED [19]	$LLaVA^{W}$ [26]	MM-Vet [40]	Avg. ↑	TFLOPs ↓
LLaVA-1.5-7B [25]	78.5	62.0	50.0	66.8	58.2	85.9	75.5	64.3	58.3	58.6	63.4	30.5	62.7	15.4
LLaVA-1.5-7B* [25]	79.1	62.7	49.0	67.8	58.6	86.3	72.8	66.2	59.3	58.5	63.7	31.7	63.0	15.4
Random sampling	69.0	57.1	37.9	67.2	48.5	82.5	65.6	55.4	48.0	51.0	55.8	23.6	55.1 (-7.9)	8.0 (-48.1%)
TopK	72.4	58.1	47.0	66.9	52.5	83.8	67.1	63.3	55.2	54.5	59.2	26.5	58.9 (-4.1)	8.0 (-48.1%)
Spatial pooling	73.9	59.6	46.5	67.7	52.5	82.3	68.5	63.3	56.6	54.9	59.7	28.3	59.5 (-3.5)	8.1(-47.4%)
EViT [24]	74.1	59.4	47.0	67.7	54.7	82.8	69.2	63.5	57.8	55.4	60.0	27.3	59.9 (-3.1)	8.0 (-48.1%)
ToMe [4]	75.1	60.0	47.1	67.5	55.3	82.4	70.4	63.9	56.5	55.2	60.5	26.6	60.0 (-3.0)	8.0 (-48.1%)
Honeybee [5]	74.8	59.0	47.2	67.8	50.9	84.0	68.7	61.6	57.8	55.2	59.4	27.1	59.5 (-3.5)	8.1 (-47.4%)
LLaMA-VID [22]	74.3	59.2	46.8	67.9	51.4	83.1	69.7	63.5	57.0	55.4	58.9	29.7	59.7 (-3.3)	8.2 (-46.8%)
Qwen-VL [3]	74.9	58.9	47.3	68.1	54.4	83.4	69.4	63.2	57.4	55.0	59.2	27.2	59.9 (-3.1)	8.1 (-47.4%)
IVTP (Ours)	77.8	60.4	47.9	67.8	58.2	85.7	72.6	66.1	57.4	56.4	62.8	30.5	62.0 (-1.0)	8.2 (-46.8%)

IVTP

Table 2 – Pruning Method Benchmarks (Vicuna-13B)

Method	VQA^{v2} [11]	GQA [14]	VisWiz [13]	SQA^{I} [28]	VQA^{T} [33]	POPE [23]	MME [9]	MMB [27]	MMB^{CN} [27]	SEED [19]	$LLaVA^{W}$ [26]	MM-Vet [40]	Avg. ↑	TFLOPs ↓
LLaVA-1.5-13B [25]	80.0	63.3	53.6	71.6	61.3	85.9	76.6	67.7	63.6	61.6	70.7	35.4	65.9	29.4
LLaVA-1.5-13B* [25]	80.0	63.4	54.5	70.4	60.0	86.4	78.4	68.3	63.1	60.8	69.4	36.8	66.0	29.4
Random sampling	72.3	56.7	46.6	68.0	51.5	83.3	64.9	58.0	54.8	53.0	58.8	24.6	57.7 (-8.3)	15.4 (47.5%)
TopK	74.7	58.5	50.8	69.3	54.2	85.4	68.0	64.5	59.6	54.5	62.8	26.6	60.7 (-5.3)	15.4 (47.5%)
Spatial pooling	75.1	59.7	51.1	69.9	55.0	84.8	71.6	64.2	60.2	54.9	63.3	27.4	61.4 (-4.6)	15.6 (46.9%)
EViT [24]	77.2	60.2	53.4	70.1	57.9	84.6	73.6	65.3	60.1	55.4	64.9	28.6	62.6 (-3.4)	15.4 (47.5%)
ToMe [4]	76.9	61.4	53.9	70.1	57.6	85.5	73.1	65.0	61.2	56.0	65.9	32.6	63.3 (-2.7)	15.4 (47.5%)
Honeybee [5]	76.2	61.2	52.1	70.5	59.7	83.6	73.5	63.2	61.2	55.7	66.5	32.0	63.0 (-3.0)	15.4 (47.5%)
LLaMA-VID [22]	76.5	61.7	52.9	70.4	57.2	83.3	74.4	64.2	60.5	55.2	66.0	32.7	62.9 (-3.1)	15.5 (47.3%)
Qwen-VL [3]	77.3	61.1	52.1	70.8	56.4	84.0	71.7	65.8	61.7	56.3	66.7	31.5	63.0 (-3.0)	15.4 (47.5%)
IVTP (Ours)	78.4	62.3	54.1	70.1	60.0	85.4	77.1	67.7	63.3	59.3	68.6	35.5	65.2 (-0.8)	15.6 (46.9%)

Table 3 – TFLOPs based on # of Text Tokens

Methods	128	256	512	1024
LLaVA-1.5-7B [25]	9.9	11.7	15.4	22.9
TopK	2.7 (-72.7%)	4.5 (-61.5%)	8.0 (-48.1%)	15.2 (-33.6%)
ToMe [4]	2.7 (-72.7%)	4.5 (-61.5%)	8.0 (-48.1%)	15.2 (-33.6%)
Honeybee [5]	2.9 (-70.7%)	4.6 (-60.7%)	8.1 (-47.4%)	15.4 (-32.8%)
Qwen-VL [3]	2.9 (-70.7%)	4.6 (-60.7%)	8.1 (-47.4%)	15.3 (-33.2%)
IVTP (Ours)	3.0 (-69.7%)	4.7 (-59.8%)	8.2 (-46.8%)	15.5 (-32.3%)

Table 4 – Dif. Ways of Calculating Attention Scores

Model	Avg. Acc (PI)	Avg. Acc	TFLOPs
Group-wise	55.1 (-7.9)	58.1 (-4.9)	8.2 (-46.8%)
Layer-wise	56.2 (-6.8)	59.5 (-3.5)	8.2 (-46.8%)
Rollout	59.6 (-3.4)	62.0 (-1.0)	8.2 (-46.8%)

Table 5 – Dif. Ways of Combining Attention Weights

Model	Avg. Acc (PI)	Avg. Acc
mean	56.1 (-6.9)	60.1 (-2.9)
max	55.9 (-7.1)	60.7 (-2.3)
multiply	56.7 (-6.3)	60.5 (-2.5)
Residual	59.6 (-3.4)	62.0 (-1.0)

Table 6 - Which Instruction-guided Components Matter

OTT	TE	RT	Avg. Acc (PI)	Avg. ACC
			56.7 (-6.3)	59.2 (-3.8)
✓			58.3 (-4.7)	60.2 (-2.8)
✓		✓	57.3 (-5.7)	59.9 (-3.1)
✓	✓	✓	59.6 (-3.4)	62.0 (-1.0)

Table 7 – Does LLM-sided Pruning Help Models

Model	Avg. Acc (PI)	Avg. Acc
TopK	54.1 (-8.9)	58.9 (-4.1)
TopK^{+}	54.9 (-8.1)	58.7 (-4.3)
ToMe [4]	57.5 (-5.5)	60.0 (-3.0)
ToMe^{+}	57.7 (-5.3)	60.1 (-2.9)
IVTP-V	58.5 (-4.5)	60.8 (-2.2)
IVTP	59.6 (-3.4)	62.0 (-1.0)

Table 8 – Changing # of Layers Per Pruning Group

Layers	Avg. Acc (PI)	Avg. Acc
ALL	56.7 (-6.3)	58.2 (-4.8)
2	58.8 (-4.2)	60.1 (-2.9)
3	59.6 (-3.4)	62.0 (-1.0)
4	59.1 (-3.9)	61.5 (-1.5)
6	58.6 (-4.4)	59.9 (-3.1)

Table 9 - Computational Complexity Comparison

Methods	ViT	LLM	extra	Total
LLaVA-1.5-7B [25]	0.361	15.003	-	15.364
TopK	0.190	7.772	-	7.962
ToMe [4]	0.190	7.772	-	7.962
Qwen-VL [3]	0.360	7.772	0.003	8.135
IVTP (our)	0.202	7.946	0.080	8.228