TSIT: A Simple and Versatile Framework for Image-to-Image Translation

ECCV 2020 Spotlight

Eunhye Lee

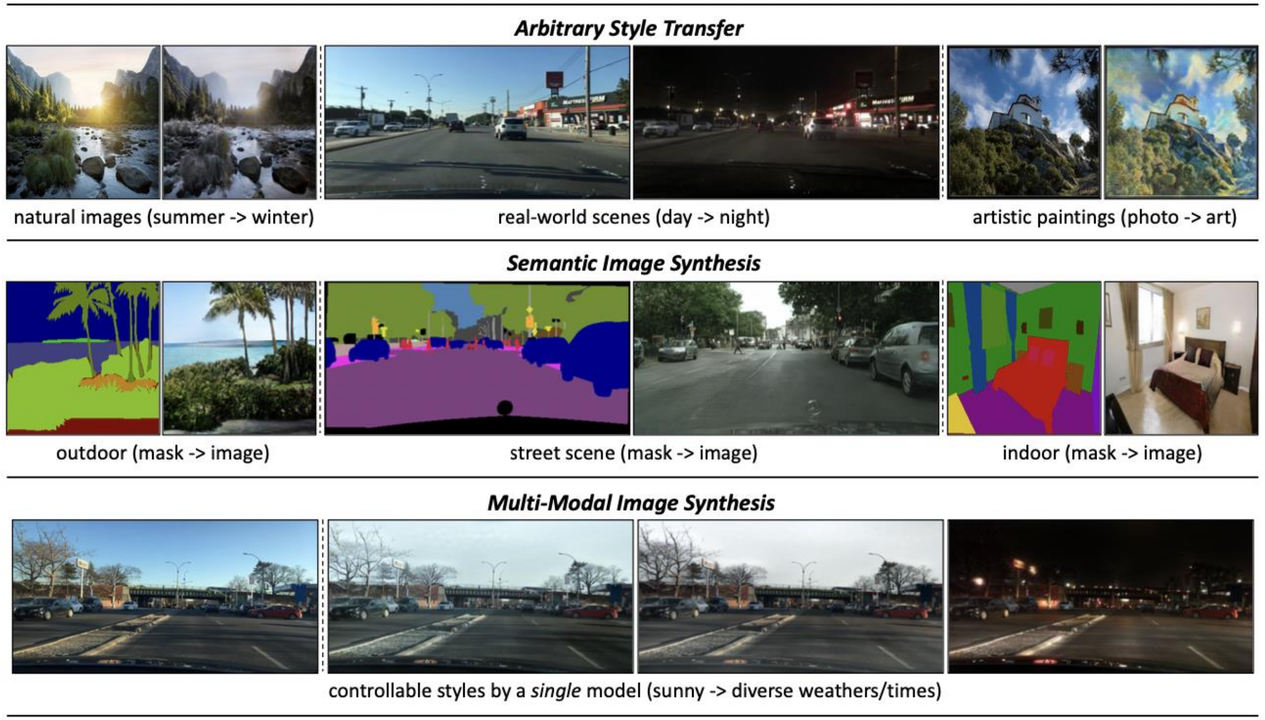
Contents

Introduction

Image-to-image translation

- Translate one image representation to another
- Ranging from arbitrary style transfer in the unsupervised setting, to semantic image synthesis in the supervised setting.

* This paper aims at devising a general and unified framework that is applicable to different image-to-image translation tasks!!



Introduction

Image-to-image translation

- Conditional image synthesis tasks(like arbitrary style transfer; Unsupervised) demand,
  - cycle consistency
  - semantic features
  - pixel gradients&values
- Semantic image synthesis(Supervised) demand,
  - Losses to minimize per-pixel distance between the generated sample and ground truth.
  - specialized structures to maintain spatial coherence and resolution.

Introduction

Key points

1. A two-stream network design that integrates both content and style effectively.

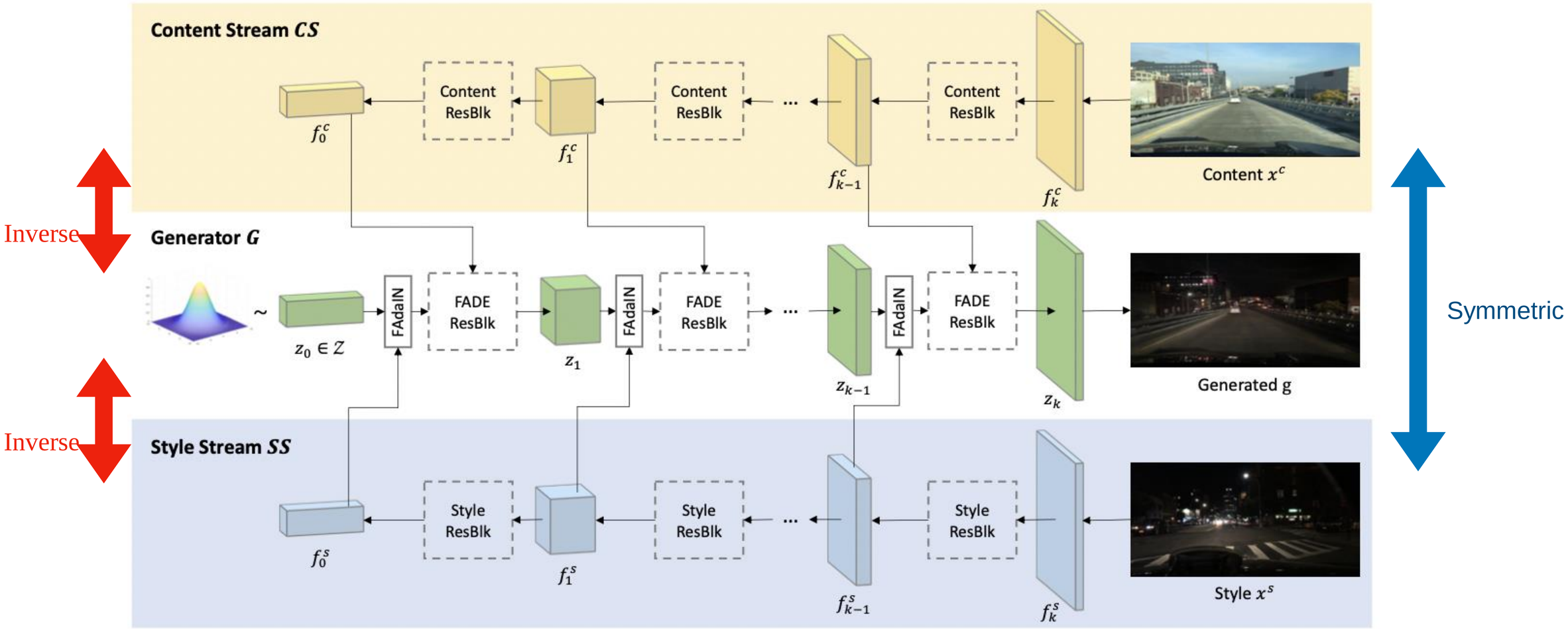
2. Multi-scale feature normalization scheme that captures coarse-to-fine structure and style information.

   - FADE

   - FAdaIN

Methodology

Two-Stream Image-to-image Translation (TSIT) framework



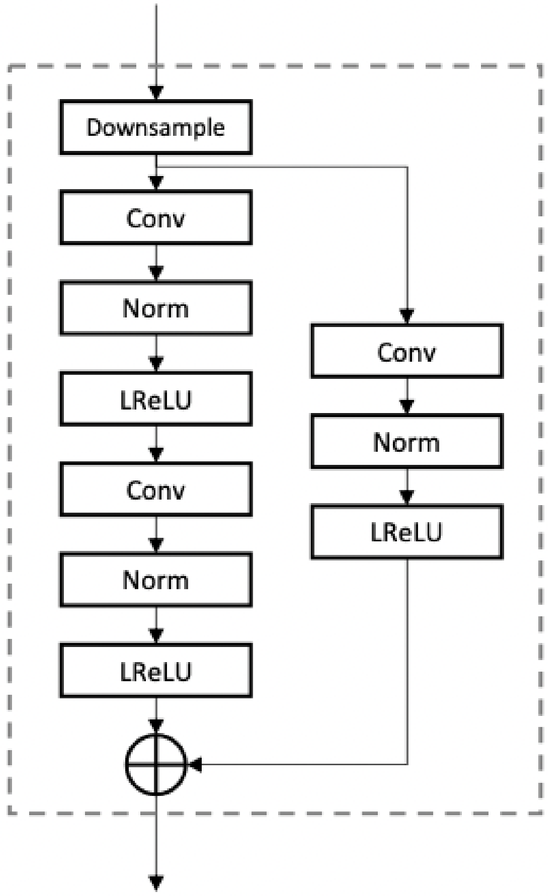
+ Discriminator

Methodology

Two-Stream Image-to-image Translation (TSIT) framework

Content/style stream

- Two streams are symmetrical with the same network structure to extract corresponding feature representations in different levels(multi-scale)
- Standard residual blocks
  - 3 conv layers
  - Skip connection
  - Leaky ReLU
  - Batch normalization



(a) Content/Style ResBlk

Methodology

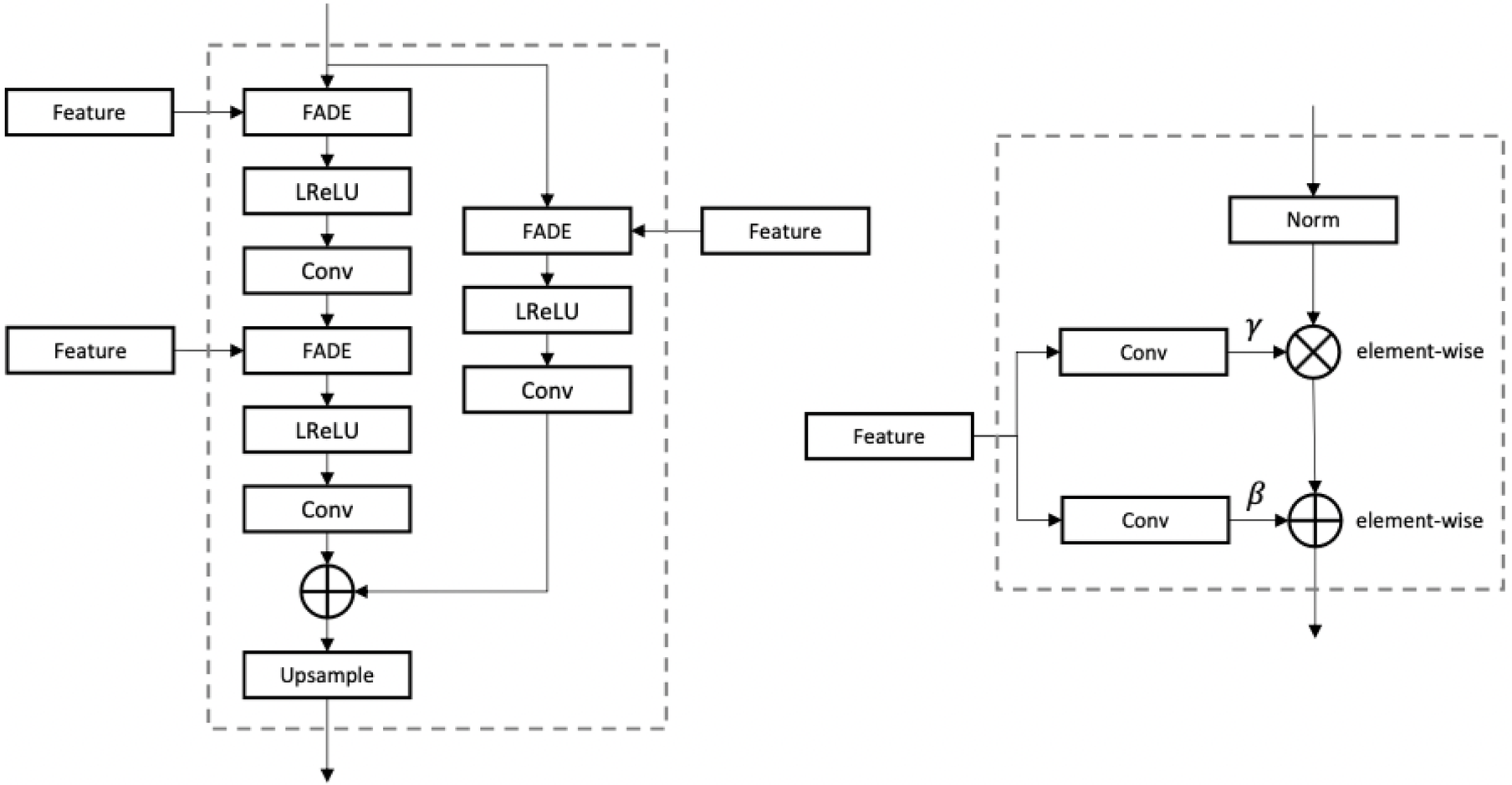
Two-Stream Image-to-image Translation (TSIT) framework

Generator

- Inverse structure of content/style stream to consistently match the level of semantic abstraction at different feature scales
- A Gaussian noise map as the latent input
- Take multi-scale feature inputs from two streams
- Feature transformation
  - FADE residual blocks(for content)
  - FAdaIN(for style)

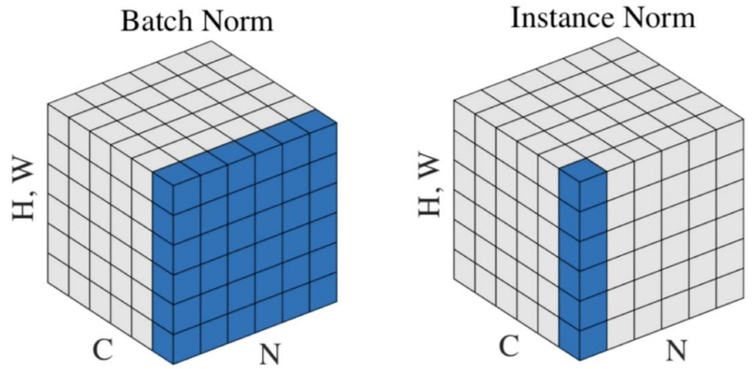
Methodology

Two-Stream Image-to-image Translation (TSIT) framework



Methodology

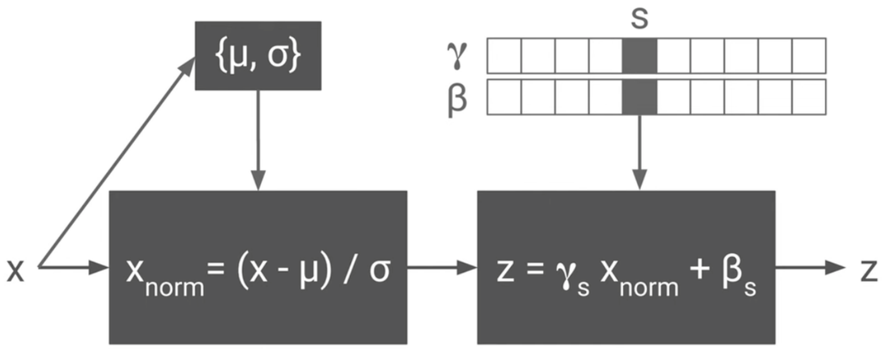
Multi-scale feature normalization scheme

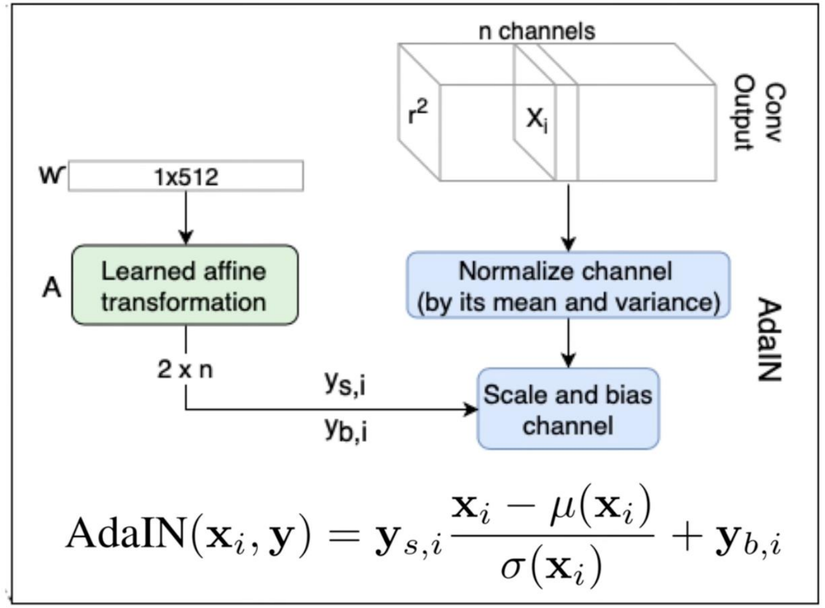


- Batch normalization
- Instance normalization

$$\mathbf{BN}(x) = \gamma\left(\frac{x-\mu(x)}{\sigma(x)}\right)+\beta \qquad \mathbf{IN}(x) = \gamma\left(\frac{x-\mu(x)}{\sigma(x)}\right)+\beta \qquad (4)$$

- Conditional batch normalization
- Adaptive instance normalization (AdaIN)

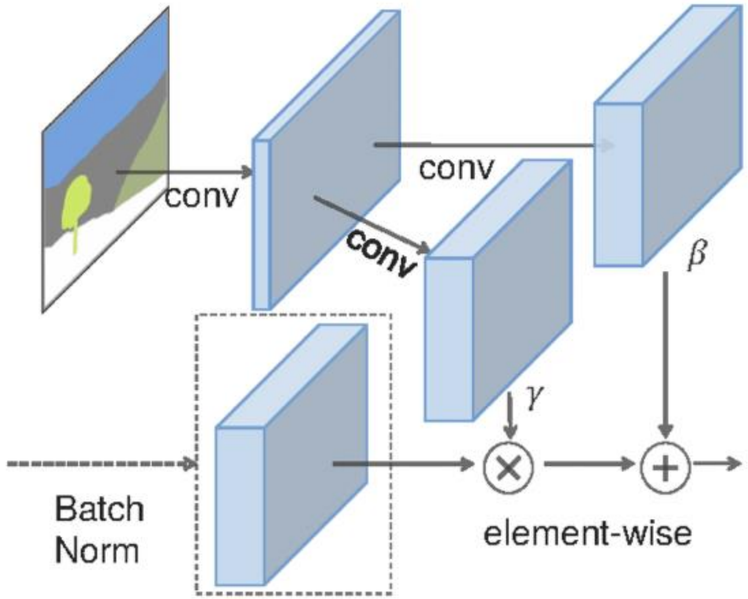




Methodology

Multi-scale feature normalization scheme

- Spatially adaptive denormalization (SPADE)*



$$\gamma^i_{c,y,x}(\mathbf{m})\frac{h^i_{n,c,y,x}-\mu^i_c}{\sigma^i_c}+\beta^i_{c,y,x}(\mathbf{m}) \qquad (1)$$

$$\mu^i_c=\frac{1}{NH^iW^i}\sum_{n,y,x}h^i_{n,c,y,x} \qquad (2)$$

$$\sigma^i_c=\sqrt{\frac{1}{NH^iW^i}\sum_{n,y,x}\left((h^i_{n,c,y,x})^2-(\mu^i_c)^2\right)}. \qquad (3)$$

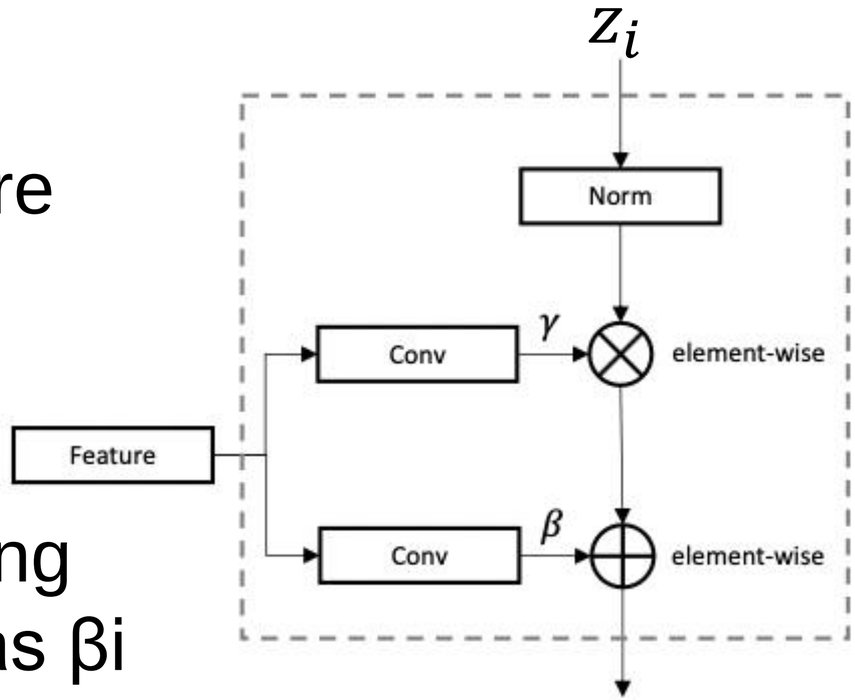
* Park, T., Liu, M.Y., Wang, T.C., Zhu, J.Y.: Semantic image synthesis with spatially-adaptive normalization. In: CVPR (2019)

Methodology

Feature adaptive denormalization(FADE)

- Inspired by SPADE
- we generalize the input to multi-scale feature representation f_i^c of the content image x_c

A. Apply batch normalization
B. Modulate the normalized feature by using the learned parameters scale γi and bias βi



Methodology

Feature adaptive denormalization(FADE)

- N: batch size
- L_i: the number of feature map channels in each layer
- Denormalized activation:

$$\gamma_i^{l,h,w} \cdot \frac{z_i^{n,l,h,w} - \mu_i^l}{\sigma_i^l} + \beta_i^{l,h,w},$$

$$\mu_i^l = \frac{1}{NH_iW_i} \sum_{n,h,w} z_i^{n,l,h,w},$$

$$\sigma_i^l = \sqrt{\frac{1}{NH_iW_i} \sum_{n,h,w} \left(z_i^{n,l,h,w}\right)^2 - \left(\mu_i^l\right)^2}.$$

Methodology

Feature adaptive instance normalization (FAdaIN)

$$\mathrm{AdaIN}(x, y) = \sigma(y)\left(\frac{x - \mu(x)}{\sigma(x)}\right) + \mu(y)$$

$$\mathrm{FAdaIN}\left(z_i, f_i^s\right) = \sigma\left(f_i^s\right)\left(\frac{z_i - \mu\left(z_i\right)}{\sigma\left(z_i\right)}\right) + \mu\left(f_i^s\right), \qquad (4)$$

Methodology

Discriminator

- The standard multi-scale patch-based discriminators
  - 3 Discriminators at different scales, with identical architecture
    - Coarsest scale for capturing global information
    - Finest scale to produce better details
  - improve the robustness in different resolutions.
  - Serve as feature extractors for the generator to optimize the feature matching loss.

Methodology

Objective function

$$\mathcal{L}_G = -\mathbb{E}\left[D\left(g\right)\right] + \lambda_P \mathcal{L}_P\left(g, x^c\right) + \lambda_{FM}\mathcal{L}_{FM}\left(g, x^s\right),$$

$$\mathcal{L}_D = -\mathbb{E}\left[\min\left(-1 + D\left(x^s\right), 0\right)\right] - \mathbb{E}\left[\min\left(-1 - D\left(g\right), 0\right)\right],$$

- For generator
  - Hinge-based adversarial loss
  - VGG-19 based Perceptual loss
  - Feature matching loss
- For discriminator
  - Hinge-based adversarial loss to distinguish real or fake image

$g = G\left(z_0, x^c, x^s\right)$

z_0: input noise map in latent space

x^c: Content image

x^s Style image

Experiments

- **Applications**
  - Arbitrary style transfer (unsupervised)
    - Winter to summer
    - Day to night
    - Photo to art
  - Semantic image synthesis (supervised)
  - Multi-modal image synthesis (enriching generation diversity)
    - Time and weather image-to-image translation

Experiments

Arbitrary style transfer

- FID: evaluating similarity of distribution between the generated images and the real images.
- IS(Inception Score): considering clarity and diversity.

Table 1. The FID and IS scores of our method compared to state-of-the-art methods in arbitrary style transfer tasks. A lower FID and a higher IS indicate better performance.

Methods	summer → winter		day → night		photo → art	
	FID ↓	IS ↑	FID ↓	IS ↑	FID ↓	IS ↑
MUNIT [14]	118.225	2.537	110.011	2.185	167.314	3.961
DMIT [49]	87.969	2.884	83.898	2.156	166.933	3.871
Ours	**80.138**	**2.996**	**79.697**	**2.203**	**165.561**	**4.020**

Experiments

Arbitrary style transfer

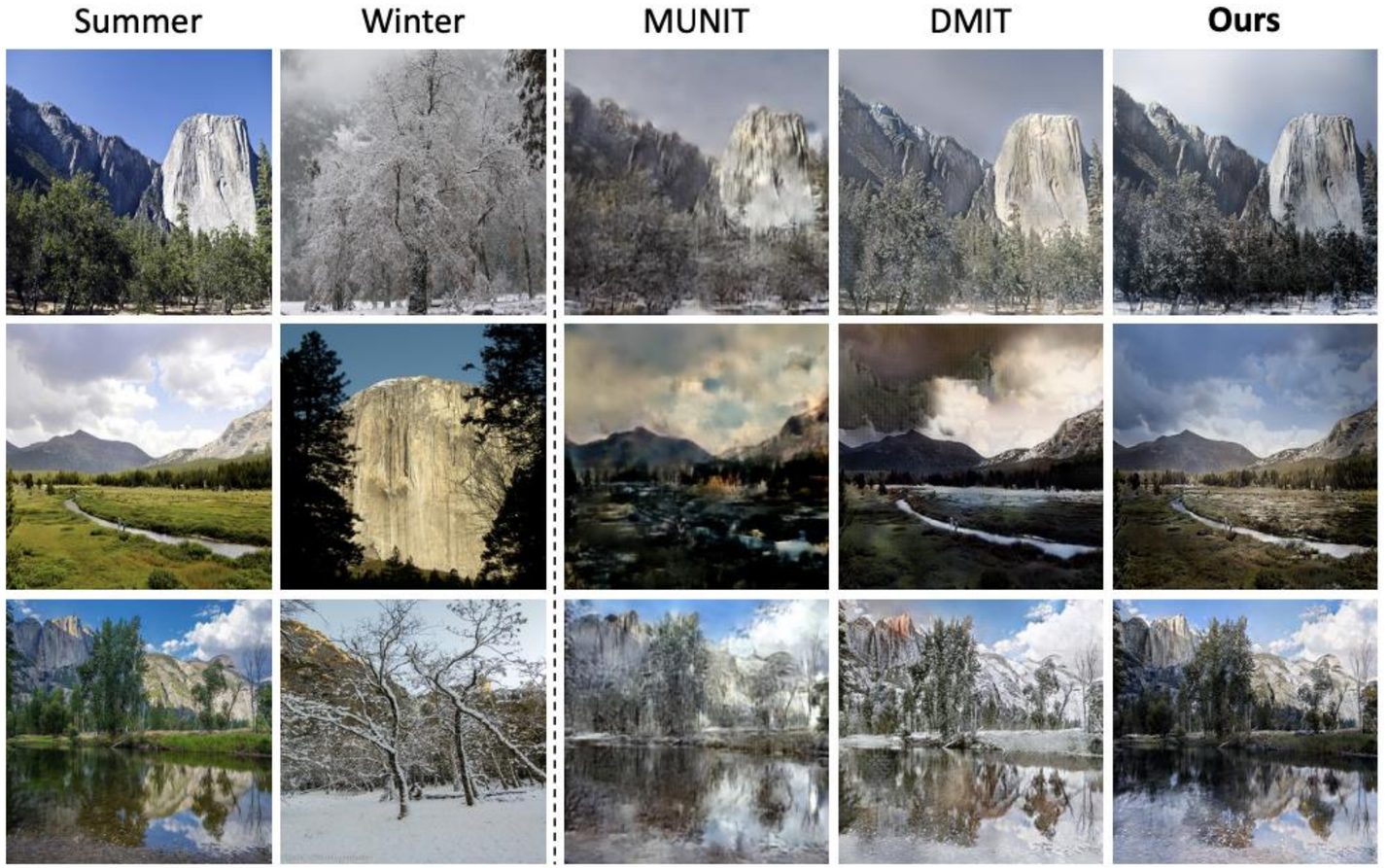


Fig. 4. Yosemite summer → winter season transfer results compared to baselines.

Experiments

Arbitrary style transfer

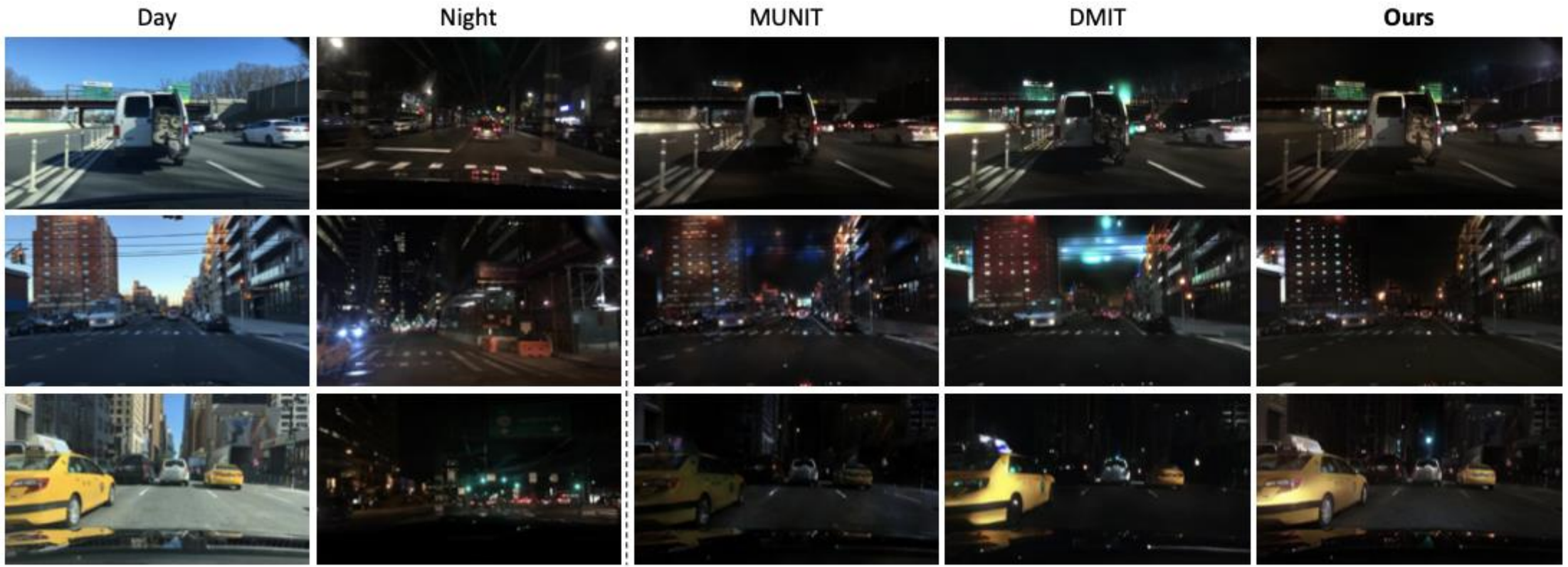


Fig. 5. BDD100K day → night time translation results compared to baselines.

Experiments

Arbitrary style transfer

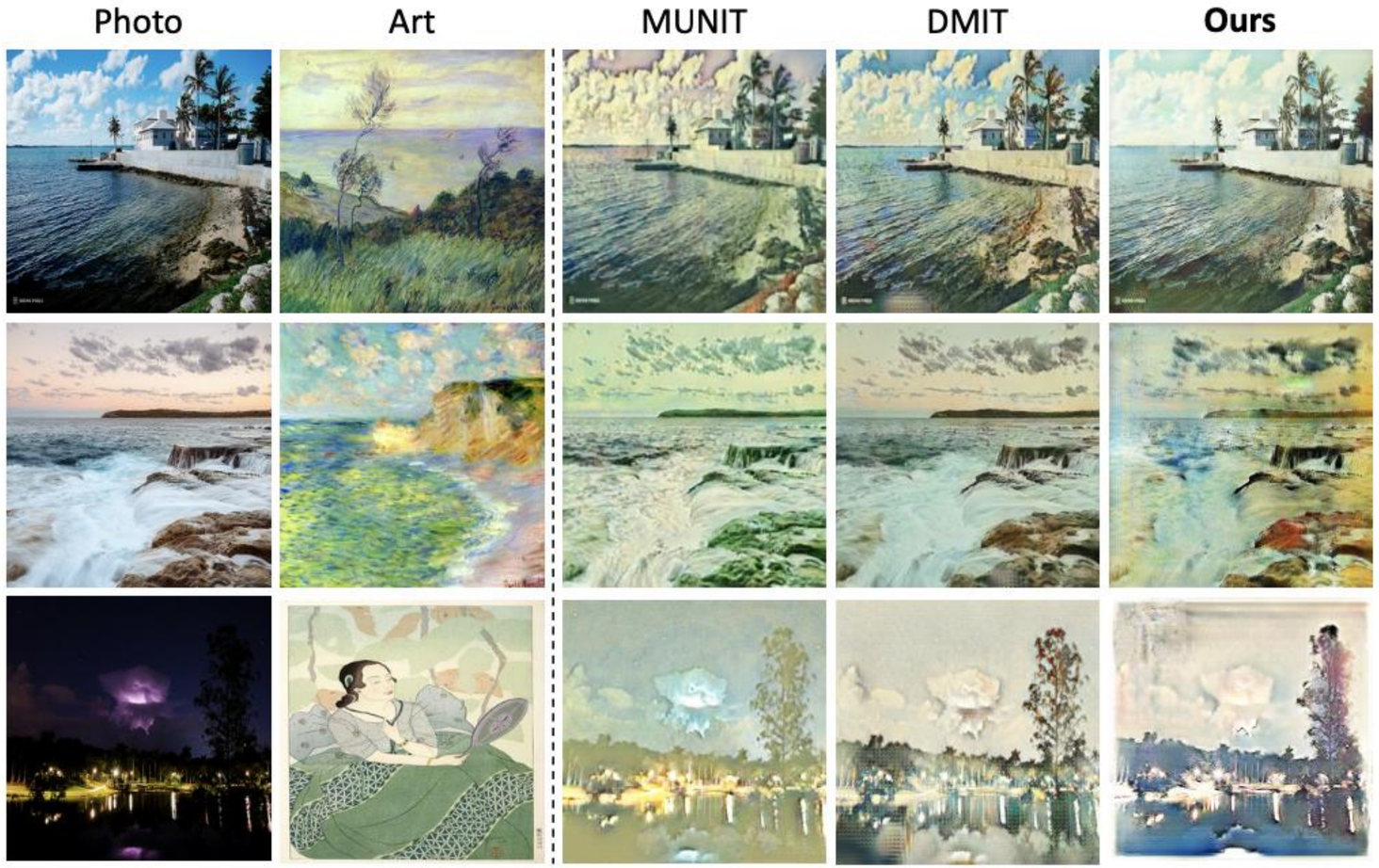


Fig. 6. Photo → art style transfer results compared to baselines.

Experiments

Semantic image synthesis

- FID
- Segmentation accuracy (mean Intersection-over-Union (mIoU)
- Pixel accuracy (accu)

Table 2. The mIoU, pixel accuracy (accu) and FID scores of our method compared to state-of-the-art methods in semantic image synthesis tasks. A higher mIoU, a higher pixel accuracy (accu) and a lower FID indicate better performance.

	Cityscapes			ADE20K		
Methods	mIoU ↑	accu ↑	FID ↓	mIoU ↑	accu ↑	FID ↓
CRN [4]	52.4	77.1	104.7	22.4	68.8	73.3
SIMS [34]	47.2	75.5	**49.7**	N/A	N/A	N/A
pix2pixHD [41]	58.3	81.4	95.0	20.3	69.2	81.8
SPADE [33]	62.3	81.9	71.8	38.5	79.9	33.9
CC-FPSE [28]	65.5	82.3	54.3	**43.7**	**82.9**	31.7
Ours	**65.9**	**94.4**	59.2	38.6	80.8	**31.6**

Experim

Semantic ima

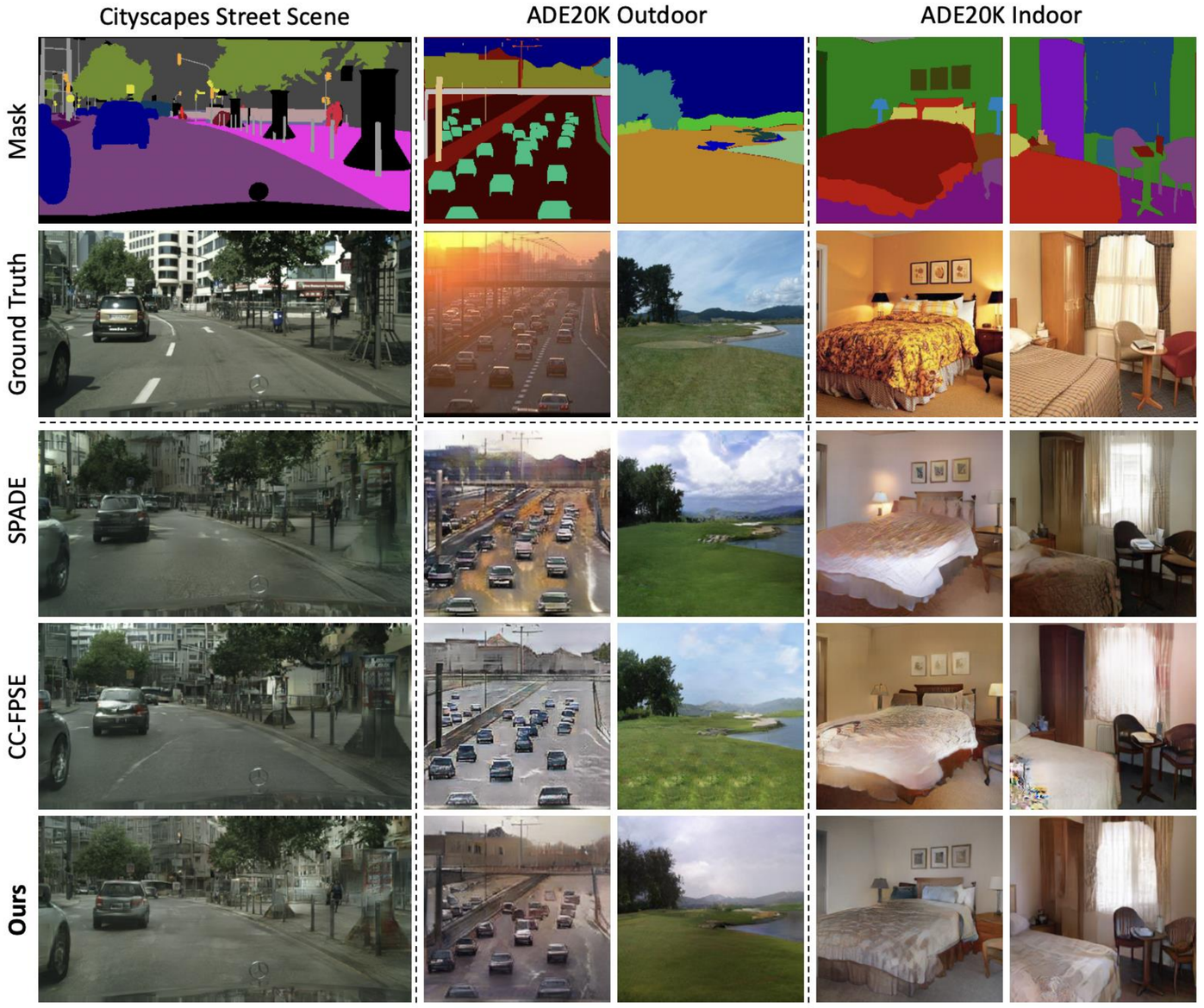


Fig. 7. Semantic image synthesis results compared to baselines.

Experiments

Multi-modal image synthesis

Fig. 8. BDD100K multi-modal image synthesis for different time and weather translation results by a *single* model.

Experiments

Fig. 9. Cross validation of ineffectiveness

Fig. 10. Ablation studies of key modules (*i.e.*, content stream (CS), style stream(SS)) and feature transformations in multi-modal image synthesis task.