北京大学 PEKING UNIVERSITY

UnZipLoRA: Separating Content and Style from a Single Image

Chang Liu Viraj Shah Aiyu Cui Svetlana Lazebnik

University of Illinois, Urbana-Champaign

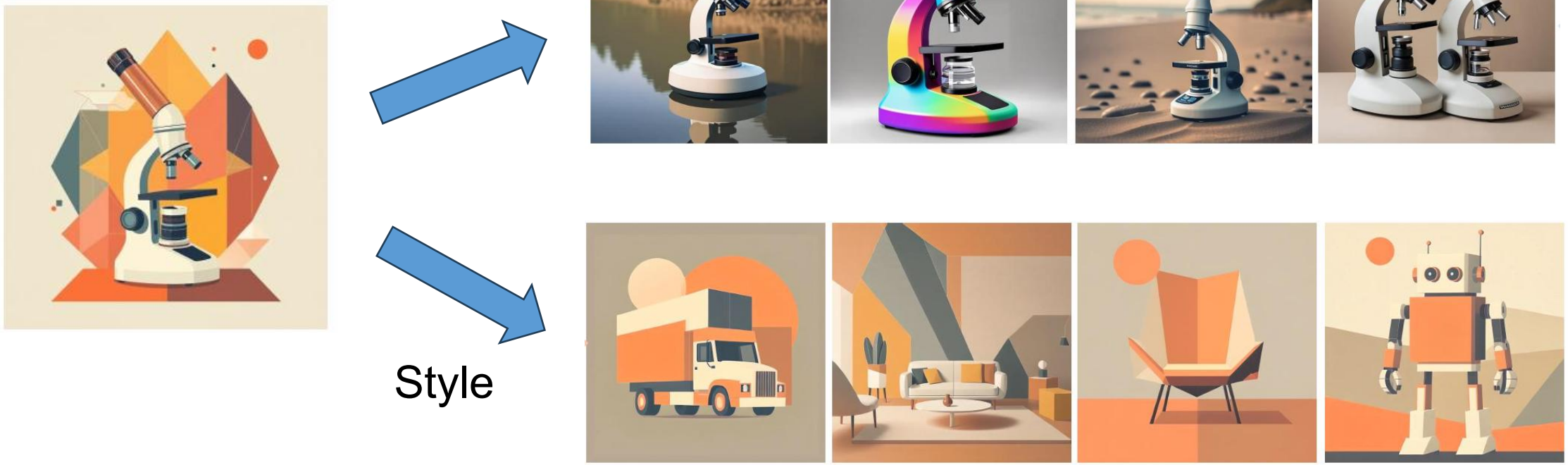
ICCV 2025 (Highlight)

Presenter: Yixuan Zou

2025.11.30

Background

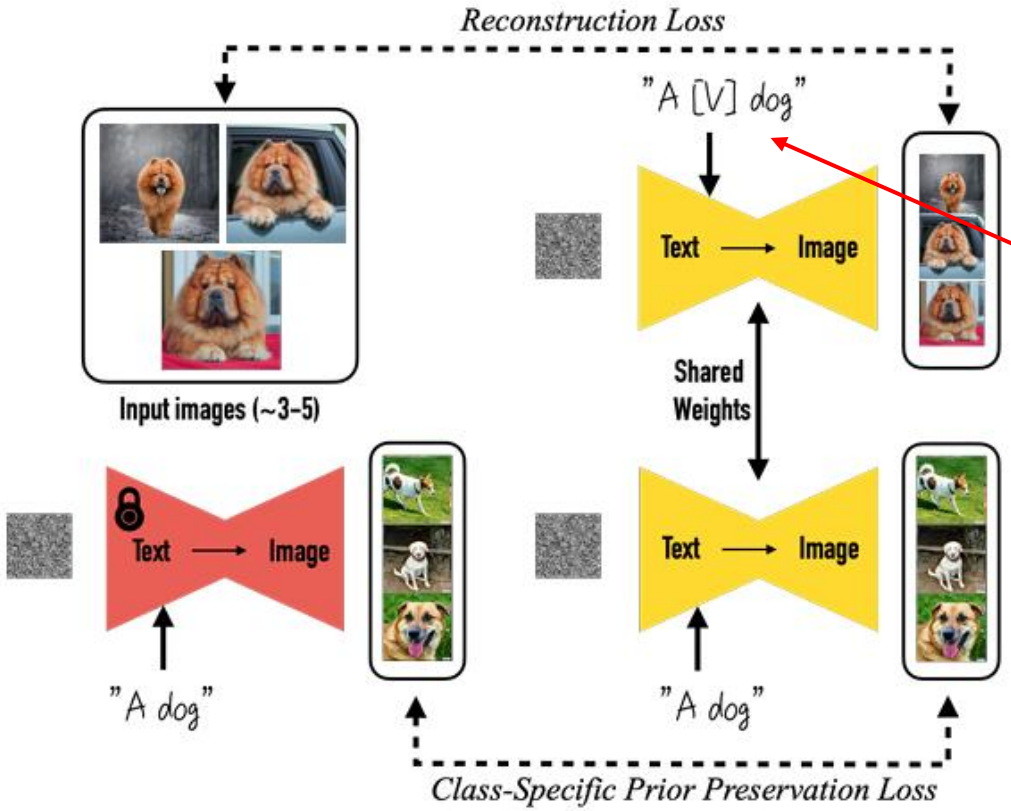
How to separate content and style from a single image ?



Background

For simplicity, let us first consider extracting the content from 3–5 images

DreamBooth



- Fine-tune a pretrained model
- For a given image set, label all images

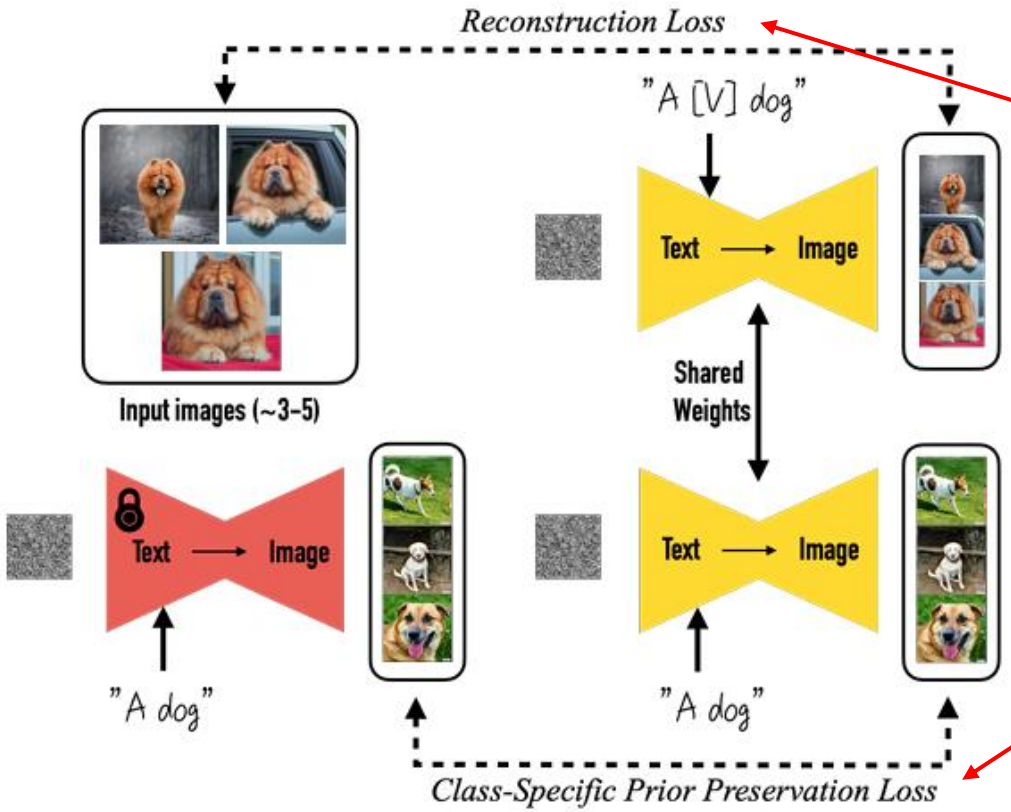
  “a [identifier] [class noun]”

  [identifier] is a rare token(e.g., sks)

  [class noun] is a class descriptor(e.g., dog)

DreamBooth: Fine Tuning Text-to-Image Diffusion Models for Subject-Driven Generation, arXiv:2208.12242

DreamBooth



- Add a class-specific prior preservation loss

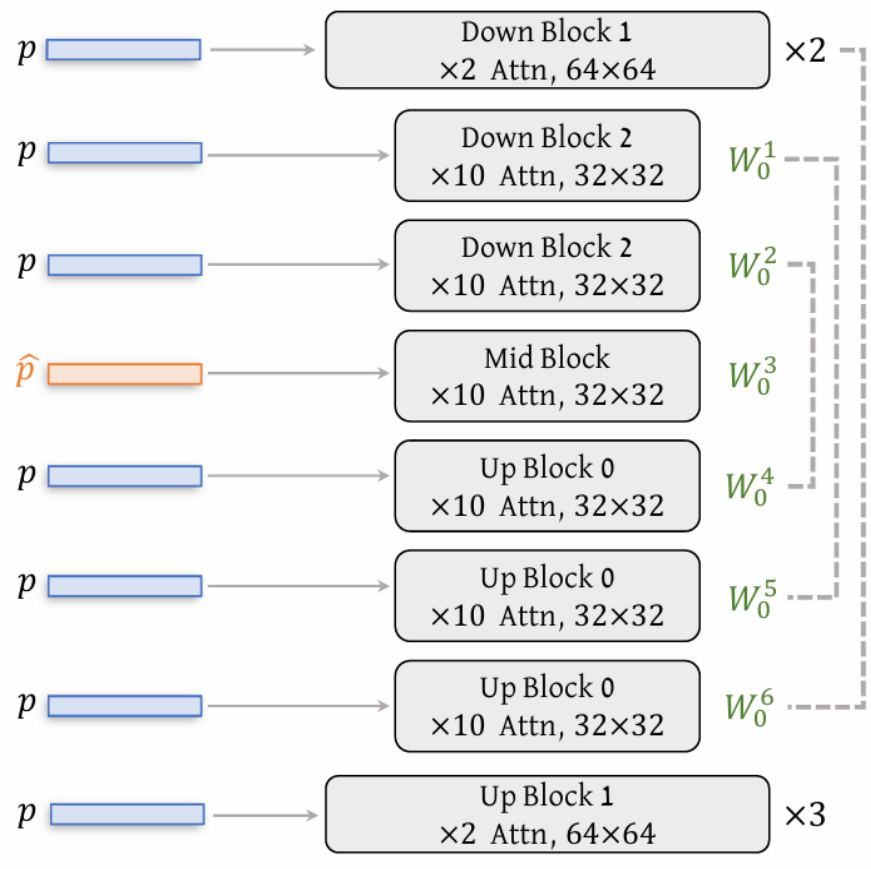
$$\mathbb{E}_{\mathbf{x},\mathbf{c},\boldsymbol{\epsilon},\boldsymbol{\epsilon}',t}\big[w_t\|\hat{\mathbf{x}}_\theta(\alpha_t\mathbf{x}+\sigma_t\boldsymbol{\epsilon},\mathbf{c})-\mathbf{x}\|_2^2+ \lambda w_{t'}\|\hat{\mathbf{x}}_\theta(\alpha_{t'}\mathbf{x}_{\text{pr}}+\sigma_{t'}\boldsymbol{\epsilon}',\mathbf{c}_{\text{pr}})-\mathbf{x}_{\text{pr}}\|_2^2\big],$$

$$\mathbf{c}_{\text{pr}} := \Gamma(f(\text{"a [class noun]"})).$$

DreamBooth: Fine Tuning Text-to-Image Diffusion Models for Subject-Driven Generation, arXiv:2208.12242

北京大学 PEKING UNIVERSITY

SDXL



- 3 Down Blocks → 1 Mid Block → 3 Up Blocks
- 2 Attention Blocks at mid-resolution stages

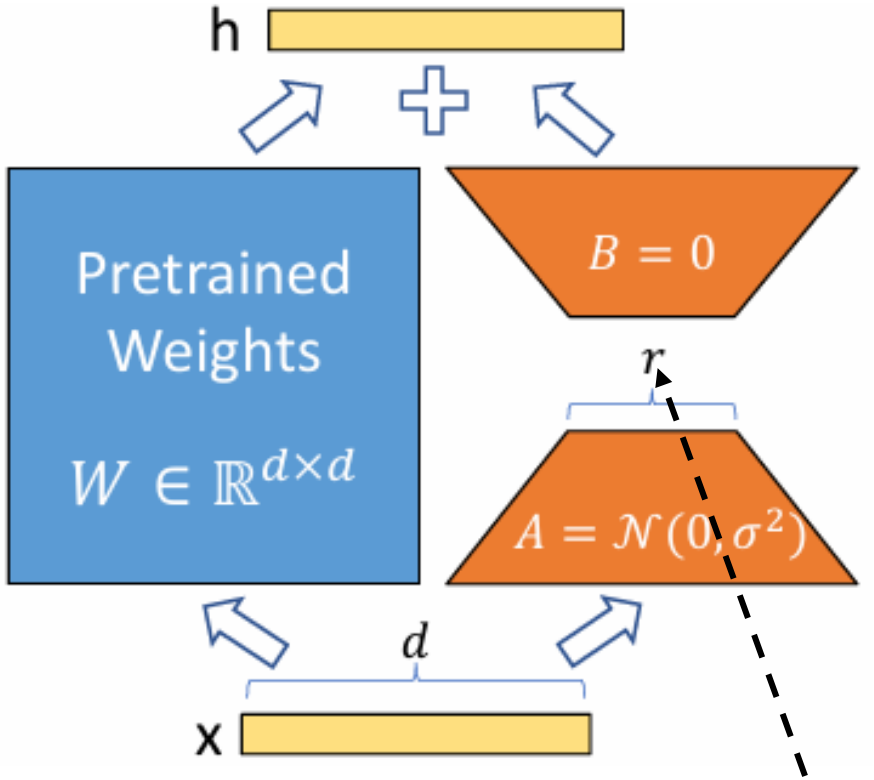
  10 Attention Blocks at low-resolution stages

Table 1: Comparison of *SDXL* and older *Stable Diffusion* models.

Model	*SDXL*	SD 1.4/1.5	SD 2.0/2.1
# of UNet params	2.6B	860M	865M
Transformer blocks	[0, 2, 10]	[1, 1, 1, 1]	[1, 1, 1, 1]
Channel mult.	[1, 2, 4]	[1, 2, 4, 4]	[1, 2, 4, 4]
Text encoder	CLIP ViT-L & OpenCLIP ViT-bigG	CLIP ViT-L	OpenCLIP ViT-H
Context dim.	2048	768	1024
Pooled text emb.	OpenCLIP ViT-bigG	N/A	N/A

SDXL: Improving Latent Diffusion Models for High-Resolution Image Synthesis, arXiv:2307.01952

Low-Rank Adaptation (LoRA)



Pre-trained weight matrix: $W_0 \in \mathbb{R}^{d\times k}$

Low-rank-parametrized update matrices:

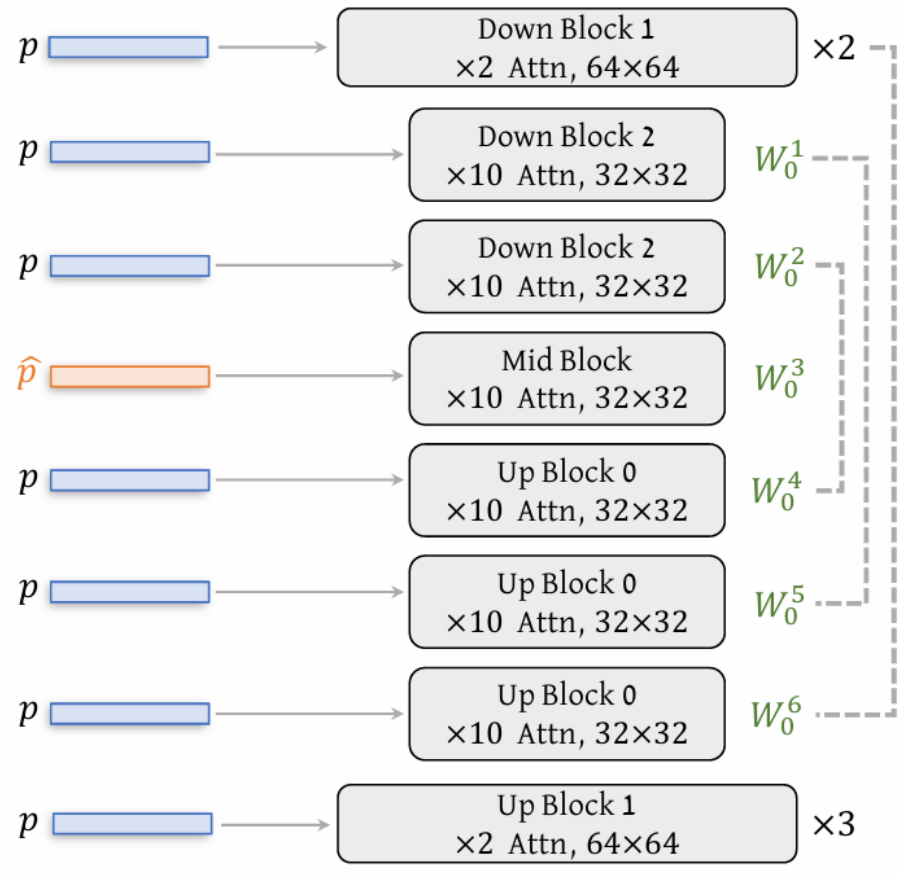
$W_0 + \Delta W = W_0 + BA$, where

$B \in \mathbb{R}^{d\times r}$, $A \in \mathbb{R}^{r\times k}$, $r \ll \min(d, k)$

LoRA: Low-Rank Adaptation of Large Language Models, arXiv:2106.09685

Background

B-LoRA



- Can we just optimize a few blocks ?
- B-LoRA finds that

  4th Block W_0^4 captures **content**

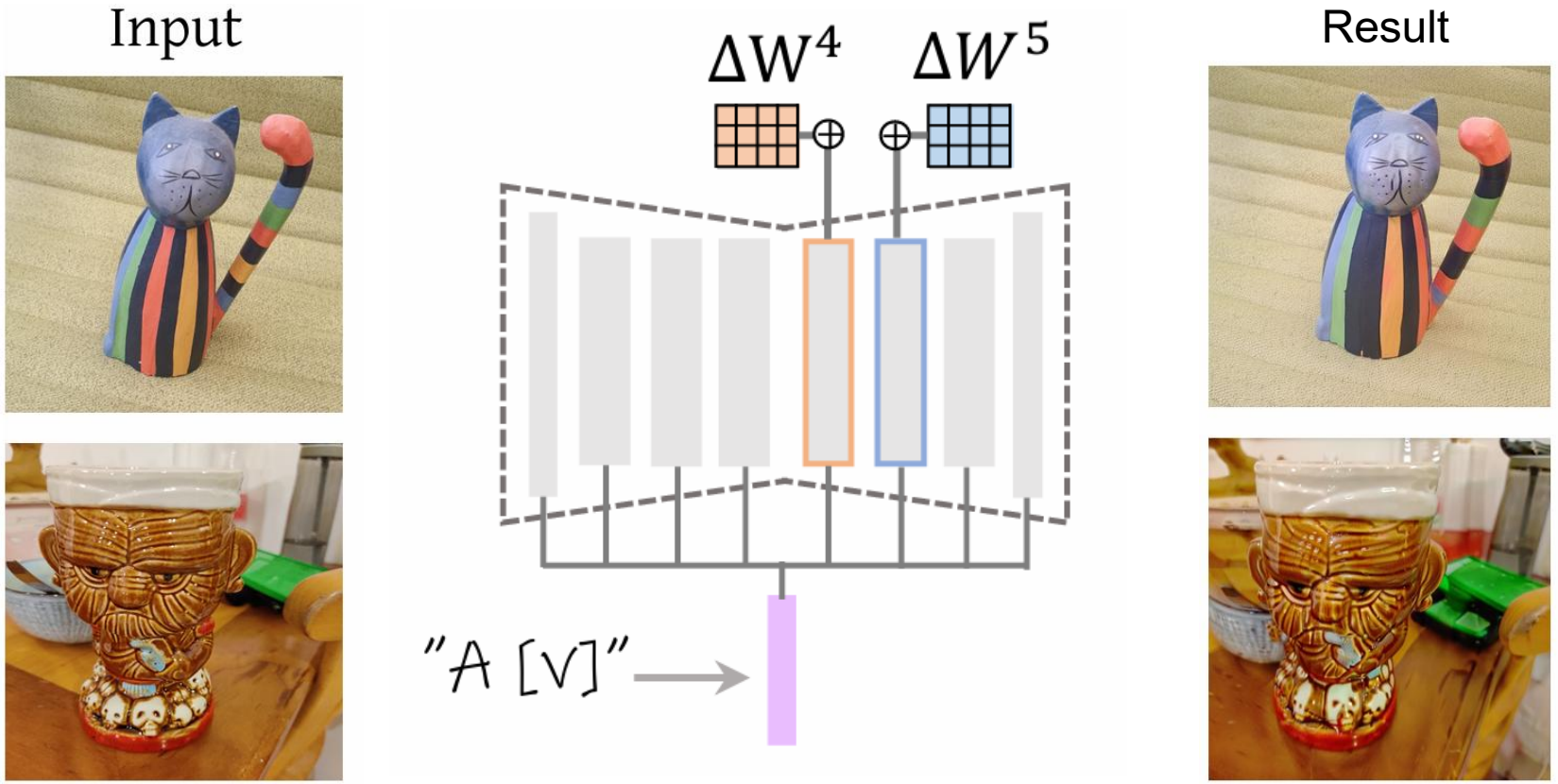
  5th Block W_0^5 captures **style**

- We can just optimize 2 blocks using LoRA

Implicit Style-Content Separation using B-LoRA, arXiv:2403.14572

B-LoRA

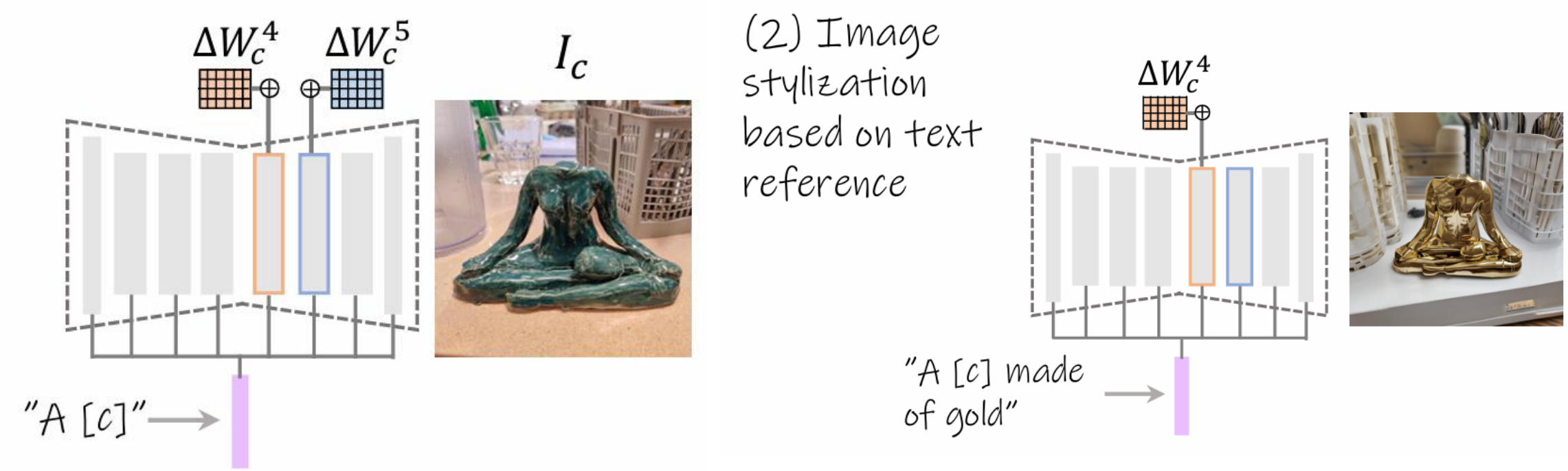
- Optimizing LoRA weights $\Delta W^4, \Delta W^5$ achieves good reconstruction performance



Implicit Style-Content Separation using B-LoRA, arXiv:2403.14572

B-LoRA

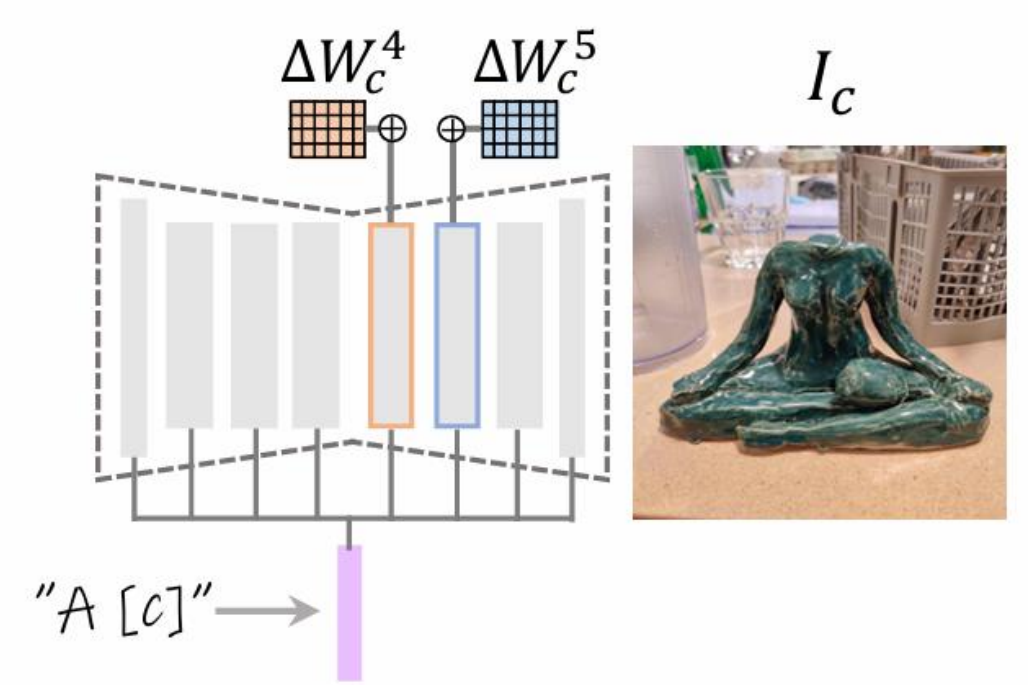
- Given an input image, we can just finetune LoRA weights $\Delta W^4, \Delta W^5$
- If we only plug ΔW^4, we can keep the **content** and change the **style**

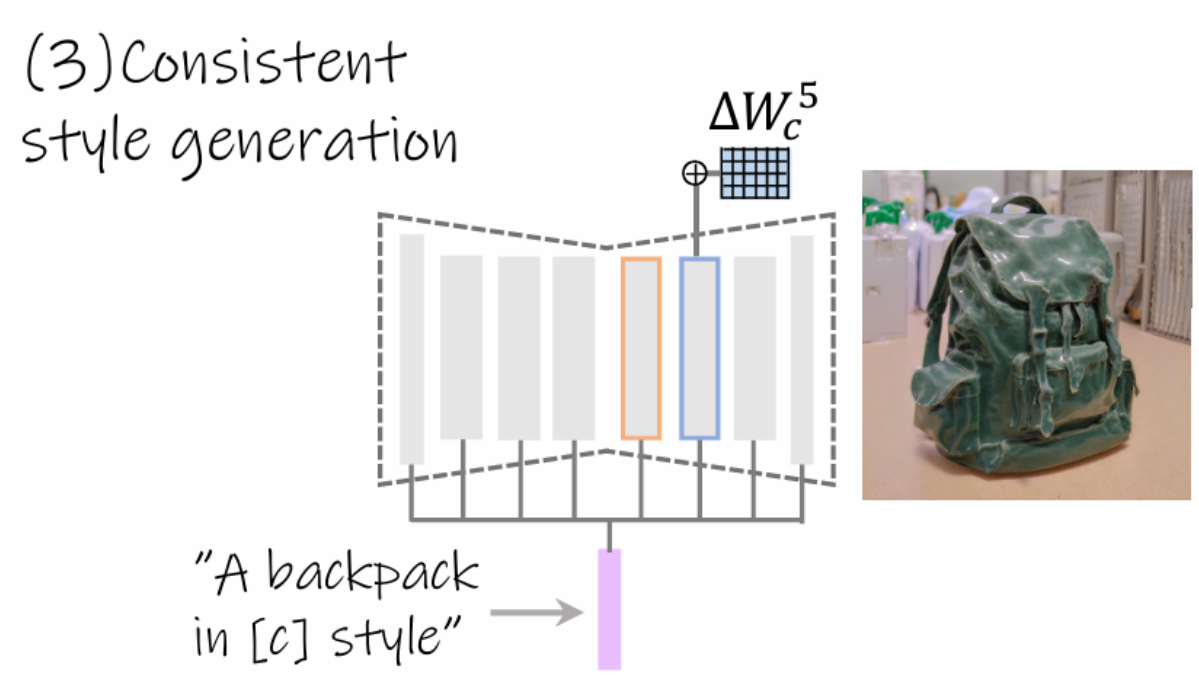


Implicit Style-Content Separation using B-LoRA, arXiv:2403.14572

B-LoRA

- Given an input image, we can just finetune LoRA weights $\Delta W^4, \Delta W^5$
- If we only plug ΔW^5, we can keep the **style** and change the **content**

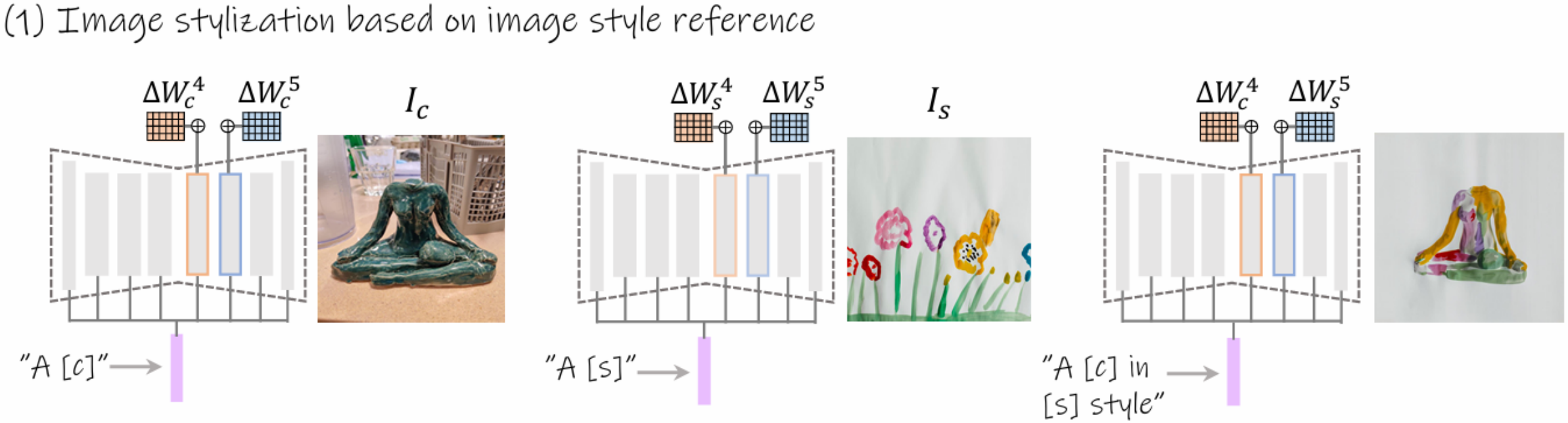




Implicit Style-Content Separation using B-LoRA, arXiv:2403.14572

B-LoRA

- Given 2 input images, we can finetune LoRA weights separately
- Combining LoRA weights from different images can mix **content** and **style**



Background

When we have 2 LoRA weights, can we just merge them ?

Content

Style

Background

Direct merge obtains bad results, but ZipLoRA obtains better results

References

A [V] toy

flat cartoon illustration

Direct Merge

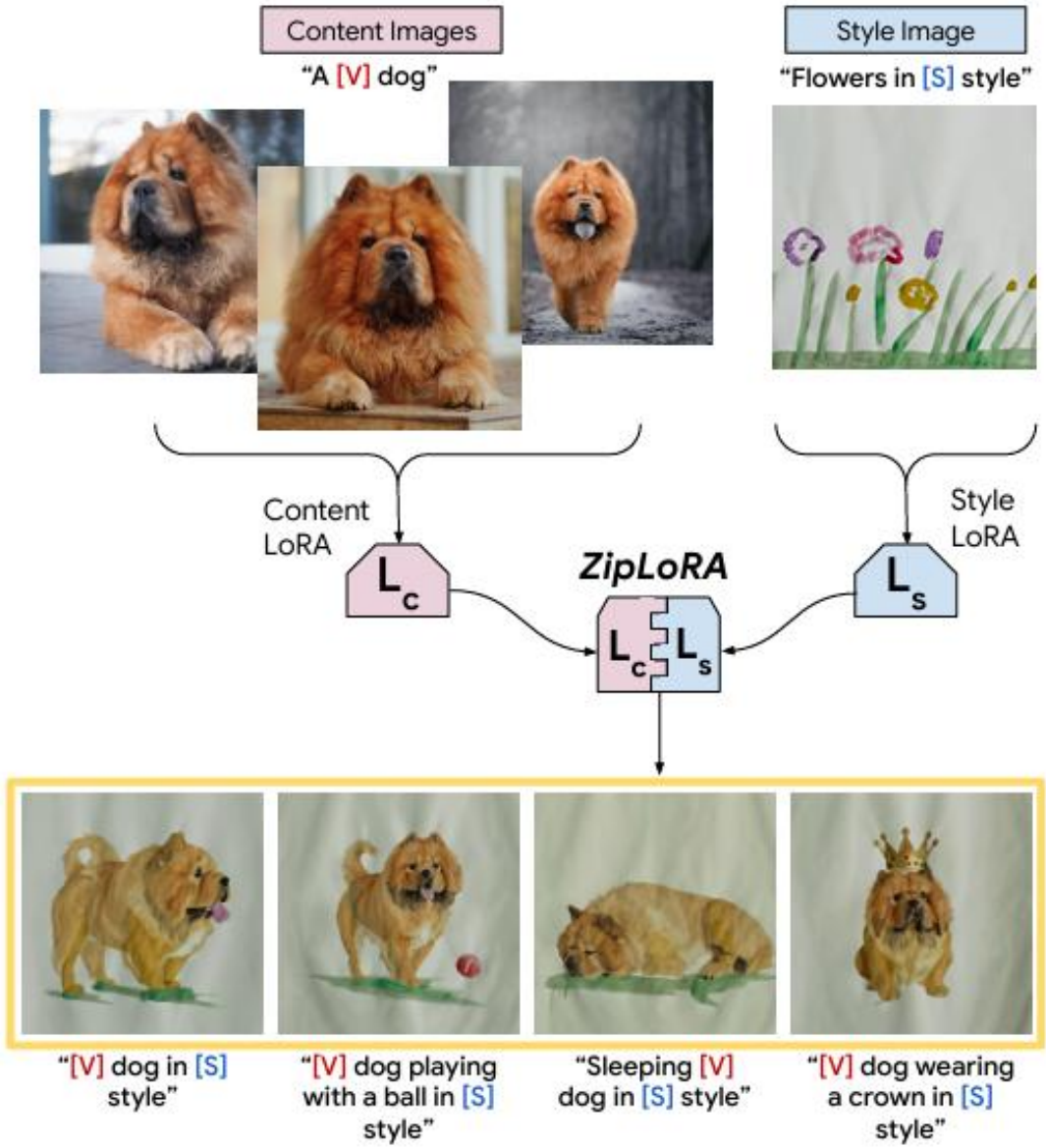
ZipLoRA

Background

ZipLoRA

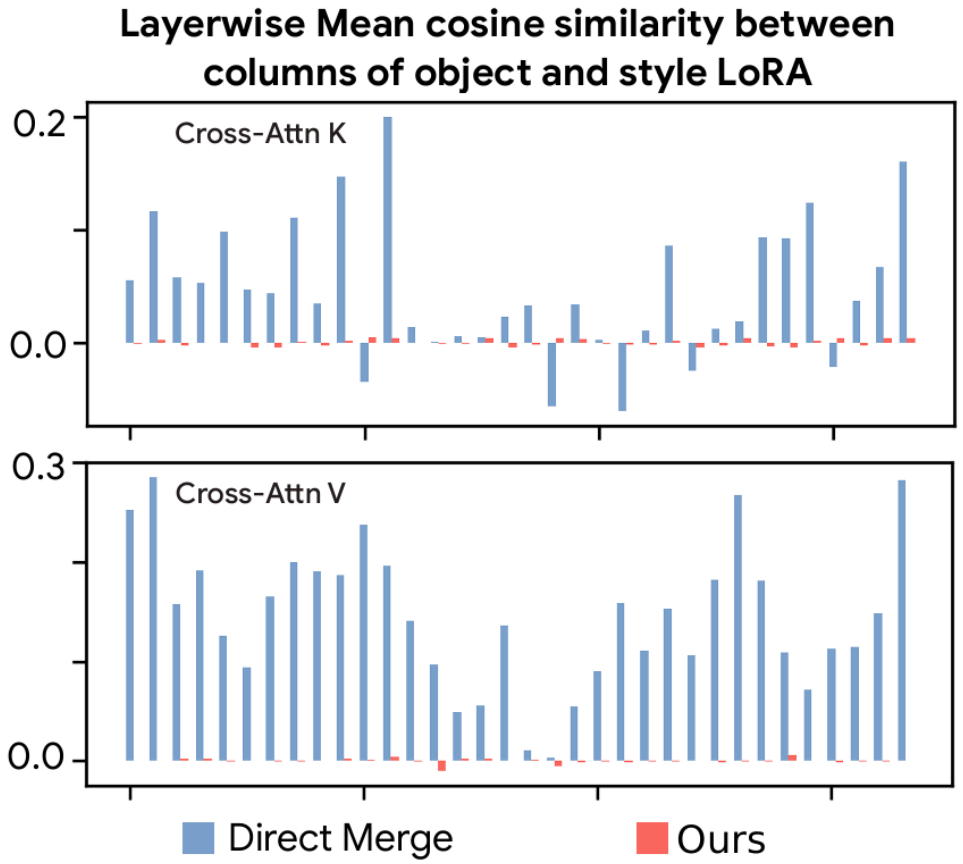
- Given content LoRA and style LoRA

  ZipLoRA learns how to merge them



ZipLoRA

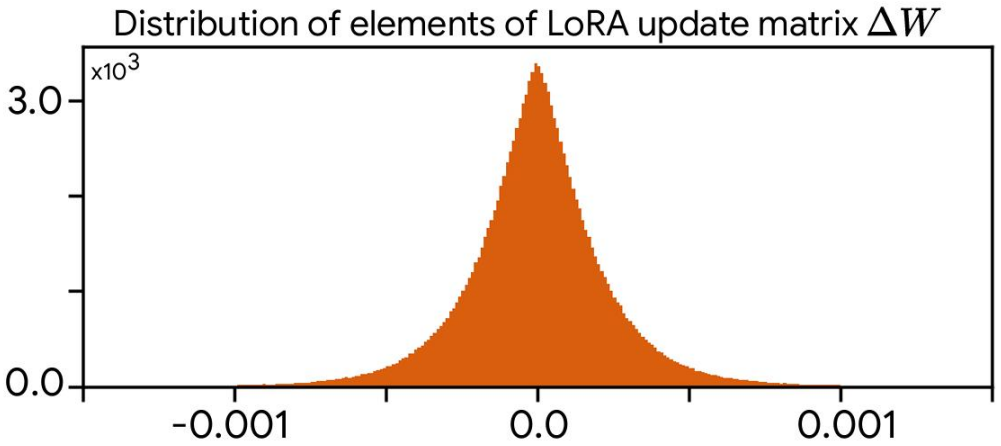
- Key insight: Highly aligned LoRA weights merge poorly
- “Aligned” means cosine similarity between columns of 2 LoRAs are large

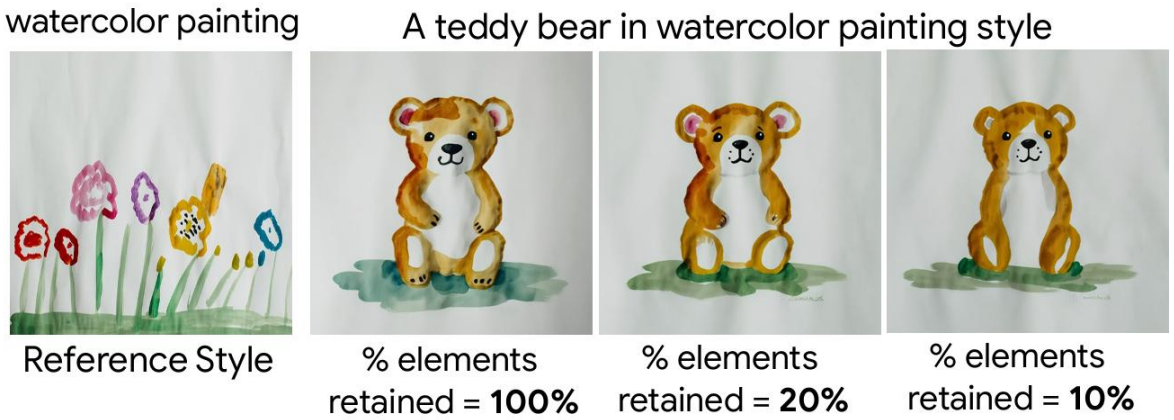


ZipLoRA: Any Subject in Any Style by Effectively Merging LoRAs, arXiv:2311.13600

ZipLoRA

- Another key insight: LoRA weight matrices are sparse

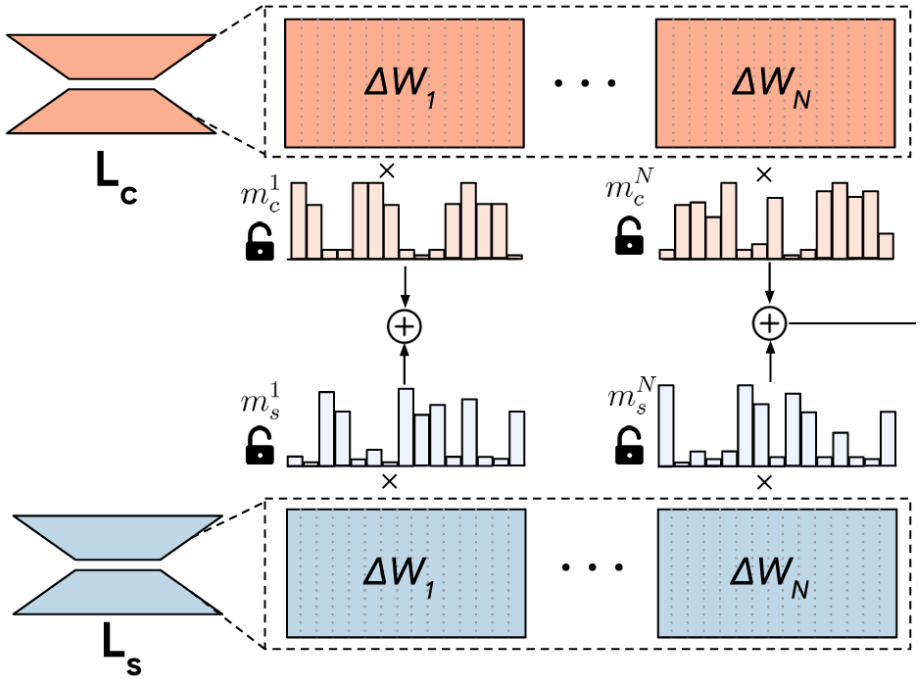




ZipLoRA: Any Subject in Any Style by Effectively Merging LoRAs, arXiv:2311.13600

ZipLoRA

- Introduce the column-wise scaling factors m_c , m_s to keep the orthogonality
- We can freeze content LoRA L_c and style LoRA L_s
- Learn column-wise scaling factors m_c , m_s

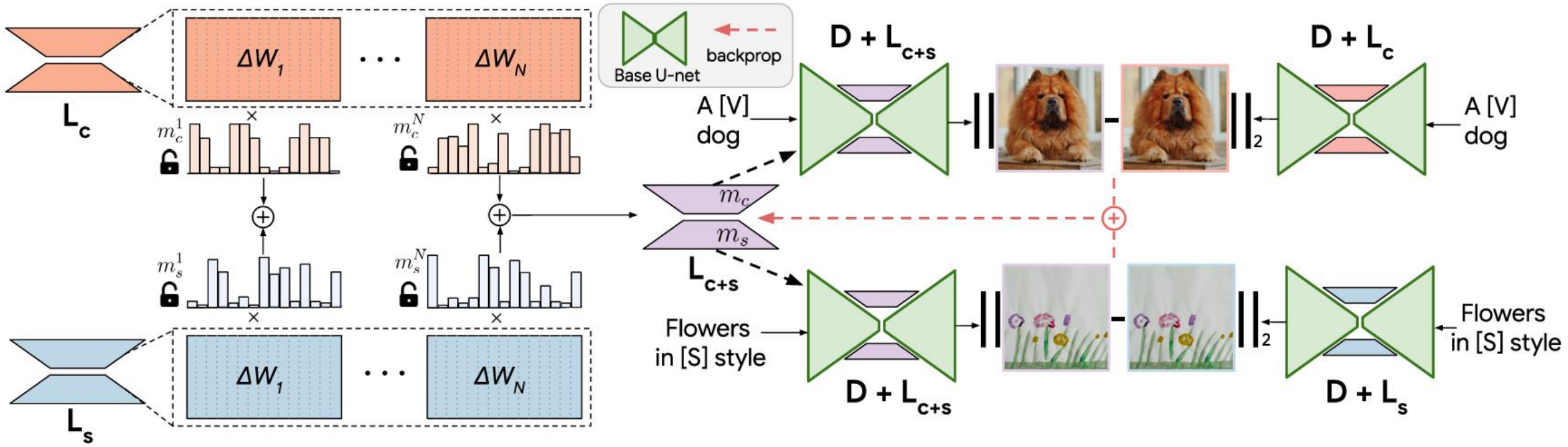


ZipLoRA: Any Subject in Any Style by Effectively Merging LoRAs, arXiv:2311.13600

ZipLoRA

- Loss: content-style loss + cosine-similarity loss

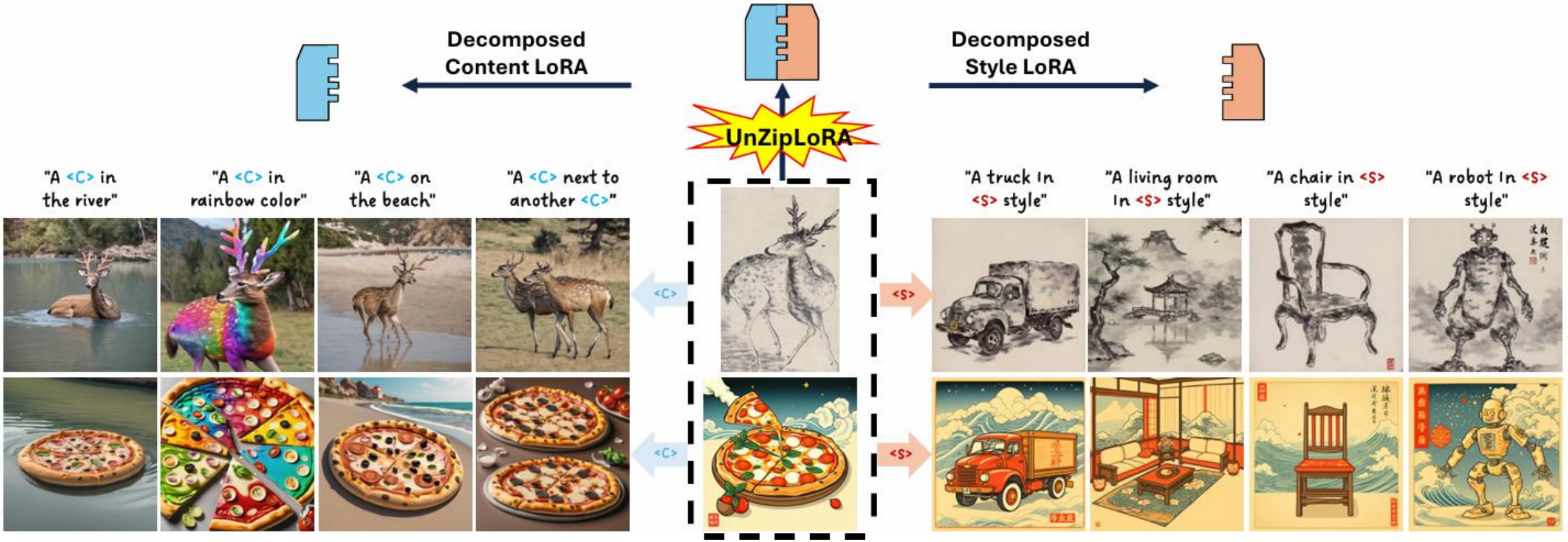
$$\mathcal{L}_{merge} = \|(D \oplus L_m)(x_c, p_c) - (D \oplus L_c)(x_c, p_c)\|_2 + \|(D \oplus L_m)(x_s, p_s) - (D \oplus L_s)(x_s, p_s)\|_2 + \lambda \sum_i |m_c^{(i)} \cdot m_s^{(i)}|,$$



ZipLoRA: Any Subject in Any Style by Effectively Merging LoRAs, arXiv:2311.13600

北京大学 PEKING UNIVERSITY

UnZipLoRA does the opposite



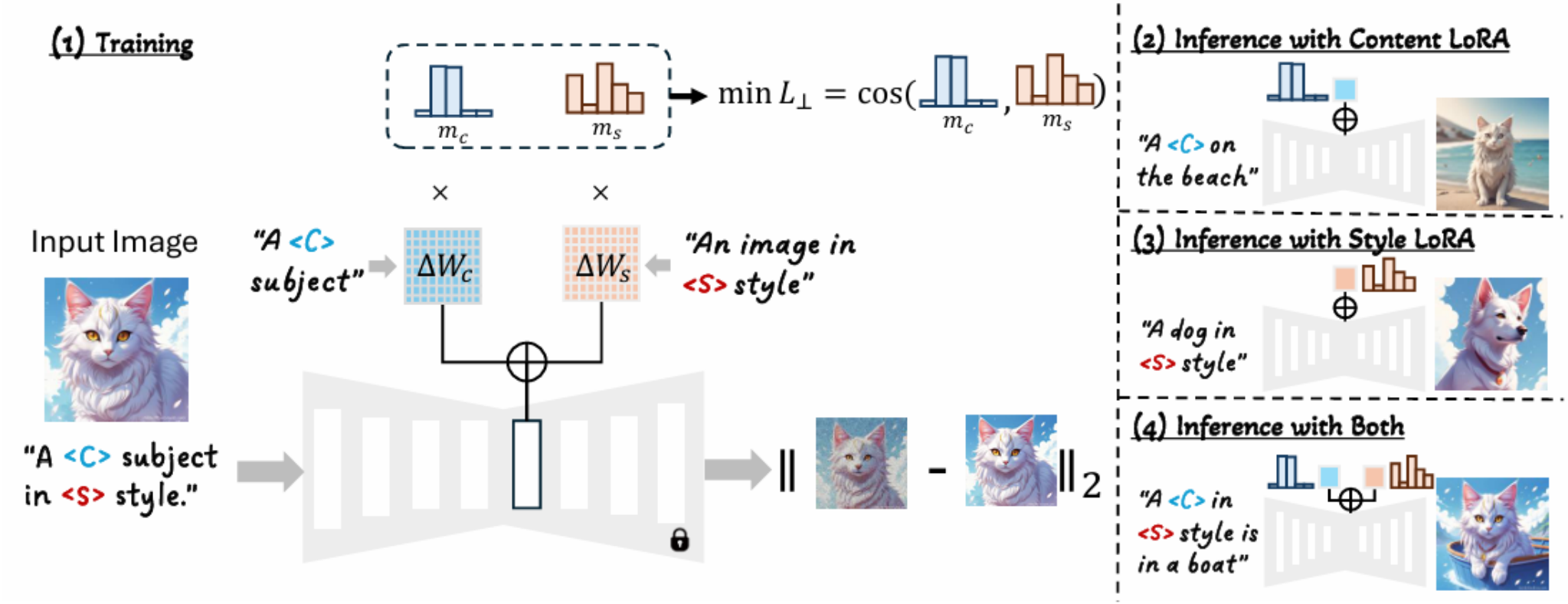
北京大学 PEKING UNIVERSITY

Problem Setup

- Given a pre-trained diffusion model with weights $\{W_0^i\}$

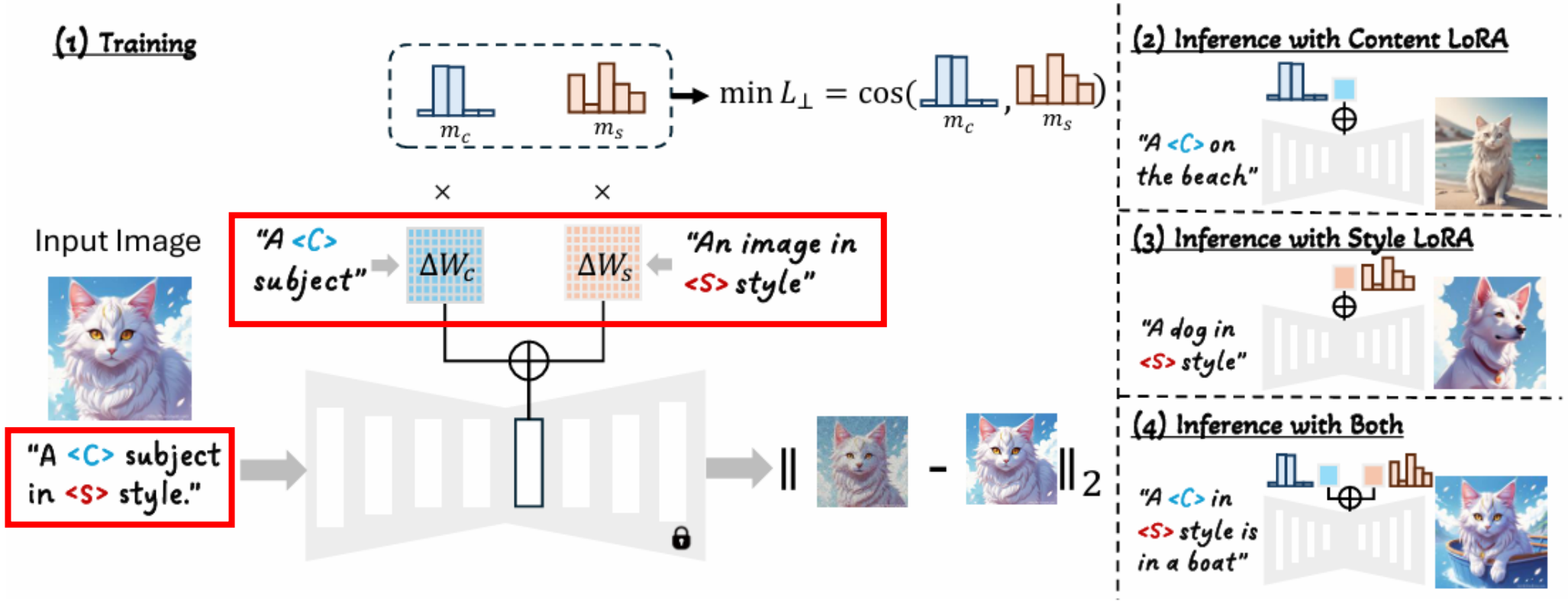
  Given a single input image
- Learn two models:
  - content LoRA $L_c = \{\Delta W_c^i\}$
  - style LoRA $L_s = \{\Delta W_s^i\}$

Overview: prompt separation, column separation, block separation



北京大学 PEKING UNIVERSITY

Prompt Separation



Prompt Separation

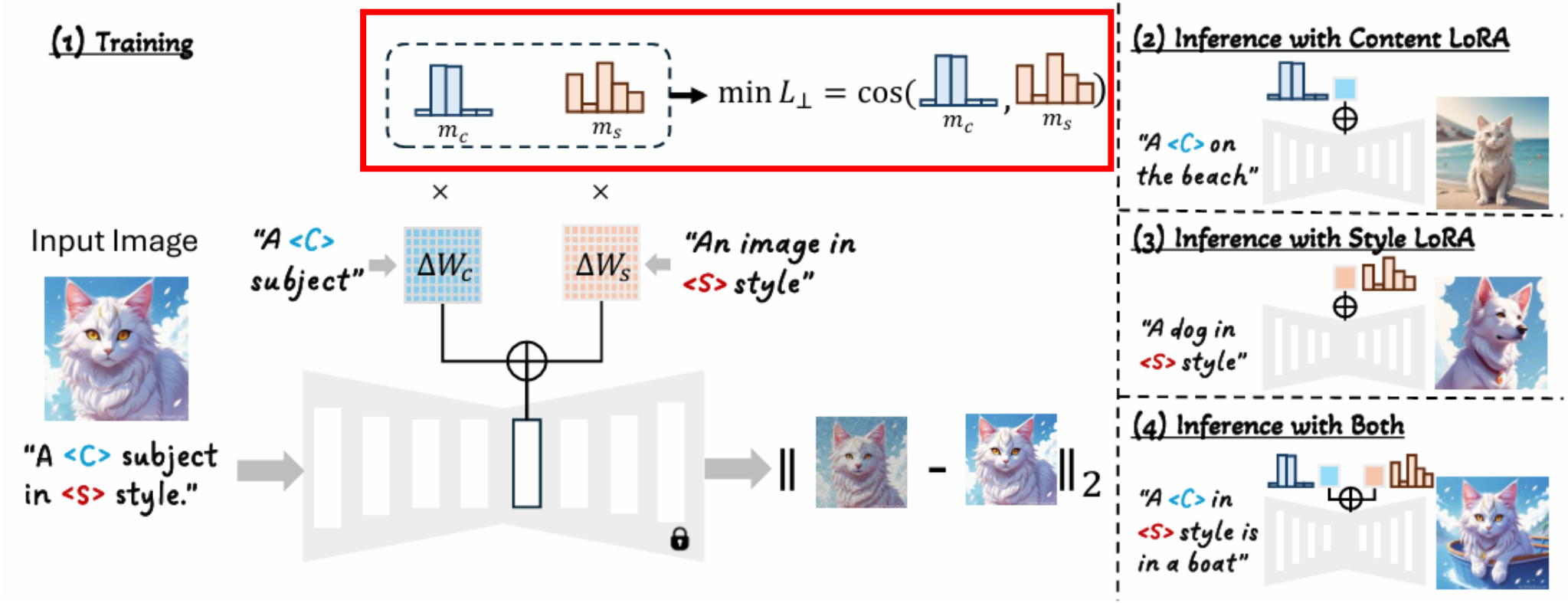
- Typical cross-attention layer $K(x)$ or $V(x) = W_0^T x$.
- Use 3 separate prompts to prevent cross-contamination

$$K \text{ or } V(x, x_s, x_c) = W_0^T x + \Delta W_s^T x_s + \Delta W_c^T x_c,$$

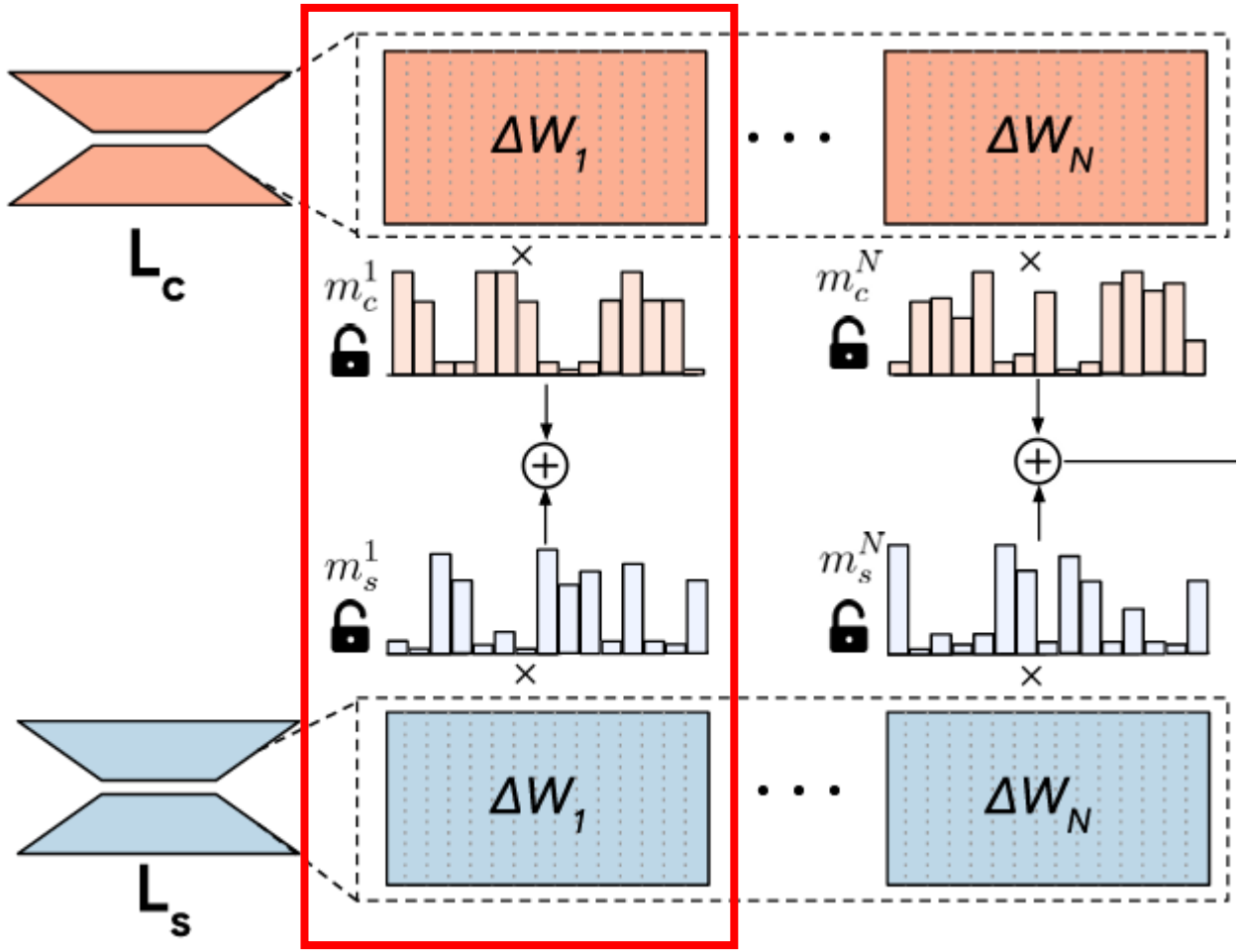
where x is “A < c > in < s > style”

x_c is < c > (e.g., ‘sks dog’) x_s is < s > (e.g., ‘watercolor painting’)

Column Separation



Recall: ZipLoRA uses column-wise scaling factors

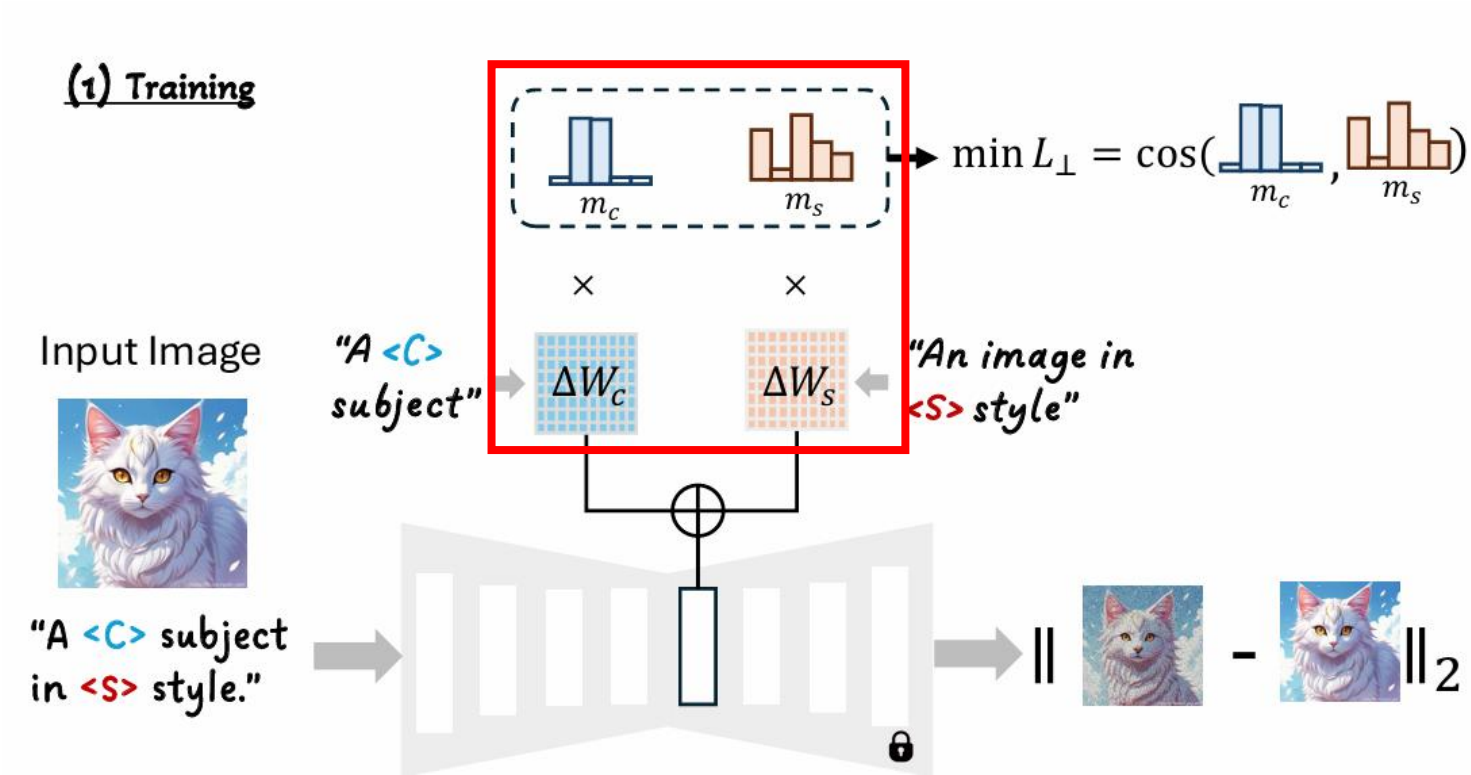


$$\mathcal{L}_{\perp} = \sum_i |m_c^i \cdot m_s^i|$$

Column Separation

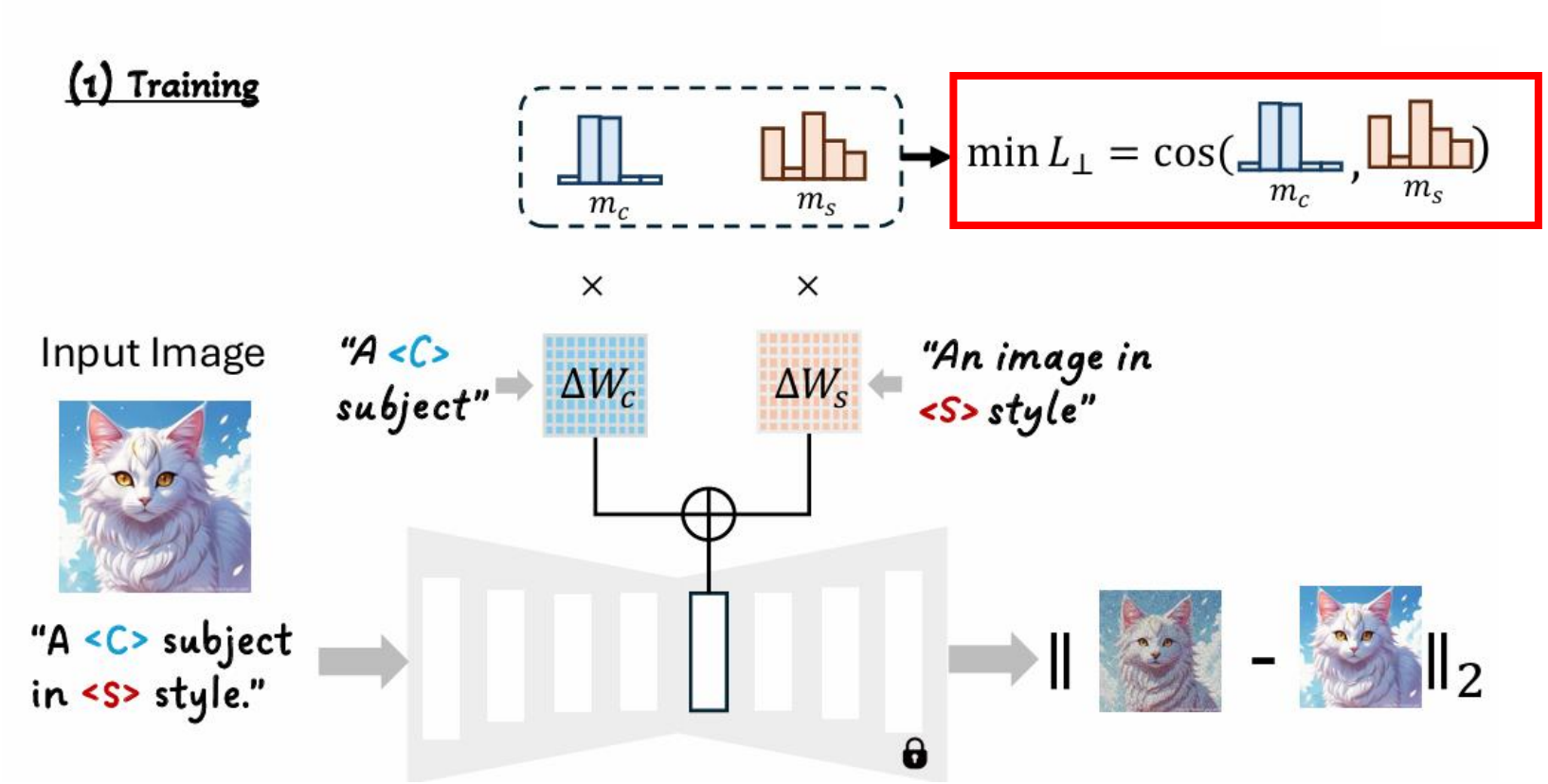
- Introduce the column-wise scaling factors for each LoRA

$$K \text{ or } V(x, x_s, x_c) = W_0^T x + m_s \Delta W_s^T x_s + m_c \Delta W_c^T x_c.$$



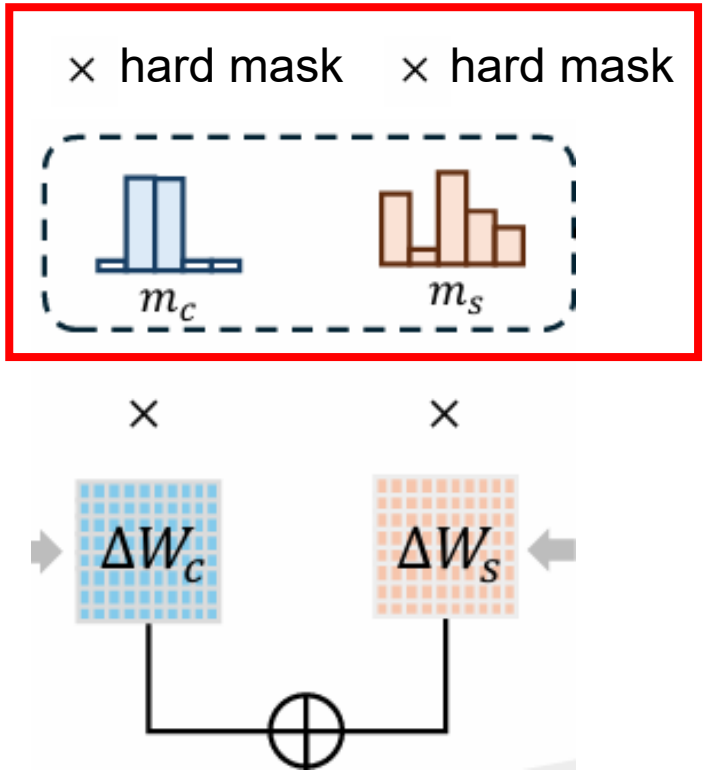
Column Separation

- Add an orthogonal loss to promote compatibility $\mathcal{L}_{\perp} = \sum_i |m_c^i \cdot m_s^i|$



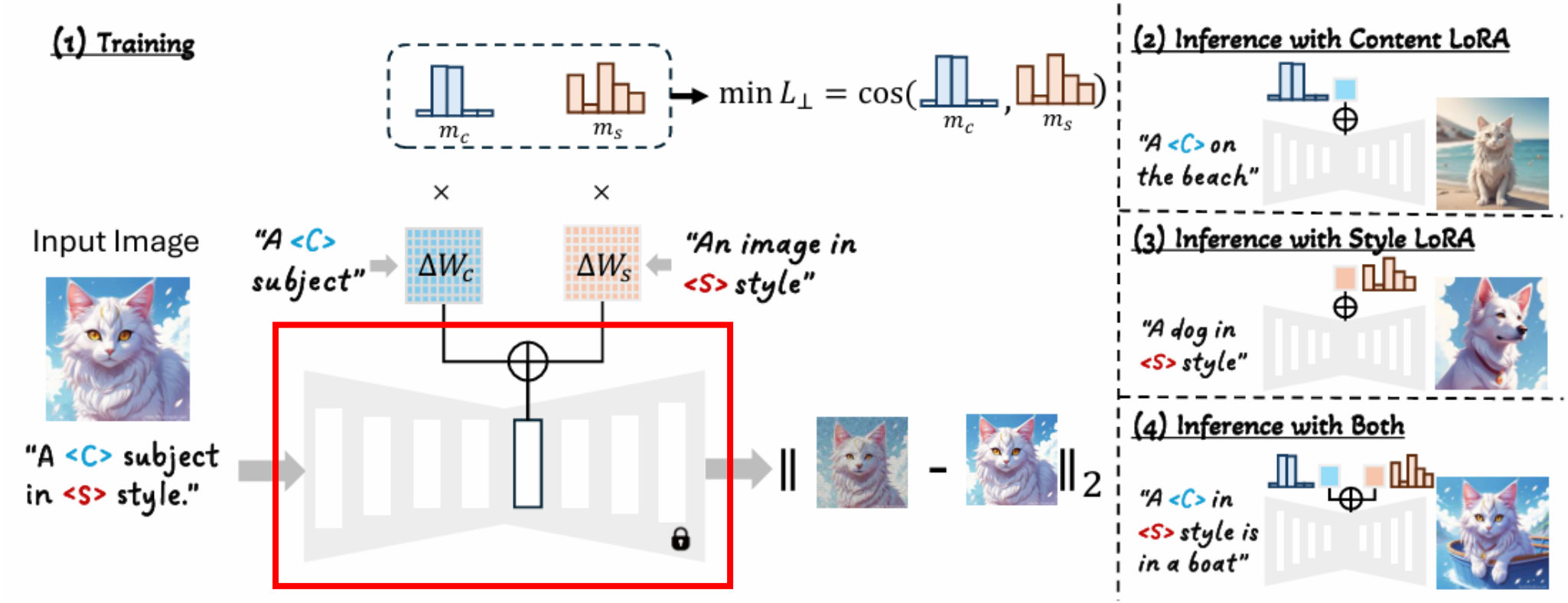
Column Separation

- UnZipLoRA finds that training a fraction of columns is sufficient
- We can just optimize a set of columns (10% → 20% → 30%)

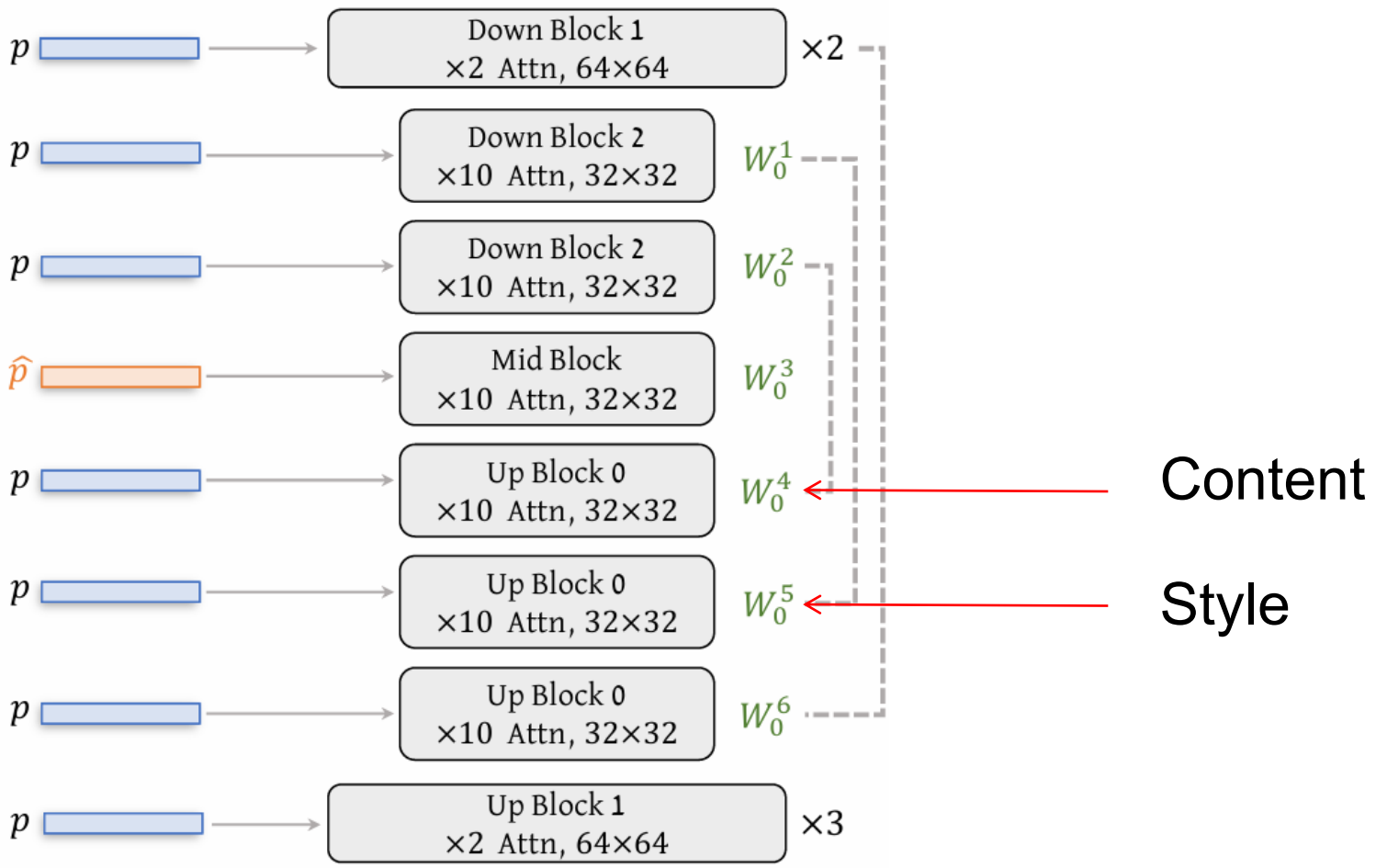


北京大学 PEKING UNIVERSITY

Block Separation

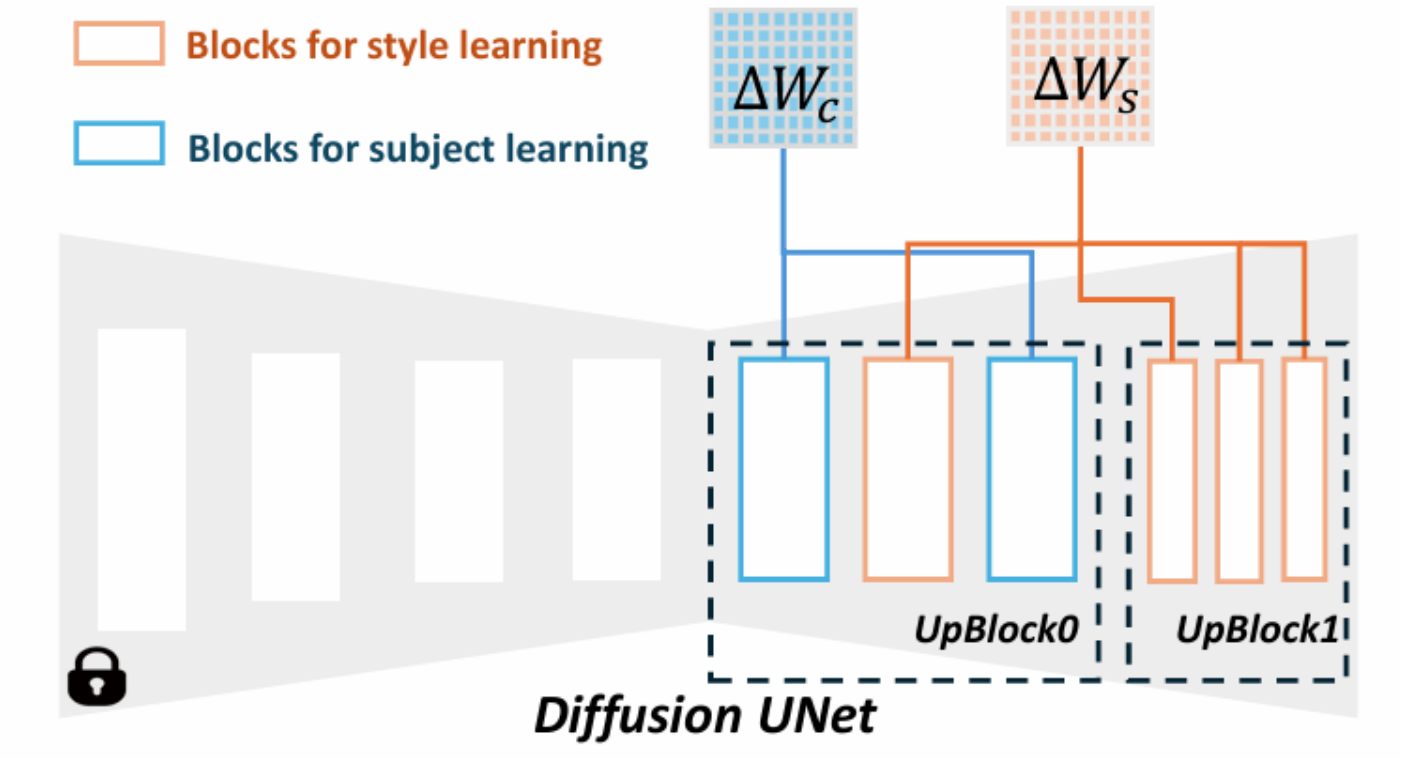


Recall: B-LoRA finds that certain blocks capture content/style



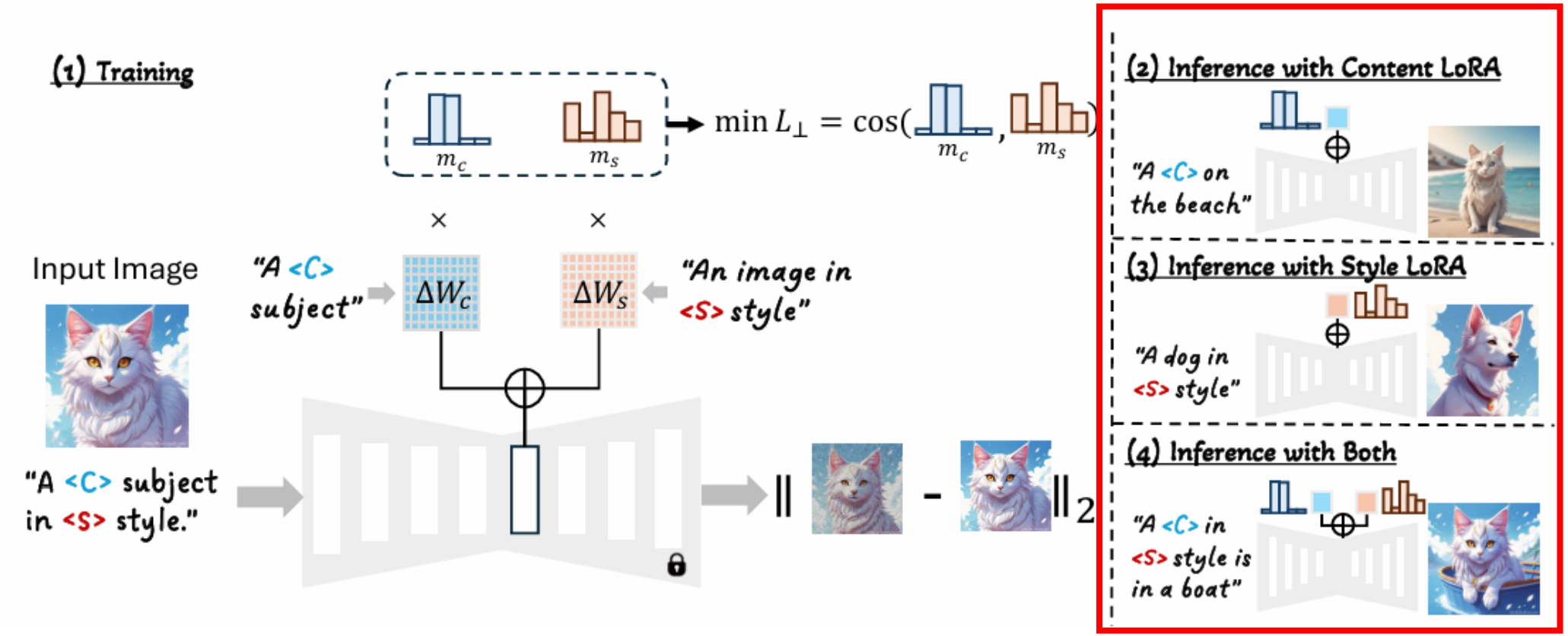
Block Separation

- In these blocks, all LoRA columns are fully trained without masks



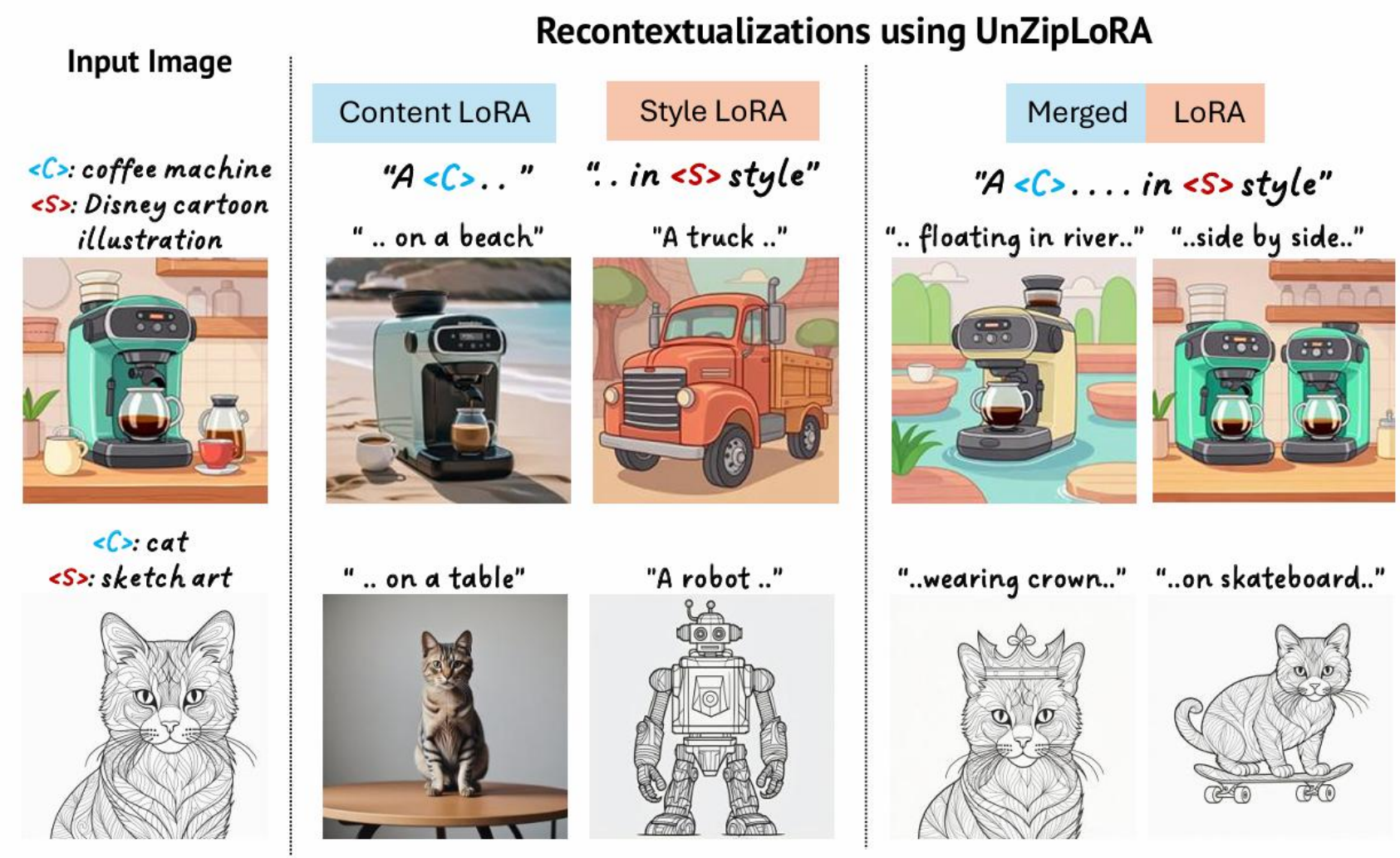
北京大学 PEKING UNIVERSITY

Inference



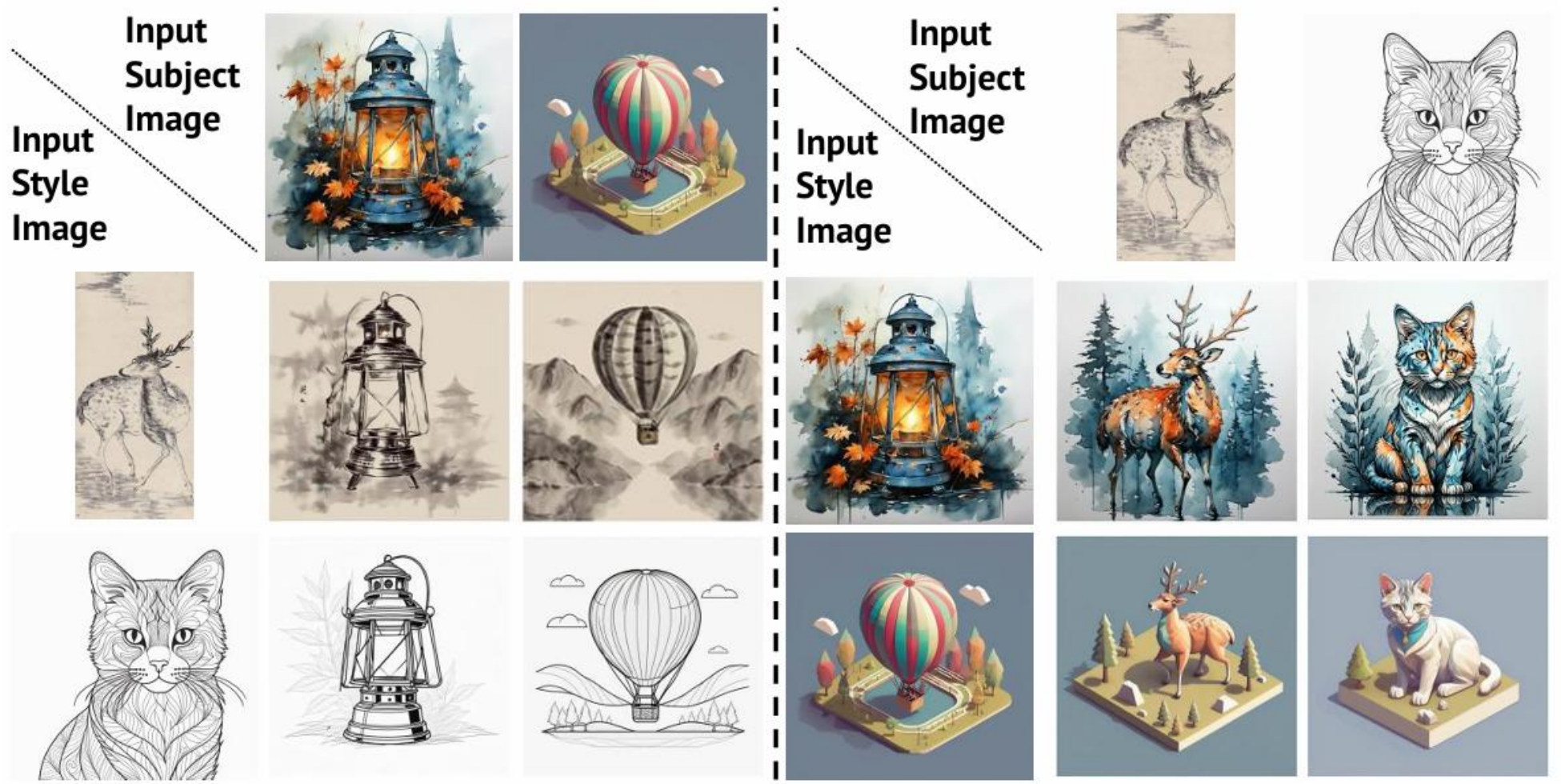
北京大学 PEKING UNIVERSITY

Qualitative Results



北京大学 PEKING UNIVERSITY

Qualitative Results



Qualitative Comparisons

Quantitative Comparison

% Preference for UnZipLoRA over:			
	DreamBooth LoRA	Inspiration Tree	B-LoRA
Decomposition	91.17%	81.53%	62.74%
Recontextualization	98.10%	79.17%	77.14%

	DB-LoRA	Insp. Tree	B-LoRA	UnZipLoRA
Style-align. (CLIP-I) ↑	0.417	0.404	0.418	**0.427**
Subject-align. (DINO) ↑	0.339	0.291	0.337	**0.349**
Style-align. (CSD) ↑	0.245	0.229	0.244	**0.265**
Subject-align. (CSD) ↑	0.338	0.334	0.342	**0.358**

Experiments

Ablation Study

Input Image

<C>: cat
<S>: anime illustration

Baseline (DreamBooth-LoRA)

Baseline + Prompt Sep. [M1]

Baseline + Prompt Sep. + Column Sep. [M2]

Baseline + Prompt Sep. + Column Sep. + Block Sep. [M3]

Failure Cases

- For highly abstract styles, UnZipLoRA may fail to destylize the subject

Conclusion

北京大学 PEKING UNIVERSITY

- Certain blocks are more responsible for content/style
- Highly aligned LoRA weights merge poorly
- LoRA weight matrices are sparse
- Training a fraction of LoRA weights is sufficient to capture concepts

Thanks for listening!

Presenter: Yixuan Zou
2025.11.30

Implementation Details

- Dataset: use a set of 40 diverse images collected from previous work
- Experimental setup:
  - Base model: SDXL v1.0
  - LoRA: rank = 64 using Adam (learning rate = 5e-5) for 600 steps
  - Column separation: t = 200, N = 30%, λ = 0.5
  - Block separation: use all upsampling blocks