



# DEFT: Decompositional Efficient Fine-Tuning for Text-to-Image Models

Komal Kumar, Rao Muhammad Anwer, Fahad Shahbaz Khan, Salman Khan,  
Ivan Laptev, Hisham Cholakkal

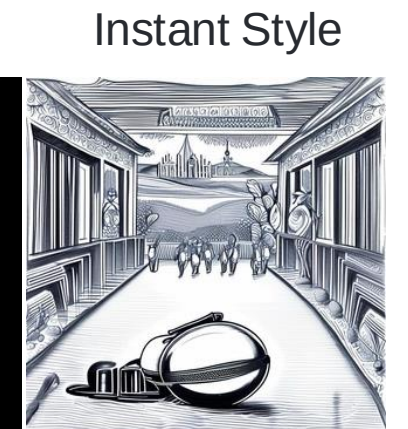
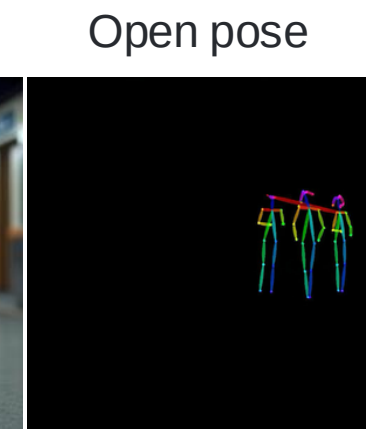
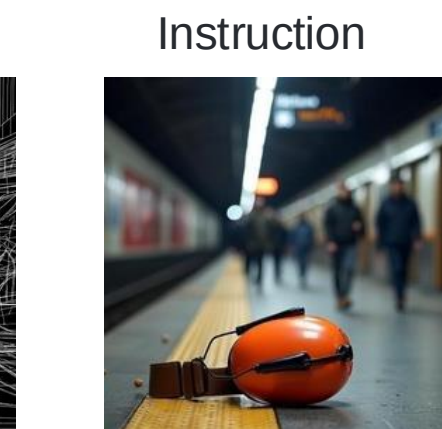
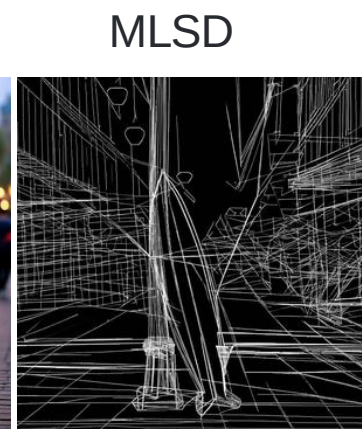
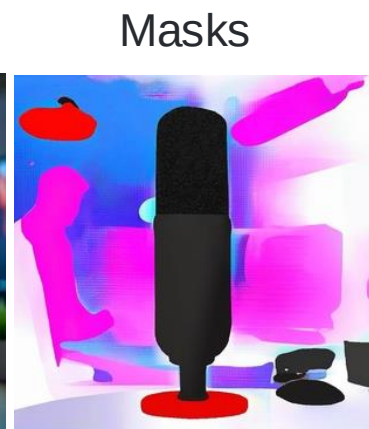
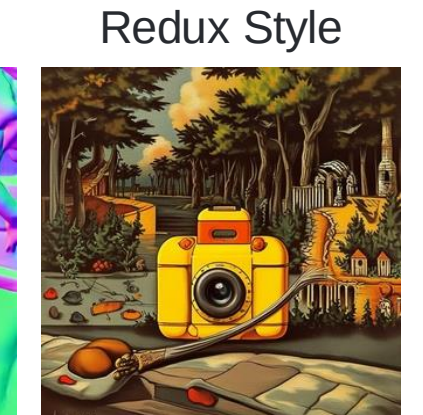
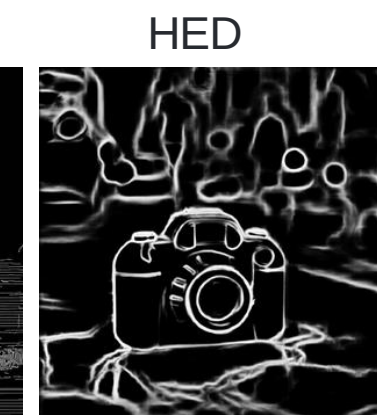
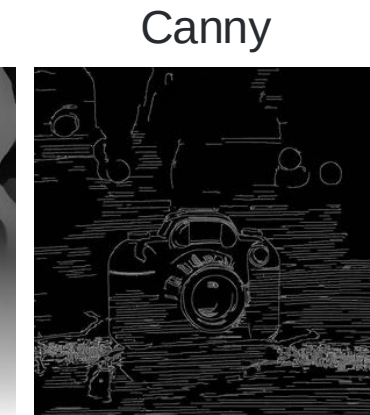
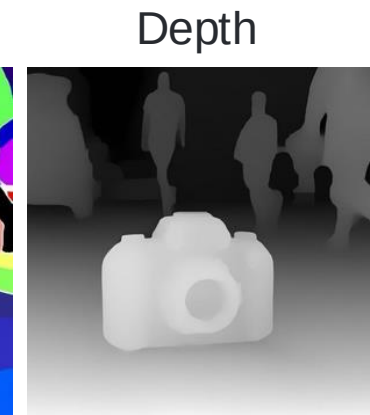
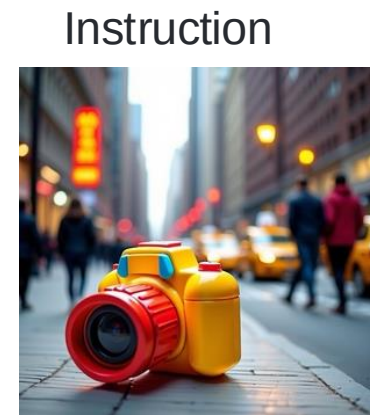
<sup>1</sup>Mohamed bin Zayed University of Artificial Intelligence Abu Dhabi, UAE  
{komal.kumar, rao.anwer, fahad.khan,  
salman.khan, ivan.laptev, hisham.cholakkal}@mbzuai.ac.ae



# Why Efficient Fine-Tuning Matters for Text-to-Image Models?

- Large T2I models (e.g., SDXL) are powerful but costly to adapt.
- Real-world use cases (personalization, domain adaptation, concept learning) require efficient fine-tuning.
- Full fine-tuning = billions of parameters → expensive, slow, and prone to forgetting.

# Why Efficient Fine-Tuning Matters for Text-to-Image Models?




**Goal:** Enable efficient, new task adaption without retraining the entire model.



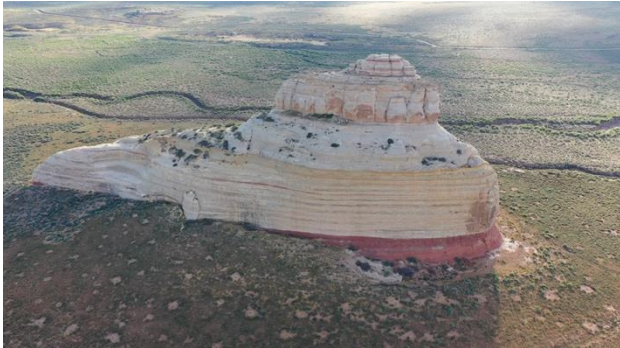
# Why Efficient Fine-Tuning Matters for Text-to-Image Models?









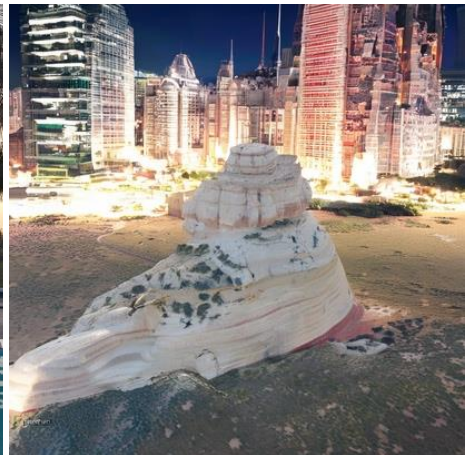



Scene Personalization

Office (indoor)



Church Rock (Outdoor)

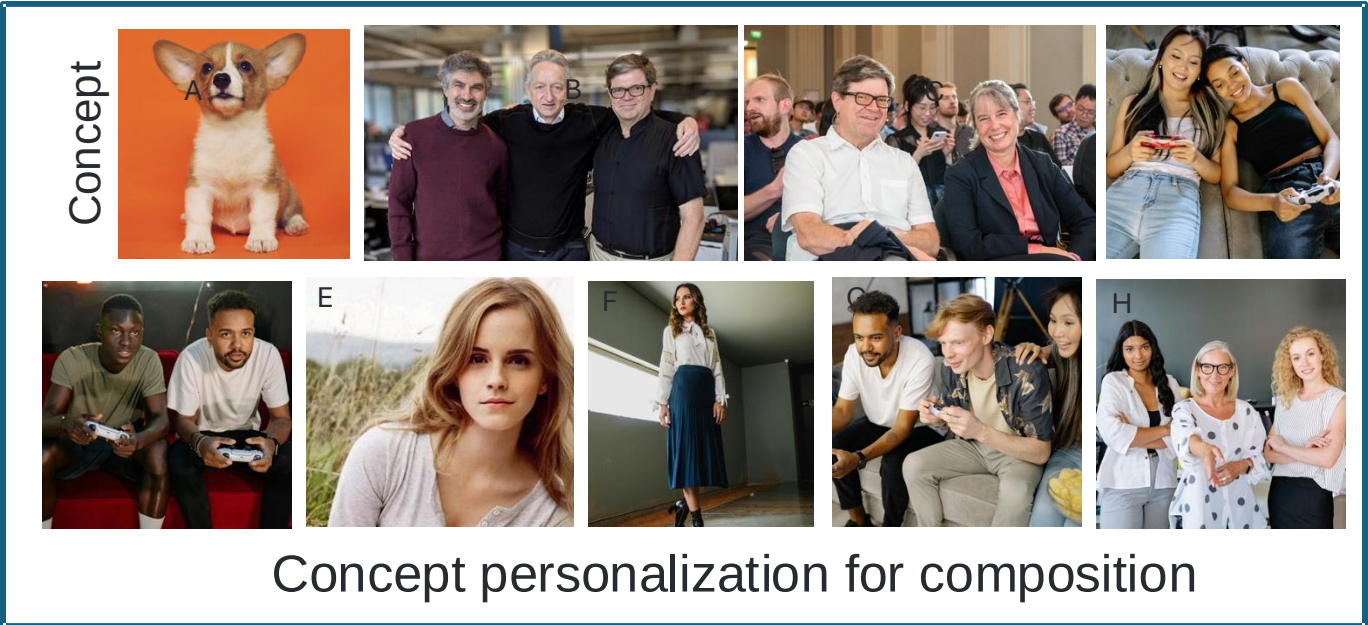


					
I = white desk+chair+computer	I+books	I+coffee cup	I+a plant beside the computer	I+papers scattered on the desk	I+a clock on the wall
					
Top view	In a pool with palm trees around	In a city at night	On a snowy mountain top	Crowded, on a beach sunset	Surrounded by autumn in forest

**Goal:** Enable efficient, new image editing and multi view editing without retraining the entire model.



# Why Efficient Fine-Tuning Matters for Text-to-Image Models?



A man in a red sweater is sitting in the library reading a book, while a woman in a white shirt next to him pets a dog. The man is shown in Image A, the woman in Image B, and the dog in the Concept.



A woman in a white long-sleeve blouse with lace details and a blue pleated skirt is walking a dog down the street. The woman is shown in Image E, the blouse and skirt in Image F, and the dog is Concept.



A man in a black shirt is reading a book and a dog (Concept) is sitting in front of his head. The man is the right man in Image D.



A dog and a short-haired woman with a wrinkled face are standing in front of a bookshelf in a library. The dog is in the Concept, and the woman is oldest woman in Image H.



A man and a woman are sitting at a classroom desk teaching a dog. The man is the man with yellow hair in Image G. The woman is the woman on the left of Image C. The dog is the one in Concept.

**Goal:** Enable efficient, new concept adaptation without retraining the entire model

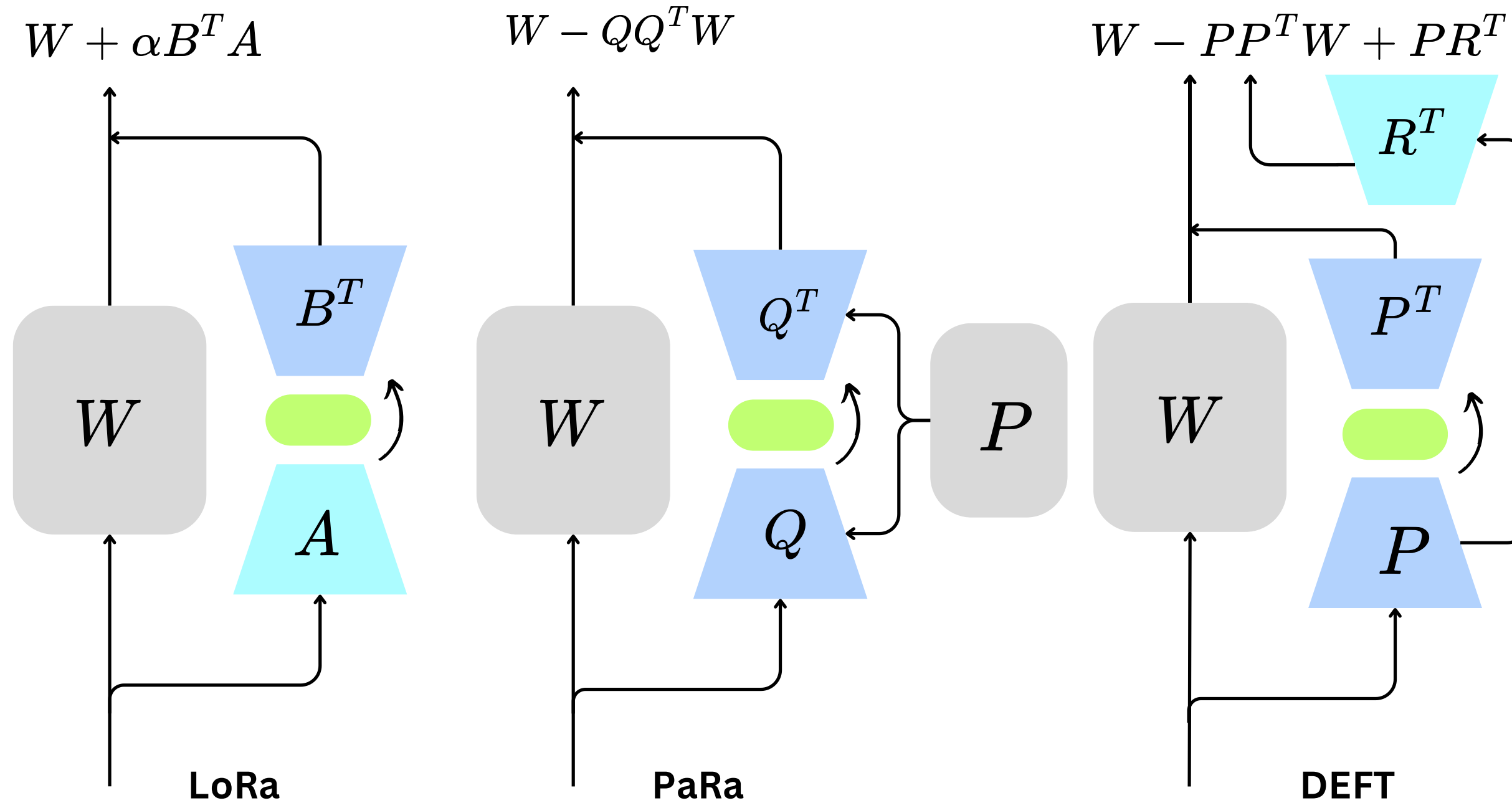
In this work, we propose a novel fine-tuning framework,

## DEFT: Compositional Efficient Fine-Tuning

- Which improve adaptability by decomposing weight updates into two components: a projection onto a low-rank subspace and a low-rank adjustment. This structure enables efficient weight editing while preserving prior knowledge.
- Supports efficient fine-tuning with minimal data, enabling personalization and adaptation to new tasks without extensive retraining or risk of overfitting.

We validate DEFT with extensive experiments on datasets such as DreamBooth, Dreambench Plus, VisualCloze, and InsDet, demonstrating its effectiveness in personalization and universal image generation.

# Comparative Diagram



- [1] Shangyu et al. "PaRa: Personalizing Text-to-Image Diffusion via Parameter Rank Reduction." ICLR 2025.  
 [2] Nataniel et al. "Dreambooth: Fine-tuning text-to-image diffusion models for subject-driven generation." CVPR 2023.

# DEFT: Method

The weight matrix  
decomposition:

$$W_{\text{total}} = (I - PP^{\top})W_0 + PR$$

1

The column space relationship\* for QR  
decomposition:

$$\text{col}(W_{\text{total}}) = \text{col}(W_{\text{reduce}}) + \text{col}(QR) \subseteq \text{col}(W_0) + \text{col}(Q)$$

2

**Condition:** The subspace is extended, enabling adaptation to new tasks:

$$\text{col}(Q) \not\subseteq \text{col}(W_0)$$

We can perform various decomposition approaches in equation (1):

QR Decomposition (orthogonality):  $P^T P = I$

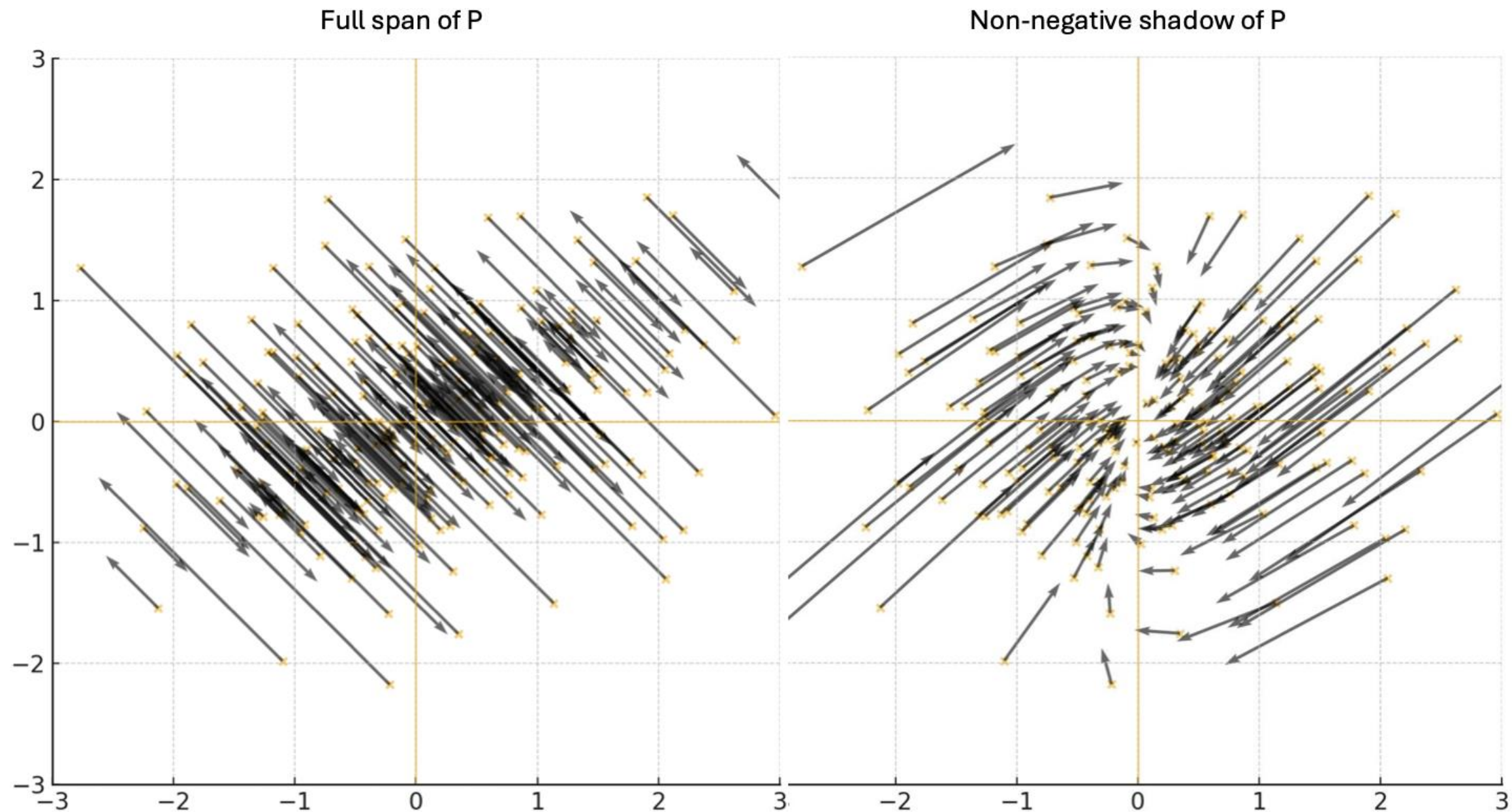
Decomposition with Truncated and Non-Truncated SVD

Non-Negative Matrix Factorization (NMF) and Low-Rank Matrix Factorization (LRMF)



# Displacement field visualization of the DEFT update

Impact of low rank non-negative matrices



**Left:** subtraction using the full span of  $P$ , where both positive and negative entries contribute to the removed subspace.  
**Right:** subtraction using only the non-negative shadow  $\text{ReLU}(P)$ , which restricts removal to additive components and yields weaker, more selective displacements.

# Results

DEFT improves Instruction following abilities

This improvement stems from DEFT’s low-rank injection, which expands the fine-tuning subspace, retaining the original model’s instruction-following capabilities, enabling more coherent image generation in personalization tasks.

Frameworks	T2I Model	CLIP-T
Textual Inversion [9]	SD v1.5	0.302
DreamBooth [34]	SD v1.5	0.323
DreamBooth LoRA [6]	SDXL v1.0	0.341
BLIP-Diffusion [24]	SD v1.5	0.286
Emu2 [40]	SDXL v1.0	0.310
IP-Adapter-Plus [52] ViT-H	SDXL v1.0	0.282
IP-Adapter [52] ViT-G	SDXL v1.1	0.309
PaRa [4]	SDXL v1.0	0.354
DreamBooth DEFT (Ours)	SDXL v1.0	<b>0.361</b>



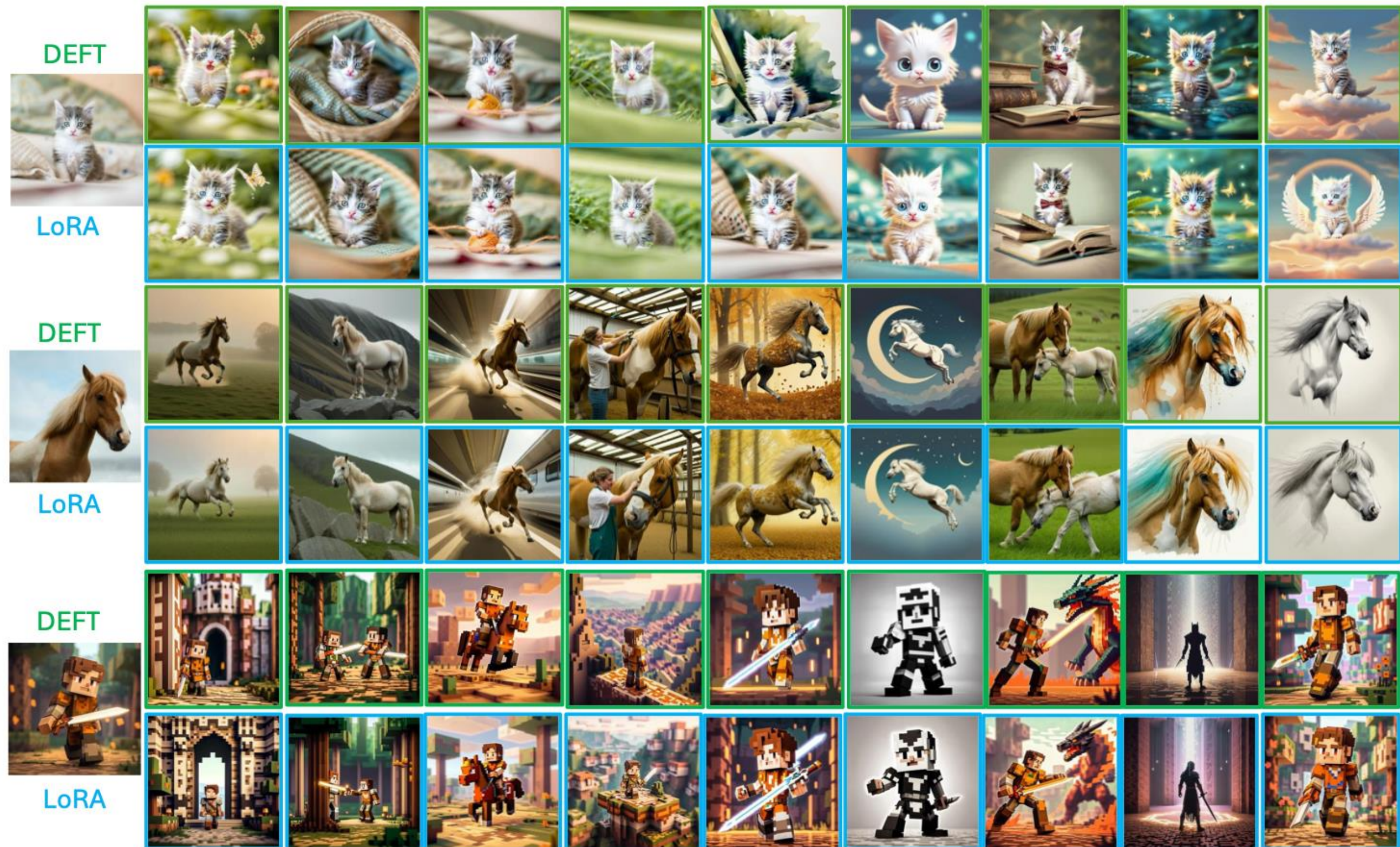
# Universal Image Generation through Adapter

Beyond personalization, we assess DEFT’s generalization by fine-tuning OmniGen.

Condition	Method	CLIP-Score	Image Consistency	
		Image	DINOv1	DINOv2
Canny	OmniGen	95.45	87.13	87.60
	VisualCloze	89.32	–	–
	<b>DEFT (Ours)</b>	<b>95.78</b>	<b>90.37</b>	<b>90.65</b>
Depth	OmniGen	92.02	85.16	77.39
	VisualCloze	87.56	–	–
	<b>DEFT (Ours)</b>	<b>93.18</b>	88.98	<b>85.75</b>
Style Type	Method	Text Score(↑)	Image Score(↑)	F1(↑)
InstantStyle	InstantStyle [44]	0.27	0.60	0.55
	OmniGen [48]	0.27	0.52	
	VisualCloze-dev [25]	0.30	0.53	
	VisualCloze-fill [25]	<b>0.29</b>	0.55	
	<b>DEFT (Ours)</b>	0.28	<b>0.69</b>	<b>0.59</b>
ReduxStyle	OmniGen [48]	0.27	0.58	0.47
	VisualCloze-dev [25]	<b>0.29</b>	0.53	
	VisualCloze-fill [25]	0.27	0.55	
	<b>DEFT (Ours)</b>	0.26	<b>0.69</b>	<b>0.49</b>

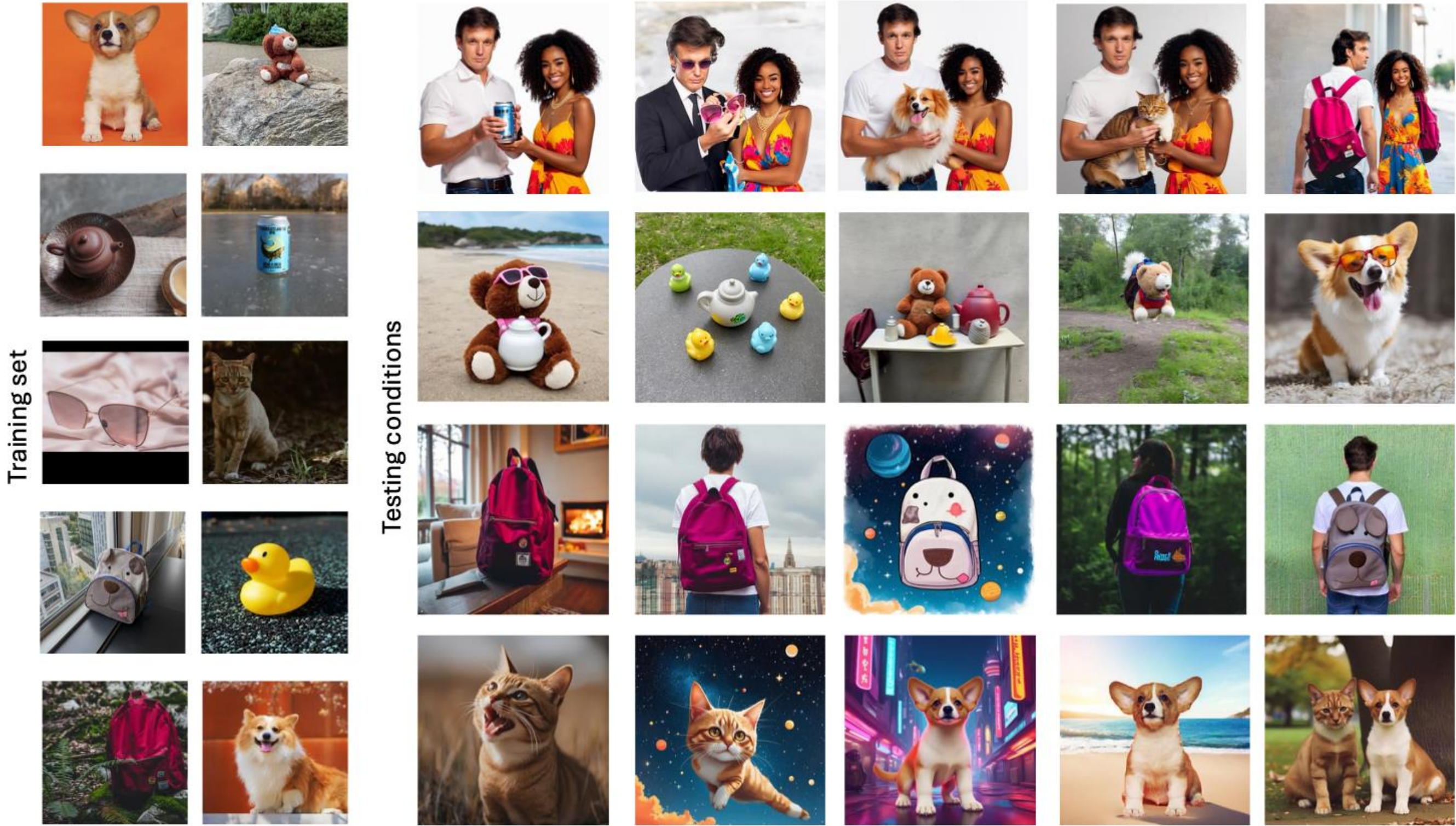


# Quality Check



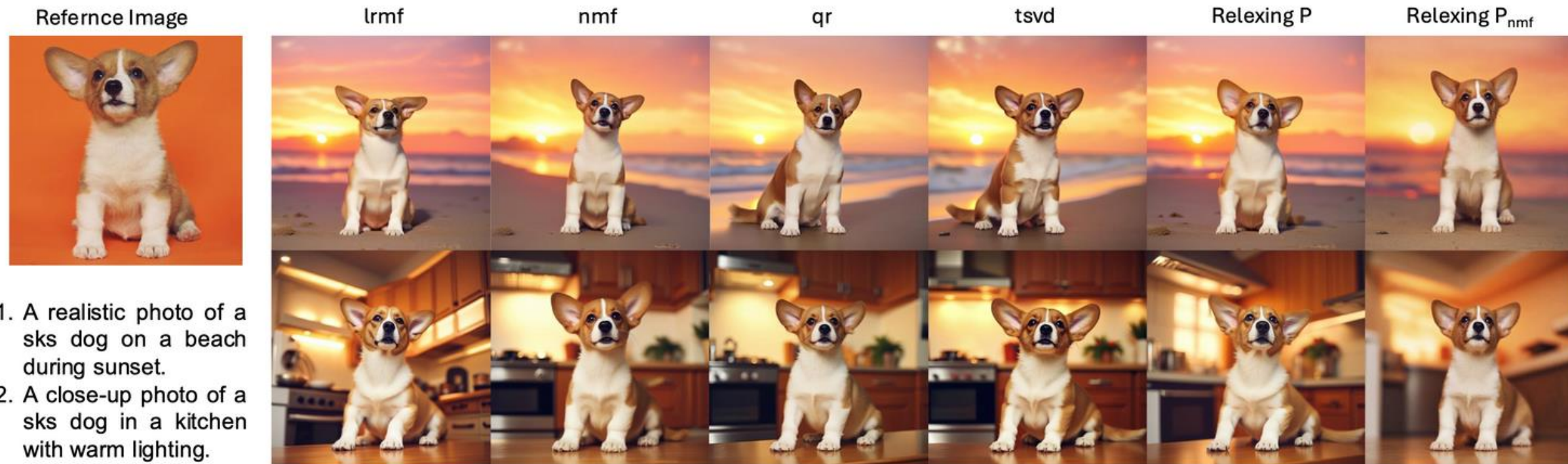


# Emergent properties





# Effect of decomposition



Method	Speed (ms)	CLIP-I	CLIP-T	DINO-V1	Aesthetic	Sharpness
LRMF	29.10	$0.827 \pm 0.038$	$0.220 \pm 0.051$	$0.307 \pm 0.040$	$0.014 \pm 0.005$	$349 \pm 414$
NMF	5.16	$0.883 \pm 0.033$	$0.222 \pm 0.042$	$0.357 \pm 0.068$	$0.015 \pm 0.003$	$206 \pm 91$
QR	5.38	$0.875 \pm 0.043$	$0.217 \pm 0.041$	$0.340 \pm 0.060$	$0.017 \pm 0.003$	$215 \pm 106$
TSVD	28.72	$0.875 \pm 0.031$	$0.223 \pm 0.041$	$0.333 \pm 0.062$	$0.016 \pm 0.004$	$331 \pm 134$
Relxing P	5.22	$0.879 \pm 0.030$	$0.235 \pm 0.041$	$0.330 \pm 0.058$	$0.015 \pm 0.003$	$340 \pm 150$
Relxing P <sub>nmf</sub>	5.22	$0.923 \pm 0.037$	$0.266 \pm 0.033$	$0.440 \pm 0.096$	$0.016 \pm 0.003$	$175 \pm 25$



# Training Steps

Method	Training Steps = 2000				Training Steps = 8000			
	CLIP-I	CLIP-T	Aesthetic	Sharpness	CLIP-I	CLIP-T	Aesthetic	Sharpness
DEFT	$0.811 \pm 0.066$	$0.302 \pm 0.026$	$0.015 \pm 0.005$	$294 \pm 261$	$0.882 \pm 0.059$	$0.319 \pm 0.025$	$0.015 \pm 0.003$	$320 \pm 364$
LORA	$0.836 \pm 0.039$	$0.286 \pm 0.033$	$0.016 \pm 0.004$	$449 \pm 334$	$0.876 \pm 0.024$	$0.219 \pm 0.023$	$0.013 \pm 0.002$	$65 \pm 3$

# Camera-Aware Generation

	CP-I	CP-T	DO-V1	Sharp
Pre	0.475	0.201	0.220	1179
DEFT <sub>INS</sub>	0.647	0.210	0.350	1813
DEFT <sub>SFM</sub>	<b>0.660</b>	<b>0.213</b>	<b>0.343</b>	<b>1852</b>

# Conclusion

- We introduced DEFT, a novel framework for fine-tuning large pretrained models.
- DEFT improves the model's adaptability by decomposing weight updates into two key components: a projection onto a low-rank subspace and a low-rank update.
- We demonstrated that T2I fine-tuning can be efficient without losing the capabilities.
- Futurework can explore its abilities to overcome catastrophic forgetting in LLMs.



*Thank You*

