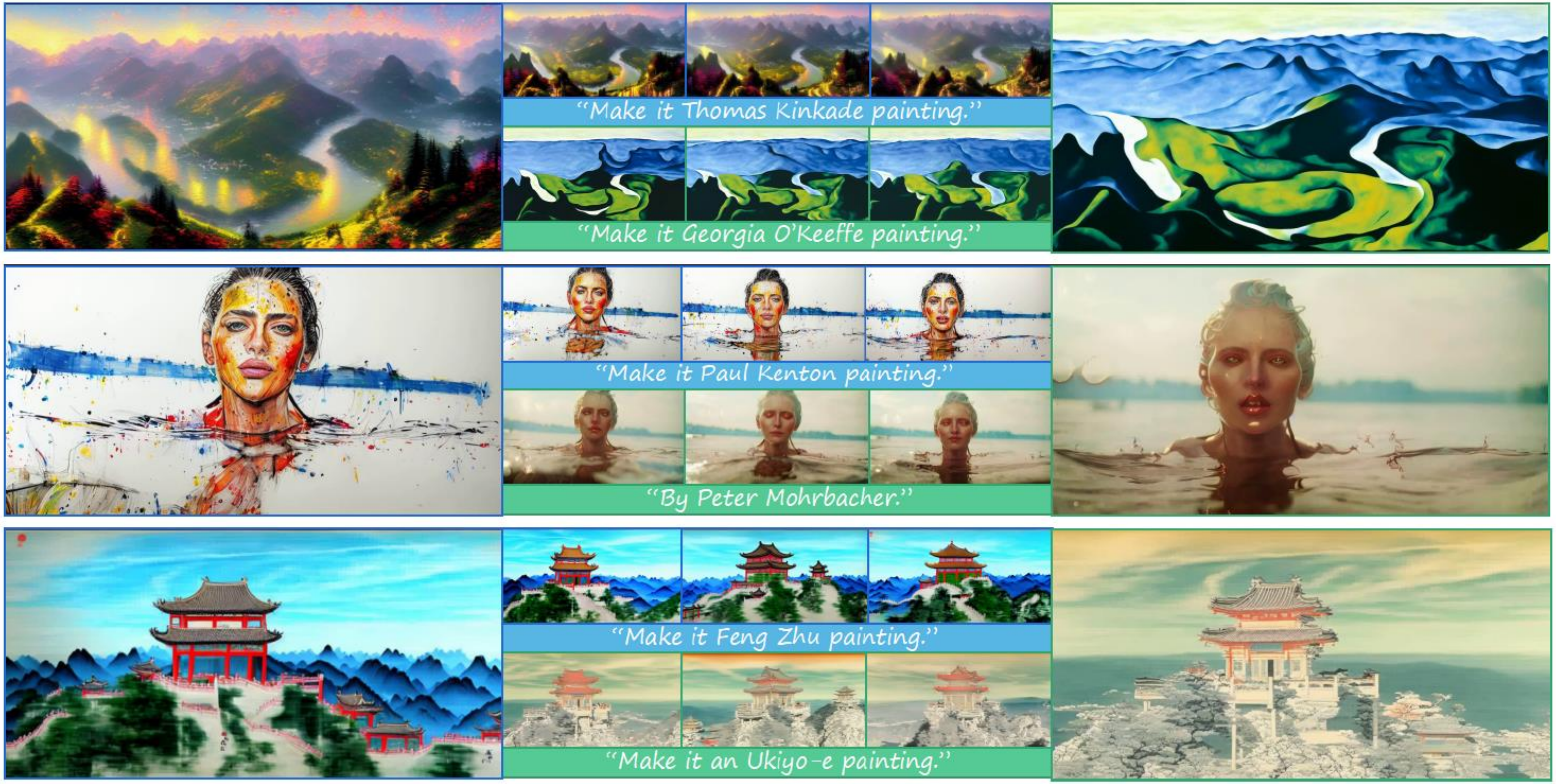
Style-A-Video : Agile Diffusion for Arbitrary Text-based Video Style Transfer

Nisha Huang[1,2] Yuxin Zhang[1,2] Weiming Dong[1,2]
[1]School of Artificial Intelligence, UCAS [2]MAIS, Institute of Automation, CAS
https://github.com/haha-lisa/Style-A-Video

Introduction



Introduction

task : text-to-video diffusion models used for stylization

difficulties :

- lack of extensive text-to-video datasets and necessary computational resources for training
- input video content is frequently tough to retain
  - the noise addition process on the input content is random and destructive

Introduction

target :

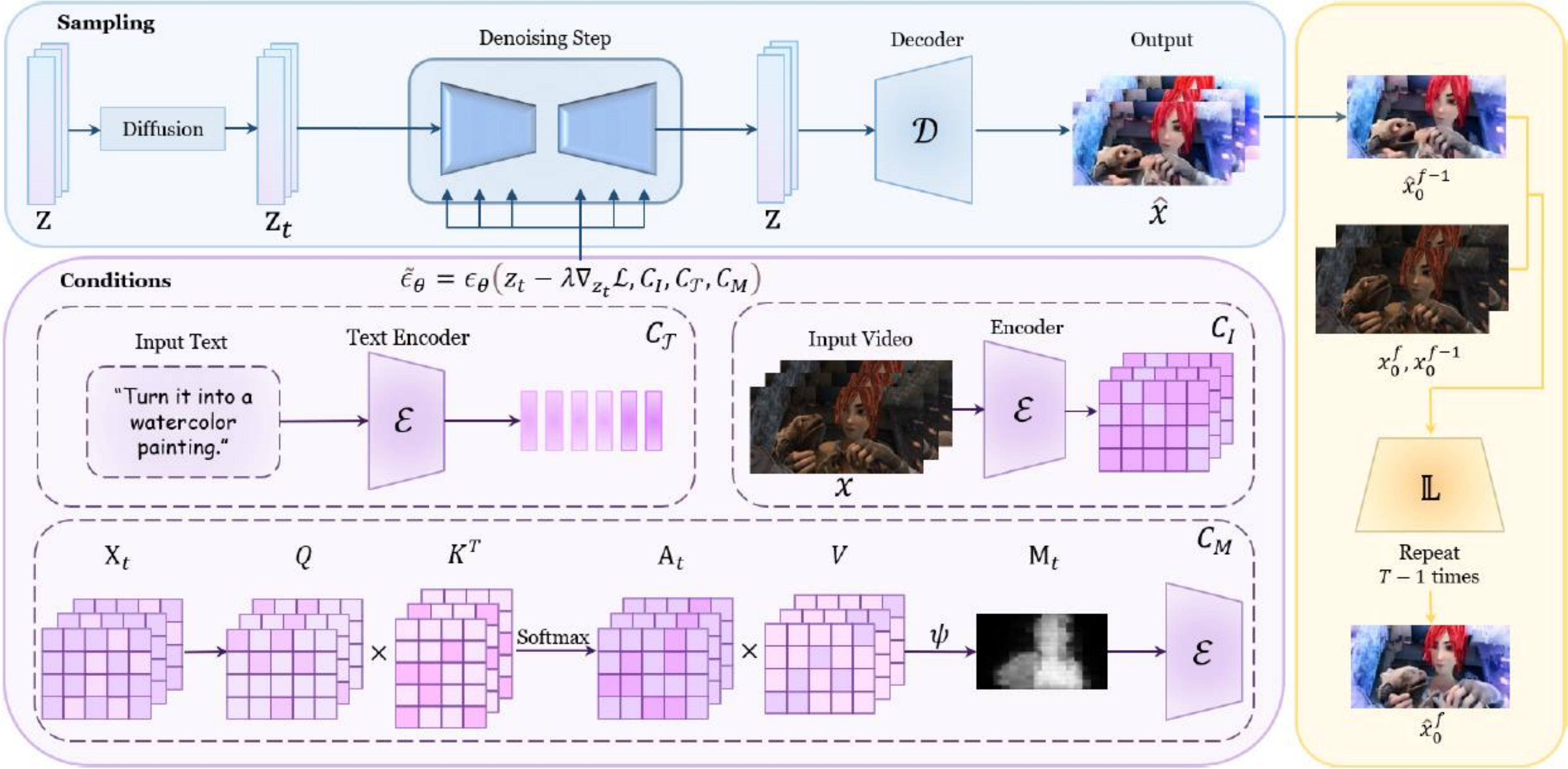
1. requirements of the stylization task
    - accomplish stylistic representation of text prompt
    - preservation of input video content
    - inter-frame consistency
2. fast optimization of the inference process

Introduction

contribution :

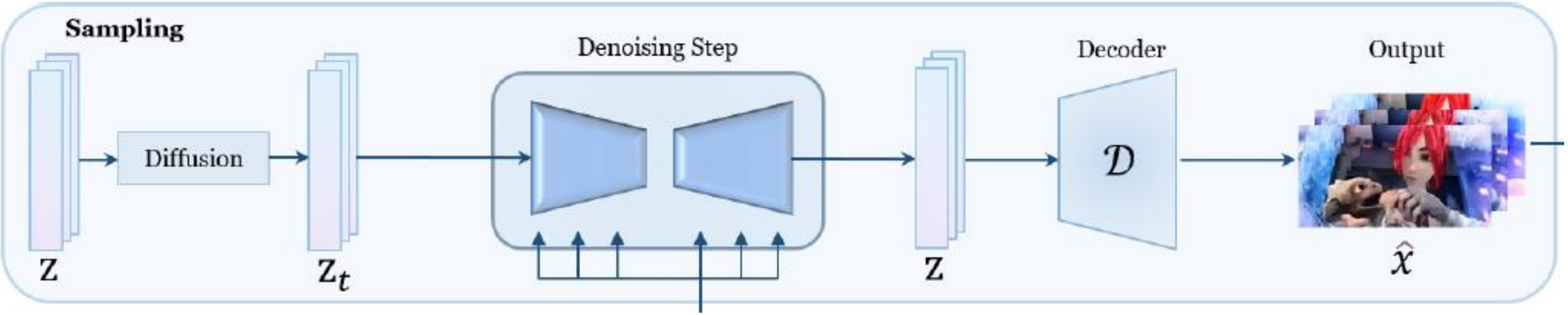
1. This work is performed entirely in inference time without additional per-video training or fine-tuning
2. Novel noise prediction guiding formulas are proposed to achieve simultaneous control of style, content, and structure
3. achieve the control of time and content consistency in the inference process

Method



Sampling
Diffusion
Denoising Step
Decoder
Output
Z
Z_t
Z
$\mathcal{D}$
$\hat{x}$
$\hat{x}_0^{f-1}$
x_0^f, x_0^{f-1}
Conditions
$\tilde{\epsilon}_\theta = \epsilon_\theta(z_t - \lambda \nabla_{z_t} \mathcal{L}, C_I, C_\mathcal{T}, C_M)$
Input Text
Text Encoder
$C_\mathcal{T}$
"Turn it into a watercolor painting."
$\mathcal{E}$
Input Video
Encoder
C_I
x
$\mathcal{E}$
C_M
X_t
Q
K^T
A_t
V
M_t
Softmax
×
×
ψ
$\mathcal{E}$
$\mathbb{L}$
Repeat
$T-1$ times
$\hat{x}_0^f$

Diffusion Model



Latent Diffusion Models (LDMs)

- improve the efficiency and quality of diffusion models by operating in the latent space of a pre-trained variational autoencoder
- We learn a network θ that predicts the noise added to the noisy latent zt given image condition cI , text prompt condition cT, and attention map condition cM

$$L = \mathbb{E}_{\mathcal{E}(x), c_I, c_{\mathcal{T}}, c_M, \epsilon \sim \mathcal{N}(0,1), t} \left[\| \epsilon - \epsilon_\theta \left(z_t, t, c_I, c_{\mathcal{T}}, c_M \right) \|_2^2 \right].$$

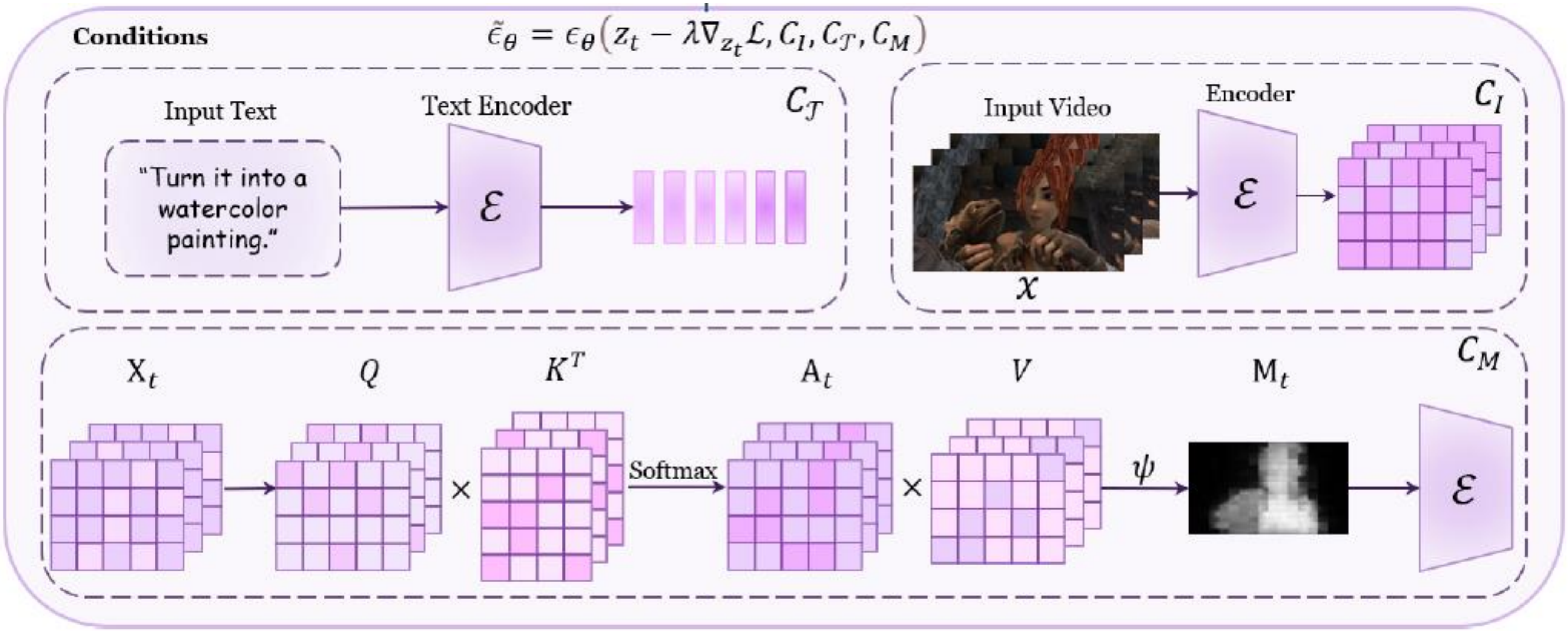
Diffusion Model

Algorithm 1 Conditions Guidance Diffusion Sampling.

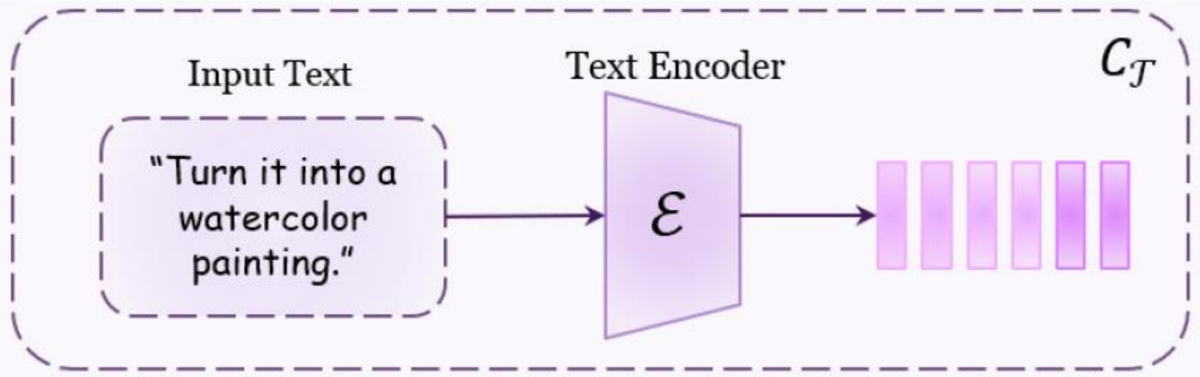
Input: The text prompt $\mathcal{T}$, video frame I,
frame number F, Diffusion model Model(z_t).
Output: Stylized frames $x_0^1, ..., x_0^F$.
$\mathrm{z}_T \sim \mathcal{N}(0, \mathrm{I})$ a unit Gaussian random variable with specific seed S

for f in $1, ..., F$ **do**
$x_t^f \leftarrow noising(x_0^f)$
$z_t = \mathcal{E}(x_t^f)$
  for t in $T, ..., 1$ **do**
    $\epsilon, \Sigma, M \leftarrow \text{Model}\,(\mathrm{z}_t)$
    $\Delta z_t = \nabla_{z_t} \mathcal{L}_s$
    $C_I, C_{\mathcal{T}}, C_M \leftarrow \text{CLIP embedding}\,(I, \mathcal{T}, M)$
    $\tilde{\epsilon}_\theta = \epsilon_\theta(z_t - \lambda \Delta z_t, C_I, C_{\mathcal{T}}, C_M)$
    $\mathrm{z}_{t-1} \sim \mathcal{N}\left(\frac{1}{\sqrt{\bar{\alpha}_t}}\left(\mathrm{z}_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\tilde{\epsilon}\right), \Sigma\right)$
  end for
  return z_0
  $\hat{x}_0^f = \mathcal{D}(\mathrm{z}_0)$
  if $f = 1$
    $\hat{x}_0^{f-1} = \varnothing$
  else
    $\hat{x}_0^f \leftarrow x_0^f, \hat{x}_0^{f-1}$
end for
return x_0^f

Condition Representation



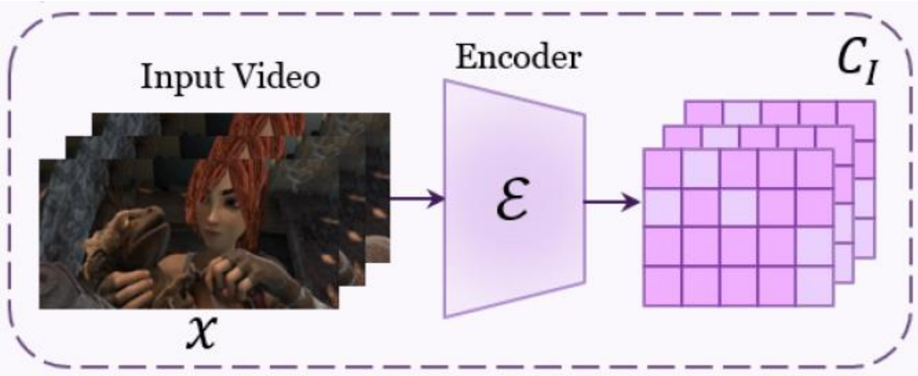
Style condition representation



- the forward process q remains unchanged while the conditioning variables c become additional inputs to the model
- replace the category condition with the textual prompt description during sampling

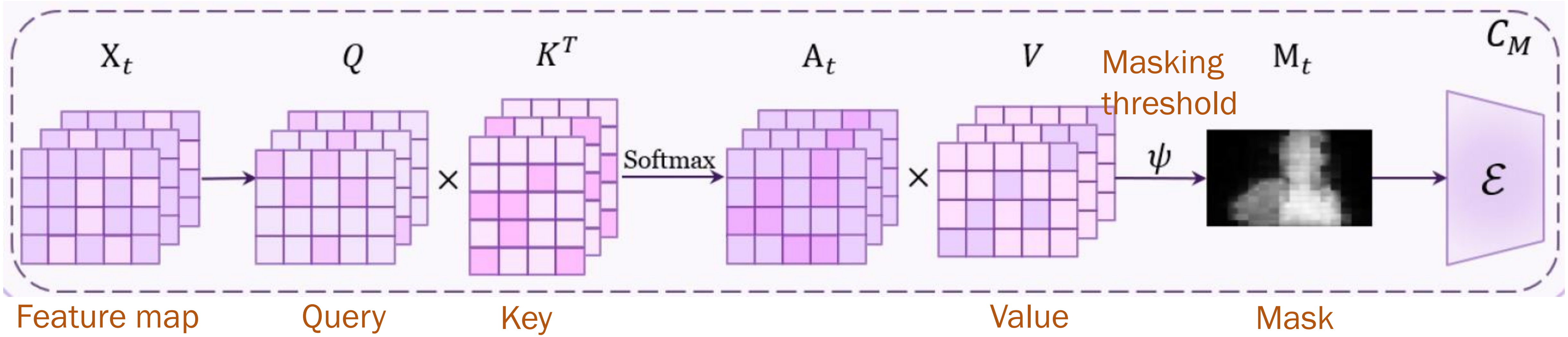
$$z \sim p_\theta(z \mid c_{\mathcal{T}}), \quad x = \mathcal{D}(z).$$

Content condition representation



- Previous works use CLIP image embedding to represent the content condition
  - difficulties in achieving the traditional stylization requirements
- add additional input channels to the first convolutional layer to concatenate latent vector zt and encoded feature vector cI
  - the final generated results have more consistency in semantic content relative to the input video

Self-features condition representation



Self-attention

- Recent large-scale diffusion incorporate conditioning by augmenting the denoising U-net $\epsilon\theta$ with the attention layer
- Self-attention relies less on external information and is better at capturing the internal relevance of self-features

Condition guidance

- improve the visual quality of generated images and to make sampled images better correspond with their conditioning
- jointly training the diffusion model for conditional and unconditional denoising, and combining the two score estimates at inference time
- with a guidance scale $s \geq 1$, the modified score estimate $\tilde{\epsilon}\theta$ is extrapolated in the direction toward the conditional $\epsilon\theta$ and away from the unconditional $\epsilon\theta$

guidance scale

$$\tilde{\epsilon_\theta}\left(z_t, c\right) = \epsilon_\theta\left(z_t, \varnothing\right) + s \cdot \left(\epsilon_\theta\left(z_t, c\right) - \epsilon_\theta\left(z_t, \varnothing\right)\right).$$

Unconditional

Condition guidance

- For our task, the scoring network $\tilde{\epsilon}\theta$ $(zt, cI, c$T) has three conditions: the input image cI , text prompt cT, and self-attention map cM
- introduce three guidance scales, sI , sT, and sM, which can be adjusted to trade off how strongly the generated samples correspond with the conditions
- Modified score estimate :

$$\begin{aligned}\tilde{\epsilon_\theta}\left(z_t, c_I, c_\mathcal{T}, c_M\right) =&\left(1 - s_I - s_\mathcal{T} - s_M\right) \cdot \epsilon_\theta\left(z_t, \varnothing, \varnothing\right)\\&+ s_I \cdot \epsilon_\theta\left(z_t, c_I, \varnothing\right)\\&+ s_\mathcal{T} \cdot \epsilon_\theta\left(z_t, \varnothing, c_\mathcal{T}\right)\\&+ s_M \cdot \epsilon_\theta\left(z_t, c_M, \varnothing\right).\end{aligned}$$

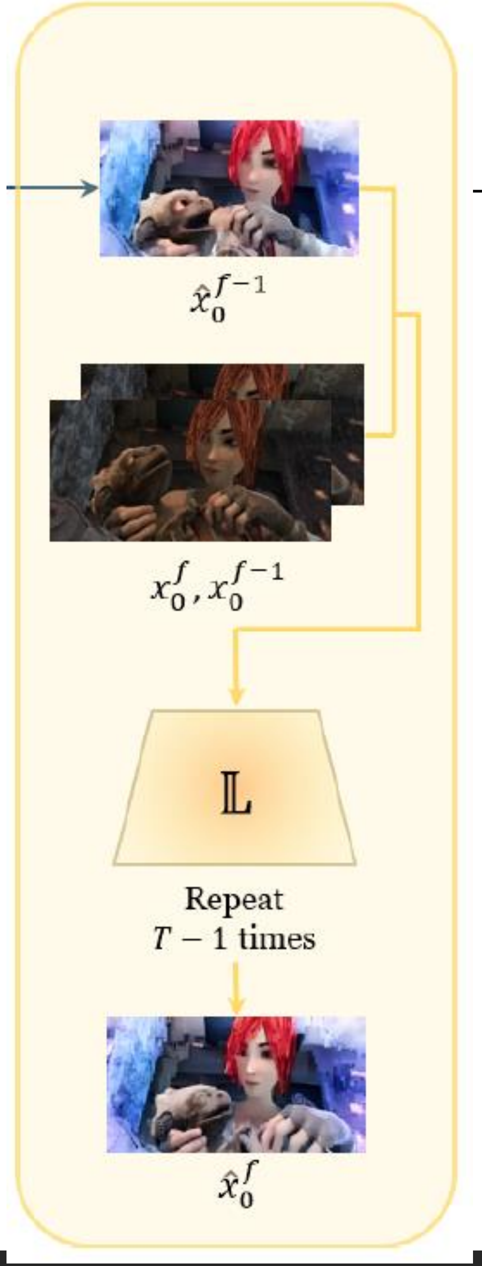
Sampling Optimization

- define our loss in CLIP feature space
  - allows us to impose additional constraints on the resulting internal CLIP representation of output Io
  - feed an image into CLIP's ViT encoder and extract its spatial tokens from the deepest layer
- Optimizes the denoising network :

$$\mathcal{L}_s = 1 - \mathcal{D}_{\cos}\left(x_0^f, x_t^f\right)$$

$$\tilde{\epsilon}_\theta = \epsilon_\theta(z_t - \lambda \Delta z_t, C_I, C_{\mathcal{T}}, C_M)$$

$$\Delta z_t = \nabla_{z_t} \mathcal{L}_s$$



Temporal Consistency

- Problem : local flicker
  - misalignment between input and atlas-based frames
- Solve : Use extra local deflicker network $\mathbb{L}$ to refine the results
  - predict the output frame $\hat{x}_0^f$ by providing two consecutive frames x_0^f, x_0^{f-1} and previous output $\hat{x}_0^{f-1}$
  - the network is trained with temporal consistency loss

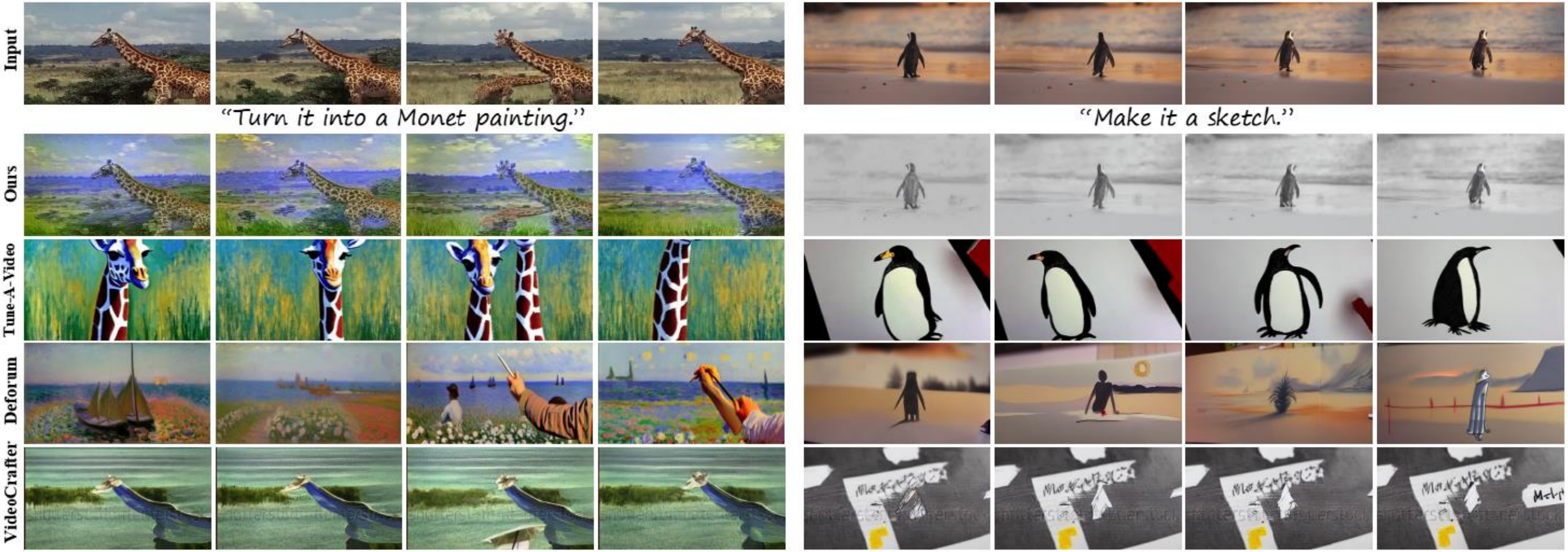
Result


"Make it Feng Zhu painting."
"Make it a cartoon."
"Make it rococo."
"Make it Edward Hopper painting."
"Make it a Claymation."
"Make it an Ukiyo-e painting."

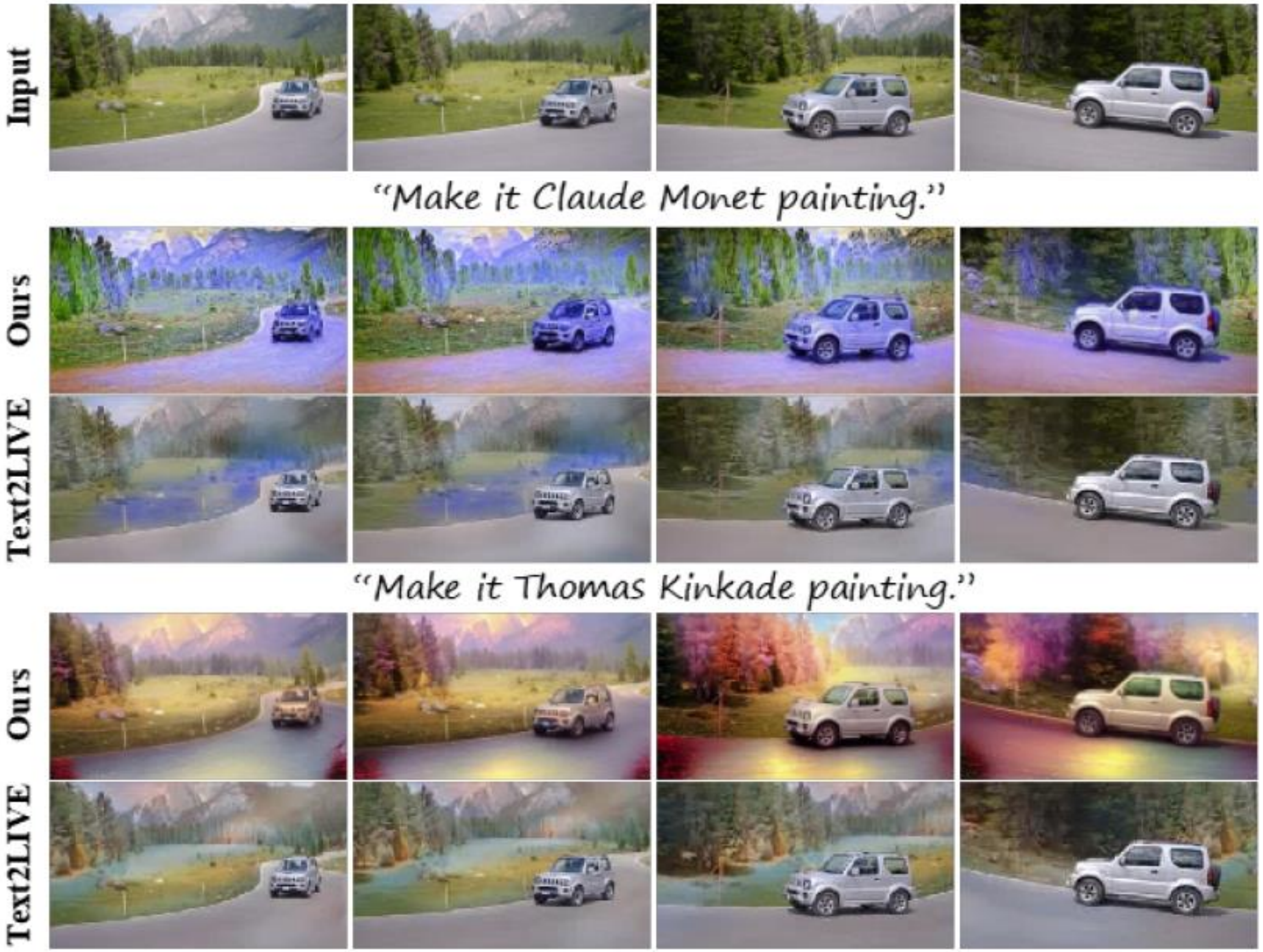
Result


"Make it Berthe Morisot painting."
"Turn it into a sand sculpture."
"Change to pixel art."
"Make it in the style of Mineraft."

Experiments



Experiments



Experiments

	Ours	Tune-A-Video	Deforum	Text2LIVE	VideoCrafter
Fra-Con↑	**0.987**	0.882	0.908	0.969	0.973
Pro-Con↑	**0.304**	0.235	0.263	0.272	0.266
Fra-Acc↑	0.983	0.75	0.872	**0.987**	0.945
Preference↑	-	0.157	0.086	0.229	0.286

Experiments