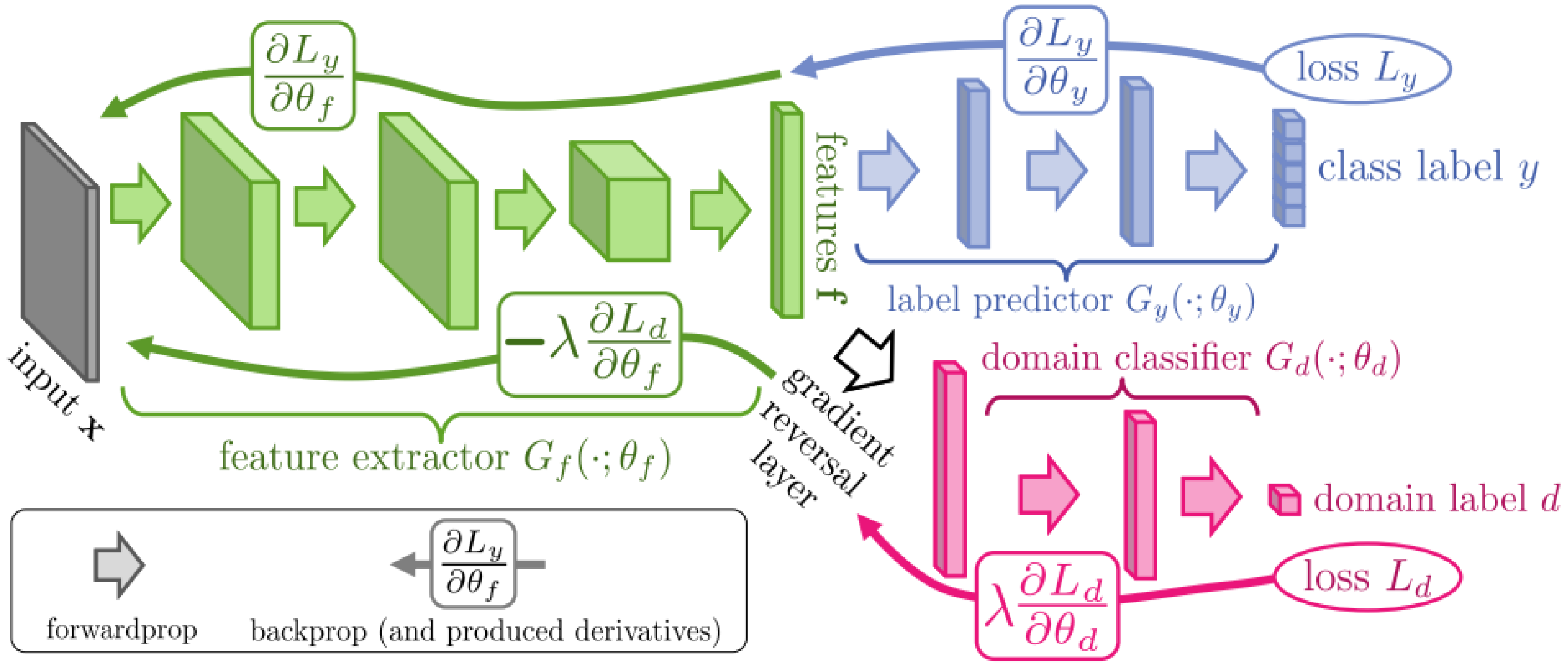
DLOW: Domain Flow for Adaptation and Generalization

Rui Gong[1], Wen Li[1], Yuhua Chen[1], Luc Van Gool[1,2]
[1]Computer Vision Laboratory, ETH Zurich [2]VISICS, ESAT/PSI, KU Leuven
gongr@student.ethz.ch {liwen, yuhua.chen, vangool}@vision.ee.ethz.ch

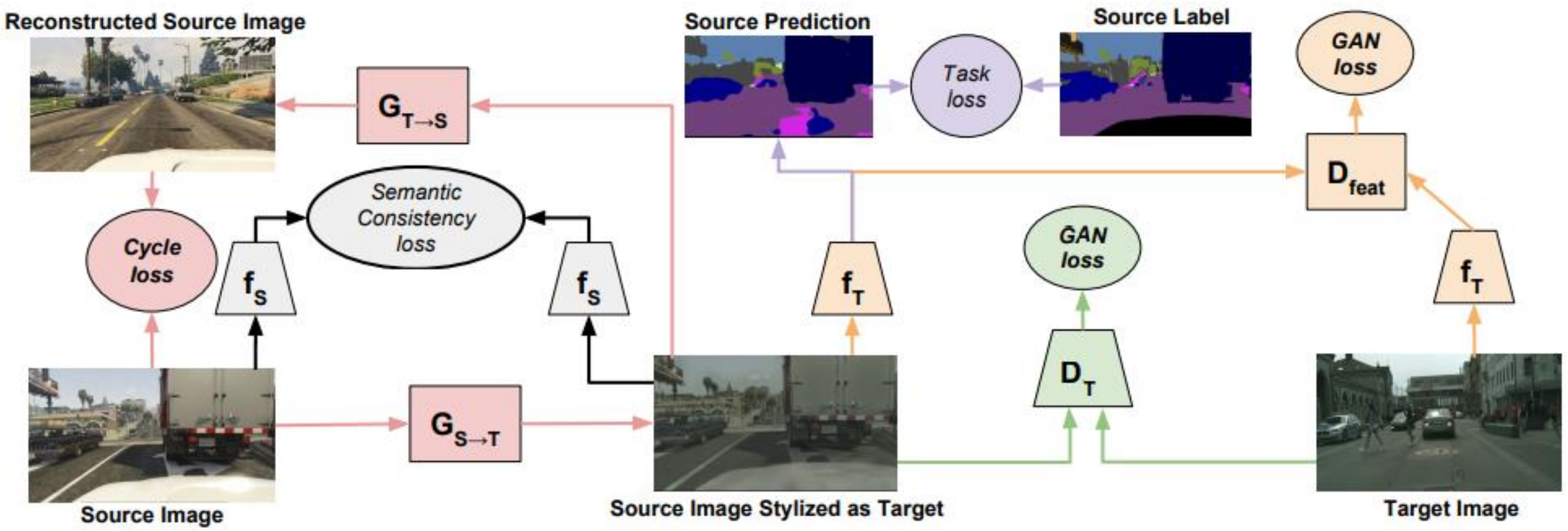
CVPR 2019 (Oral paper)

발표자 : 김용규

Domain adaptation vs Domain Generalization

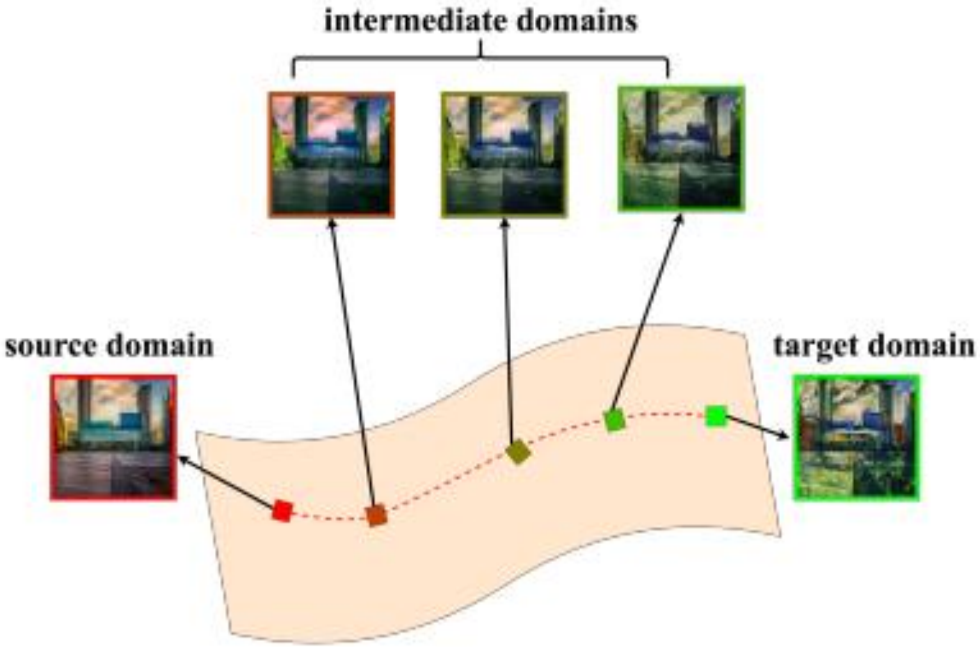


CyCADA

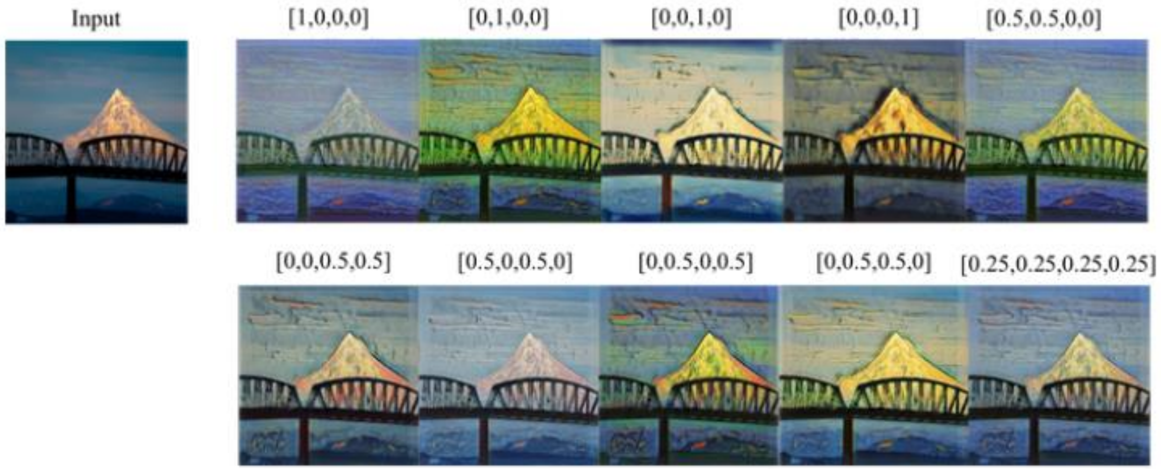


Reconstructed Source Image
$G_{T \to S}$
Source Prediction
Task loss
Source Label
GAN loss
D_{feat}
Cycle loss
f_S
Semantic Consistency loss
f_S
f_T
GAN loss
f_T
D_T
$G_{S \to T}$
Source Image
Source Image Stylized as Target
Target Image

Introduction



Source domain에서 target domain으로
가는 방향에 있는 intermediate domain 추출



다양한 domain의 style을 비율 별로 섞어서 적용

Contribution

- Intermediate domain image는 target domain image를 생성하도록 하는 pixel-level domain adaptation 성능 증가에 도움

- Style generalization은 one-to-one mapping 이지만, 여러 target domain과 관련이 있는 intermediate domain image를 생성할 수 있도록 한다. (Van Gogh, Monet, etc.)

Method

1. Conventional CycleGAN review

2. Formulate intermediate domain adaptation problem based on data distribution distance

3. Develope DLOW model based on CycleGAN

   - How to improve existing domain adaptation models with the images generated from DLOW model

   - How to transfer images into arbitrarily mixed styles when there are multiple target domains

Method (CycleGAN)

$$G_{ST} : \mathcal{S} \rightarrow \mathcal{T}$$
$$G_{TS} : \mathcal{T} \rightarrow \mathcal{S}$$

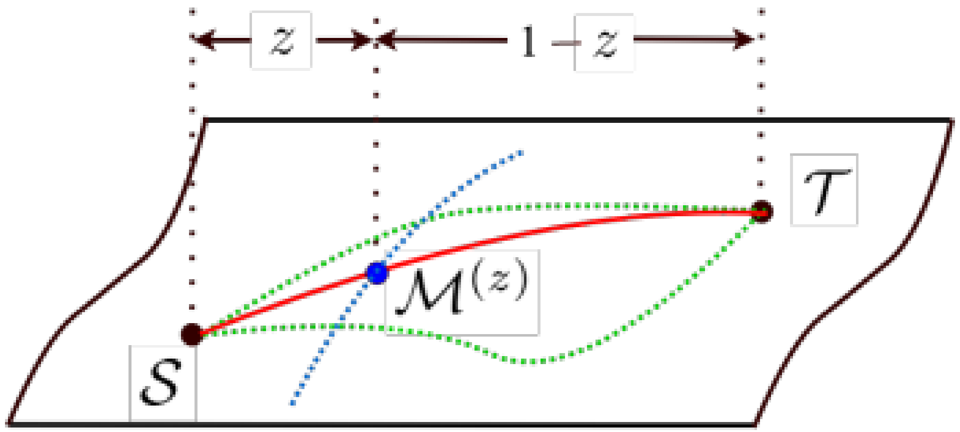
Adversarial training module :

$$\min_{G_{ST}} \max_{D_T} \quad \mathbb{E}_{\mathbf{x}^t \sim P_T} \left[\log(D_T(\mathbf{x}^t))\right] + \mathbb{E}_{\mathbf{x}^s \sim P_S} \left[\log(1 - D_T(G_{ST}(\mathbf{x}^s)))\right]$$

Reconstruction module :

$$\min_{G_{ST}} \quad \mathbb{E}_{\mathbf{x}^s \sim P_S} \left[\|G_{TS}(G_{ST}(\mathbf{x}^s)) - \mathbf{x}^s\|_1\right]$$

Method (Modeling Intermediate Domains)



Intermediate domain : $\mathcal{M}^{(z)}$

Domainness parameter : $z \in [0, 1]$

$z = 0$, $\mathcal{M}^{(z)}$ is source domain

$z = 1$, $\mathcal{M}^{(z)}$ is source domain

We expect the domain flow $\mathcal{M}^{(z)}$ to be the shortest geodesic path

$$\frac{dist\left(P_S, P_M^{(z)}\right)}{dist\left(P_T, P_M^{(z)}\right)} = \frac{z}{1-z}$$

$$\mathcal{L} = (1-z) \cdot dist\left(P_S, P_M^{(z)}\right) + z \cdot dist\left(P_T, P_M^{(z)}\right)$$

Method (DLOW Model)

DLOW model : $G_{ST}(\mathbf{x}^s, z) : \mathcal{S} \times \mathcal{Z} \rightarrow \mathcal{M}^{(z)}$

Adversarial Loss :

$D_S(\mathbf{x})$ to distinguish $\mathcal{M}^{(z)}$ and $\mathcal{S}$

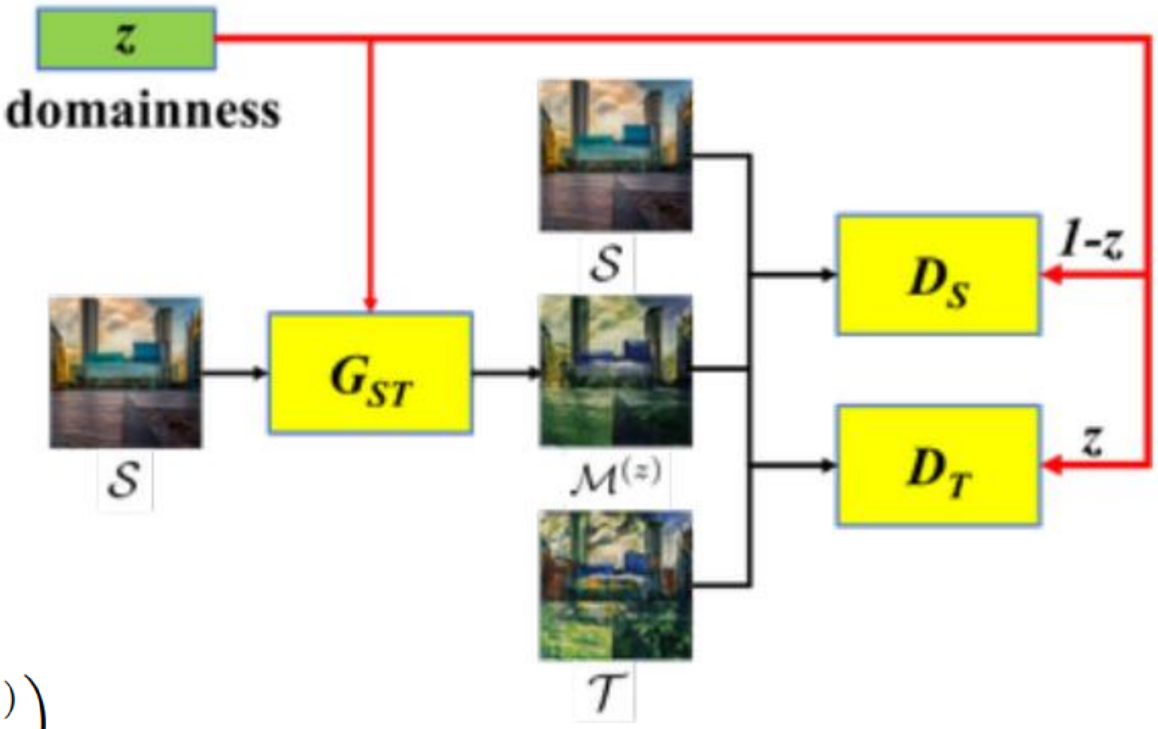
$D_T(\mathbf{x})$ to distinguish $\mathcal{M}^{(z)}$ and $\mathcal{T}$

$$\mathcal{L}_{adv}(G_{ST}, D_S) = \mathbb{E}_{\mathbf{x}^s \sim P_S}[\log(D_S(\mathbf{x}^s))] + \mathbb{E}_{\mathbf{x}^s \sim P_S}[\log(1 - D_S(G_{ST}(\mathbf{x}^s, z)))]$$

$$\mathcal{L}_{adv}(G_{ST}, D_T) = \mathbb{E}_{\mathbf{x}^t \sim P_T}[\log(D_T(\mathbf{x}^t))] + \mathbb{E}_{\mathbf{x}^s \sim P_S}[\log(1 - D_T(G_{ST}(\mathbf{x}^s, z)))]$$

$$\mathcal{L} = (1-z) \cdot dist\left(P_S, P_M^{(z)}\right) + z \cdot dist\left(P_T, P_M^{(z)}\right)$$

$$\rightarrow \mathcal{L}_{adv} = (1-z)\mathcal{L}_{adv}(G_{ST}, D_S) + z\mathcal{L}_{adv}(G_{ST}, D_T)$$



Method (DLOW Model)

First, the domainness parameter z is taken as the input of the generator G_{ST}. This is implemented with the Conditional Instance Normalization (CN) layer [1, 23]. We first use one deconvolution layer to map the domainness parameter z to the vector with dimension $(1, 16, 1, 1)$, and then use this vector as the input for the CN layer. Moreover,

3.3. Conditional Instance Normalization

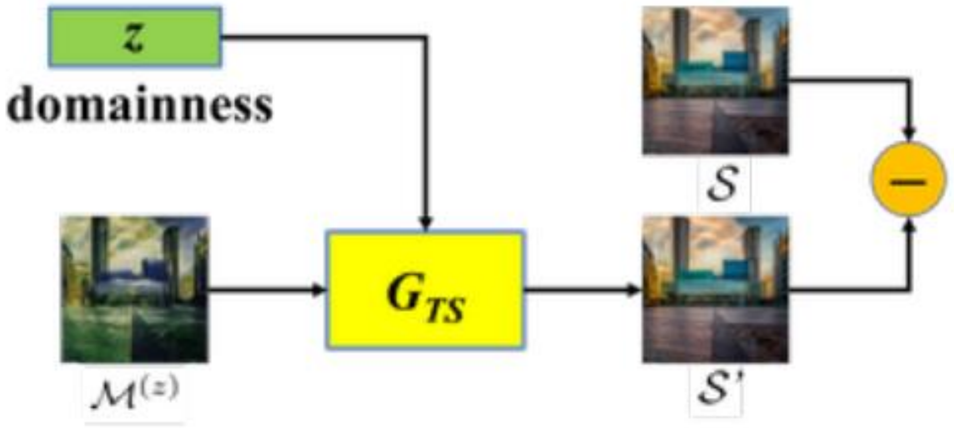
Instead of learning a single set of affine parameters γ and β, Dumoulin *et al*. [11] proposed a *conditional instance normalization* (CIN) layer that learns a different set of parameters γ^s and β^s for each style s:

$$\mathrm{CIN}(x; s) = \gamma^s \left(\frac{x - \mu(x)}{\sigma(x)} \right) + \beta^s \qquad (7)$$

Method (DLOW Model)

Image Cycle Consistency Loss :

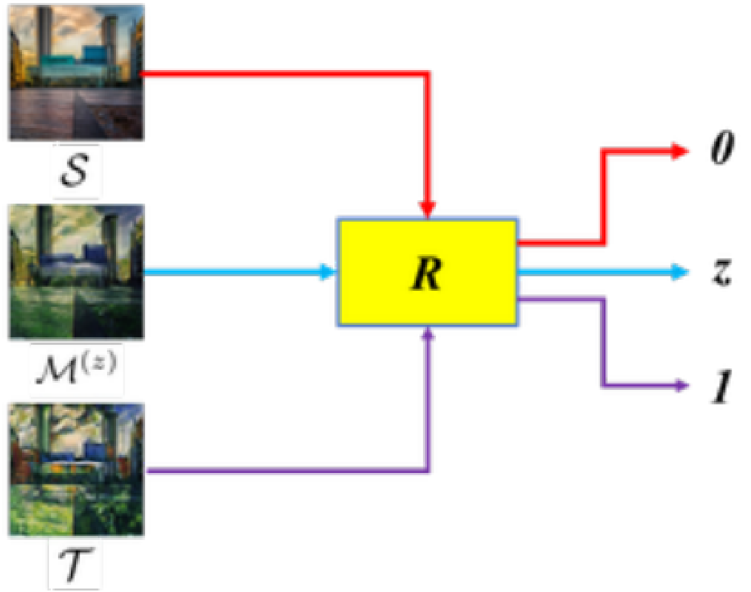
$$L_{cyc} = \mathbb{E}_{\mathbf{x}^s \sim P_s} [\| G_{TS}(G_{ST}(\mathbf{x}^s, z), z) - \mathbf{x}^s \|_1]$$



Domainness Cycle Consistency Loss :

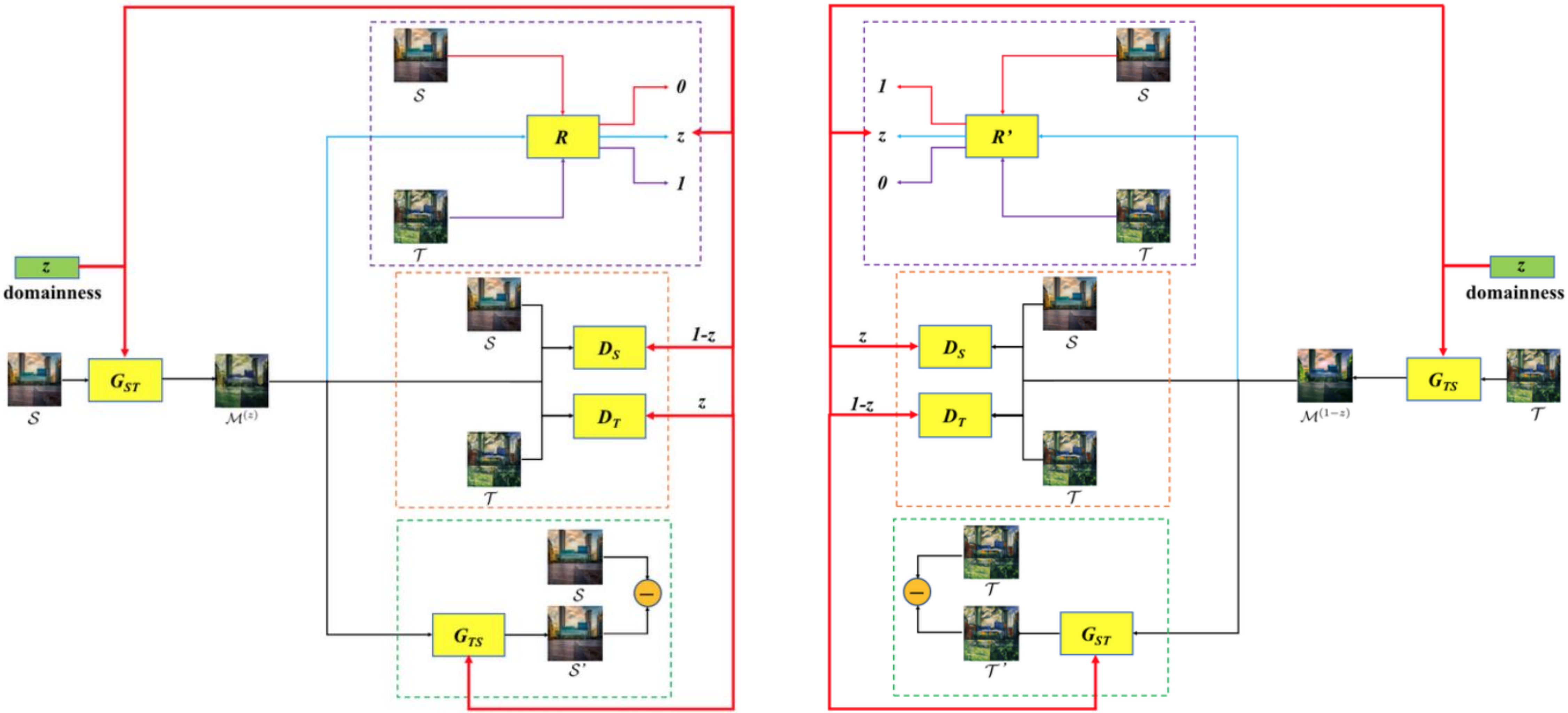
regressor $R_S : \mathcal{M}^{(z)} \rightarrow z$

$$\mathcal{L}_{dns} = -\mathbb{E}_{\mathbf{x}^t \sim P_T} [\log(R_S(\mathbf{x}^t))] \\ -\mathbb{E}_{\mathbf{x}^s \sim P_S} [\log(1 - R_S(\mathbf{x}^s)] \\ +\mathbb{E}_{\mathbf{x}^s \sim P_S} [\| R_S(G_{ST}(\mathbf{x}^s, z)) - z \|_2]$$



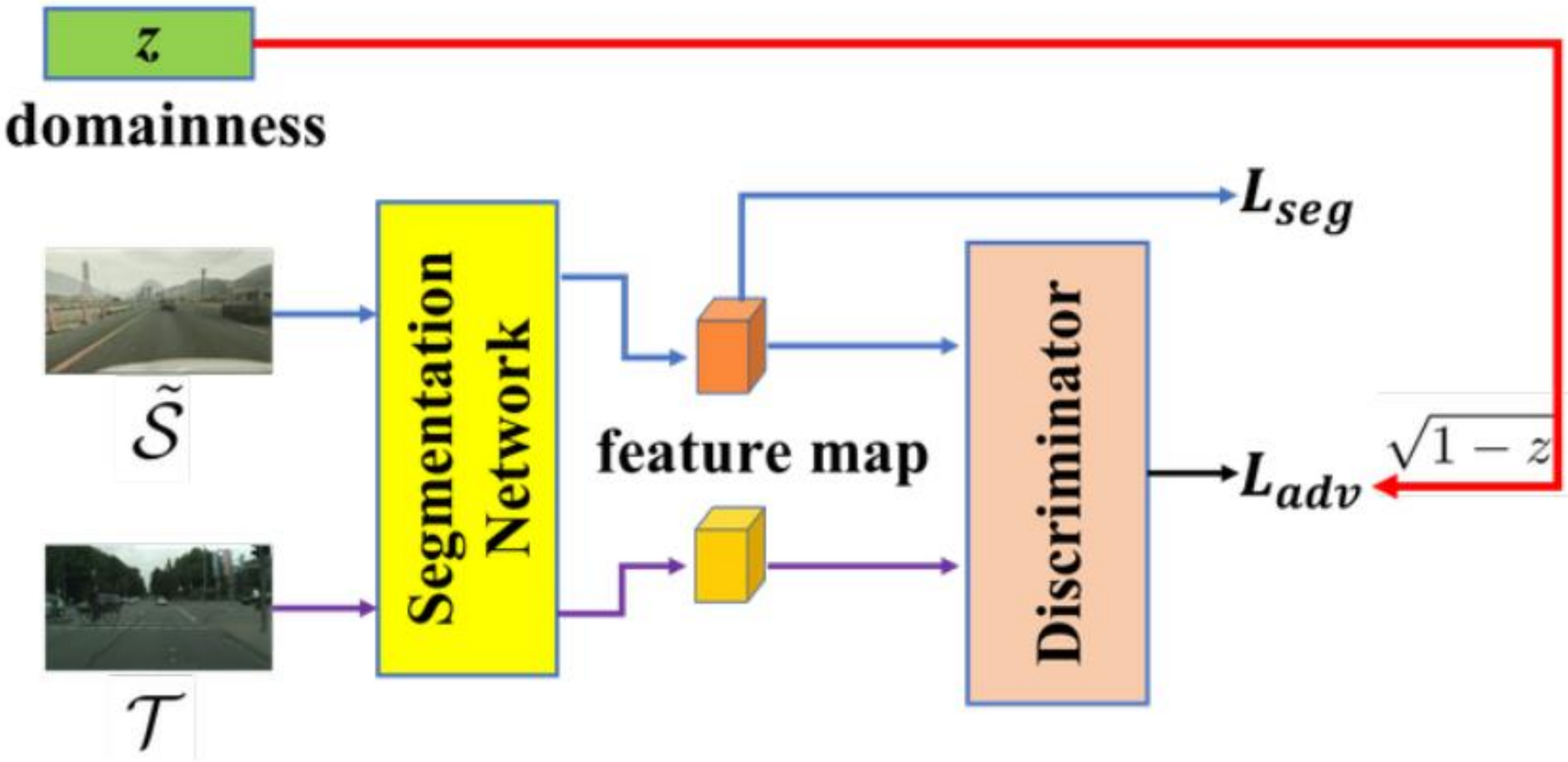
Method (DLOW Model)

Full Objective : $\mathcal{L} = \mathcal{L}_{adv} + \lambda_1 \mathcal{L}_{cyc} + \lambda_2 \mathcal{L}_{dns}$



Method (Boosting Domain Adaptation Models)

$$\tilde{S} = \{(\tilde{\mathbf{x}}_i^s, y_i)|_{i=1}^n\} \quad \tilde{\mathbf{x}}_i^s = G_{ST}(\mathbf{x}_i^s, z_i)$$



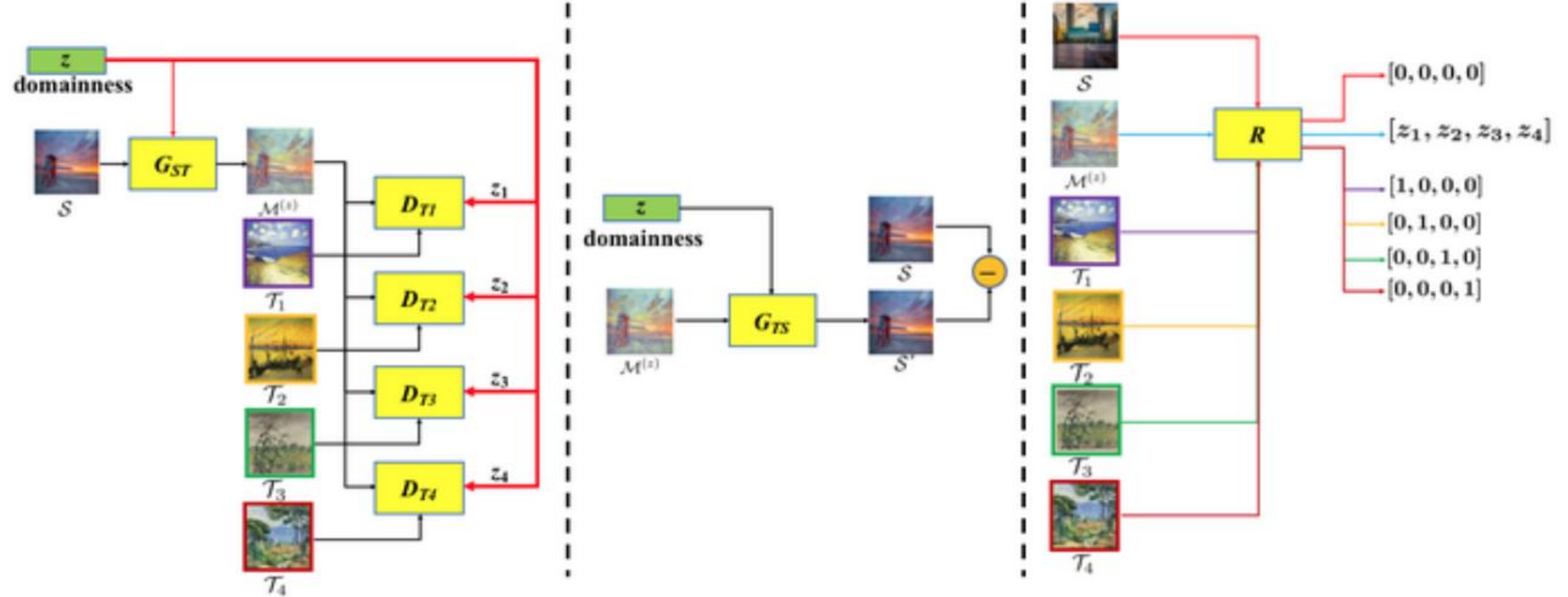
Training data is helpful to learn a domain-invariant models

Method (Style Generalization)

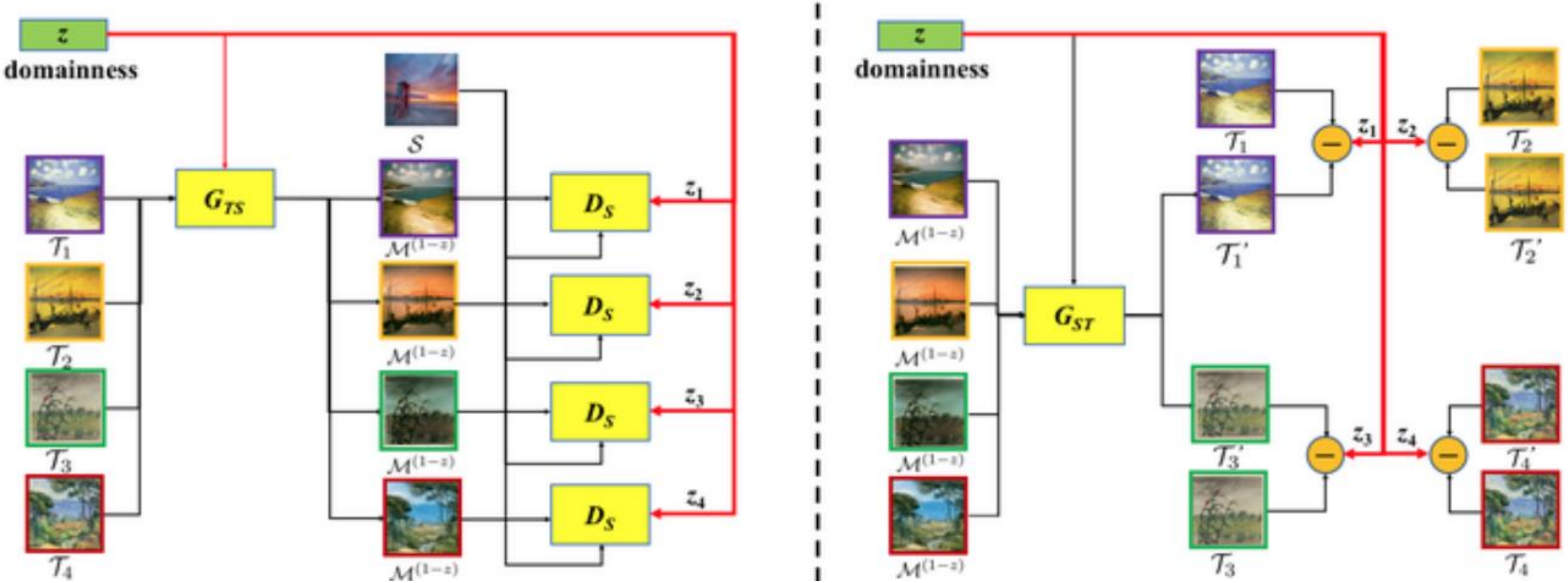
K target domains $\mathcal{T}_1, \ldots, \mathcal{T}_K$

K-dim vector $\mathbf{z} = [z_1, \ldots, z_K]'$ $\sum_{k=1}^{K} z_k = 1$

$$\mathcal{L} = \sum_{k=1}^{K} z_k \cdot dist(P_M, P_{T_k}), \quad \text{s.t.} \quad \sum_{1}^{K} z_k = 1$$



Method (Style Generalization)



Experiments (Boosting Domain Adaptation Models)

(a) $z = 0$ (b) $z = 0.3$ (c) $z = 0.6$ (d) $z = 0.8$ (c) $z = 1$

Figure 5: Examples of intermediate domain images from GTA5 to Cityscapes

Table 1: Results of segmentation segmentation on the CityScapes dataset based on DeepLab-v2 model with ResNet-101 backbone using the images translated with different models.

GTA5 → Cityscapes																				
Method	road	sidewalk	building	wall	fence	pole	traffic light	traffic sign	vegetation	terrian	sky	person	rider	car	truck	bus	train	motorbike	bicycle	mIoU
NonAdapt[50]	75.8	16.8	77.2	12.5	**21.0**	25.5	**30.1**	20.1	81.3	24.6	70.3	53.8	**26.4**	49.9	17.2	25.9	**6.5**	**25.3**	**36.0**	36.6
CycleGAN[19]	81.7	27.0	**81.7**	**30.3**	12.2	28.2	25.5	27.4	82.2	**27.0**	77.0	**55.9**	20.5	**82.8**	30.8	38.4	0.0	18.8	32.3	41.0
DLOW($z = 1$)	**88.5**	**33.7**	80.7	26.9	15.7	27.3	27.7	**28.3**	80.9	26.6	74.1	52.6	25.1	76.8	30.5	27.2	0.0	15.7	**36.0**	40.7
DLOW	87.1	33.5	80.5	24.5	13.2	**29.8**	29.5	26.6	**82.6**	26.7	**81.8**	**55.9**	25.3	78.0	**33.5**	**38.7**	0.0	22.9	34.5	**42.3**

Experiments (Boosting Domain Adaptation Models & Style Generalization)

Table 2: Comparison of the performance of AdaptSeg-Net [50] when using original source images and intermediate domain images translated with our DLOW model for semantic segmention under domain adaptation (1st column) and domain generalization (2nd to 4th columns) scenarios

	Cityscapes	KITTI	WildDash	BDD100K
Original [50]	42.4	30.7	18.9	37.0
DLOW	**44.8**	**36.6**	**24.9**	**39.1**

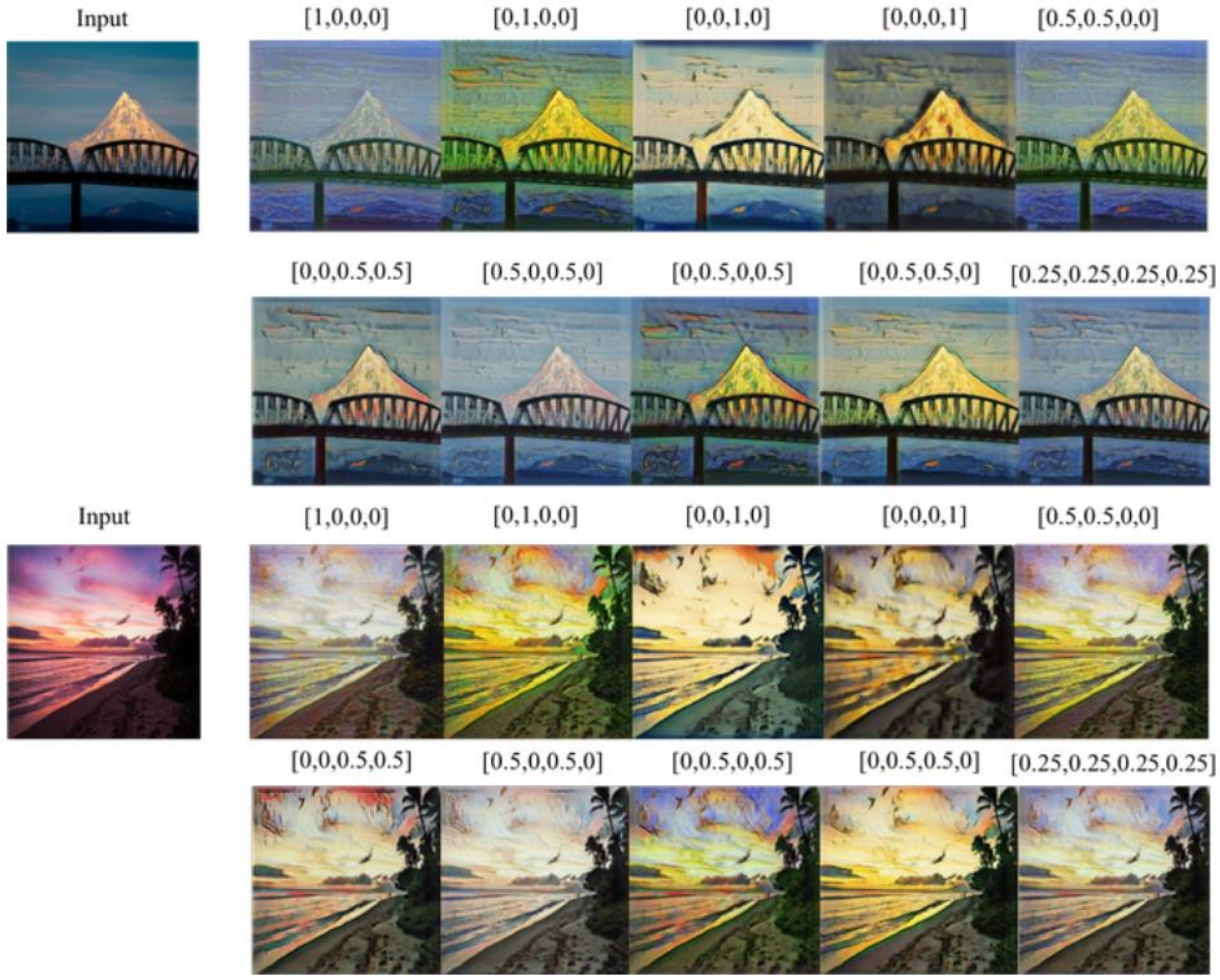


Figure 6: Examples of style generalization results. The vectors above each images are the domainness.

Experiments (Style Generalization)

Thank you