On the Difficulty of Unpaired Infrared-to-Visible Video Translation: Fine-Grained Content-Rich Patches Transfer

Zhenjie Yu[1], Shuang Li[1,✉], Yirui Shen[1], Chi Harold Liu[1], Shuigen Wang[2]

[1]Beijing Institute of Technology [2]Yantai IRay Technologies Lt. Co.

{zjyu, shuangli, yiruishen, chiliu}@bit.edu.cn shuigen.wang@iraytek.com

Clear day

Visible camera

Dark night

Overexposure

Rainy day

Foggy day

Infrared sensor

Translation Task:

Transfer

Subsequent Tasks (e.g., object detection, semantic segmentation)

Overview

- Unpaired infrared-to-visible video translation.
- We achieve a **fine-grained content-rich patches transfer**.
- The experimental results on subsequent tasks confirm the success of translation.

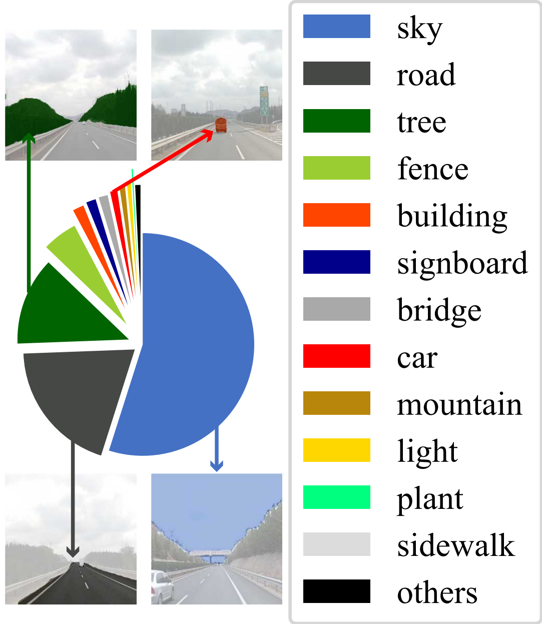
Infrared　　Visible　　Generated Results

Q1: What are the **content-rich patches**?

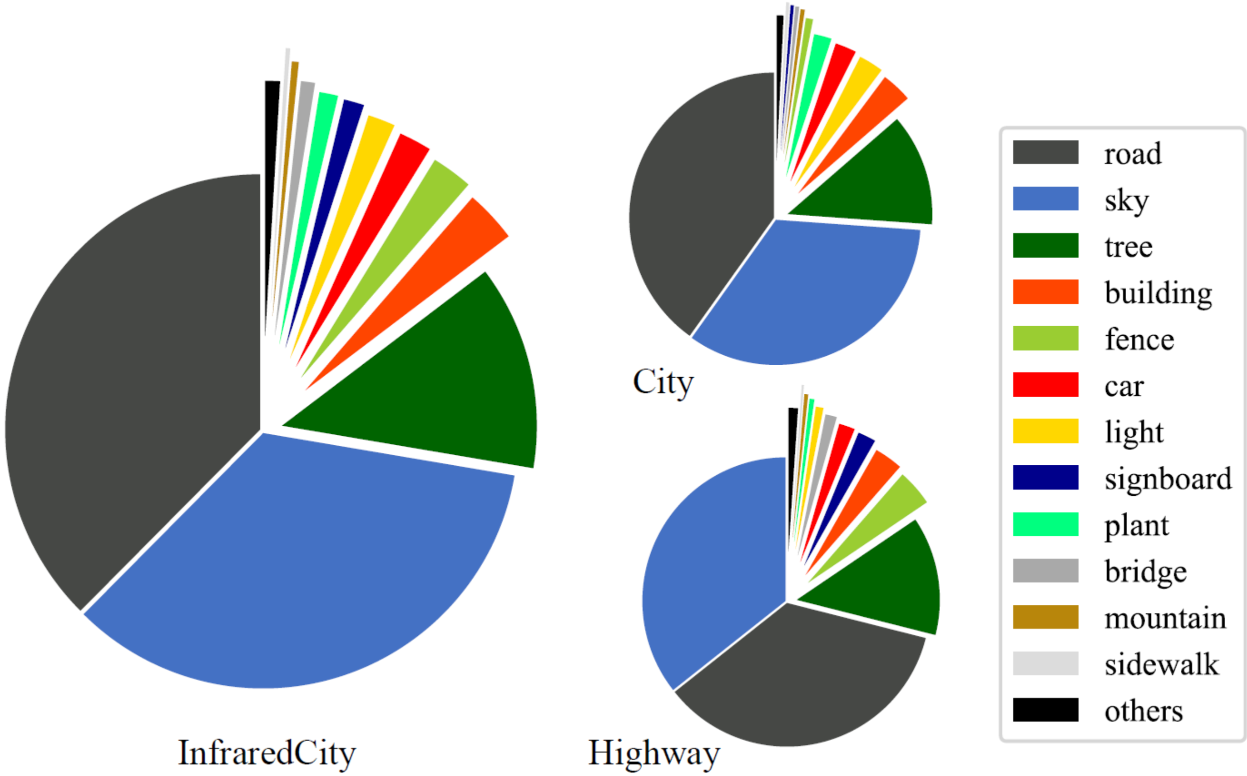
Content-rich Patches vs. Content-lacking Patches

Visual Details!

Q2: Why are the content-rich patches **not fine-grained**?



IRVI dataset



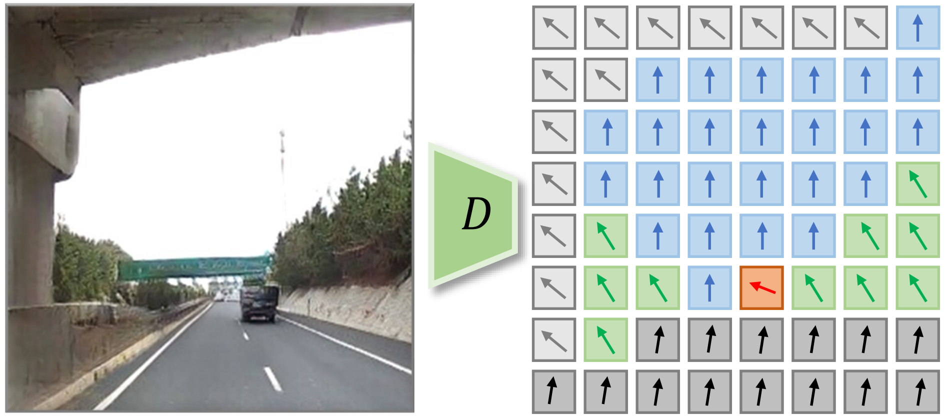
InfraredCity-Lite dataset

Q2: Why are the content-rich patches **not fine-grained**?

$$\mathcal{L}_{adv}^{patch} = \mathbb{E}_y[\log D(y)] + \mathbb{E}_x\left[\log\left(1 - D\big(G(x)\big)\right)\right]$$
$$= \mathbb{E}_y\left[\frac{1}{N}\sum_{i=1}^{N}\log p_i\right] + \mathbb{E}_x\left[\frac{1}{N}\sum_{j=1}^{N}\log\left(1-\tilde{p}_j\right)\right] \quad (1)$$

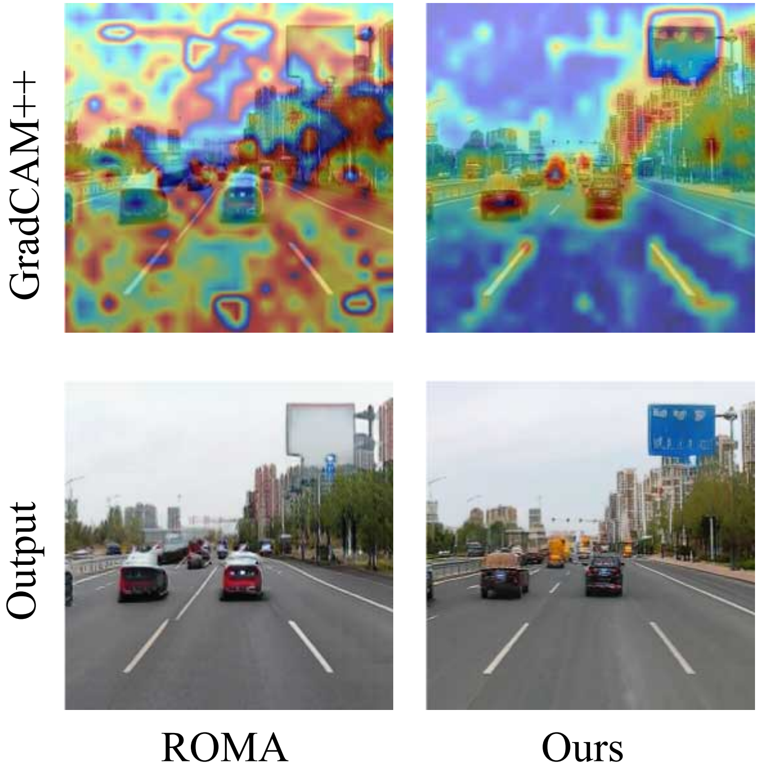
$$\nabla_{\theta_D}\mathcal{L}_{adv}^{patch} = \mathbb{E}_y\left[\frac{1}{N}\sum_{i=1}^{N}\nabla_{\theta_D}\log p_i\right] + \mathbb{E}_x\left[\frac{1}{N}\sum_{j=1}^{N}\nabla_{\theta_D}\log\left(1-\tilde{p}_j\right)\right] \quad (2)$$

Example:



Gradients

Q2: Why are the content-rich patches **not fine-grained**?



Equally optimization on all patches

+

Long-tail effect on images

||

Prejudice on optimization

Q3: What can we do to address the challenging issue?

Key idea of the CPTrans

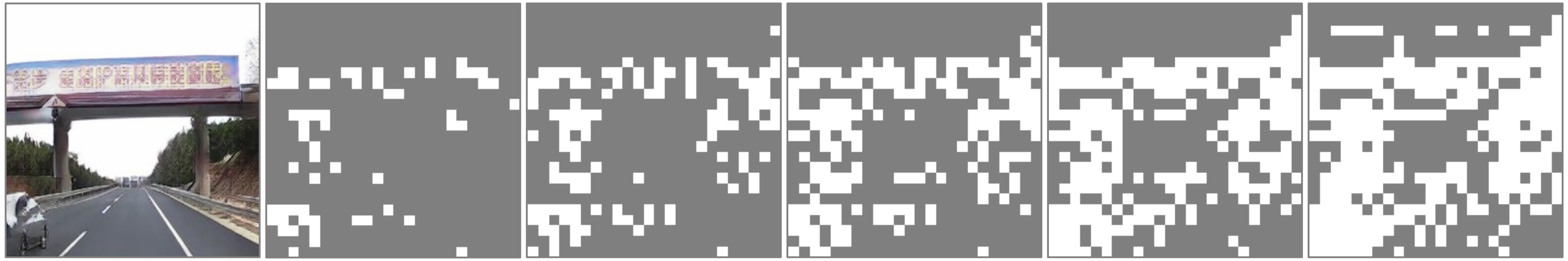
1. Find the content-rich patches.
2. Augmenting the model's focus on these patches.

Method

➢ Find the content-rich patches.

- Gradients from different content patches tend to vary [1, 2].
- Real-world training data usually exhibits long-tailed distribution [3, 4].
- The optimization of the model is **more favorable to the content-lacking regions** and **diverges from the optimization of the content-rich regions**.

The most deviated parts of patches without Content-aware Optimization.

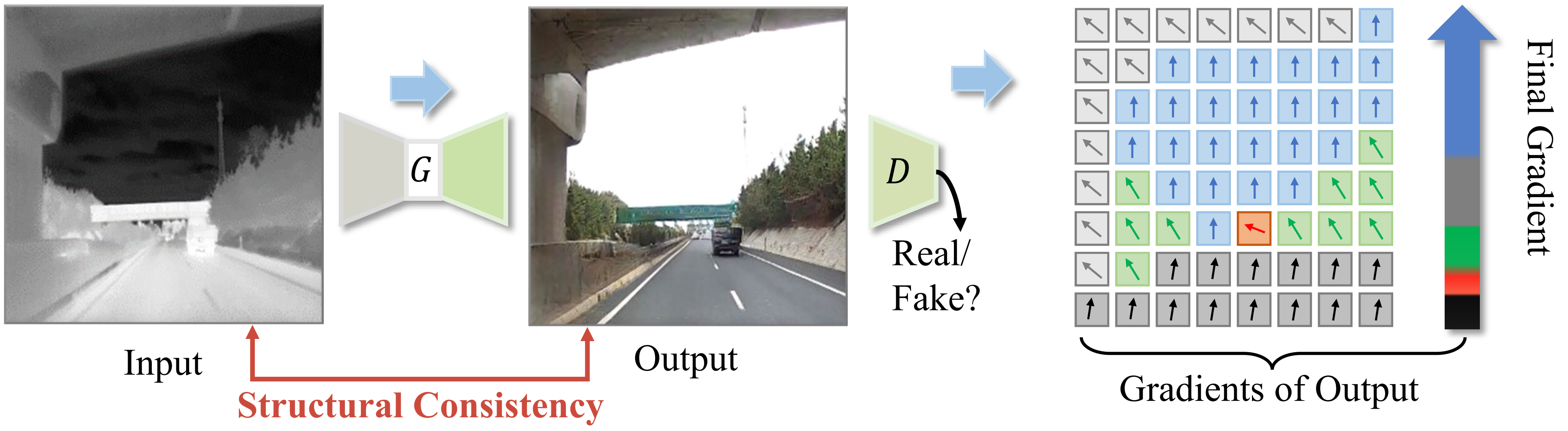


Output $\lambda_{\text{ratio}} = 10\%$ 20% 30% 40% 50%

[1] Aleksandar Armacki, Dragana Bajovic, Dusan Jakovetic, and Soummya Kar. Gradient based clustering. In ICML, pages 929–947, 2022.
[2] Michael Rapp, Eneldo Loza Menc´ıa, Johannes F¨urnkranz, and Eyke H¨ullermeier. Gradient-based label binning in multi-label classification. In ECML/PKDD, pages 462–477, 2021.
[3] Shuang Li, Kaixiong Gong, Chi Harold Liu, Yulin Wang, Feng Qiao, and Xinjing Cheng. Metasaug: Meta semantic augmentation for long-tailed visual recognition. In CVPR, pages 5212–5221, 2021.
[4] Tong Wu, Ziwei Liu, Qingqiu Huang, Yu Wang, and Dahua Lin. Adversarial robustness under long-tailed distribution. In CVPR, pages 8659–8668, 2021.

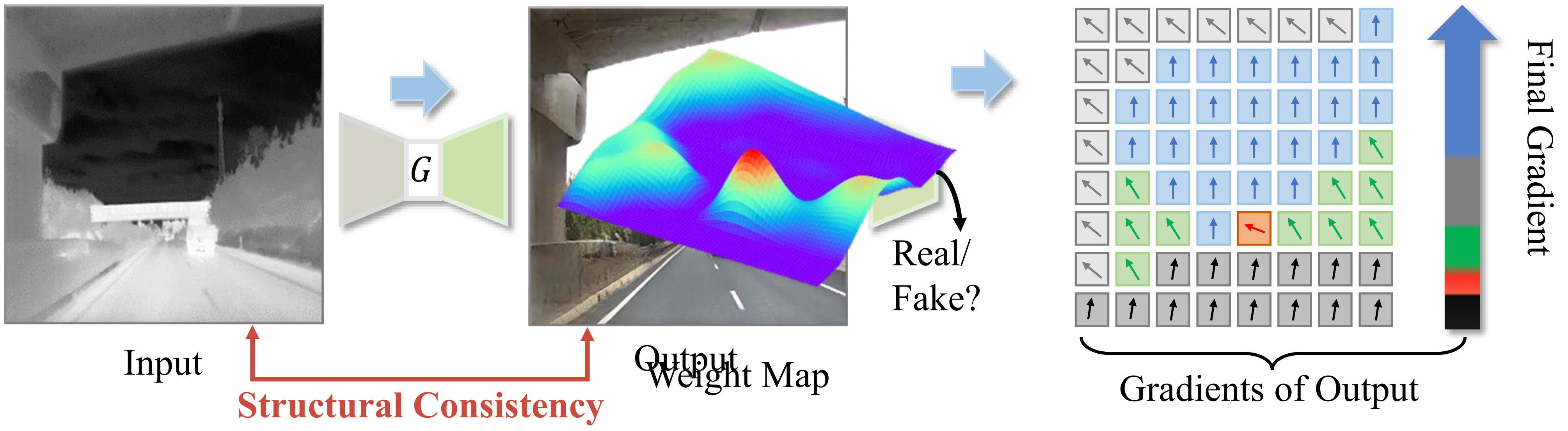
Method

➢ Find the content-rich patches.



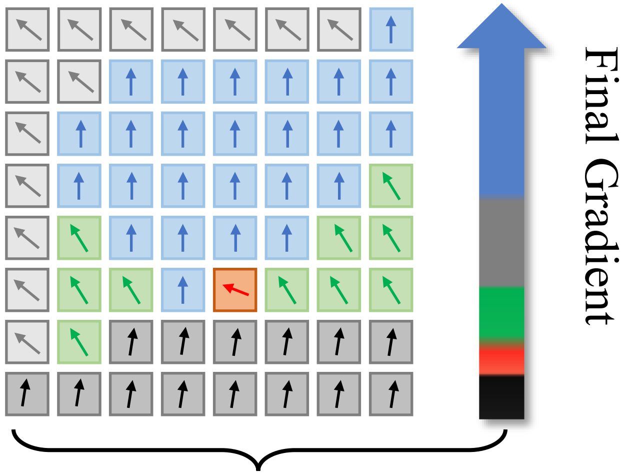
Method

➢ Find the content-rich patches.

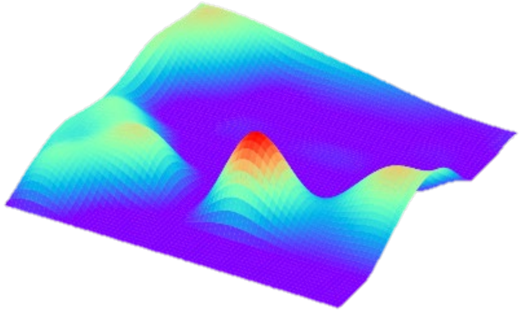


Method

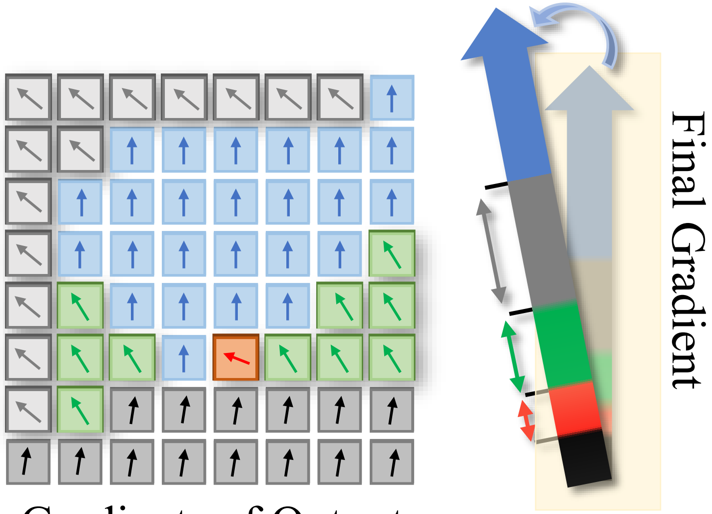
➤ Augmenting the model's focus on these patches.



Gradients of Output



Weight Map w



Gradients of Output

$$\nabla_{\theta_D}\mathcal{L}_{adv}^{patch}$$
$$= \mathbb{E}_y\left[\frac{1}{N}\sum_{i=1}^{N}\boxed{\nabla_{\theta_D}\log p_i}\right] + \mathbb{E}_x\left[\frac{1}{N}\sum_{j=1}^{N}\nabla_{\theta_D}\log\left(1-\tilde{p}_j\right)\right]$$

$$\delta_i = \cos\left(\nabla_{\theta_D}\log p_i, \nabla_{\theta_D}\frac{1}{N}\sum_{j=1}^{N}\log p_j\right)$$

$$w_i = \frac{\lambda_{inc}}{\exp(|\delta_i|)}$$

Method

- Augmenting the model's focus on these patches.

$$\delta_i = \cos\left(\nabla_{\theta_D} \log p_i, \nabla_{\theta_D} \frac{1}{N}\sum_{j=1}^{N} \log p_j\right), \qquad w_i = \frac{\lambda_{inc}}{\exp(|\delta_i|)}$$

$$\nabla_{\theta_D} \mathcal{L}_{adv}^{patch} = \mathbb{E}_y\left[\frac{1}{N}\sum_{i=1}^{N} \nabla_{\theta_D} \log p_i\right] + \mathbb{E}_x\left[\frac{1}{N}\sum_{j=1}^{N} \boxed{\nabla_{\theta_D} \log(1-\tilde{p}_j)}\right] \Rightarrow \tilde{w}_i = \frac{\lambda_{inc}}{\exp(|\tilde{\delta}_i|)}$$

$$w_i \nabla_\theta \log p_i = \nabla_\theta w_i \log p_i, \qquad \tilde{w}_i \nabla_\theta \log \tilde{p}_i = \nabla_\theta \tilde{w}_i \log \tilde{p}_i$$

Content-aware Optimization

$$\mathcal{L}_{co-adv}^{patch} = \mathbb{E}_y\left[\frac{1}{N}\sum_{i=1}^{N} w_i \log p_i\right] + \mathbb{E}_x\left[\frac{1}{N}\sum_{j=1}^{N} \tilde{w}_j \log(1-\tilde{p}_j)\right]$$

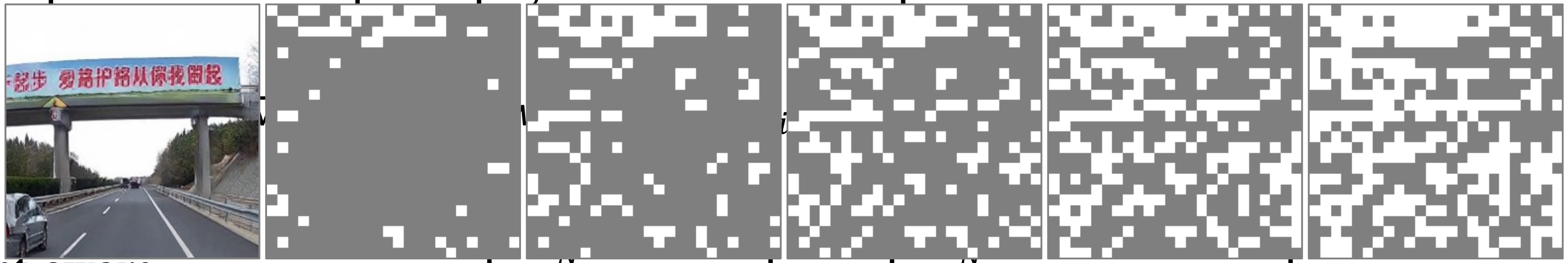
Method

➢ Augmenting the model's focus on these patches.

$$\delta_i = \cos\left(\nabla_{\theta_D}\log p_i, \nabla_{\theta_D}\frac{1}{N}\sum_{j=1}^{N}\log p_j\right), \qquad w_i = \frac{\lambda_{inc}}{\exp(|\delta_i|)}$$

$$\nabla_{\theta_D}\mathcal{L}_{adv}^{patch} = \mathbb{E}_y\left[\frac{1}{N}\sum_{i=1}^{N}\nabla_{\theta_D}\log p_i\right] + \mathbb{E}_x\left[\frac{1}{N}\sum_{j=1}^{N}\boxed{\nabla_{\theta_D}\log\left(1-\tilde{p}_j\right)}\right] \Rightarrow \tilde{w}_i = \frac{\lambda_{inc}}{\exp(|\tilde{\delta}_i|)}$$

The most deviated parts of patches with Content-aware Optimization



Output | $\lambda_{ratio} = 10\%$ | 20% | 30% | 40% | 50%

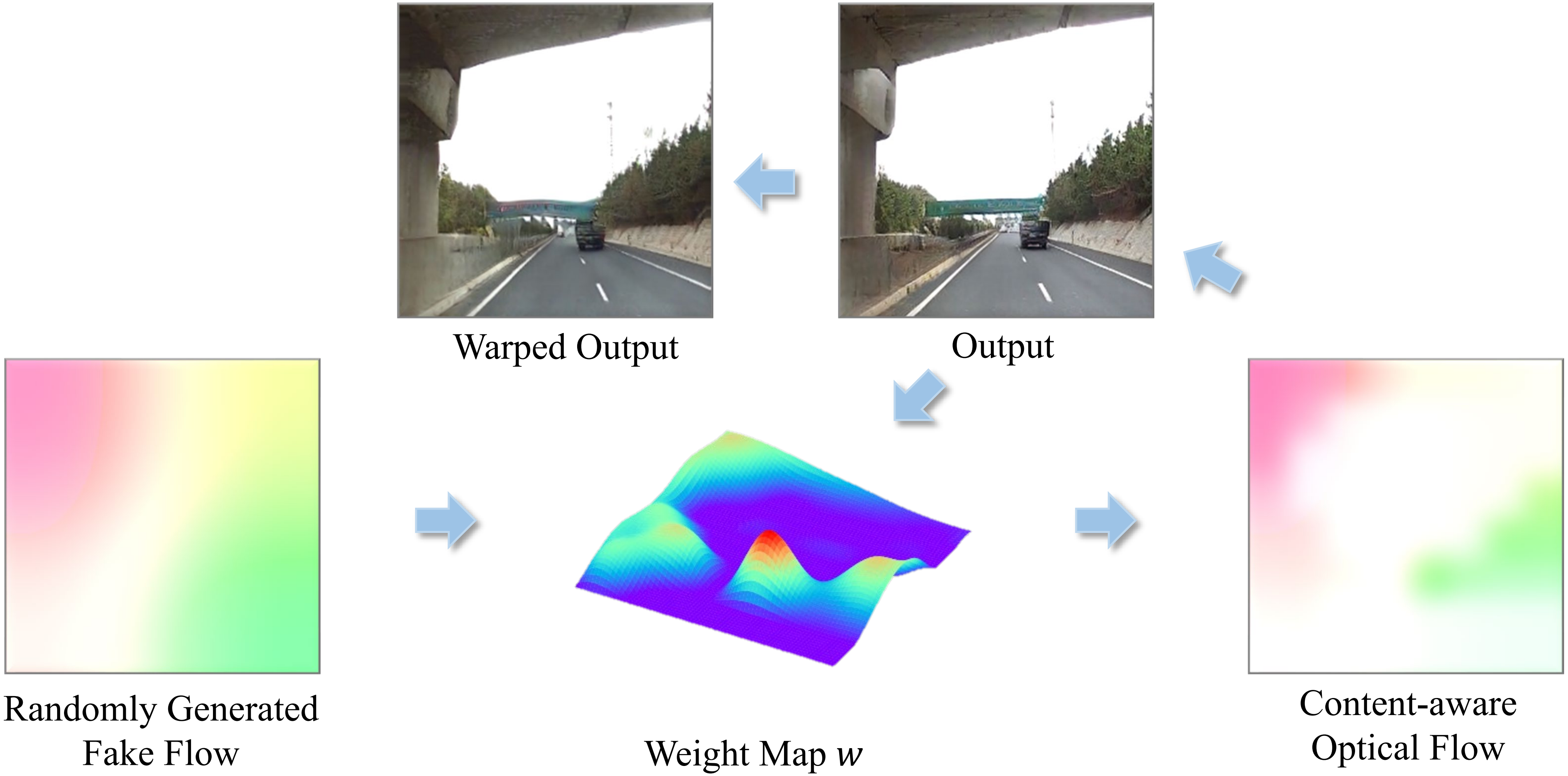
Content-aware Optimization

$$\mathcal{L}_{co-adv}^{patch} = \mathbb{E}_y\left[\frac{1}{N}\sum_{i=1}^{N}w_i\log p_i\right] + \mathbb{E}_x\left[\frac{1}{N}\sum_{j=1}^{N}\tilde{w}_j\log\left(1-\tilde{p}_j\right)\right]$$

Random!

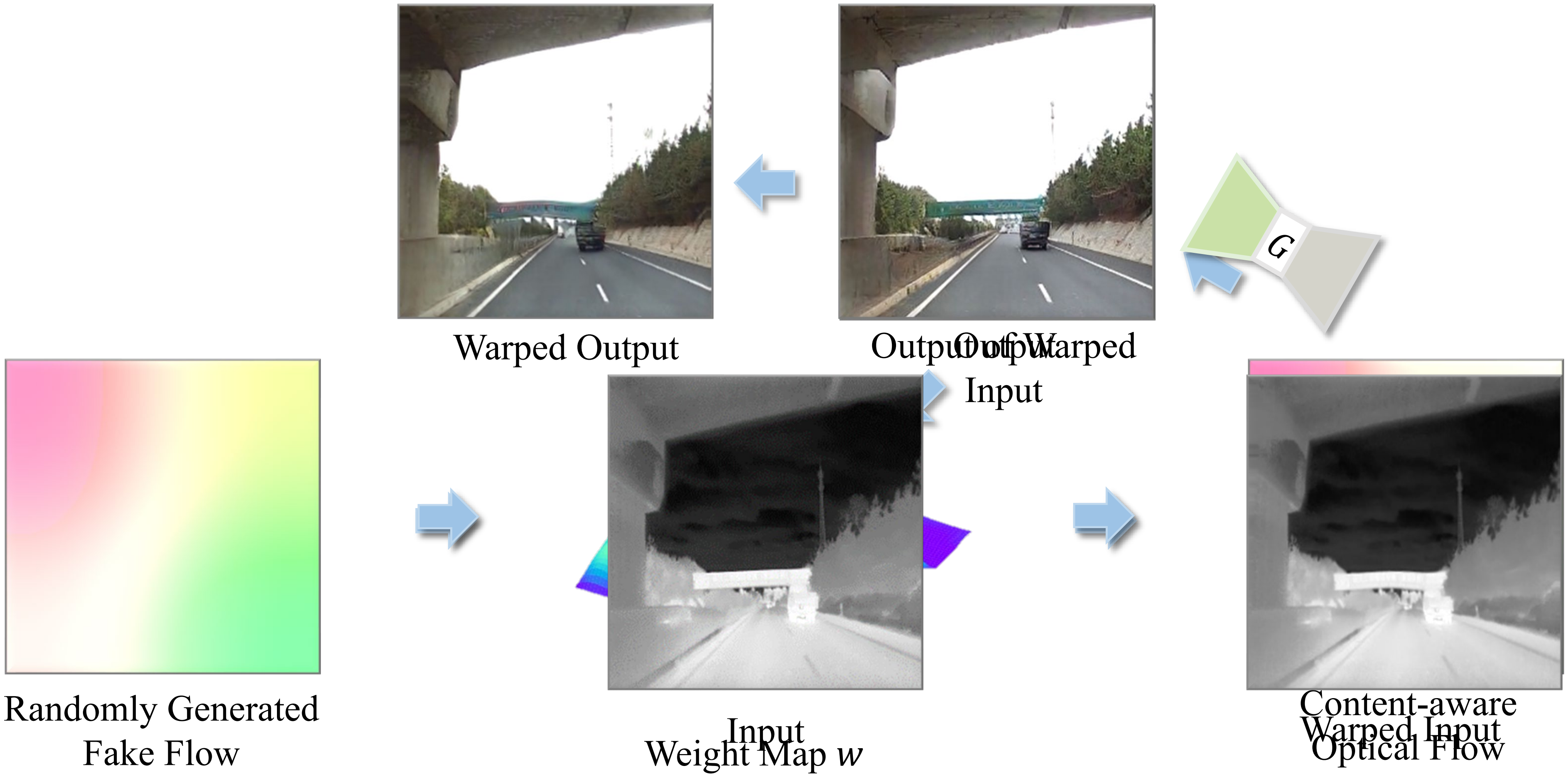
Method

➢ Content-aware Temporal Normalization



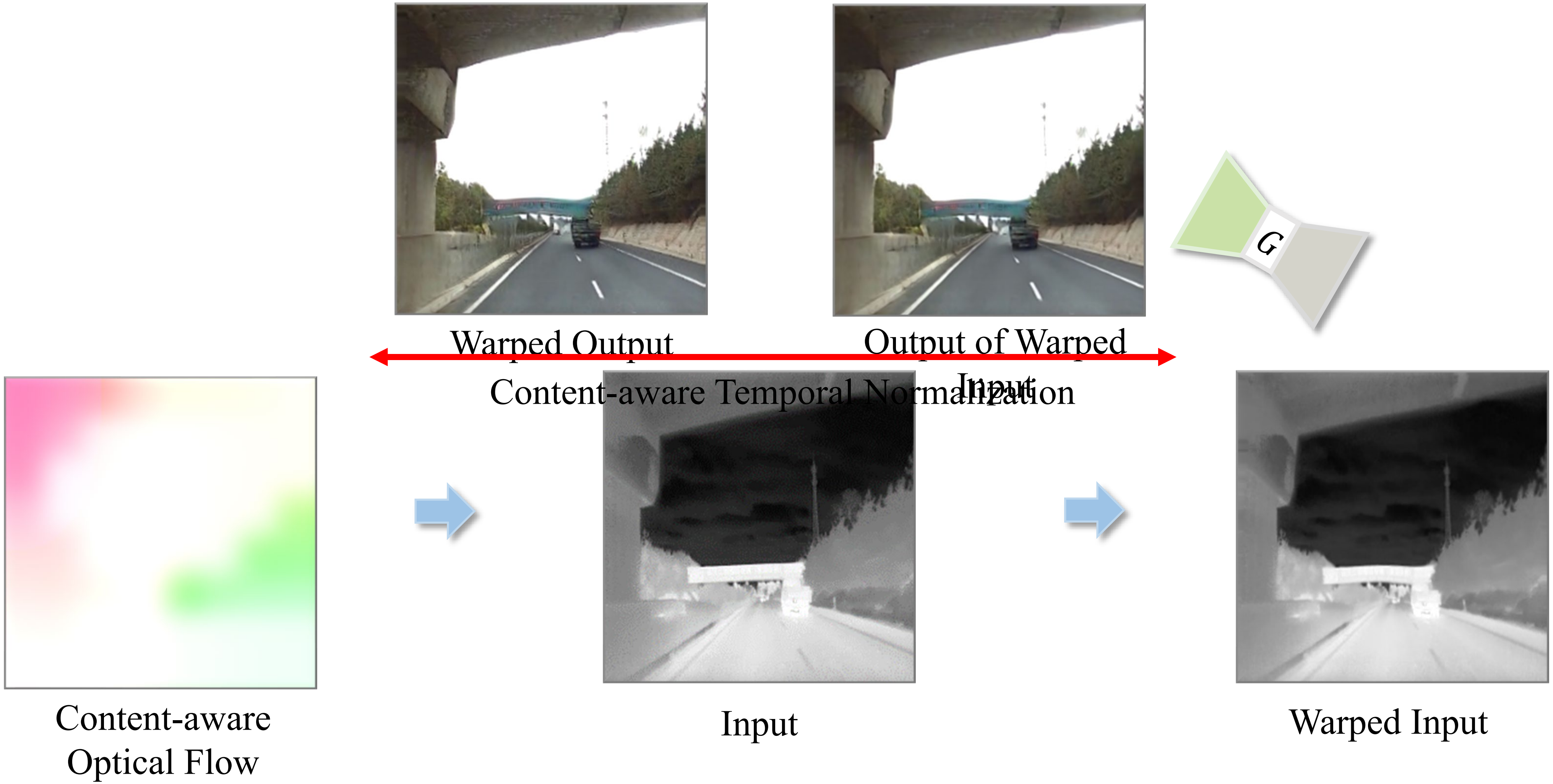
Method

➢ Content-aware Temporal Normalization

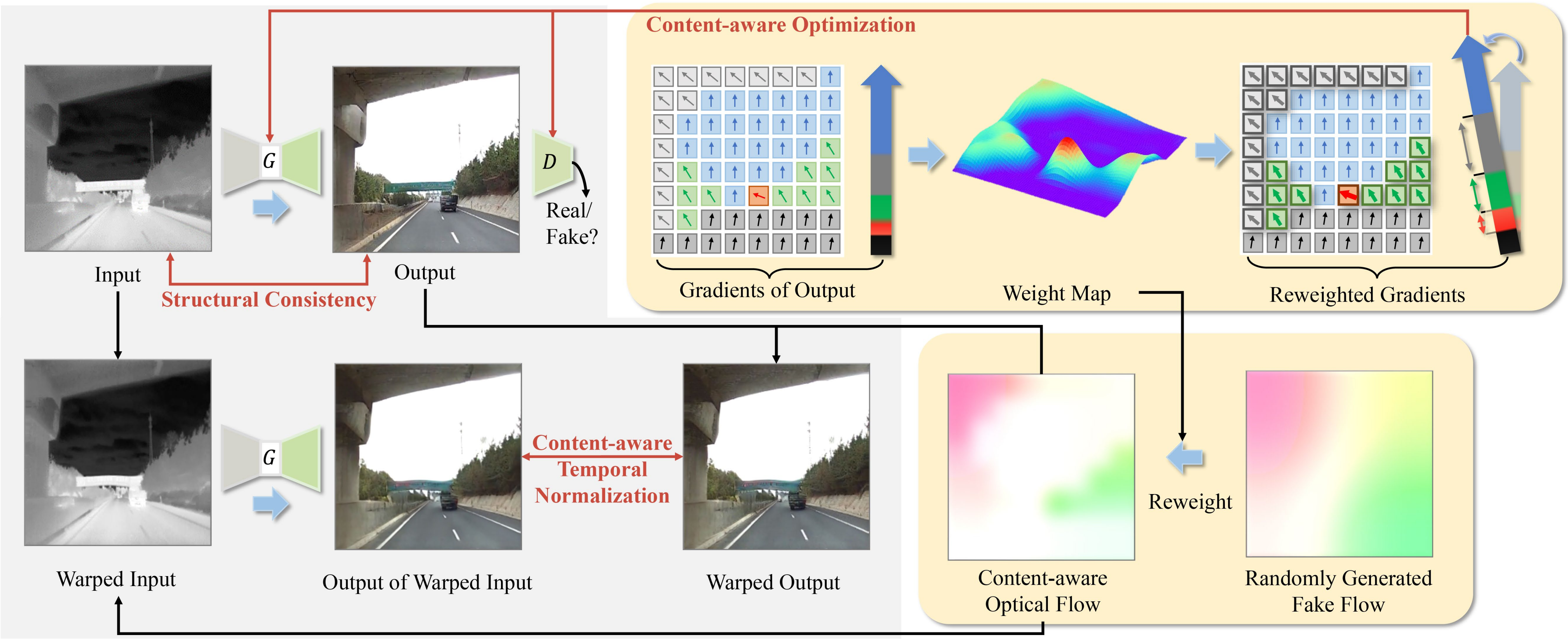


Method

- ➢ Content-aware Temporal Normalization



Overview of CPTrans Framework



Dataset: InfraredCity-Adverse

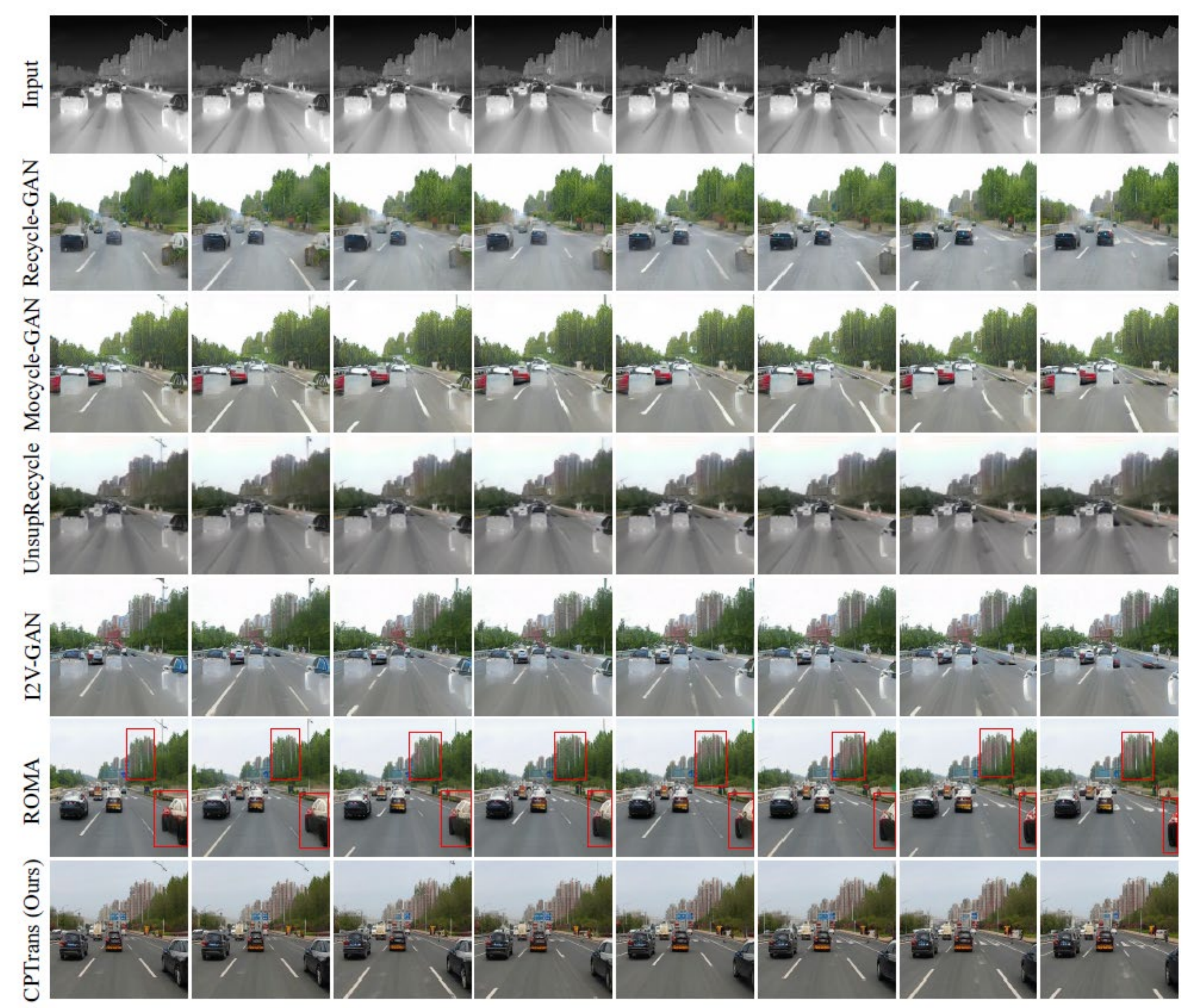
Rain

Snow

Experiments

Figure 4. Qualitative comparisons with different methods on diverse scenes, including clearday, overcast, rain, and snow, respectively, from top to bottom. Our outputs show cleaner and sufficient visual information compared with other results, especially on the adverse scenes. Additionally, our CPTrans dramatically improves the quality of content-rich patches. Best view when zoom in.

Experiments



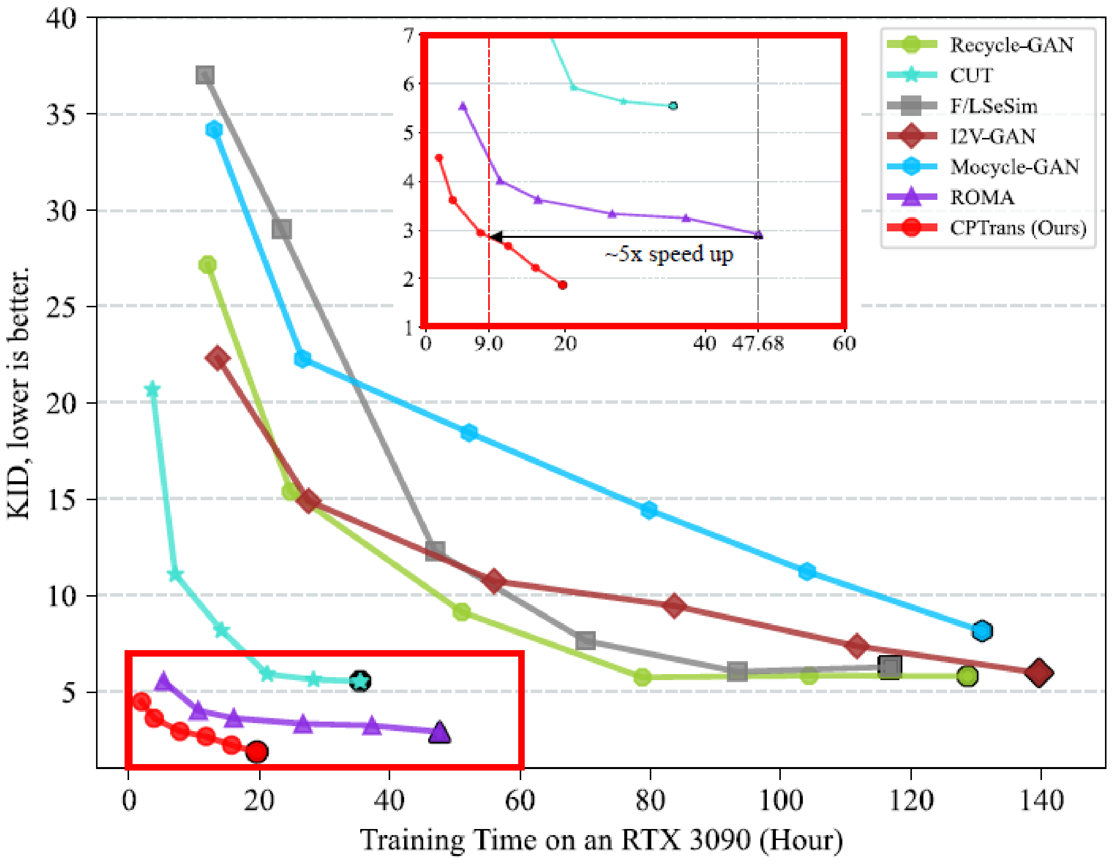
Experiments

Table 1. Comparison on InfraredCity-Lite. Our method achieve state-of-the-art scores with respect to both FID and KID on all scenes.

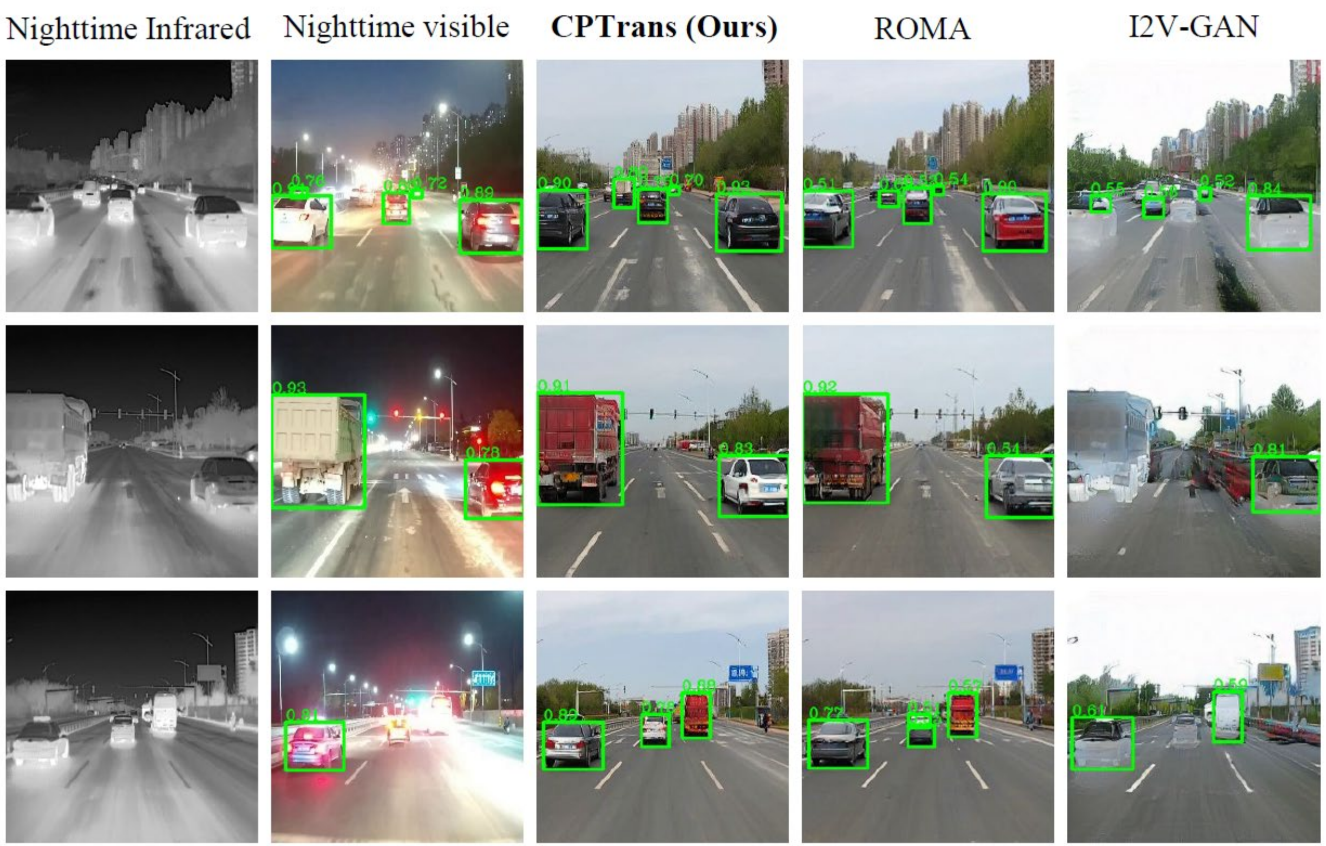
Method	Traffic														Monitoring	
	City						Highway						all			
	clear		overcast		all		clear		overcast		all					
	FID↓	KID↓	FID↓	KID↓	FID↓	KID↓	FID↓	KID↓	FID↓	KID↓	FID↓	KID↓	FID↓	KID↓	FID↓	KID↓
CUT [31]	0.5809	5.9174	0.5607	5.2185	0.6086	7.6174	0.4544	4.4742	0.5133	5.6331	0.4739	6.2903	0.4089	1.8202	0.9785	1.9126
CycleGAN [19]	0.6299	6.1114	0.5879	5.9409	0.7125	6.3001	0.4787	4.6475	0.5489	5.4571	0.4920	4.6128	0.4204	2.0781	0.8129	0.8728
F/LSeSim [51]	0.4984	3.9748	0.5369	6.1659	0.4834	4.3672	0.5108	5.9615	0.5288	5.2294	0.4809	4.9801	0.2724	1.9895	0.8984	0.8283
Recycle-GAN [3]	0.5942	5.3031	0.5974	6.2001	0.5969	5.3129	0.5173	6.1773	0.5998	8.2207	0.5101	5.2925	0.3431	3.0240	0.9433	0.9928
Mocycle-GAN [7]	0.5117	4.5128	0.5346	5.2772	0.5011	4.0732	0.5029	5.5982	0.5976	7.4907	0.4791	6.1446	0.3163	3.1973	0.7298	1.4637
UnsupRecycle [40]	0.7519	5.7289	0.9816	7.5554	0.8050	5.7288	0.4907	6.3411	0.5328	6.1268	0.4307	5.9160	0.3206	2.9047	0.8142	0.9785
I2V-GAN [25]	0.5052	4.2976	0.5574	5.9438	0.4649	4.1209	0.5064	5.9077	0.5105	6.3017	0.4515	4.7805	0.2872	2.4127	0.7039	1.8313
ROMA [49]	0.4018	3.8081	0.5149	5.7762	0.3929	3.3665	0.3325	3.9694	0.3823	4.9334	0.3444	4.3441	0.2002	0.6787	0.5488	0.7058
baseline	0.4332	4.0315	0.5258	5.8336	0.4038	3.5282	0.3474	4.3295	0.4245	5.3277	0.3916	4.5129	0.2324	1.0197	0.5731	0.8114
Ours w/o co	0.3890	3.2683	0.4762	5.0883	0.3891	3.3113	0.3453	3.3077	0.3712	4.3453	0.3389	3.7821	0.1835	0.4210	0.5303	0.6828
Ours w/o ctn	0.3824	3.3423	0.4779	4.9855	0.3867	3.5157	0.3267	3.3171	0.3642	3.9793	0.3343	3.8776	0.1816	0.2665	0.4949	0.6308
Ours	**0.3728**	**2.7573**	**0.4393**	**4.4034**	**0.3632**	**3.1693**	**0.3208**	**2.9591**	**0.3475**	**3.0938**	**0.3234**	**3.4399**	**0.1738**	**0.1826**	**0.4742**	**0.4570**

Experiments

Method	IRVI				InfraredCity-Adverse			
	Traffic		Monitoring		Rain		Snow	
	FID↓	KID↓	FID↓	KID↓	FID↓	KID↓	FID↓	KID↓
CUT [31]	0.5739	5.7356	1.0893	6.2651	0.5236	5.9084	0.5244	7.8449
CycleGAN [19]	0.6714	6.8587	0.8792	6.9381	0.5723	6.1525	0.5557	6.8426
F/LSeSim [51]	0.4321	5.3427	0.9232	5.0691	0.5775	6.0347	0.5926	6.4179
Recycle-GAN [3]	0.5255	4.9063	1.0609	5.0650	0.6133	5.8008	0.5730	5.9962
Mocycle-GAN [7]	0.7911	7.1380	1.0515	6.8002	0.8872	8.1459	0.6650	6.5410
UnsupRecycle [40]	0.6831	6.2315	0.9821	6.5123	0.7041	8.1372	0.5822	5.8795
I2V-GAN [25]	0.4425	4.5102	0.8715	4.6178	0.5917	5.6455	0.5693	5.5491
ROMA [49]	0.3467	3.0880	0.7334	3.3972	0.5577	2.5185	0.5393	4.9271
baseline	0.3652	3.6835	0.7689	3.5101	0.5751	2.9861	0.5520	5.1179
Ours w/o CO	0.3193	2.7356	0.7250	2.8762	0.5056	1.9855	0.5174	3.6446
Ours w/o CTN	0.3211	2.5720	0.7131	2.5886	0.4981	2.3112	0.4962	4.6301
Ours	**0.2936**	**1.9178**	**0.7004**	**2.3760**	**0.4760**	**1.7907**	**0.4952**	**2.6382**

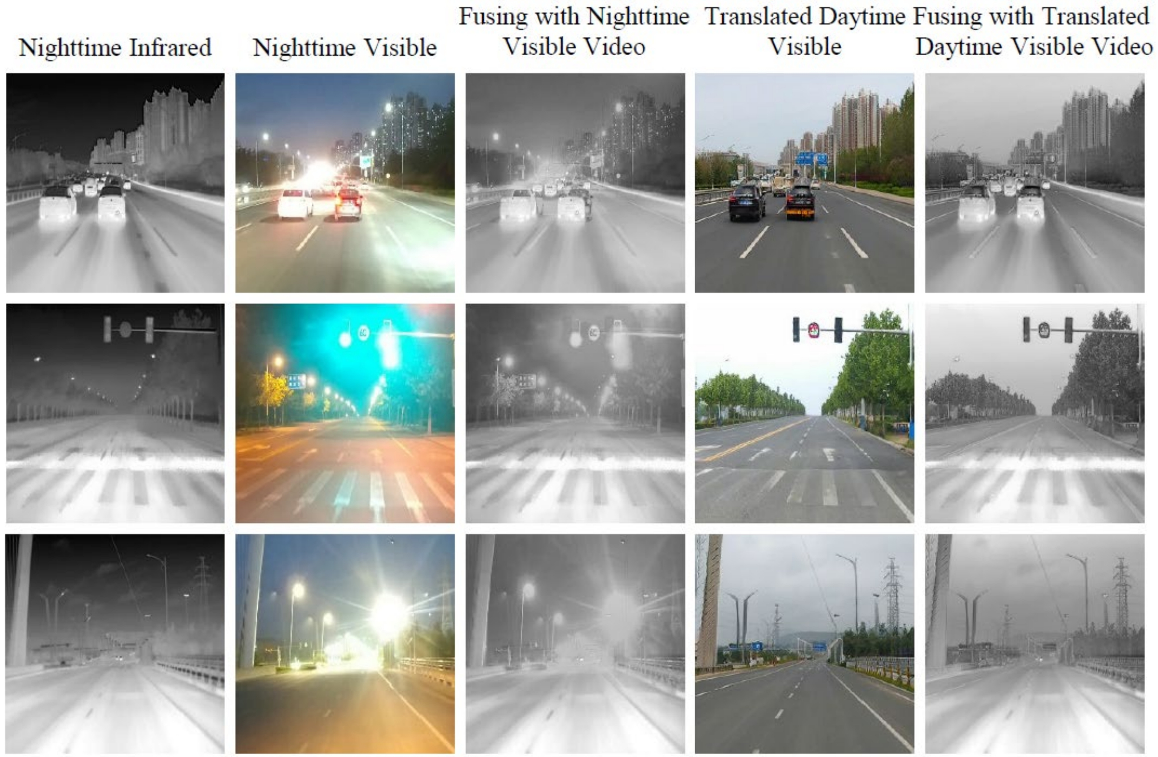


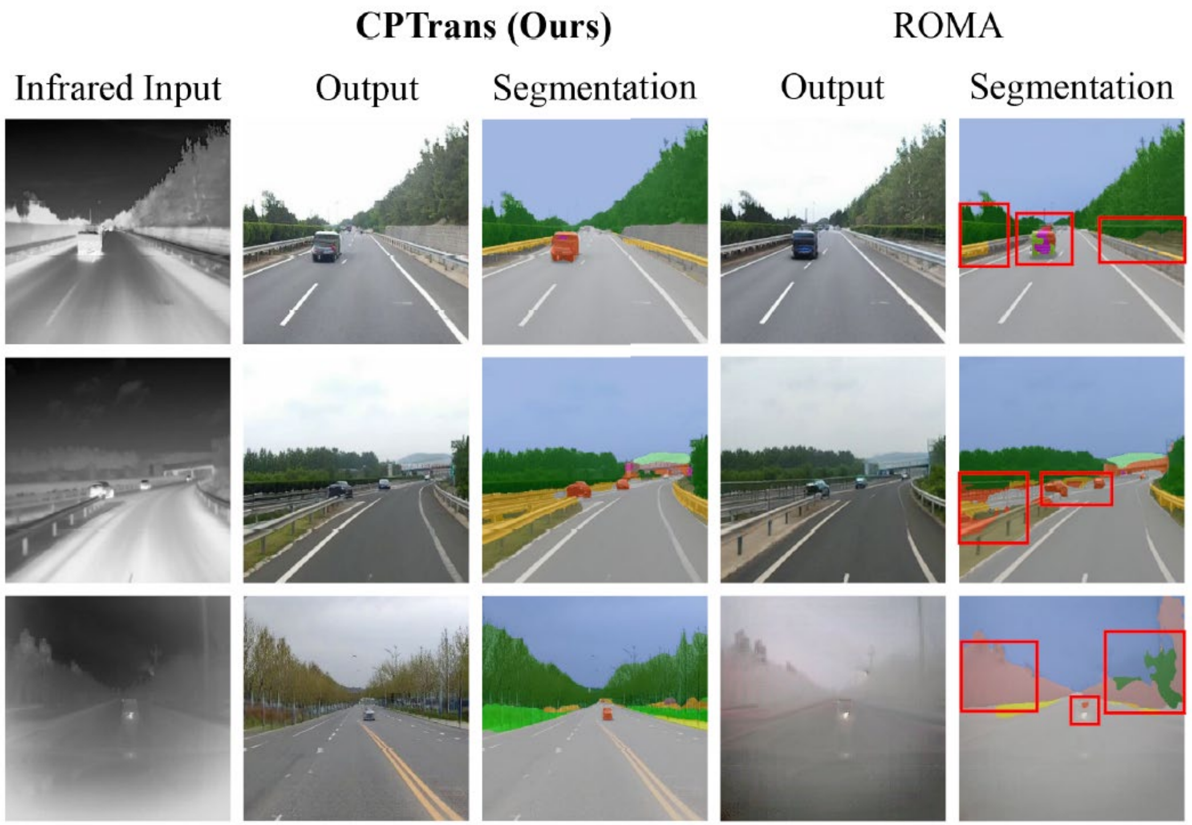
Experiments



Scenes	Nighttime Infrared	Nighttime Visible	I2V-GAN	ROMA	CPTrans (Ours)
AP	25.0	26.1	32.2	50.1	**58.1**

Experiments





Thanks!

GitHub: https://github.com/BIT-DA/I2V-Processing