RobustNet: Improving Domain Generalization in Urban-Scene Segmentation via Instance Selective Whitening

CVPR 2021, Oral

Sungha Choi[*1,3] Sanghun Jung[*2] Huiwon Yun[4] Joanne T. Kim[3]
Seungryong Kim[3] Jaegul Choo[2]

[1]LG AI Research [2]KAIST [3]Korea University [4]Sogang University

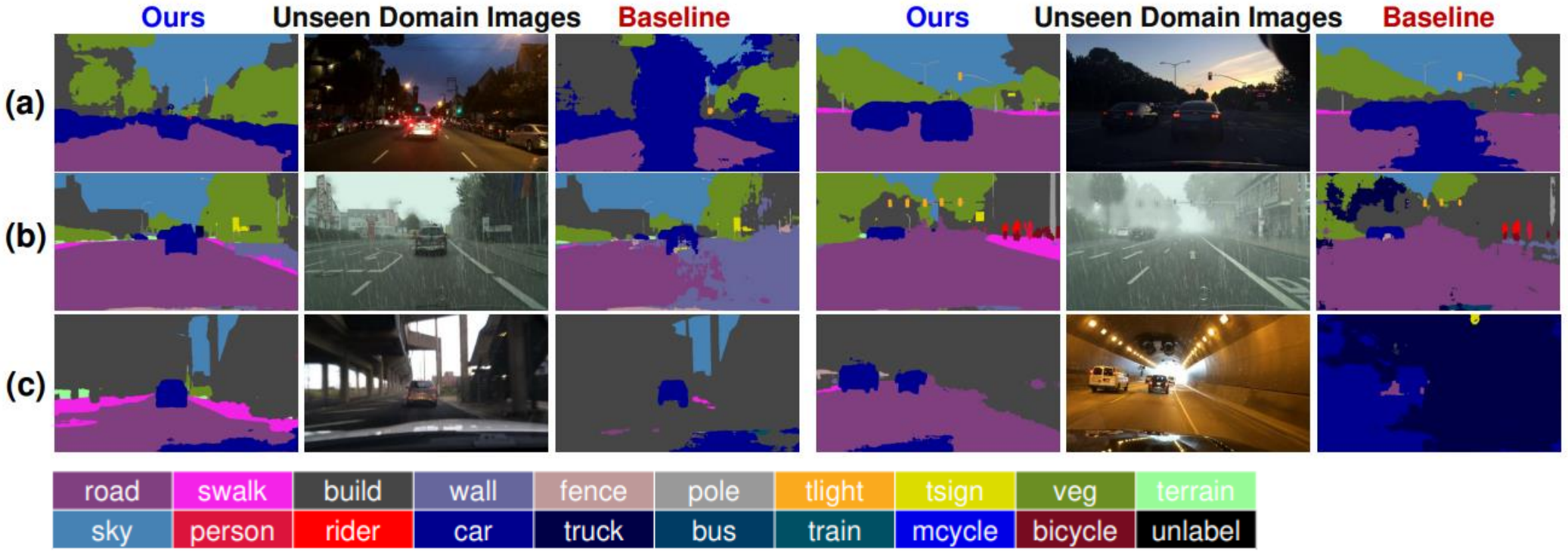
2023.06.30 내부미팅

이소영

Contents

Abstract

- **Instance selective whitening (ISW) loss**: robustness of the segmentation networks for unseen domains
  - Disentangles the **domain-specific style** and **domain-invariant content** : using feature covariance
  - Selectively **removes only the style** information causing domain shift
- Reasonable predictions for (a) low-illuminated, (b) rainy, and (c) unseen structures



Abstract

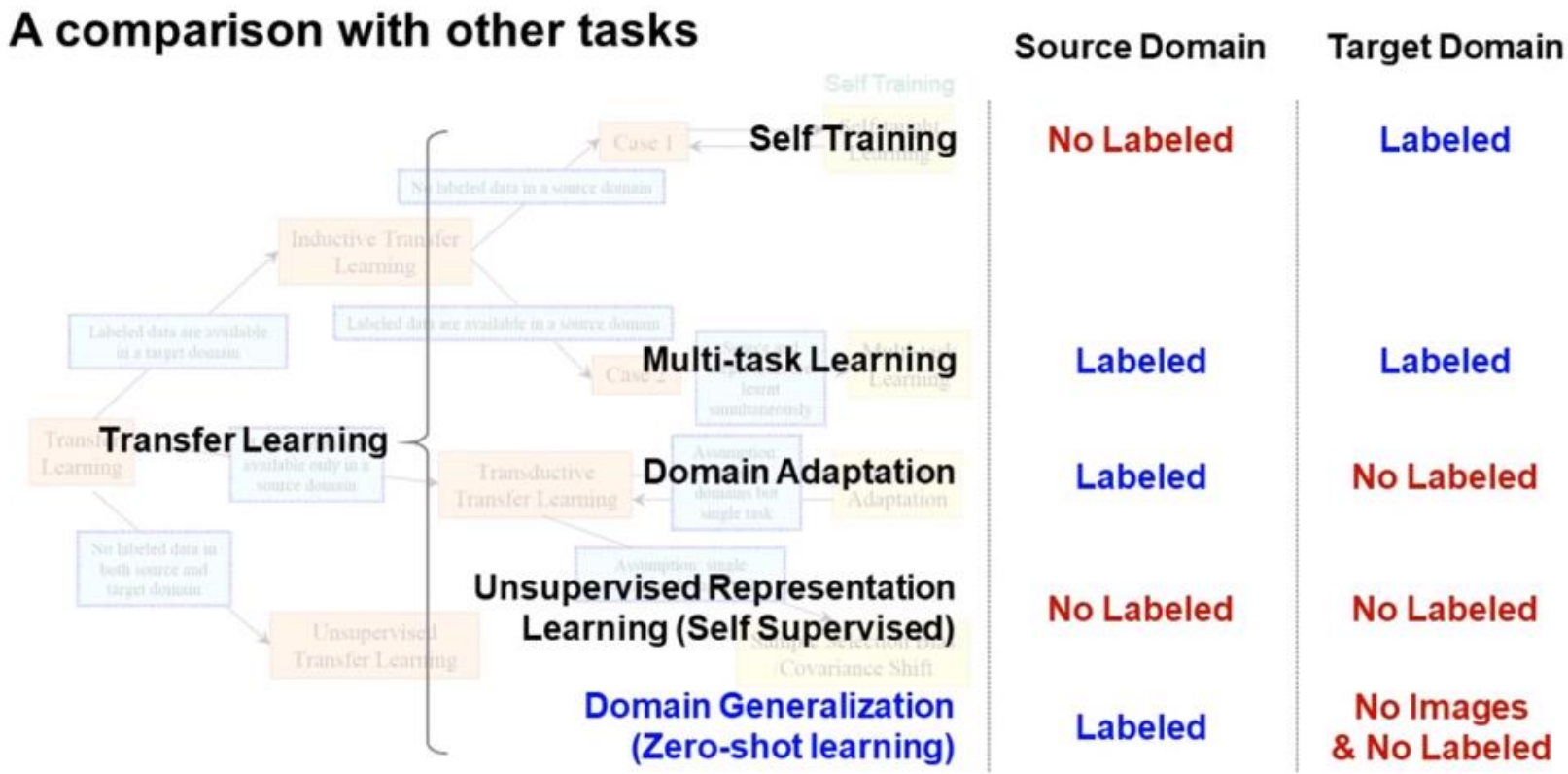
- **Instance selective whitening (ISW) loss**: robustness of the segmentation networks for unseen domains
  - Disentangles the **domain-specific style** and **domain-invariant content** : using feature covariance
  - Selectively **removes only the style** information causing domain shift
- Reasonable predictions for (a) low-illuminated, (b) rainy, and (c) unseen structures
- **Simple to use, effective**
- Urban-scene segmentation experiments show the superiority of our approach to existing work

Introduction & Related Work

- When deploying unseen data, fail to perform properly due to **domain shift**
- Need to **reducing the domain gap** between source and target domain

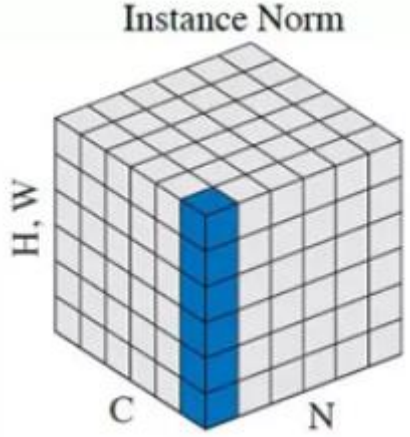
Introduction & Related Work

- Domain Adaptation (DA), Domain Generalization (DG)
  - DA: access to the samples in the target domain



Introduction & Related Work

- Domain Generalization (DG): robustness to arbitrary unseen domain
  - Shared across multiple source domains: costly, labor-intensive, highly depends on the number of source datasets
- More effective?
  - IBN-Net: exploiting **instance normalization**
  - Instance normalization: standardizes features while **not considering the correlation between channels**
  - **Feature covariance** contains **domain-specific style** (texture, color) → instance norm not sufficient



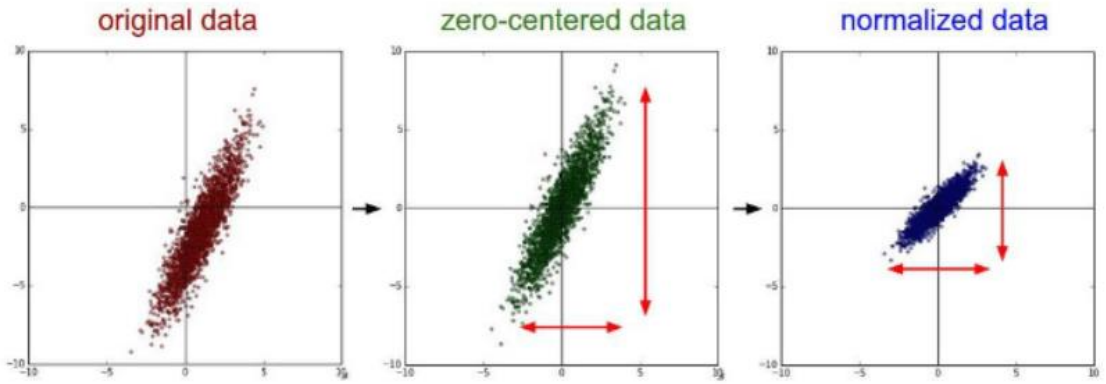
Introduction & Related Work

- **Whitening transformation**: removes feature correlation, each feature have unit variance
  - Feature whitening effectively eliminates domain-specific style information
  - Shown in image translation, style transfer, domain adaptation
  - **Not yet fully explored in DG**
- Semantic segmentation in DG: majority of the DG methods mainly focused on image classification
- Main scope of this paper
  - Adopting whitening transformation to DG
  - Decoupling the two factors: **domain-specific style** and **domain-invariant content**
  - **Selectively removing the domain-specific style**
  - Proposed loss can **easily be used**
  - Show **superiority to urban-scene segmentation** in DG (qualitative, quantitative)

Preliminaries

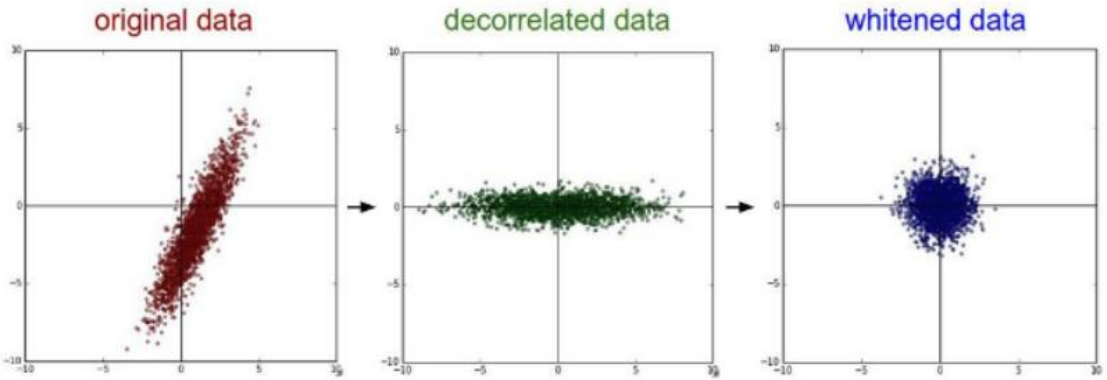
- Normalization (Standardization)
  - 평균=0, 표준편차=1이 되게 변환

$$Z = \frac{X - \mu}{\sigma}$$



- **Whitening**
  - **데이터의 평균=0, 공분산=단위행렬** 로 갖는 정규분포 형태로 변환 $\mathbf{x} \sim \mathcal{N}_D(\boldsymbol{\mu}, \boldsymbol{\Sigma}) \overset{\text{whiten}}{\mapsto} \mathbf{z} \sim \mathcal{N}_D(0, \mathbf{I})$
  - Decorrelation + Standardization

$$\begin{aligned} Z &= W(X - \mu), \\ W^\intercal W &= \Sigma^{-1} \\ \Sigma &= U \Lambda U^\intercal \\ \Sigma^{-1/2} &= U \Lambda^{-1/2} U^\intercal, \\ W &= Q U \Lambda^{-1/2} U^\intercal, = Q\, \Sigma^{-1/2} \end{aligned}$$



https://cvml.tistory.com/5
https://joelouismarino.github.io/posts/2017/08/statistical_whitening/

Preliminaries

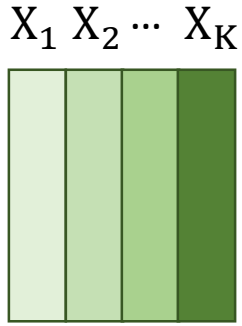
- **Covariance matrix**

Definition Let X be a $K \times 1$ random vector. The covariance matrix of X, or variance-covariance matrix of X, denoted by $\mathrm{Var}[X]$, is defined as follows:

$$\mathrm{Var}[X] = \mathrm{E}[(X - \mathrm{E}[X])(X - \mathrm{E}[X])^{\mathsf{T}}]$$

Let X_1, ..., X_K denote the K components of the vector X.

From the definition of $\mathrm{Var}[X]$, it can easily be seen that $\mathrm{Var}[X]$ is a $K \times K$ matrix with the following structure:



$$\mathrm{Var}[X] = \mathrm{E}\begin{bmatrix} (X_1 - \mathrm{E}[X_1])(X_1 - \mathrm{E}[X_1]) & \dots & (X_1 - \mathrm{E}[X_1])(X_K - \mathrm{E}[X_K]) \\ \vdots & \ddots & \vdots \\ (X_K - \mathrm{E}[X_K])(X_1 - \mathrm{E}[X_1]) & \dots & (X_K - \mathrm{E}[X_K])(X_K - \mathrm{E}[X_K]) \end{bmatrix}$$

$$= \begin{bmatrix} \mathrm{E}\left[(X_1 - \mathrm{E}[X_1])^2\right] & \dots & \mathrm{E}[(X_1 - \mathrm{E}[X_1])(X_K - \mathrm{E}[X_K])] \\ \vdots & \ddots & \vdots \\ \mathrm{E}[(X_K - \mathrm{E}[X_K])(X_1 - \mathrm{E}[X_1])] & \dots & \mathrm{E}\left[(X_K - \mathrm{E}[X_K])^2\right] \end{bmatrix}$$

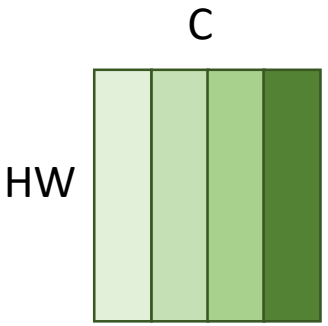
$$= \begin{bmatrix} \mathrm{Var}[X_1] & \dots & \mathrm{Cov}[X_1, X_K] \\ \vdots & \ddots & \vdots \\ \mathrm{Cov}[X_K, X_1] & \dots & \mathrm{Var}[X_K] \end{bmatrix}$$

https://www.statlect.com/fundamentals-of-probability/covariance-matrix

Preliminaries

- Whitening transformation (WT)

$$\mathbf{X} \in \mathbb{R}^{C \times HW}$$



- WT is a linear transformation that makes
  - Variance term of each channel equal to 1
  - Covariances between each pair of channels equal to 0

$$\Rightarrow \tilde{\mathbf{X}} \cdot \tilde{\mathbf{X}}^{\top} = (HW) \cdot \mathbf{I} \in \mathbb{R}^{C \times C}$$

Covariance matrix

$$\mathrm{Var}[X] = \mathrm{E}[(X - \mathrm{E}[X])(X - \mathrm{E}[X])^{\top}]$$

$$C = \frac{1}{m} X X^{T}$$

Covariance matrix

$$\begin{bmatrix} \mathrm{Var}[X_1] & \cdots & \mathrm{Cov}[X_1, X_K] \\ \vdots & \ddots & \vdots \\ \mathrm{Cov}[X_K, X_1] & \cdots & \mathrm{Var}[X_K] \end{bmatrix}$$

https://www.statlect.com/fundamentals-of-probability/covariance-matrix

Preliminaries

- **Whitening transformation (WT)**

$$\mathbf{x} \sim \mathcal{N}_D(\boldsymbol{\mu}, \boldsymbol{\Sigma}) \overset{\text{whiten}}{\mapsto} \mathbf{z} \sim \mathcal{N}_D(0, \mathbf{I})$$

$$\tilde{\mathbf{X}} = \boldsymbol{\Sigma}_{\mu}^{-\frac{1}{2}} \left(\mathbf{X} - \boldsymbol{\mu} \cdot \mathbf{1}^{\top} \right) \qquad \boldsymbol{\mu} = \frac{1}{HW} \mathbf{X} \cdot \mathbf{1} \in \mathbb{R}^{C \times 1}$$

$$\boldsymbol{\Sigma}_{\mu} = \frac{1}{HW} \left(\mathbf{X} - \boldsymbol{\mu} \cdot \mathbf{1}^{\top} \right) \left(\mathbf{X} - \boldsymbol{\mu} \cdot \mathbf{1}^{\top} \right)^{\top} \in \mathbb{R}^{C \times C}$$

- Compute $\boldsymbol{\Sigma}_{\mu}^{-\frac{1}{2}}$? → using eigen decomposition
  - The covariance matrix is real and symmetric → can be diagonalized using eigen decomposition

$$\boldsymbol{\Sigma}_{\mu} = \mathbf{Q} \boldsymbol{\Lambda} \mathbf{Q}^{\top}$$

$\mathbf{Q} \in \mathbb{R}^{C \times C}$: Orthogonal matrix of eigenvectors

$\boldsymbol{\Lambda} \in \mathbb{R}^{C \times C}$: Diagonal matrix each eigenvalue of the corresponding eigenvector from Q

$$\Rightarrow \quad \boldsymbol{\Sigma}_{\mu}^{-\frac{1}{2}} = \mathbf{Q} \boldsymbol{\Lambda}^{-\frac{1}{2}} \mathbf{Q}^{\top}$$

https://www.youtube.com/watch?v=DgZo7WkJECo

Preliminaries

- Limitations of WT
  - Eigenvalue decomposition : computationally expensive, prevents the gradient back-propagation

Universal Style Transfer via Feature Transforms

NeurIPS, 2017

Yijun Li
UC Merced
yli62@ucmerced.edu

Chen Fang
Adobe Research
cfang@adobe.com

Jimei Yang
Adobe Research
jimyang@adobe.com

Zhaowen Wang
Adobe Research
zhawang@adobe.com

Xin Lu
Adobe Research
xinl@adobe.com

Ming-Hsuan Yang
UC Merced, NVIDIA Research
mhyang@ucmerced.edu

Whitening transform. Before whitening, we first center f_c by subtracting its mean vector m_c. Then we transform f_c linearly as in (2) so that we obtain $\hat{f}_c$ such that the feature maps are uncorrelated ($\hat{f}_c \hat{f}_c^{\top} = I$),

$$\hat{f}_c = E_c \, D_c^{-\frac{1}{2}} \, E_c^{\top} \, f_c \,, \tag{2}$$

where D_c is a diagonal matrix with the eigenvalues of the covariance matrix $f_c \, f_c^{\top} \in \Re^{C \times C}$, and E_c is the corresponding orthogonal matrix of eigenvectors, satisfying $f_c \, f_c^{\top} = E_c D_c E_c^{\top}$.

Efficiency. In Table 2 (3rd row), we also compare our approach with other methods in terms of efficiency. The method by Gatys et al. [9] is slow due to loops of optimization and usually requires at least 500 iterations to generate good results. The methods [27] and [15] are efficient as the scheme is based on one feed-forward pass with a trained network. The approach [3] is feed-forward based but relatively slower as the feature swapping operation needs to be carried out for thousands of patches. Our approach is also efficient but a little bit slower than [27, 15] because we have a eigenvalue decomposition step in WCT. But note that the computational cost on this step will not increase along with the image size because the the dimension of covariance matrix only depends on filter numbers (or channels), which is at most 512 (Relu_5_1). Currently the decomposition step is implemented based on CPU. Our future work includes more efficient GPU implementations of the proposed algorithm.

Preliminaries

- **Without** the eigen-decomposition?
  1. Approximating the whitening transformation matrix using Newton's iteration (ex, IterNorm)
  2. Deep Whitening Transformation (DWT) in GDWCT : **the covariance matrix close to the identity matrix**
     - Domain-specific style and domain-invariant content are **simultaneously encoded in the covariance** of the feature map
     - Whitening all covariance elements → diminish feature discrimination and distort the boundary of an object

$$\mathcal{L}_{\mathrm{DWT}} = \mathbb{E}[\|\boldsymbol{\Sigma}_{\mu} - \mathbf{I}\|_1],$$

Proposed Method – (1)

1. Instance Whitening Loss (IW)

- DWT loss decompose $\boldsymbol{\Sigma}_\mu \swarrow \begin{matrix} \boldsymbol{\Sigma}_{\mu\,(i,i)} \\ \boldsymbol{\Sigma}_{\mu\,(i,j)} \end{matrix}$

$$\mathcal{L}_{\text{DWT}} = \mathbb{E}[\|\boldsymbol{\Sigma}_\mu - \mathbf{I}\|_1], \swarrow \begin{matrix} \|\boldsymbol{\Sigma}_{\mu\,(i,i)} - 1\|_1 = \|\frac{\mathbf{x}_i^\top \cdot \mathbf{x}_i}{HW} - 1\|_1 = \|\frac{|\mathbf{x}_i||\mathbf{x}_i|\cos 0^\circ}{HW} - 1\|_1 \\ \|\boldsymbol{\Sigma}_{\mu\,(i,j)}\|_1 = \|\frac{\mathbf{x}_i^\top \cdot \mathbf{x}_j}{HW}\|_1 = \|\frac{|\mathbf{x}_i||\mathbf{x}_j|\cos\theta}{HW}\|_1 \end{matrix}$$

  - **Diagonal** covariance matrix → **1**
  - **Off-diagonal** of the covariance matrix → **0**
  - It is **difficult to optimize** both at the same time

Proposed Method – (1)

1. Instance Whitening Loss (IW)

- To address: feature map X can first be standardized using **instance normalization**

$$\mathbf{X_s} = (\mathrm{diag}(\boldsymbol{\Sigma}_\mu))^{-\frac{1}{2}} \odot (\mathbf{X} - \boldsymbol{\mu} \cdot \mathbf{1}^\top)$$

$$\boldsymbol{\Sigma_s} = \frac{1}{HW}(\mathbf{X_s})(\mathbf{X_s})^\top \in \mathbb{R}^{C \times C}$$

- WT is a linear transformation that makes
  - Variance term of each channel equal to 1
  - Covariances between each pair of channels equal to 0

$$\tilde{\mathbf{X}} \cdot \tilde{\mathbf{X}}^\top = (HW) \cdot \mathbf{I} \in \mathbb{R}^{C \times C}$$

Covariance matrix

$$\mathrm{Var}[X] = \mathrm{E}[(X - \mathrm{E}[X])(X - \mathrm{E}[X])^\top]$$

$$\begin{bmatrix} \mathrm{Var}[X_1] & \cdots & \mathrm{Cov}[X_1, X_K] \\ \vdots & \ddots & \vdots \\ \mathrm{Cov}[X_K, X_1] & \cdots & \mathrm{Var}[X_K] \end{bmatrix}$$

- **Instance Whitening Loss**
  - Covariance matrix is symmetric → **loss** can be applied only to the **strict upper triangular part**

$$\mathcal{L}_{\mathrm{IW}} = \mathbb{E}[\|\boldsymbol{\Sigma_s} \odot \mathbf{M}\|_1]$$

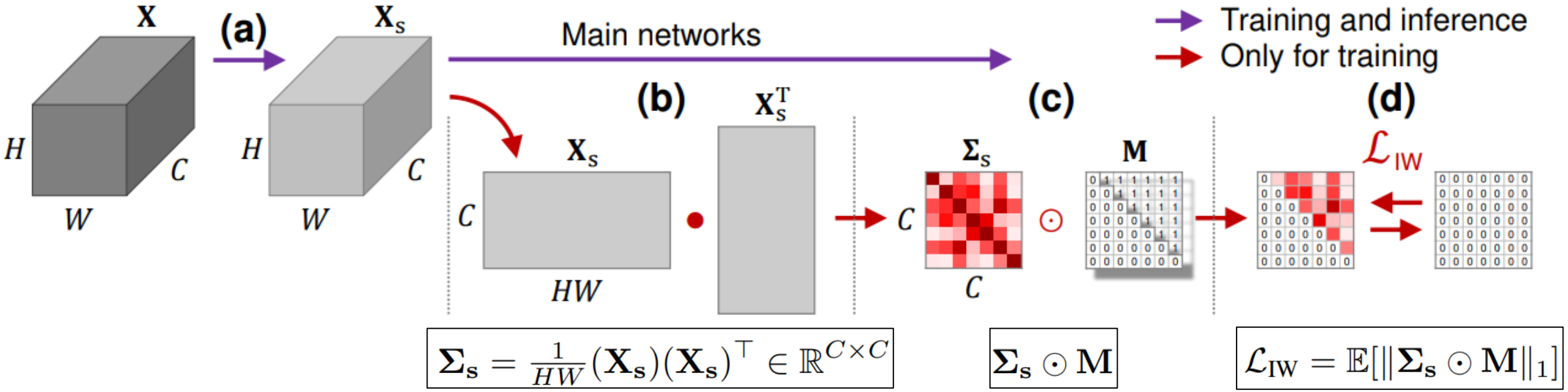
$$\mathbf{M}_{i,j} = \begin{cases} 0, & \text{if } i \geq j \\ 1, & \text{otherwise} \end{cases} \qquad 0 \leq i, j < C$$

M

Proposed Method – (1)

1. Instance Whitening Loss (IW)

$$\mathbf{X_s} = (\mathrm{diag}(\boldsymbol{\Sigma}_\mu))^{-\frac{1}{2}} \odot (\mathbf{X} - \boldsymbol{\mu} \cdot \mathbf{1}^\top)$$



$$\boldsymbol{\Sigma_s} = \frac{1}{HW}(\mathbf{X_s})(\mathbf{X_s})^\top \in \mathbb{R}^{C \times C}$$

$$\boldsymbol{\Sigma_s} \odot \mathbf{M}$$

$$\mathcal{L}_{\mathrm{IW}} = \mathbb{E}[\|\boldsymbol{\Sigma_s} \odot \mathbf{M}\|_1]$$

Proposed Method – (2)

2. Margin-based relaxation of whitening loss (IRW)

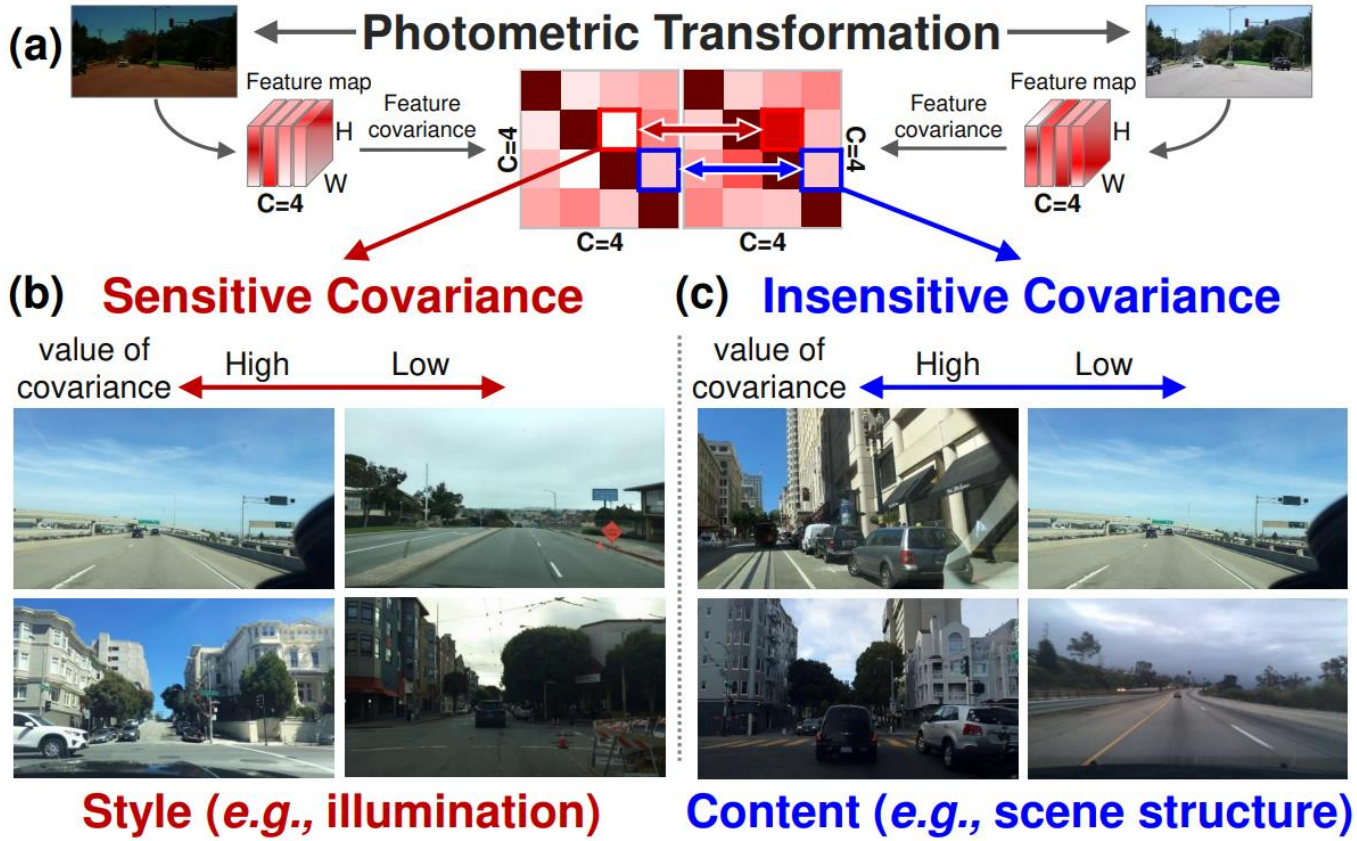
$$\mathcal{L}_{\text{IRW}} = \max(\mathbb{E}[\|\boldsymbol{\Sigma}_{\mathbf{s}} \odot \mathbf{M}\|_1] - \delta, 0)$$

Models (GTAV)	C	B	M	S	G
Baseline	28.95	25.14	28.18	26.23	**73.45**
Ours (IRW), δ=1/16	32.49	32.53	37.51	**27.77**	72.18
Ours (IRW), δ=1/32	33.30	33.17	38.03	27.43	71.96
Ours (IRW), δ=1/64	**33.57**	**33.18**	**38.42**	27.29	71.96
Ours (IRW), δ=1/128	32.85	32.40	37.36	27.43	72.21
Ours (IRW), δ=1/256	32.45	32.32	37.93	27.48	72.12
Ours (IW)	33.21	32.67	37.35	27.57	72.06

Proposed Method – (3)

3. Separation Covariance Elements (ISW)

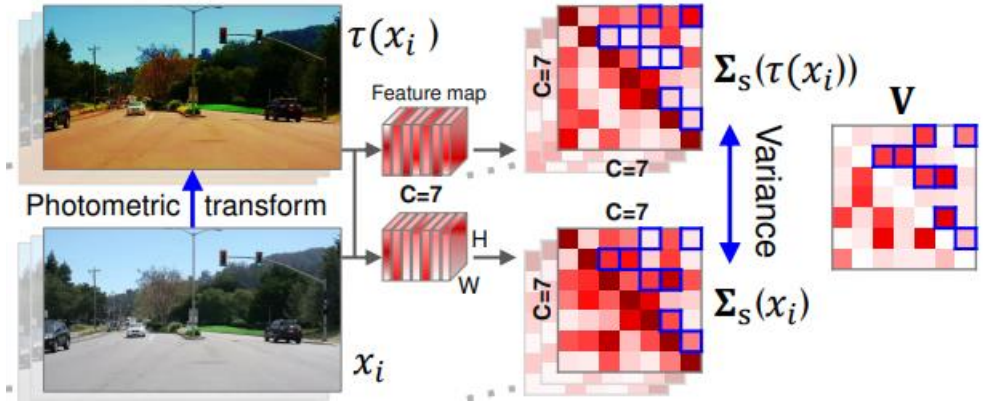
- Separate the covariance terms into two groups: domain-specific style and domain-invariant content
- Selectively **remove only the style-encoded covariances** that cause the domain shift.



Proposed Method – (3)

3. Separation Covariance Elements (ISW)

- Original / photometric transformed image → each covariance matrices → differences → variance matrix (V)



$$\mathbf{V} = \frac{1}{N}\sum_{i=1}^{N}\boldsymbol{\sigma}_i^2$$

$$\boldsymbol{\mu}_{\boldsymbol{\Sigma}_i} = \tfrac{1}{2}\left(\boldsymbol{\Sigma}_{\mathbf{s}}(x_i) + \boldsymbol{\Sigma}_{\mathbf{s}}(\tau(x_i))\right)$$

$$\boldsymbol{\sigma}_i^2 = \tfrac{1}{2}\left(\left(\boldsymbol{\Sigma}_{\mathbf{s}}\left(x_i\right) - \boldsymbol{\mu}_{\boldsymbol{\Sigma}_i}\right)^2 + \left(\boldsymbol{\Sigma}_{\mathbf{s}}\left(\tau\left(x_i\right)\right) - \boldsymbol{\mu}_{\boldsymbol{\Sigma}_i}\right)^2\right)$$

- Apply **k-means clustering** (k =3, m = 1)

$$C = \{c_1, c_2, \ldots, c_k\} \begin{cases} G_{low} = \{c_1, \ldots, c_m\} \\ G_{high} = \{c_{m+1}, \ldots, c_k\} \end{cases}$$

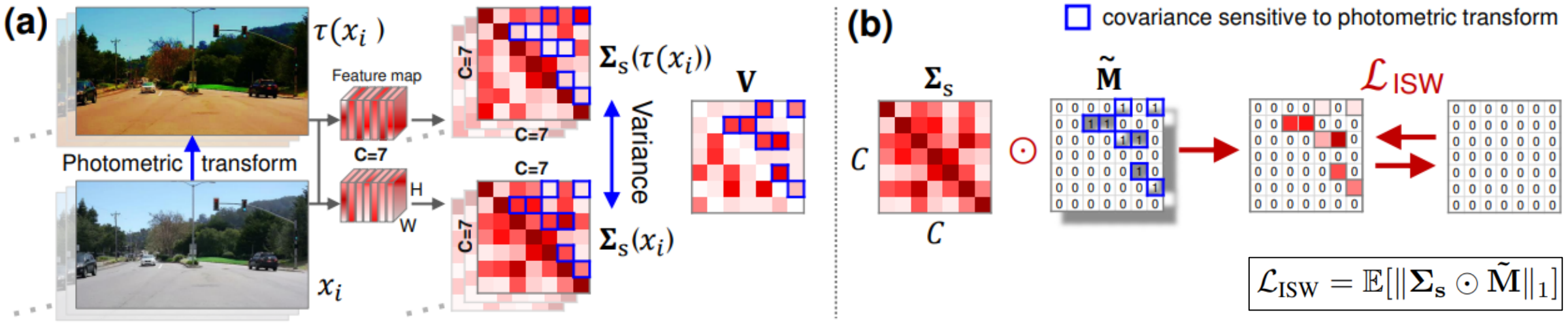
$$\tilde{\mathbf{M}}_{i,j} = \begin{cases} 1, & \text{if } \mathbf{V}_{i,j} \in G_{high} \\ 0, & \text{otherwise} \end{cases}$$

$$\mathcal{L}_{\text{ISW}} = \mathbb{E}\left[\left\|\boldsymbol{\Sigma}_{\mathbf{s}} \odot \tilde{\mathbf{M}}\right\|_1\right]$$

Models (GTAV)	C	B	M	S	G
Baseline	28.95	25.14	28.18	26.23	**73.45**
Ours (ISW), k=2	35.46	35.00	39.38	27.70	72.08
Ours (ISW), k=3	**36.58**	**35.20**	**40.33**	**28.30**	72.10
Ours (ISW), k=5	34.84	33.58	39.25	27.52	72.31
Ours (ISW), k=10	33.58	33.76	38.96	27.68	72.24
Ours (ISW), k=20	33.66	33.29	38.70	27.47	72.10
Ours (IW)	33.21	32.67	37.35	27.57	72.06

Proposed Method – (3)

3. Separation Covariance Elements (ISW)



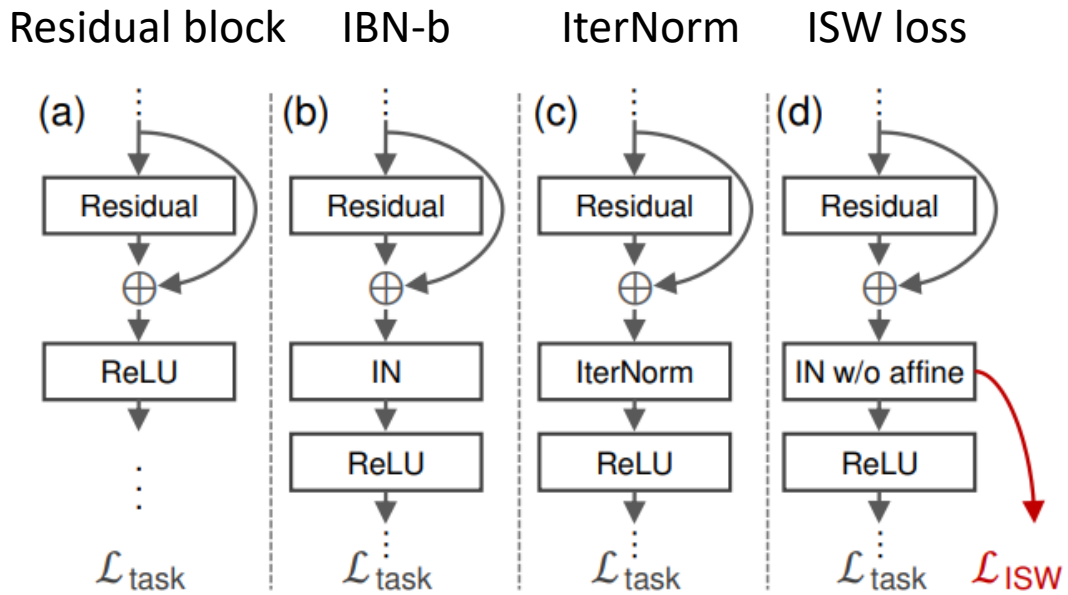
Proposed Method

Models (GTAV)	C	B	M	S	G
Ours (ISW), γ=0.4	35.60	34.07	38.98	28.10	71.96
Ours (ISW), γ=0.6	**36.58**	**35.20**	**40.33**	**28.30**	**72.10**
Ours (ISW), γ=0.8	35.73	34.01	39.69	27.44	71.96

- Network architecture with proposed ISW loss
- Simply add our proposed ISW loss to the instance normalization layer
  - γ = 0.6, L (layer) = 3 (IBN-Net)

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{task}} + \lambda\left(\frac{1}{L}\sum_{i}^{L}\mathcal{L}_{\text{ISW}}^{i}\right)$$

- Architecture comparison with other methods



Experiments

- Implementation
  - Baseline architecture: **DeepLabV3+**
  - Photometric transformation in ISW: **color jittering, gaussian blur**
- Datasets
  - **Real-world** datasets: Cityscapes, BDD-100K, Mapillary
  - **Synthetic** datasets: GTAV, SYNTHIA

Experiments

- Effectiveness of ISW (1)
  - Metric: mIoU (%)
  - Cityscapes (C), BDD-100K (B), Mapillary (M), SYNTHIA (S), GTAV (G)

Models (GTAV)	C	B	M	S	G
Baseline	28.95	25.14	28.18	26.23	73.45
†SW [45]	29.91	27.48	29.71	27.61	**73.50**
†IBN-Net [44]	33.85	32.30	37.75	27.90	72.90
†IterNorm [22]	31.81	32.70	33.88	27.07	73.19
Ours (IW)	33.21	32.67	37.35	27.57	72.06
Ours (IRW)	33.57	33.18	38.42	27.29	71.96
Ours (ISW)	**36.58**	**35.20**	**40.33**	**28.30**	72.10

Models (Cityscapes)	B	M	G	S	C
Baseline	44.96	51.68	42.55	23.29	**77.51**
†SW [45]	48.49	55.82	44.87	26.10	77.30
†IBN-Net [44]	48.56	57.04	45.06	26.14	76.55
†IterNorm [22]	49.23	56.26	**45.73**	25.98	76.02
Ours (IW)	48.19	58.90	45.21	25.81	76.06
Ours (IRW)	48.67	**59.20**	45.64	26.05	76.13
Ours (ISW)	**50.73**	58.64	45.00	**26.20**	76.41

Baseline: DeeplabV3+

Backbone: ResNet-50 with an output stride of 16

† : own re-implemented models

Experiments

- Effectiveness of ISW (2)
  - Metric: mIoU (%)
  - Cityscapes (C), BDD-100K (B), Mapillary (M), SYNTHIA (S), GTAV (G)

Models (GTAV)	C	B	M	S	G
Baseline	25.56	22.17	28.60	23.33	**66.47**
†IBN-Net [44]	27.10	31.82	34.89	**25.56**	65.44
Ours (ISW)	**30.98**	**32.06**	**35.31**	24.31	64.99
Baseline	25.92	25.73	26.45	24.03	**68.12**
†IBN-Net [44]	30.14	27.66	27.07	**24.98**	67.66
Ours (ISW)	**30.86**	**30.05**	**30.67**	24.43	67.48

Baseline: DeeplabV3+

Backbones: ShuffleNetV2 (up), MobileNetV2 (down)

† : own re-implemented models

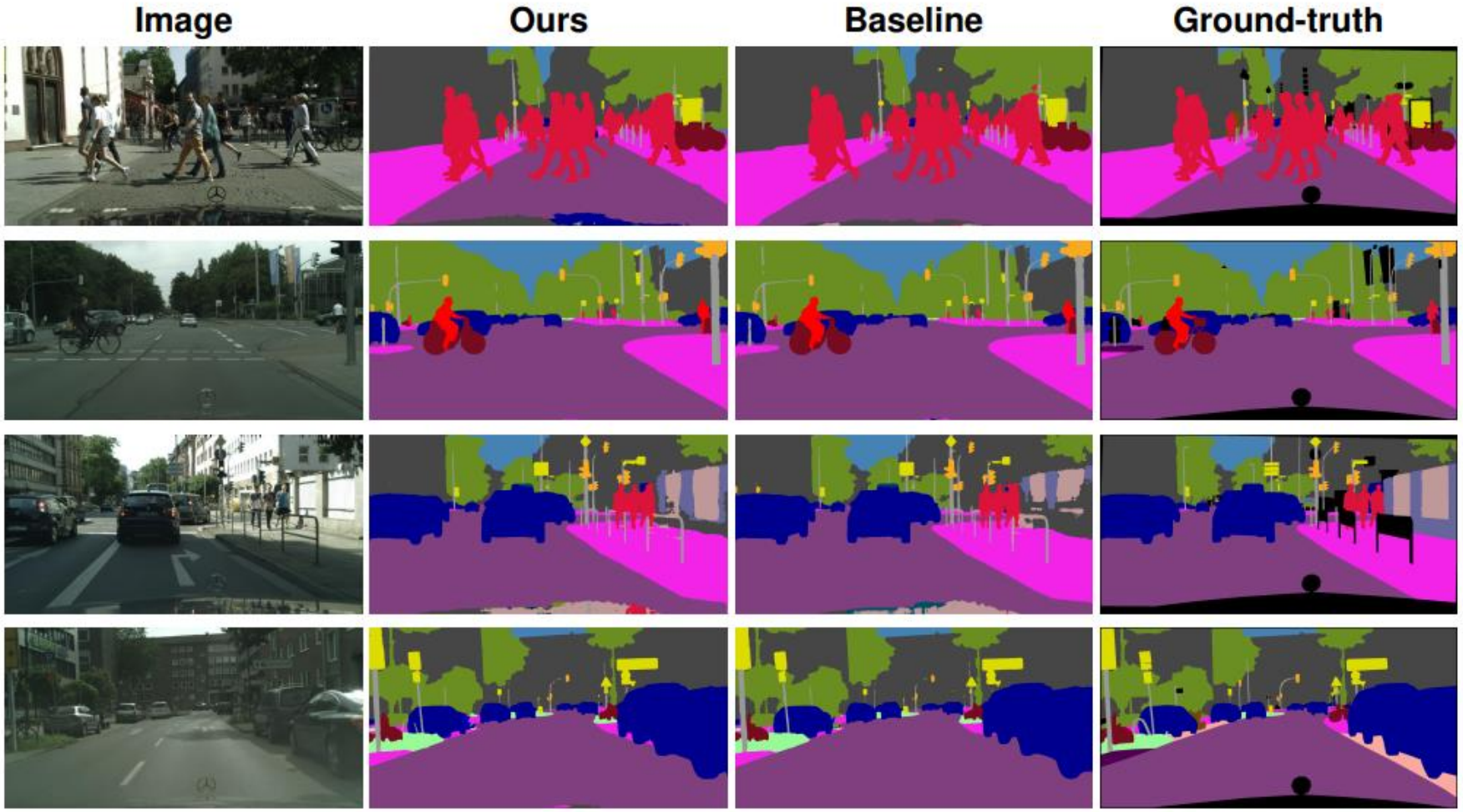
Models (G + S)	C	B	M	G	S
Baseline	35.46	25.09	31.94	68.48	67.99
IBN-Net	35.55	32.18	38.09	**69.72**	66.90
Ours	**37.69**	**34.09**	**38.49**	68.26	**68.77**

Backbone: ResNet-50 with an output stride of 16

Experiments

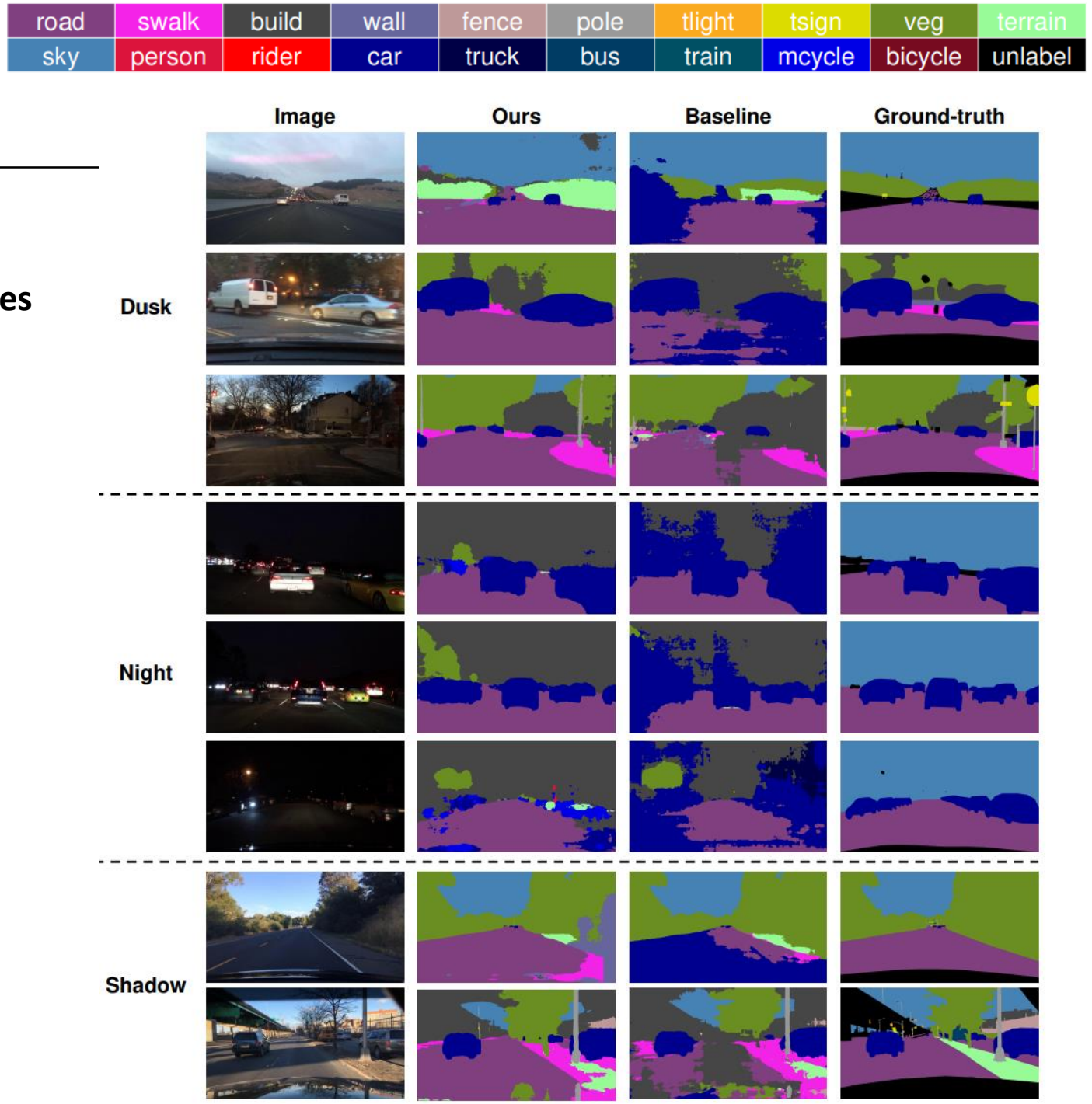
road	swalk	build	wall	fence	pole	tlight	tsign	veg	terrain
sky	person	rider	car	truck	bus	train	mcycle	bicycle	unlabel

- Effectiveness of ISW (2)
  - Segmentation results on **seen domain** images (Cityscape)



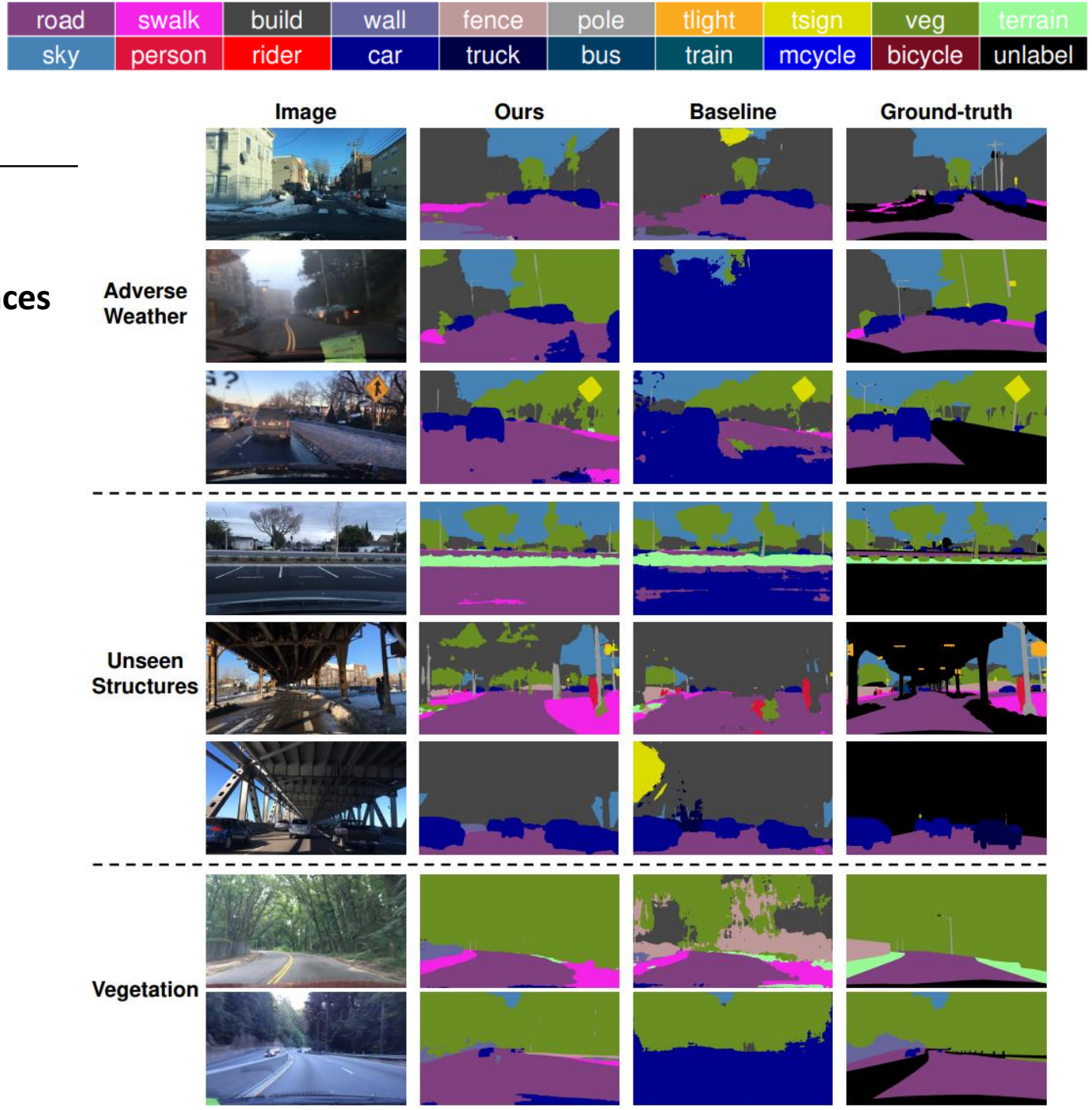
Experiments

- Effectiveness of ISW (2)
  - Segmentation results under **illumination changes**
  - Train C, inference B



Experiments

- Effectiveness of ISW (2)
  - Segmentation results under **various circumstances**
  - Train C, inference B

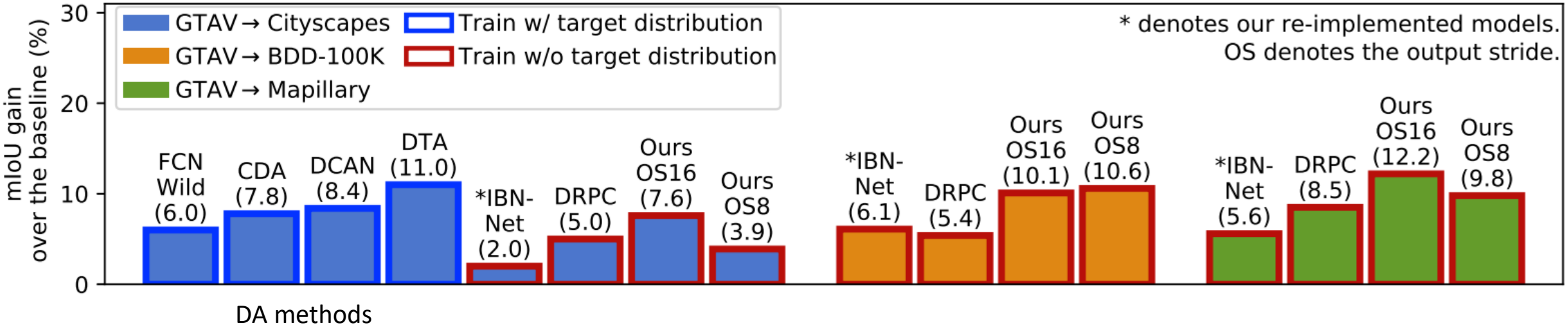


Experiments

- Comparison with other DG and DA methods

DG methods

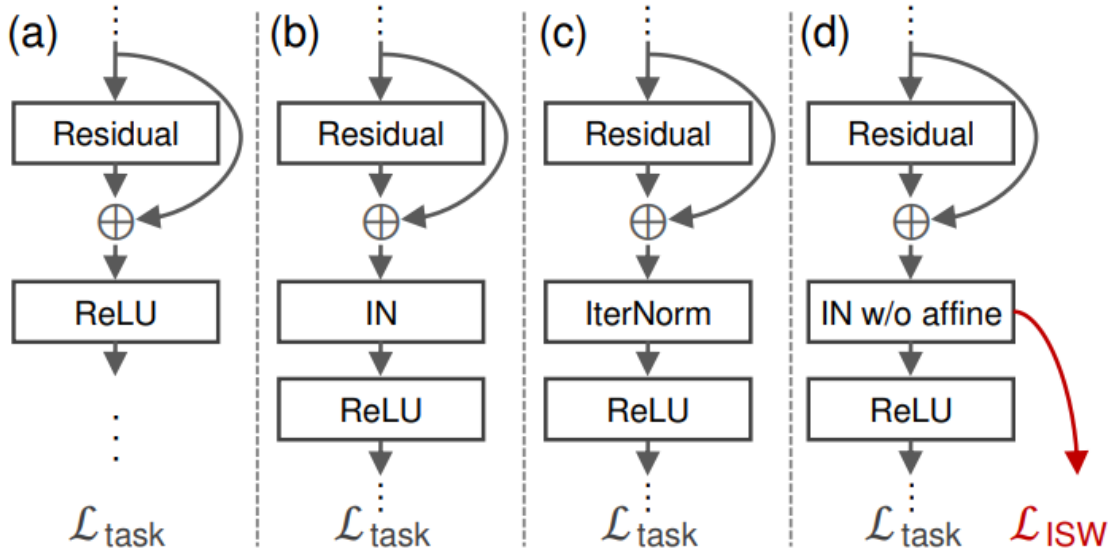
Models (GTAV)	C		B		M	
Baseline	22.20	7.40 ↑	N/A		N/A	
IBN-Net [44]	29.60					
Baseline	32.45	4.97↑	26.73	5.41↑	25.66	8.46↑
DRPC [64]	**37.42**		32.14		34.12	
Baseline	28.95	**7.63**↑	25.14	**10.06**↑	28.18	**12.15**↑
Ours (ISW)	36.58		**35.20**		**40.33**	



Experiments

- Computational cost analysis

Residual block IBN-b IterNorm ISW loss



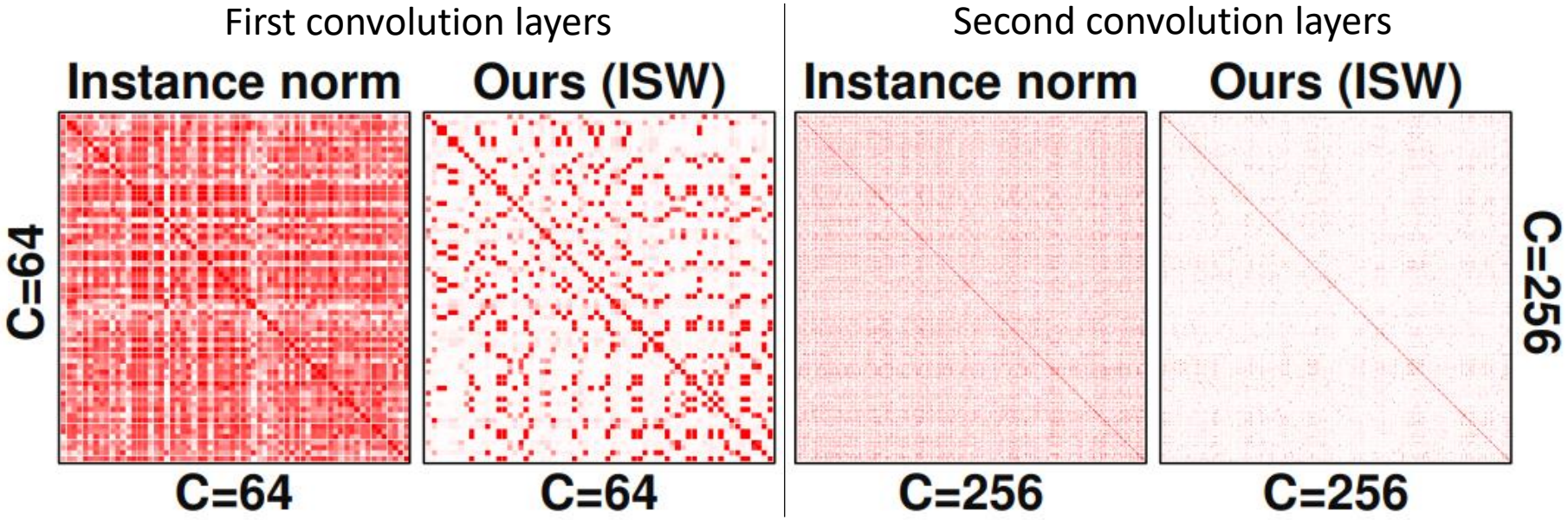
Share the same network architecture
Different **normalization methods**

Models	# of Params	GFLOPS	Inference Time (ms)
Baseline	45.082M	554.31	10.48
†IBN-Net [44]	45.083M	554.31	10.51
†IterNorm [22]	45.081M	554.31	40.31
Ours	45.081M	554.31	10.43

→ whitening transformation
without additional computational cost

Experiments

- Comparison of **covariance matrices**



ISW selectively eliminates the covariance

Experiments

- **Reconstructing** images with whitened features
  - Using U-Net

Image contents are properly maintained
Style (such as illumination and colors) vanish

Discussions

- Affine parameters
  - Adding affine parameter or 1x1 convolution layer after the normalization layer: not improve
  - Conjecture: above approach do not have sufficient complexity in recovering the original distribution
- Photometric transformation
  - We found that applying color transform and Gaussian blur does not harm the content information
  - Expect various photometric augmentation techniques

Conclusions

- Focused on solving the **domain generalization** problem in **urban-scene segmentation**

- A novel **instance selective whitening (ISW) loss**
  - Disentangling the covariances (of the intermediate features): the style- and content-related ones
  - **Suppressing only the style-related covariances** → learn the domain-invariant feature representation

END