StyleCrafter: Taming Stylized Video Diffusion with Reference-Augmented Adapter Learning

GONGYE LIU, Tsinghua University, China
MENGHAN XIA*, YONG ZHANG, and HAOXIN CHEN, Tencent AI Lab, China
JINBO XING, The Chinese University of Hong Kong, China
YIBO WANG, Tsinghua University, China
XINTAO WANG and YING SHAN, Tencent AI Lab, China
YUJIU YANG*, Tsinghua University, China

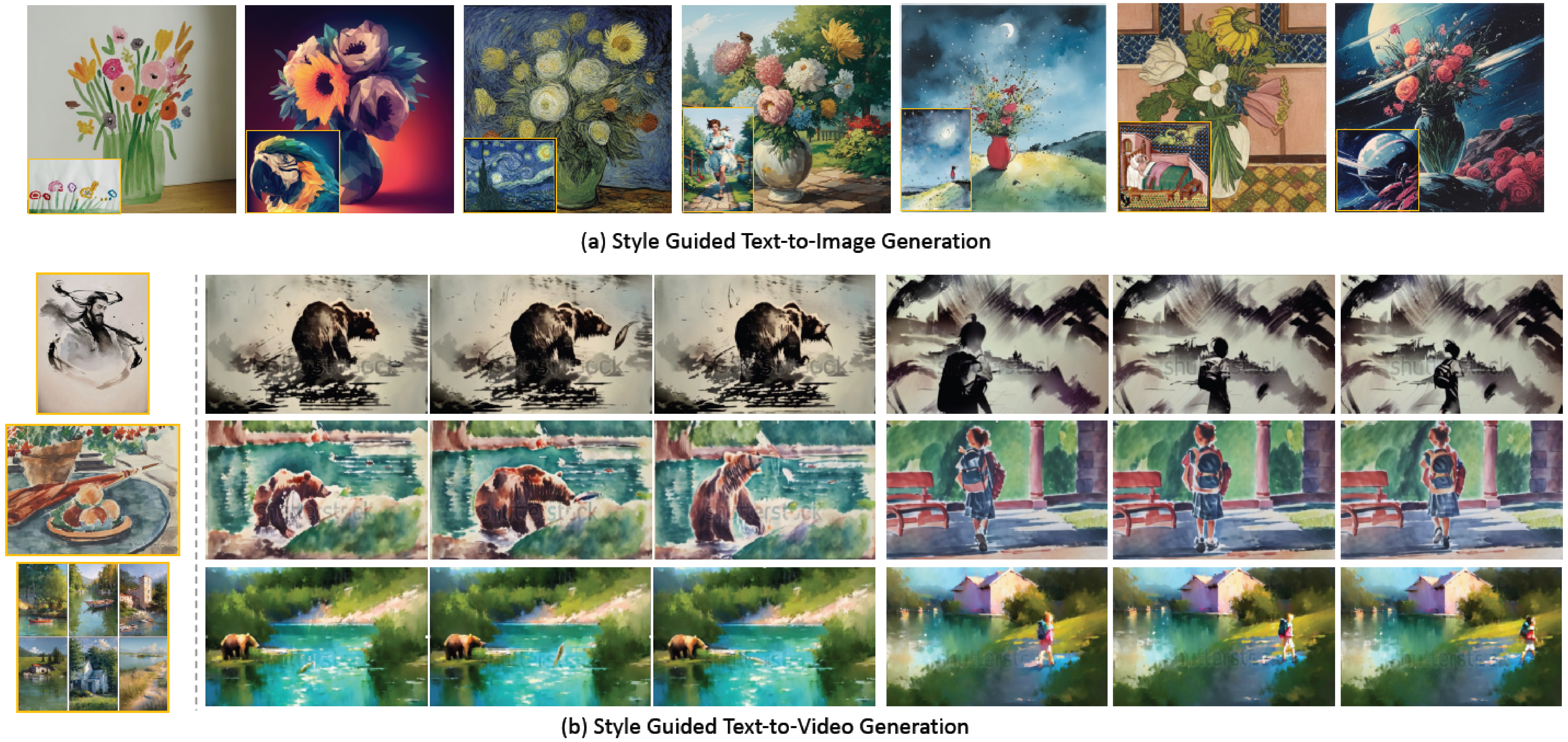


Fig. 1. Stylized Generation Results Produced by StyleCrafter

Points

- Stylized video dataset 부족
- Style-Content Decoupling에 집중

- Pre-trained T2V model 활용 (Add style adapter)
- Two-stage Training Strategy

1. Model

Inputs

- **text prompt**: content
- **style image(s)**: style reference

Style Adapter

- style feature extractor
- dual cross-attention module
- context-aware scale factor predictor

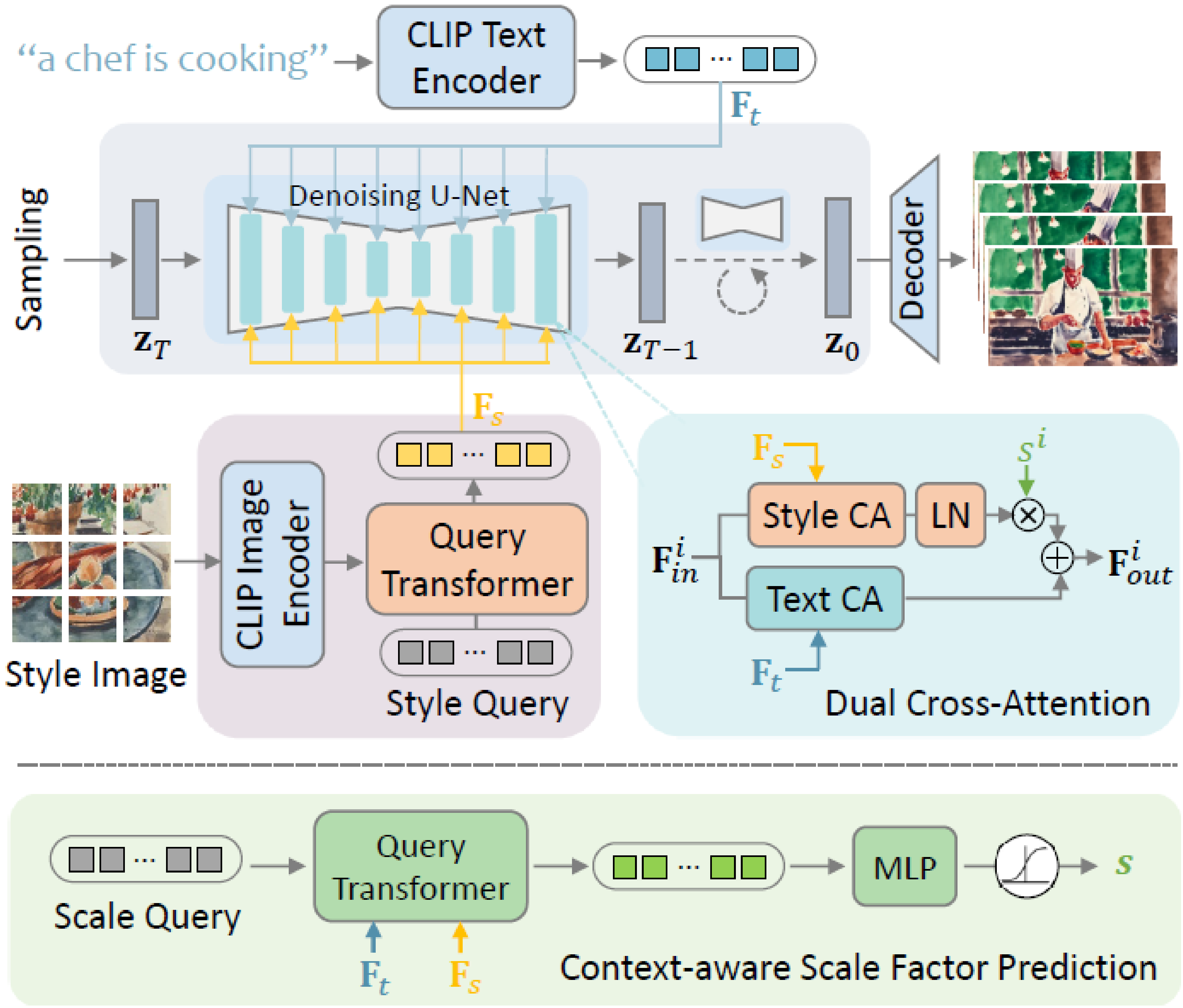


Fig. 2. Overview of our proposed style adapter. It consists of three components, i.e. style feature extractor, dual cross-attention module, and context-aware scale factor predictor.

Pre-trained T2V Model

Style Adapter

1. Model

Style Feature Extractor

style ref. image

CLIP Image Encoder

→ Global semantic & full local tokens

Q-former (Query Transformer)

→ F_s

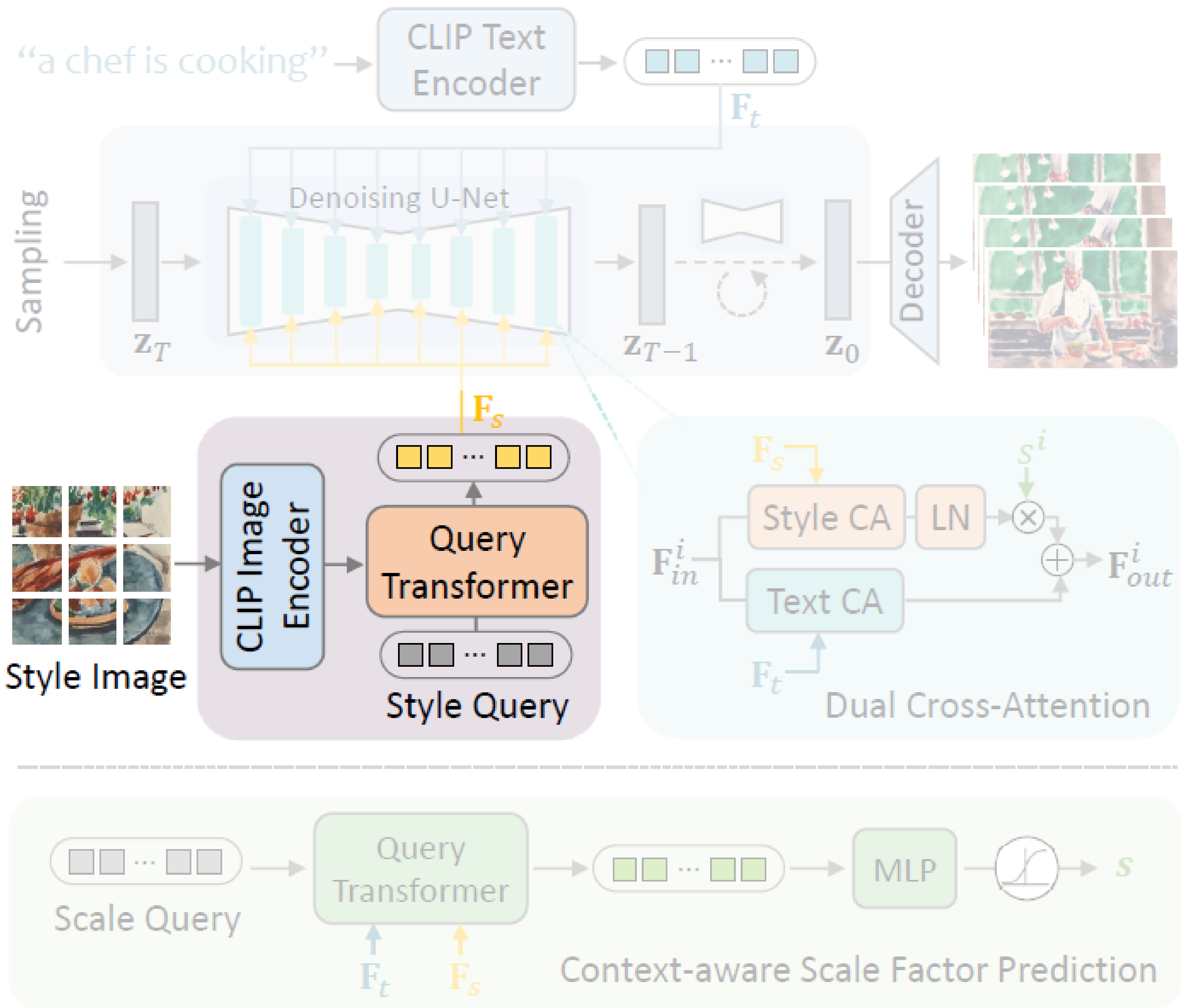


Fig. 2. Overview of our proposed style adapter. It consists of three components, i.e. style feature extractor, dual cross-attention module, and context-aware scale factor predictor.

1. Model

Dual Cross-attention Module

Denoising U-Net에서, style embedding을 위한 새로운 cross-attention module을 추가, text feature + style feature ⇒ F_out

* **attach-to-text**: text embedding에 style embedding을 붙여서 기존 cross-attention에 입력

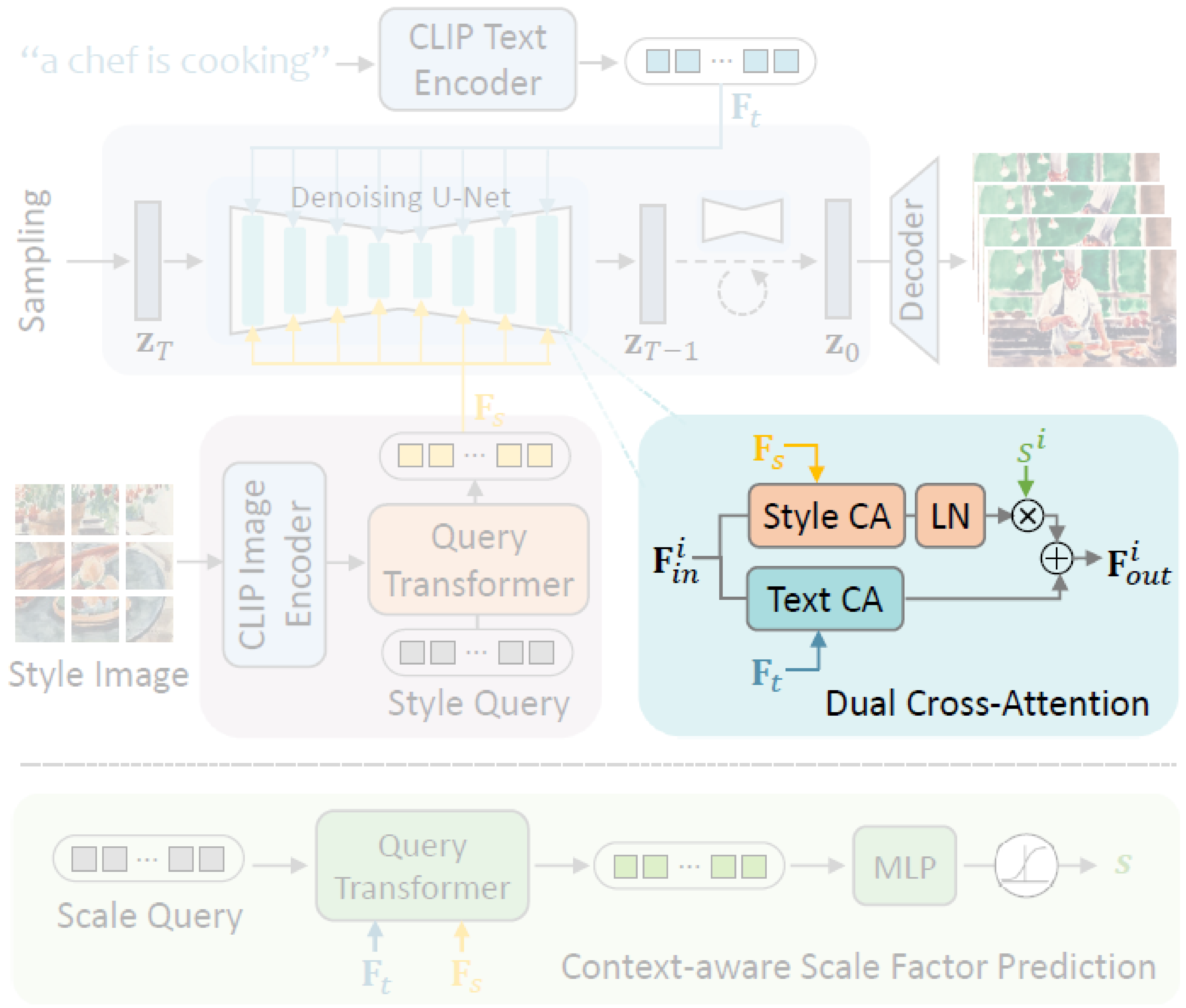


Fig. 2. Overview of our proposed style adapter. It consists of three components, i.e. style feature extractor, dual cross-attention module, and context-aware scale factor predictor.

1. Model

Context-Aware Scale Factor Predictor

Text feature와 Style feature를 합칠 때, scale을 조절하는 “scale factor prediction network”를 학습시킴.

$$\mathbf{F}^i_{out} = \underbrace{\mathrm{TCA}(\mathbf{F}^i_{in}, \mathbf{F}_t)}_{\text{text-based cross attention}} + \underbrace{s^i}_{\text{scale factor}} * \mathrm{LN}(\underbrace{\mathrm{SCA}(\mathbf{F}^i_{in}, \mathbf{F}_s)}_{\text{style-based cross attention}}),$$

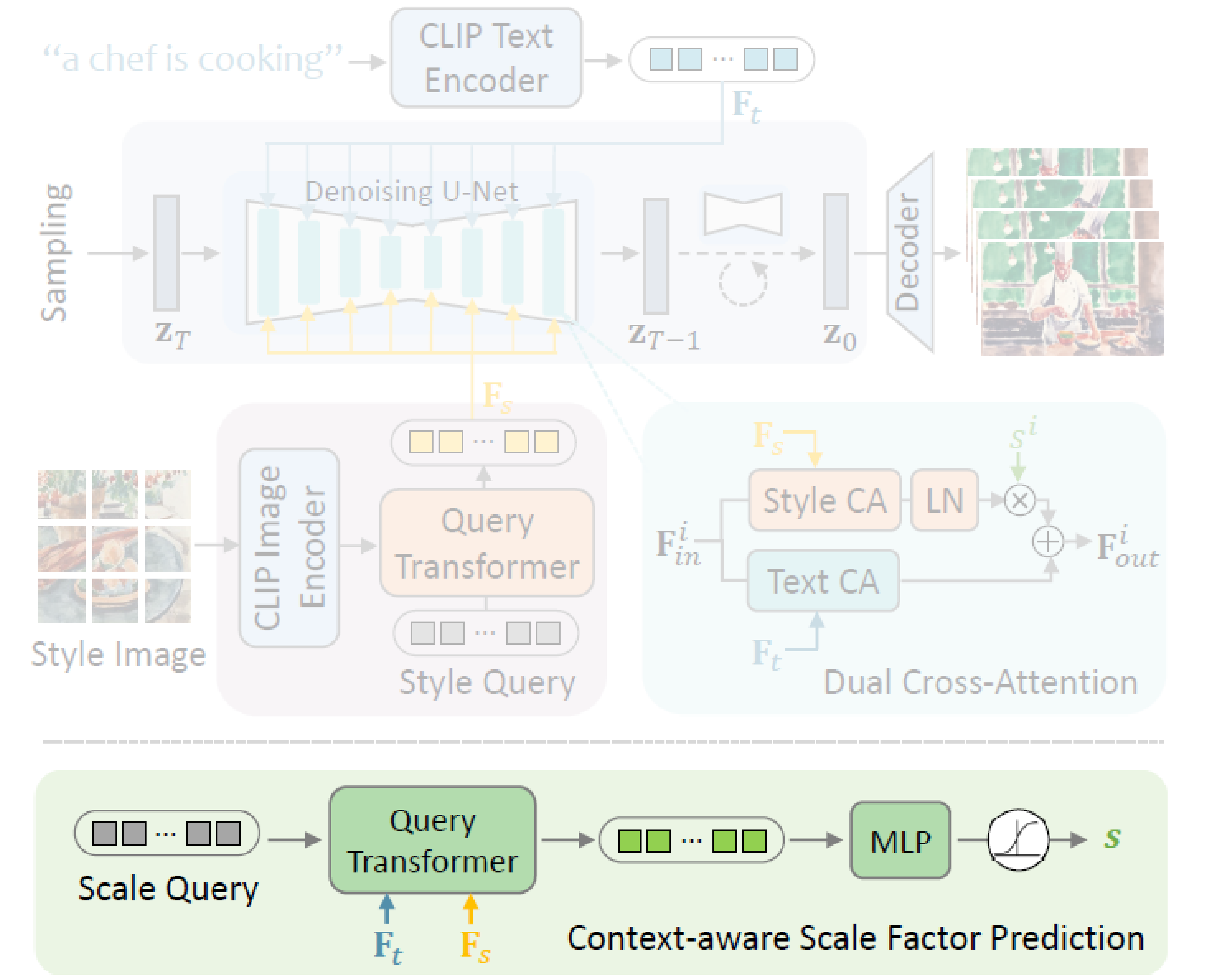


Fig. 2. Overview of our proposed style adapter. It consists of three components, i.e. style feature extractor, dual cross-attention module, and context-aware scale factor predictor.

1. Model

Context-Aware Scale Factor Predictor

Text feature와 Style feature를 합칠 때, scale을 조절하는 “scale factor prediction network”를 학습시킴.

$$\mathbf{F}^i_{out} = \mathrm{TCA}(\mathbf{F}^i_{in}, \mathbf{F}_t) + s^i * \mathrm{LN}(\mathrm{SCA}(\mathbf{F}^i_{in}, \mathbf{F}_s)),$$

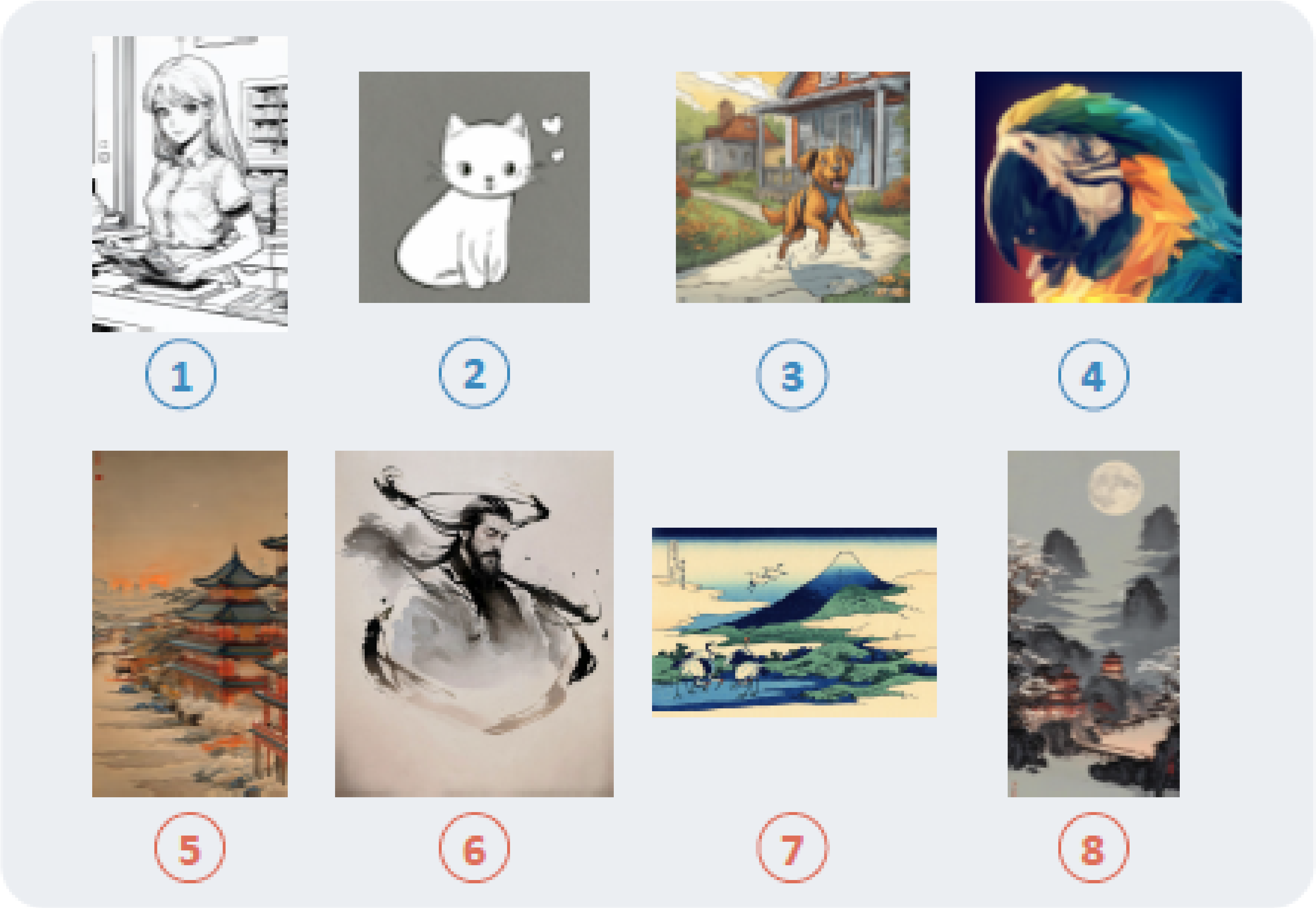
text-based cross attention

scale factor

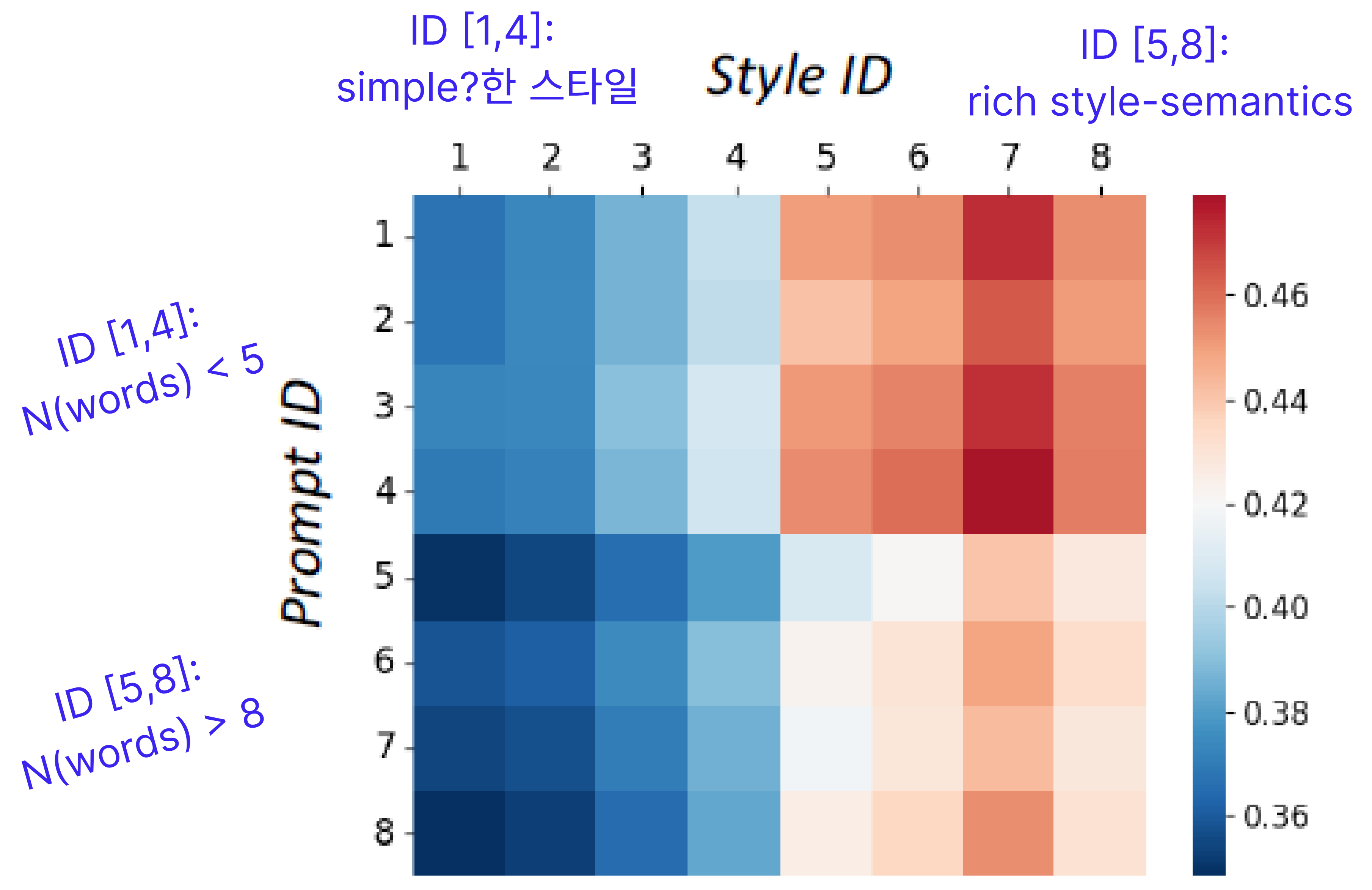
style-based cross attention

rich style → higher scales

complex prompt → lower scale factors



(a) Style References



(b) Scale Factors across various pairs

2. Two Stage Training Strategy

Base T2V Model

VideoCrafter

Step 1) Style Adapter Training

Style image dataset으로
Style Adapter 학습

Dataset: WikiArt, Laion-Aesthetics-6.5

T2V에 적용하면, 시간에 따라 떨리는 현상 발생
⇒ T2V model의 temporal self-attention finetuning 필요 (*step 2*)

Step 2) Temporal blocks Finetuning

Temporal blocks of VideoCrafter 학습,
나머지 파트는 frozen

jointly train image datasets & video datasets (subset of WebVid-10M)

3. **Results -** Single-Reference Style Guided

VideoComposer
- style reference의 content를 가져온다. (*invalid style-content decoupling*)
- 움직임이 거의 없음.

VideoCrafter
- style 반영 미흡

Ours
- style을 잘 반영한 좋은 결과

Fig. 5. Visual comparison of single-reference guided T2V generation. Vid.Comp.: VideoComposer, Vid.Craf.: VideoCrafter

Methods	CLIP-Text ↑	CLIP-Style ↑	Temporal Consistency	
			CLIP-Temp ↑	W.E.($\times 10^{-3}$) ↓
VideoComposer	0.0468	**0.7306**	0.9853	**9.903**
VideoCrafter*	0.2209	0.3124	0.9757	61.41
Ours	**0.2726**	0.4531	**0.9892**	18.73

3. **Results -** Multi-Reference Style Guided

AnimateDiff

- Close-to-realism style
- temporal artifacts

Ours (Multi-ref.)

- 시간적 일관성
- style, context 모두 잘 반영
- Ours (Single-ref.)보다 좋은 성능

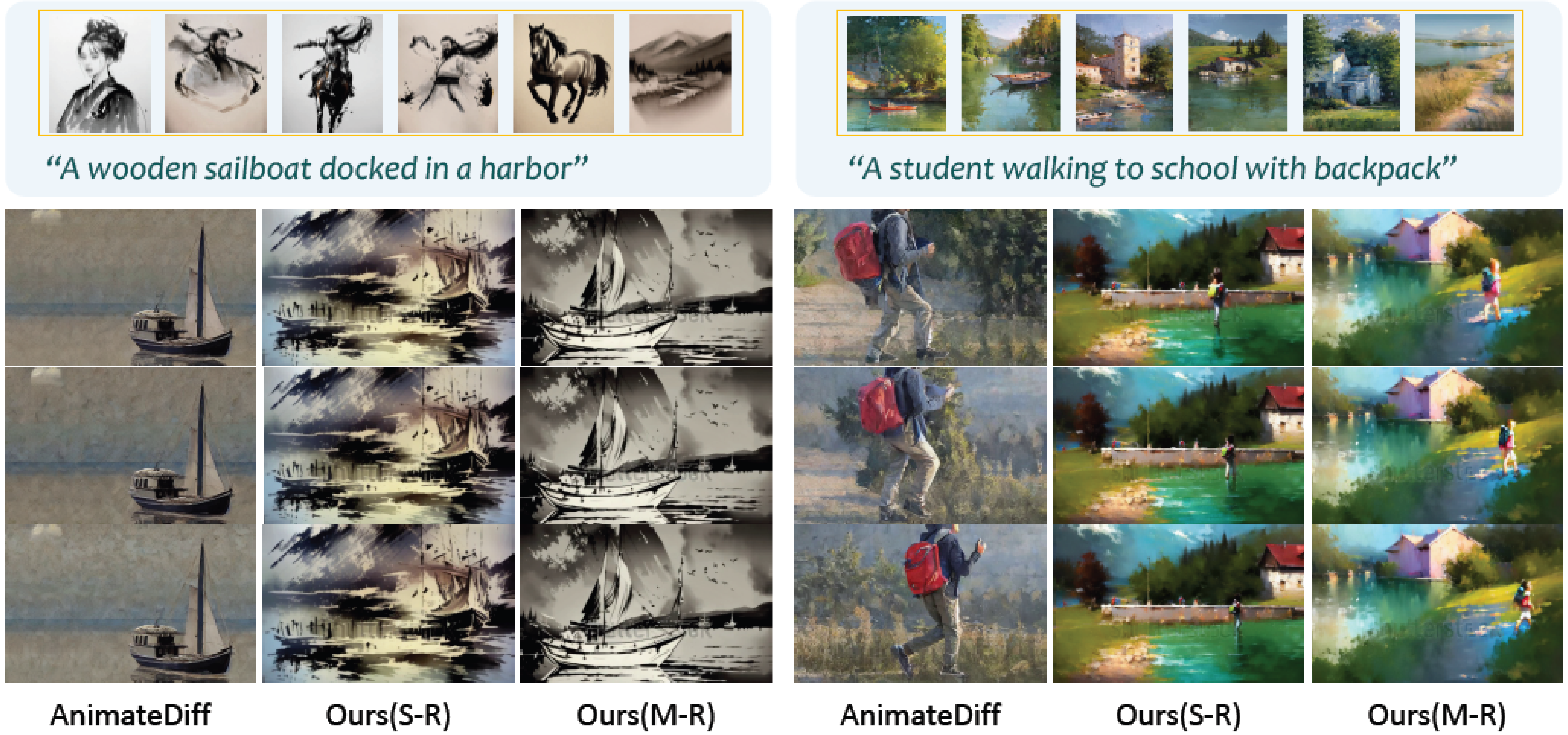


Fig. 6. Qualitative comparison of multi-reference style-guided T2V generation. S-R: Single-Reference, M-R: Multi-Reference

4. Ablation Study

Dual Cross Attention

attatch-to-text(하나의 cross attention에 text와 style features 입력)
⇒ content와 style을 분리하지 못함.

(i) *"A bear fishing in a river"*

(ii) *"A bouquet of flowers in a vase"*

Table 4. Ablation studies on style modulation designs. The performance is evaluated based on the style-guided T2I generation.

	Methods	CLIP-Text ↑	CLIP-Style ↑
dual cross attention	Ours	0.3028	0.4836
	w/o Data Augmentation	0.3173	0.4005
attach-to-text	w/o Dual Cross Attention	0.0983	0.7332
	w/o Adaptive Fusion	0.2807	0.4925

(i)
(ii)

w/o Dual CA | w/o Data Aug | Ours

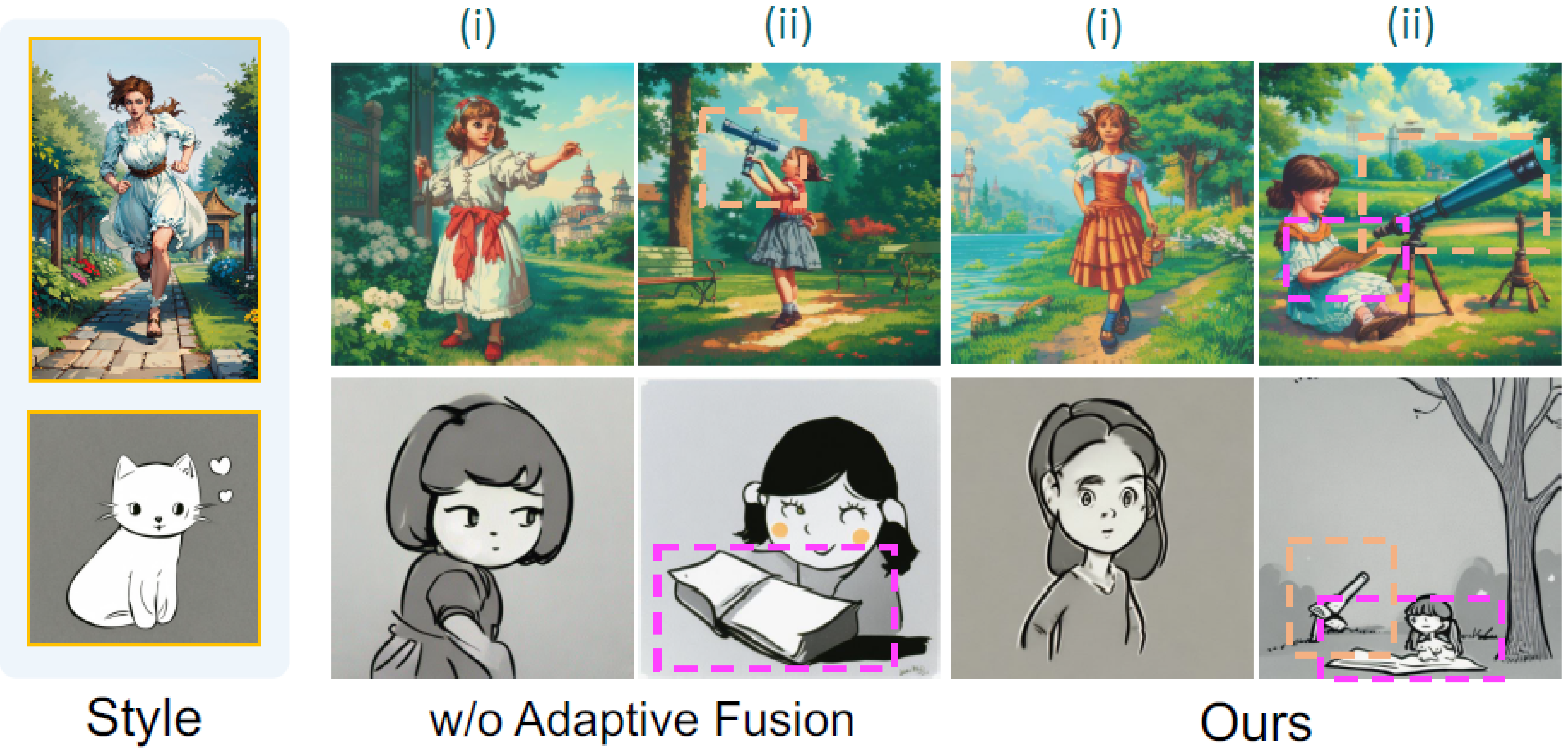
attach-to-text | dual cross attention

4. Ablation Study

Adaptive Style-Content Fusion

content(text)와 image(style)의 features를 더하는 scale 조절.

없으면, text prompt가 길 때, content가 일부 손실된다.



Text Prompts

(i) A little girl

(ii) A little girl ***reading a book*** in the park,
with ***a telescope*** nearby pointed at the sky

Table 4. Ablation studies on style modulation designs. The performance is evaluated based on the style-guided T2I generation.

Methods	CLIP-Text ↑	CLIP-Style ↑
Ours	0.3028	0.4836
w/o Data Augmentation	0.3173	0.4005
w/o Dual Cross Attention	0.0983	0.7332
w/o Adaptive Fusion	0.2807	0.4925

4. Ablation Study

Two-Stage Training Scheme

Style adapter training → Temporal block finetuning

(i) Without Temporal block Finetuning (only style adapter training) → temporal consistency 성능 감소

(ii) Joint Training (style adapter & temporal blocks 동시에) → style embedding extraction (style adapter) 성능 감소

Table 5. Ablation study on our two-stage training scheme.

Methods	CLIP-Text ↑	CLIP-Style ↑	Temporal Consistency	
			CLIP-Temp ↑	W.E.($\times 10^{-3}$) ↓
w/o Temporal Adaption	0.2691	0.3923	0.9612	47.88
Joint Training	0.3138	0.2226	0.9741	24.74
Two-Stage(ours)	**0.2726**	**0.4531**	**0.9892**	**18.73**

Thank You

Classifier-Free Guidance for Multiple Conditions

$$\hat{\epsilon}(z_t, c_t, c_s) = \epsilon(z_t, \varnothing) + \lambda_s(\epsilon(z_t, c_t, c_s) - \epsilon(z_t, c_t)) + \lambda_t(\epsilon(z_t, c_t) - \epsilon(z_t, \varnothing)), \tag{2}$$

3. Results - T2I

Table 1. Quantitative comparison on single-reference style-guided T2I generation. We conduct evaluation on a test set of 400 pairs. **Bold**: Best.

Method	Stable Diffusion 2.1 based				SDXL based			
	Dreambooth	InST	SD*	Ours	IP-Adapter-Plus	Style-Aligned	SDXL*	Ours(SDXL)
CLIP-Text ↑	**0.3047**	0.3004	0.2766	0.3028	0.2768	0.2254	0.2835	**0.2918**
CLIP-Style ↑	0.3459	0.3708	0.4183	**0.4836**	0.5182	0.5515	0.4348	**0.5615**
DINO-Style ↑	0.2278	0.2587	0.2890	**0.3652**	0.4367	0.4395	0.2912	**0.4514**

Style Reference (a) DreamBooth (b) InST (c) IP-Adapter (d) Style Aligned (e) SD* (f) SDXL* (g) Ours(SD 2.1) (h) Ours(SDXL)

Fig. 4. Visual comparison on style-guided T2I generation. Blue: methods based on SD 2.1. Green: based on SDXL. Prompt: A rabbit nibbling on a carrot.

5. Limitations

•

여백 등 디테일한 style 반영 부족

데이터 부족의 한계가 여전히 존재 (전체적인 성능의 아쉬움?)