# Similarity Min-Max: Zero-Shot Day-Night Domain Adaptation (Supplementary Material)

## Contents



## 1. Supplementary Material Overview

In the following supplementary material, we first provide the implementation details of our proposed approach (Sec. 2). Then we describe the details of comparison methods and ablation studies (Sec. 3). Finally, we provide additional experiments and empirical analysis (Sec. 4).

## 2. Implementation Details

### 2.1. Architecture: Darkening Module

The architecture of our learnable darkening module is a U-Net [23] with three downsampling and three upsampling layers, as shown in Figure 1.

For all vision tasks, we train the darkening module on the corresponding daytime dataset. The input exposure map has identical entries uniformly sampled from $[0, 0.5]$. Loss weights are set to $\lambda_D^{sim} = 0.1, \lambda_{c-exp} = 10, \lambda_{col} = 25, \lambda_{ltv} = 1600$, $\lambda_{flex} = 5$, and $\alpha$ in $\mathcal{L}_{ltv}$ is set to 0.02. We use a single RTX 2080 Ti GPU for training.

When synthesizing nighttime images, the exposure map $E'$ is first initialized with identical entries uniformly sampled between $[0, 0.2]$ to simulate nighttime illumination. Then, we inject both pixel-wise and patch-wise to $E'$, *i.e.*,

$$E' = \mathcal{U}(0, 0.2) + z_1 + z_2, \tag{1}$$

where

$$z_1 \in \mathbb{R}^{h \cdot w} \sim \mathcal{N}(0, \alpha_1), \tag{2}$$

is the pixel-wise Gaussian noise, and

$$\begin{aligned} \bar{z}_2 &\in \mathbb{R}^{\frac{h}{d} \cdot \frac{w}{d}} \sim \mathcal{N}(0, \alpha_2), \\ z_2 &= \text{interpolate}(\bar{z}_2, h, w), \end{aligned} \tag{3}$$

is the patch-wise Gaussian noise. $h, w$ is the input's height and width, $d$ the downsampling scale, and $\alpha_1, \alpha_2$ the noise intensity. Both $\alpha_1$ and $\alpha_2$ are set to $0.025$.

Note that the above settings are task-agnostic, *i.e.*, shared across all tasks.

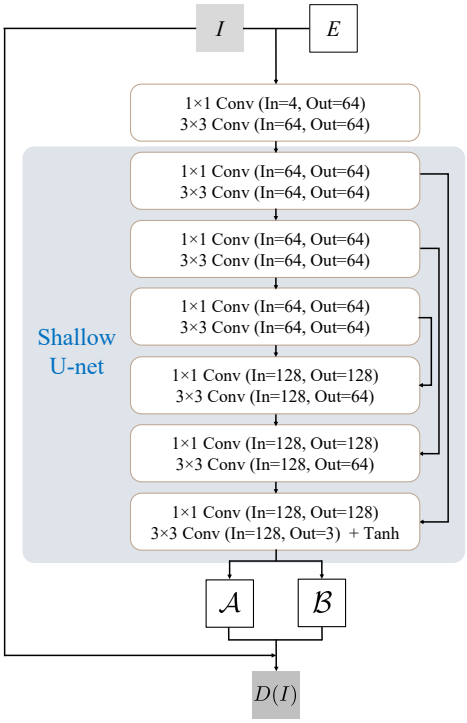



Figure 1. Network architecture of our darkening module.

## 2.2. Architecture: BYOL Heads

The BYOL [10] head consists of a projection head $q$, $q$'s duplicate $q'$, and a prediction head $z$. Both $q$ and $z$ are MLPs with a single hidden layer, except that we prepend a convolutional block before $q$ in the segmentation task. This extended version is denoted by "BYOL Conv head". The architecture of these modules is shown in Figure 2. Note that BYOL heads are only used in the adaptation stage.

## 2.3. Implementation: Nighttime Image Classification

For image classification, we adopt the ResNet-18 [11] backbone (w/o ImageNet [7] pre-train). Features for similarity losses are extracted after the global average pooling layer, as shown in Figure 3a. The downsampling scale $d$ in Eq. (3) is set to 16. We train the darkening module for 15 epochs by SGD optimizer with an initial learning rate of $0.0001$ and decays at the $5^{\text{th}}, 10^{\text{th}}$ epoch by $0.1$. The batch size is set to 16.

In the adaptation stage, the BYOL head is appended after the global average pooling layer. Loss weights are set to $\lambda_F^{sim} = 0.1$ and $\lambda_{task} = 1$. We train the model for 90 epochs with SGD optimizer, batch size 32, and initial learning rate $0.001$. Learning rate decays at the $30^{\text{th}}, 60^{\text{th}}$ epoch by $0.1$. Data augmentations include resize-crop, horizontal flip, and color jitter. We use a single RTX 2080Ti GPU for training.

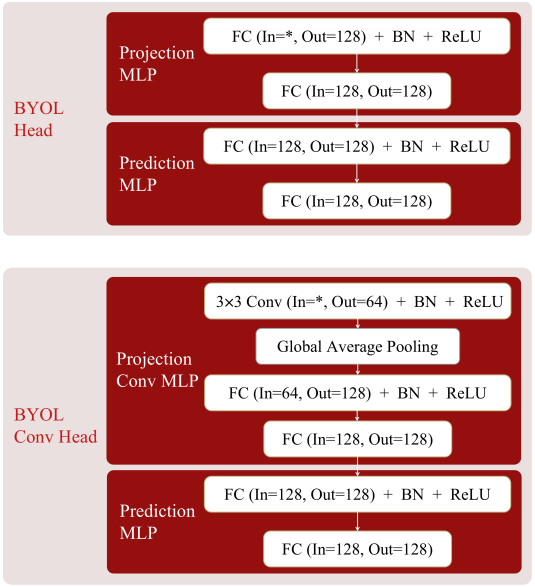



Figure 2. Network architecture of our BYOL heads.

### 2.4. Implementation: Nighttime Semantic Segmentation

We adopt the RefineNet [18] architecture with ResNet-101 backbone for semantic segmentation. Features are extracted after each of the four ResNet blocks, as shown in Figure 3b. The downsampling scale $d$ in Eq. (3) is set to 64. We train the darkening module for 15 epochs by SGD optimizer with an initial learning rate of $0.0001$, which decays at the $5^{\text{th}}$, $10^{\text{th}}$ epoch by $0.1$. Input images are cropped to $256 \times 256$. The batch size is set to 4.

In the adaptation stage, BYOL Conv heads are appended after all ResNet blocks. Loss weights are set to $\lambda_F^{sim} = 0.1$ and $\lambda_{task} = 1$. We train the model for 100 epochs with SGD optimizer, batch size 4, and initial learning rate $0.001$. Learning rate decays at the $30^{\text{th}}$, $60^{\text{th}}$, and $90^{\text{th}}$ epoch by $0.1$. All input images are resized to $1024 \times 512$ and then cropped to $768 \times 384$. Other data augmentations include horizontal flip, color jitter, Gaussian blur, and cutout [8]. We use two RTX 2080Ti GPUs for training. Due to the limit of computational resources, the nighttime data are generated in an offline manner, *i.e.*, we generate the nighttime dataset in advance instead of generating them during the training process.

### 2.5. Implementation: Visual Place Recognition

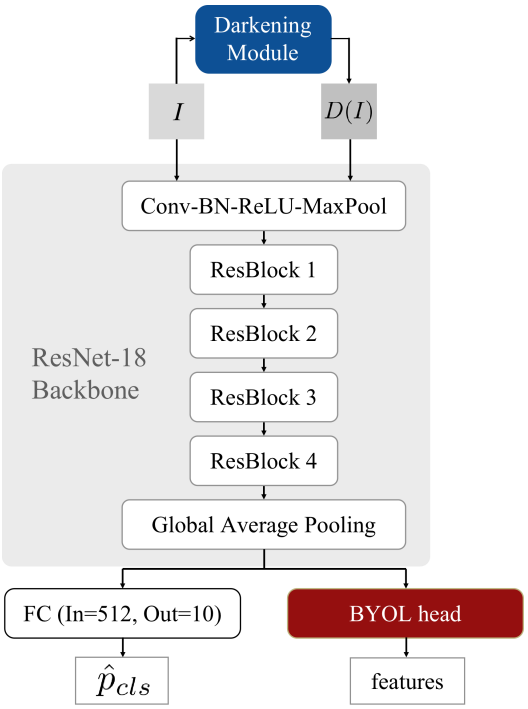
For visual place recognition, we adopt the GeM [22] framework with ResNet-101 backbone. Features are extracted after the average pooling layer, as shown in Figure 3c. The downsampling scale $d$ in Eq. (3) is set to 32. We train the darkening module for 5000 iterations by SGD optimizer with an initial learning rate of $0.0001$. Learning rate decays at the $1000^{\text{th}}$, $3000^{\text{th}}$ iteration by $0.1$. The batch size is set to 8.

In the adaptation stage, we extend our method as follows. The original GeM framework receives a tuple of images $\{p, q, n_1, \cdots, n_k\}$ as input, in which the query $q$ only matches $p$. In our implementation, we consider $D(p)$ as an additional matching for $p$, *i.e.*, an input tuple contains two positive samples (instead of one) and $k$ negative samples. We train the model for five epochs with Adam optimizer and learning rate $5 \cdot 10^{-7}$. Other hyperparameters are identical to that of GeM.

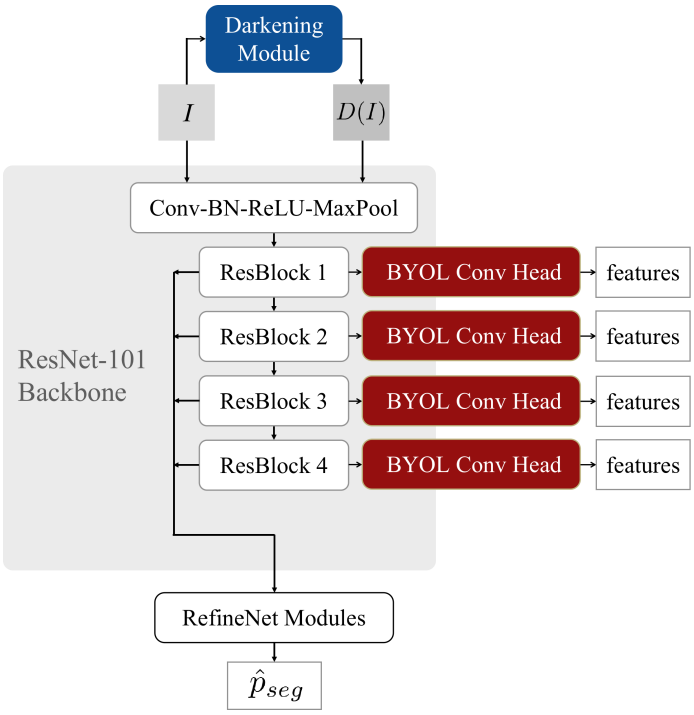
### 2.6. Implementation: Low-Light Video Action Recognition

For low-light action recognition, we adopt the I3D [2] action recognizer based on 3D-ResNet [9], as shown in Figure 3d. Features are extracted after the 3D average pooling layer. The downsampling scale $d$ in Eq. (3) is set to 16. We train the darkening module for 5000 iterations by SGD optimizer with an initial learning rate of $0.0001$. Learning rate decays at the $1000^{\text{th}}$, $3000^{\text{th}}$ iteration by $0.1$. The batch size is set to 2.

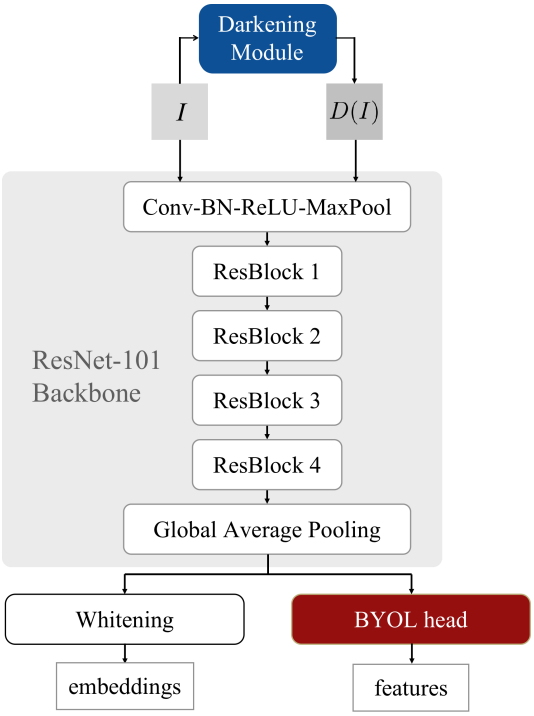
In the adaptation stage, projection and prediction heads are appended after the 3D average pooling layer. Loss weights are set to $\lambda_F^{sim} = 0.1$ and $\lambda_{task} = 1$. We train the whole model for 3- epochs with SGD optimizer, batch size 36, and initial learning rate $0.01$. Learning rate decays at the $20000^{\text{th}}$, $40000^{\text{th}}$ iteration by $0.1$. Since we cannot access low-light data, we use normal-light data's mean and standard deviation values for input normalization. We follow [29] for data augmentation



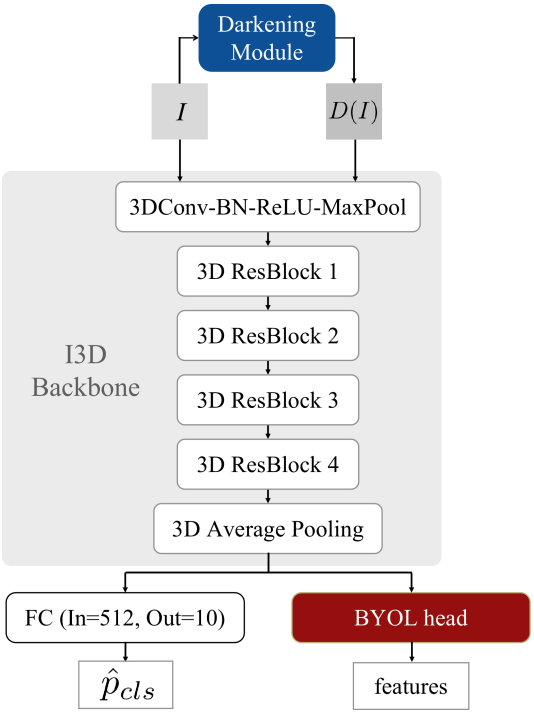

(a) Nighttime Image Classification




(b) Nighttime Semantic Segmentation




(c) Visual Place Recognition




(d) Low-Light Video Action Recognition

Figure 3. Our network architecture for various vision tasks.

and hyperparameter settings. Due to the huge computational cost of synthesizing videos, we transfer videos to low light in advance.

# 3. Benchmarking Details

## 3.1. Comparison Methods Settings

Table 1 shows the code sources of all comparison methods.

Table 1. The code sources of comparison methods.

| Method | Link |
|---|---|
| EnlightenGAN [13] | https://github.com/VITA-Group/EnlightenGAN |
| Zero-DCE++ [16] | https://github.com/Li-Chongyi/Zero-DCE_extension |
| RUAS [19] | https://github.com/KarelZhang/RUAS |
| SCI [20] | https://github.com/vis-opt-group/SCI |
| URetinexNet [28] | https://github.com/andersonyong/uretinex-net |
| LEDNet [33] | https://github.com/xiaoyufenfei/LEDNet |
| StableLLVE [30] | https://github.com/zkawfanx/StableLLVE |
| SMOID [12] | https://github.com/MichaelHYJiang/Learning-to-See-Moving-Objects-in-the-Dark |
| SGZ [31] | https://github.com/ShenZheng2000/Semantic-Guided-Low-Light-Image-Enhancement |
| IRM [1] | https://github.com/thuml/Transfer-Learning-Library |
| MixStyle [32] | https://github.com/thuml/Transfer-Learning-Library |
| RobustNet [3] | https://github.com/shachoi/RobustNet |
| SAN-SAW [21] | https://github.com/leolyj/SAN-SAW |
| MAET [5] | https://github.com/cuiziteng/ICCV_MAET |
| CIConv [15] | https://github.com/Attila94/CIConv |
| ARID [29] | https://github.com/xuyu0010/ARID_v1 |
| GeM [22] | https://github.com/Attila94/cnnimageretrieval-pytorch |

For enhancement methods, we directly use their released enhancement models as a pre-processor to enhance the testing images, then we apply the daytime baseline model. Same settings are adopted in [20, 27]. For other methods, we use their pre-trained models if available. Otherwise, we run their code for re-implementation. We follow CIConv [15] and ARID [29] for other benchmarking details.

## 3.2. Ablation Studies Settings

We compare our module $D$ with heuristic and learnable darkening methods in Sec. 4.2 of the main text on classification and Sec. 4.1 of the supplementary on segmentation. For heuristic methods, we implement them by randomly sampling a darkening parameter in a fixed pre-defined range. We test various ranges and report the best performance. Specifically, we use $\beta = 2$ for brightness adjustment ($f(x) = \beta \cdot x$) and $\gamma = 3$ for gamma correction ($f(x) = x^3$) in our implementation.

For the learnable methods, we test the gamma curve ($f(x, \alpha) = x^{\frac{1}{\alpha}}, \alpha \in (0, 1]$) and the reciprocal curve ($f(x, \alpha) = \frac{(1-\alpha)\cdot x}{1-\alpha \cdot x}, \alpha \in [0, 1)$). For these two methods, we replace $\mathcal{L}_{ltv}$ with the standard total variance loss (*i.e.*, $h(x) = x$ in $\mathcal{L}_{ltv}$) and use a lower learning rate due to their unstable training dynamics.

## 3.3. Implementation of MMD Statistics

Given a fixed feature extractor, we extract the features of all images from the "test-day" split and "test-night" split. Then we calculate the Maximum Mean Discrepancy (MMD) between these two sets of features using Radical Basis Function (RBF) kernel with bandwidth $10, 15, 20$, and $50$.

# 4. Additional Experiments

## 4.1. Ablation Studies

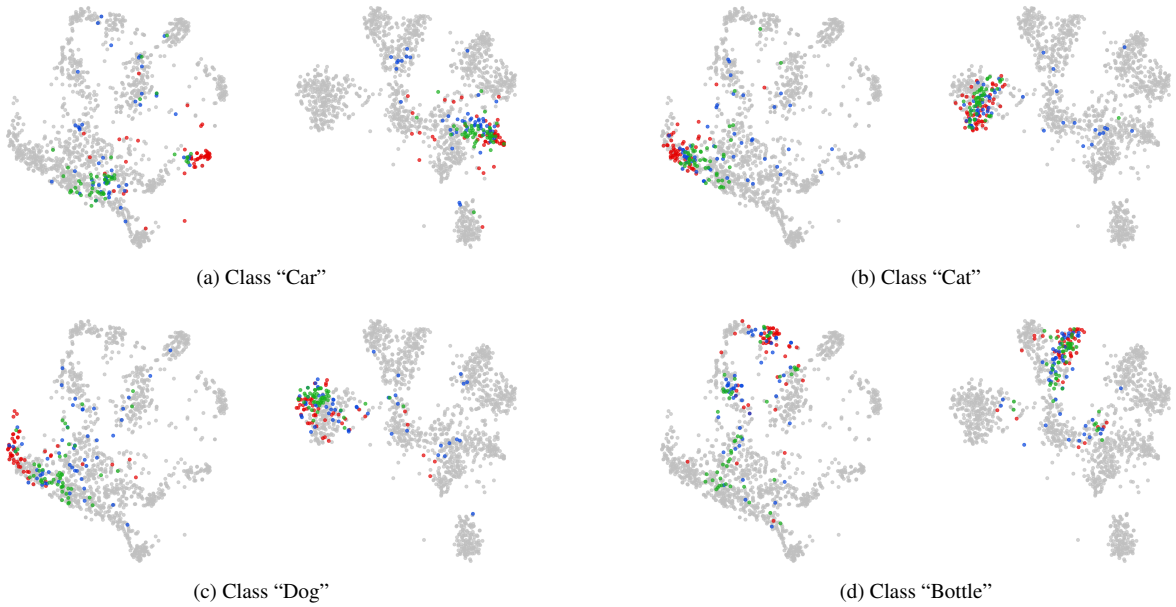
Besides classification, we also ablate our method on the segmentation task (Nighttime Driving [6] and Dark-Zurich [24]). The full results are shown in Table 2. Our proposed approach achieve the best results on all nighttime unseen target domains, demonstrating the superiority of our framework.

Table 2. Ablation studies for module $D$ and similarity losses on classification (CODaN [15]) and segmentation (Nighttime Driving [6] and Dark-Zurich [24]). We report Top-1 accuracy for the former and mIoU for the latter.

| Category | Method | CODaN | Nighttime Driving | Dark-Zurich |
|---|---|---|---|---|
| Baseline | - | 53.32 | 34.3 | 30.6 |
| Module $D$ Heuristic | Brightness adjustment | 57.96 | 42.8 | 37.0 |
| | Gamma correction | 63.96 | 40.4 | 35.1 |
| Module $D$ Learnable | Reciprocal curve | 62.60 | 43.3 | 39.5 |
| | Gamma curve | 64.16 | 38.7 | 33.8 |
| Similarity Loss | w/o $\mathcal{L}_D^{sim}$ and $\mathcal{L}_F^{sim}$ | 64.08 | 40.3 | 37.4 |
| | w/o $\mathcal{L}_D^{sim}$ | 64.56 | 42.4 | 39.5 |
| | w/o $\mathcal{L}_F^{sim}$ | 64.88 | 43.0 | 39.4 |
| Full version | - | **65.87** | **44.9** | **40.2** |

## 4.2. Empirical Analysis on Nighttime Image Classification

**T-SNE Clustering Visualization.** Firstly, we visualize images' features extracted by the original daytime model and our adapted model on CODaN [15]. Red, blue, and green dots stand for the feature of daytime, *synthesized* nighttime, and *real* nighttime images, respectively.



(a) Class "Car" (b) Class "Cat"

(c) Class "Dog" (d) Class "Bottle"

Figure 4. t-SNE [26] visualization of images' features extracted by the original daytime model (left) and our adapted model on CODaN [15] (right). Red, blue, and green dots stand for the feature of daytime, *synthesized* nighttime, and *real* nighttime images, respectively. Zoom in for better visual quality.

As shown in the left of each group in Figure 4, the synthetic nighttime images (green dots) are very similar to the real-world nighttime images (blue dots) on the feature-level. Besides, by comparing the red and blue dots, we find our method could significantly reduce the feature-level gap between daytime and nighttime images, indicating our model's effectiveness in the night time.

**Saliency Map Visualization.** We visualize the saliency map of images generated by our proposed darkening module $D$ trained with and without $\mathcal{L}_D^{sim}$ using XRAI [14] in Figure 5. The brighter the region's color, the greater importance that the model attaches. Compared with $D$ trained without $\mathcal{L}_D^{sim}$, we find introducing $\mathcal{L}_D^{sim}$ to the training process could enable $D$ to fool the daytime classifier, thus providing more valuable knowledge for the subsequent adaptation stage. For example, the image on the left illustrates a cup in front of a computer screen. In the original image, the classifier responds notably to the cup and outputs the correct category. Then we darken the image by module $D$ trained without $\mathcal{L}_D^{sim}$. Although the image's appearance is drastically changed, the classifier can still determine the cup's appearance and make the correct prediction. This phenomenon indicates that simply reducing the illumination without considering machine vision is inadequate in changing the image's feature response, thus providing limited benefit for day-night adaptation. On the other hand, adding $\mathcal{L}_D^{sim}$ to $D$ could induce semantic shifts despite the darkened results seeming identical, thus disabling the classifier to locate the correct region.

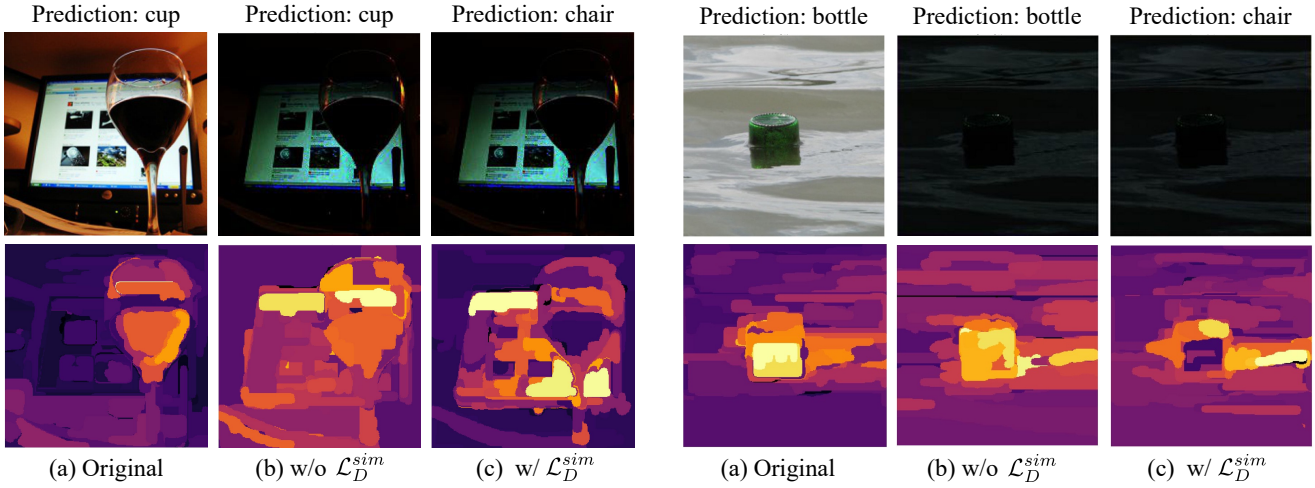



Figure 5. Saliency maps of a daytime classifier on different images. The brighter the region's color, the greater importance that the model attaches. In each group, three columns represent the original image, darkened images generated by $D$ trained w/o $\mathcal{L}_D^{sim}$, and w/ $\mathcal{L}_D^{sim}$, respectively.

**Cross-Architectual Analysis** We study the effect of using different model architectures for image-level darkening and model-level adaptation. Specifically, we try two alternate model architectures: VGG16 and AlexNet, for darkening and keeping the model-level backbone as ResNet-18. The accuracy is 63.52% and 62.28%, respectively, which are slightly worse than dropping the similarity loss $\mathcal{L}_D^{sim}$ (64.56%), but significantly outperforming the baseline (53.32%). These results show that our model-level stage remains effective under the cross-architecture scenario.

### 4.3. Results on Nighttime Segmentation

**Quantitative Segmentation Results.** Additional results on the source daytime domain are provided in Table 3. Our approach does not attenuate and even improve the model's performance in the daytime, justifying that contrastive learning in the model-level adaptation stage not only narrows the day-night domain gap and but also enhances the representation.

The per-class IoU (Intersection over Union) scores on Dark-Zurich [24] are shown in Table 4. Our method improves segmentation results across nearly all classes.

We also evaluate our method on the NightCity dataset [25]. The mIoU of the baseline (RefineNet [18]), previous SoTA (CIConv [15]), and ours are 23.0%, 25.1%, and 28.5%, respectively, which further justifies the superiority of our method.

**Qualitative Segmentation Results.** We provide additional qualitative segmentation results in Figures 6 and 7. We show that low-light enhancement methods perform poorly on nighttime street scenes, thus yielding unsatisfactory results. On the

other hand, our method can better extract information hidden by the darkness and generate more accurate semantic maps.

### 4.4. Results on Visual Place Recognition

We provide additional qualitative visual place recognition results in Figure 8. Compared with the baseline model [22] that often gets deceived by the nighttime appearance, our model can extract features more robust to illumination and thus retrieve the correct daytime image showing the same scene as the query image.

### 4.5. Results on Low-Light Action Recognition

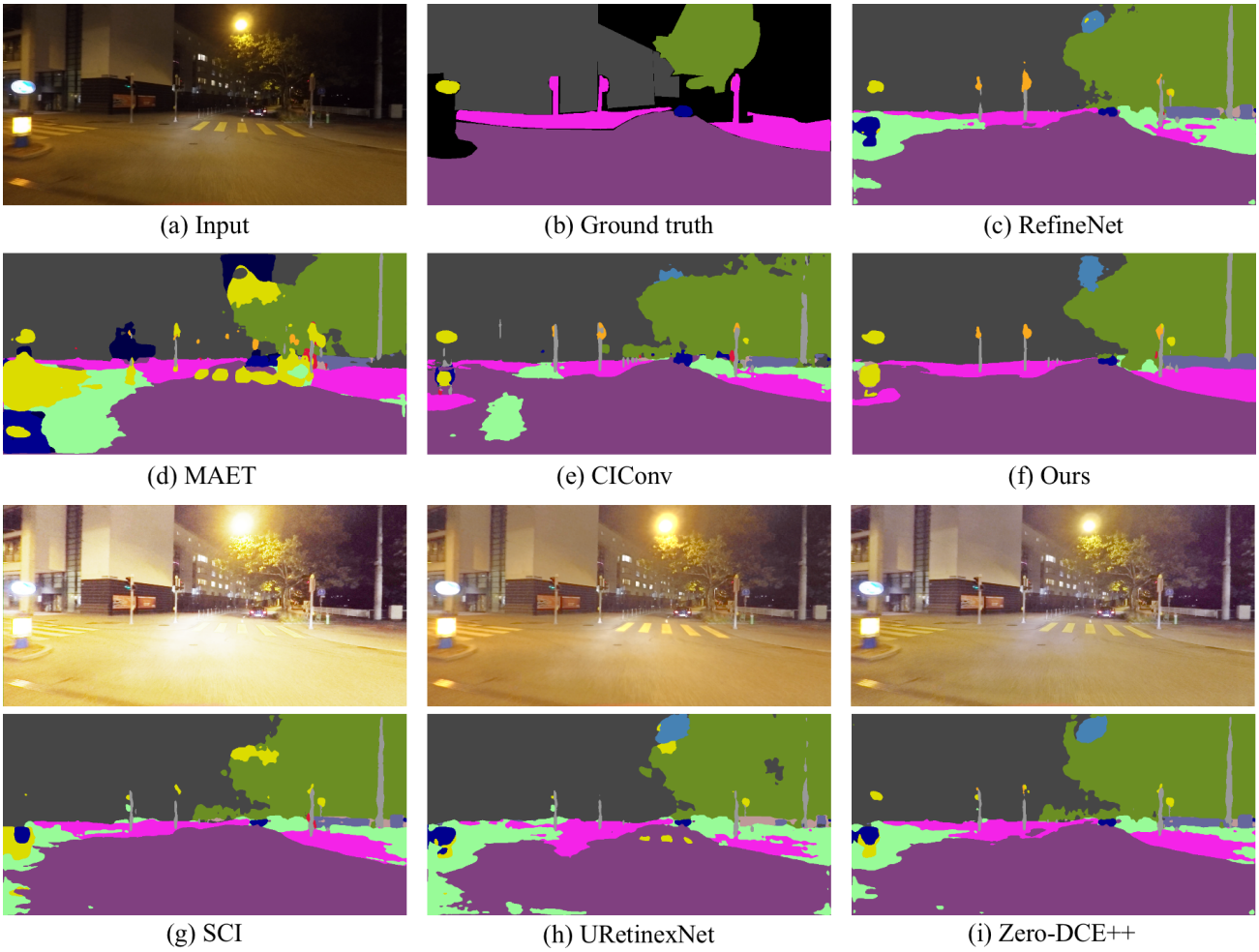
We provide qualitative video action recognition results in Figures 9 and 10. For instance, Figure 9 demonstrates a video about a person running. All video enhancement methods perform poorly and therefore mislead the classifier. Meanwhile, our adapted model correctly classifies the video with more than 99% confidence.

Table 3. Semantic segmentation results on Cityscapes [4] (daytime), Nighttime Driving [6] and Dark-Zurich [24], reported as mIoU scores.

| Category | Method | Cityscapes | Nighttime Driving | Dark-Zurich |
|---|---|---|---|---|
| Baseline | RefineNet [18] | 66.9 | 34.3 | 30.6 |
| Low-Light Enhancement | EnlightenGAN [13] | - | 25.2 | 24.9 |
| | Zero-DCE++ [16] | - | 32.7 | 28.3 |
| | RUAS [19] | - | 25.1 | 23.4 |
| | SCI [20] | - | 28.6 | 25.7 |
| | URetinexNet [28] | - | 28.1 | 24.0 |
| | LEDNet [33] | - | 27.6 | 26.6 |
| Domain Generalization | AdaBN [17] | 66.1 | 37.2 | 31.1 |
| | RobustNet [3] | 71.5 | 33.0 | 34.5 |
| | SAN-SAW [21] | 62.7 | 28.1 | 16.0 |
| Zero-Shot Day-Night Domain Adaptation | MAET [5] | 56.1 | 28.1 | 26.4 |
| | CIConv [15] | 64.4 | 41.2 | 34.5 |
| | **Ours** | 69.1 | 44.9 | 40.2 |

Table 4. Pre-class segmentation result of the best trail on Dark-Zurich [24], reported as IoU.

| Method | road | sidew. | build. | wall | fence | pole | light | sign | veget. | terrain | sky | person | rider | car | truck | bus | train | motroc. | bicycle | **mIoU** |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| RefineNet [18] | 86.2 | 34.8 | 62 | 26 | 12.8 | 30.9 | 14.4 | 27.7 | 38.4 | 10.0 | 3.1 | 38.3 | 34.5 | 49.1 | 6.0 | 0.0 | 55.4 | 31.1 | 20.4 | 30.6 |
| URetinexNet [28] | 81.9 | 34.9 | 59.8 | 11.1 | 11.2 | 22.3 | 1.3 | 14.9 | 36.7 | 4.4 | 11.2 | 29.2 | 10.7 | 53.3 | 2.3 | 0.0 | 37 | 12 | 21.2 | 24.0 |
| RUAS [19] | 73.8 | 25.2 | 58.6 | 7.9 | 9.9 | 20.8 | 2.2 | 6.5 | 37.4 | 3.3 | 5.6 | 29.3 | 27.2 | 47 | 13.8 | 0.0 | 41.7 | 12.8 | 20.9 | 23.4 |
| Zero-DCE++ [16] | 85.3 | 37.8 | 61.8 | 12.1 | 11.9 | 26.9 | 2.1 | 20.4 | 37.9 | 6.1 | 12.6 | 32.9 | 35.4 | 60.1 | 10.4 | 0.0 | 49.4 | 12.1 | 22.9 | 28.3 |
| SCI [20] | 83.0 | 36.9 | 58.9 | 8.8 | 12.2 | 24.4 | 1.2 | 18.9 | 39.9 | 4.4 | 13.0 | 29.9 | 27.5 | 51.6 | 0.0 | 0.2 | 43.9 | 12.1 | 21.6 | 25.7 |
| EnlightenGAN [13] | 71.9 | 25.1 | 59.0 | 9.4 | 12.2 | 23.0 | 3.5 | 3.3 | 42.4 | 2.8 | 15.8 | 32.7 | 27.0 | 51.6 | 7.1 | 0.0 | 52.8 | 14.1 | 19.6 | 24.9 |
| LEDNet [33] | 77.8 | 32.8 | 61.8 | 11.2 | 14.2 | 29.9 | 4.6 | 5.9 | 48.1 | 5.7 | 32.0 | 43.0 | 10.7 | 40.1 | 1.6 | 0.0 | 50.1 | 15.0 | 20.0 | 26.6 |
| AdaBN [17] | 85.9 | 38.5 | 62.9 | 23.7 | 11.6 | 31.0 | 12.1 | 23.6 | 39.0 | 11.5 | 3.7 | 42.1 | 37.1 | 55.0 | 6.1 | 0.0 | 60.3 | 22.2 | **24.5** | 31.1 |
| SAN-SAW [21] | 60.8 | 10.7 | 44.8 | 14.3 | 5.2 | 12.2 | 12.0 | 22.4 | 30.0 | 6.9 | 0.6 | 18.4 | 11.5 | 19.0 | 4.0 | **0.6** | 16.0 | 10.0 | 4.5 | 16.0 |
| RobustNet [3] | 84.9 | 41.3 | 52.3 | 13.6 | **18.3** | 38.5 | **31.2** | 33.1 | **53.9** | 8.0 | 26.9 | 44.5 | 41.1 | 53.3 | 0.2 | 0.0 | **61.5** | 28.9 | 22.3 | 34.5 |
| MAET [5] | 80.7 | 39.0 | 56.8 | 24.8 | 15.7 | 28.9 | 6.3 | 6.7 | 34.8 | 7.6 | 2.8 | 30.9 | 24.3 | 45.0 | 1.1 | 0.2 | 45.4 | **40.7** | 9.9 | 26.4 |
| CIConv [15] | **90.3** | 48.3 | 57.8 | 29.3 | 11.1 | 36.3 | 24.4 | 30.2 | 45.8 | 7.6 | 8.0 | 37.6 | 40.1 | 69.7 | 10.1 | 0.0 | 55.0 | 37.4 | 16.0 | 34.5 |
| Ours | 89.7 | **52.0** | **67.2** | **32.5** | 14.8 | **39.4** | 10.4 | **34.7** | 46.9 | **11.8** | **37.3** | **46.5** | **44.8** | **76.0** | **58.6** | 0.0 | 55.3 | 30.0 | 20.4 | **40.5** |

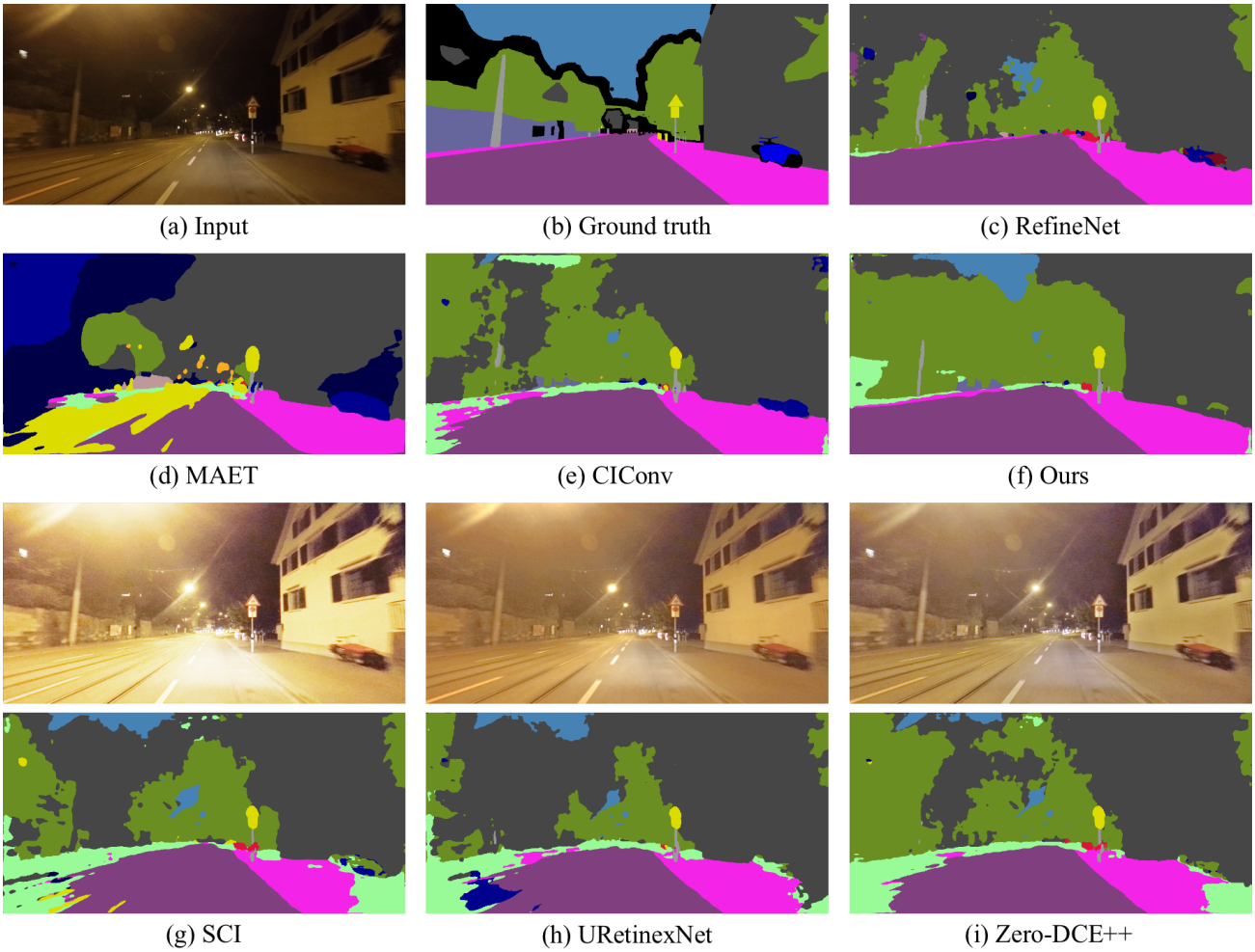


(a) Input (b) Ground truth (c) RefineNet

(d) MAET (e) CIConv (f) Ours

(g) SCI (h) URetinexNet (i) Zero-DCE++

Figure 6. Qualitative segmentation results on the Nighttime Driving [6] dataset. Comparison methods include baseline RefineNet [18], zero-shot day-night adaptation methods [5, 15], and low-light enhancement methods [16, 20, 28].



(a) Input (b) Ground truth (c) RefineNet

(d) MAET (e) CIConv (f) Ours

(g) SCI (h) URetinexNet (i) Zero-DCE++

Figure 7. Qualitative segmentation results on the Dark-Zurich [24] dataset. Comparison methods include baseline RefineNet [18], zero-shot day-night adaptation methods [5, 15], and low-light enhancement methods [16, 20, 28].


Figure 8. Qualitative Visual Place Retrieval Results. For each group, the query image is shown on the left; the first two images (*i.e.*, two images that have the highest similarity with the query image) are shown on the right. Compared with the baseline model GeM [22] that often gets deceived by the nighttime appearance, our model can extract features more robust to illumination and thus retrieve the correct daytime image showing the same scene as the query image.


Figure 9. Qualitative low-light action recognition results. We compare our method with low-light video enhancement methods SGZ [31], SMOID [12], and StableLLVE [30].

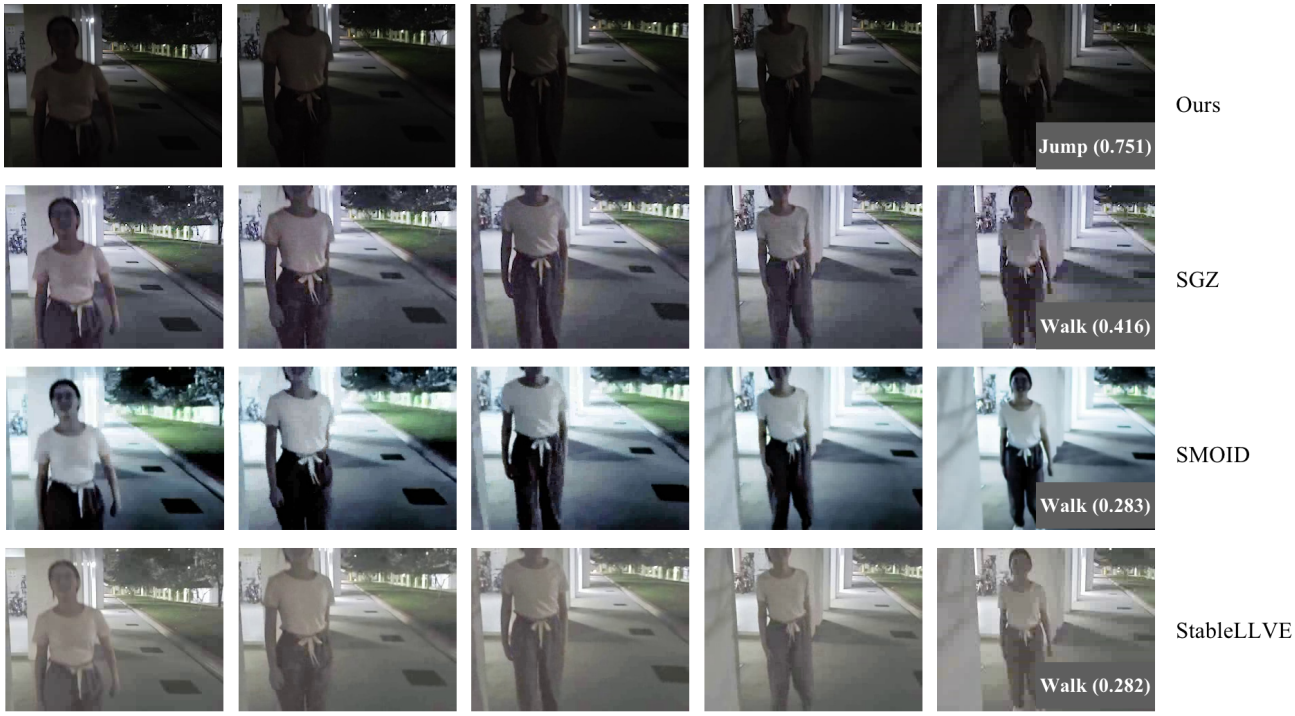



Figure 10. Qualitative low-light action recognition results. We compare our method with low-light video enhancement methods SGZ [31], SMOID [12], and StableLLVE [30].

## References


[1] Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. Invariant risk minimization. *arXiv*, 2019. 5

[2] João Carreira and Andrew Zisserman. Quo vadis, action recognition? A new model and the kinetics dataset. In *CVPR*, 2017. 3

[3] Sungha Choi, Sanghun Jung, Huiwon Yun, Joanne T. Kim, Seungryong Kim, and Jaegul Choo. Robustnet: Improving domain generalization in urban-scene segmentation via instance selective whitening. In *CVPR*, 2021. 5, 8, 9

[4] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *CVPR*, 2016. 8

[5] Ziteng Cui, Guo-Jun Qi, Lin Gu, Shaodi You, Zenghui Zhang, and Tatsuya Harada. Multitask aet with orthogonal tangent regularity for dark object detection. In *ICCV*, 2021. 5, 8, 9, 10, 11

[6] Dengxin Dai and Luc Van Gool. Dark model adaptation: Semantic image segmentation from daytime to nighttime. In *ITSC*, 2018. 6, 8, 10

[7] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009. 2

[8] Terrance Devries and Graham W. Taylor. Improved regularization of convolutional neural networks with cutout. In *CoRR*, 2017. 3

[9] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *ICCV*, 2019. 3

[10] Jean-Bastien Grill, Florian Strub, Florent Altché, C. Tallec, Pierre H. Richemond, Elena Buchatskaya, C. Doersch, Bernardo Avila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, B. Piot, K. Kavukcuoglu, Rémi Munos, and Michal Valko. Bootstrap your own latent: A new approach to self-supervised learning. In *NeurIPS*, 2020. 2

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 2

[12] Haiyang Jiang and Yinqiang Zheng. Learning to see moving objects in the dark. In *ICCV*, 2019. 5, 13, 14

[13] Yifan Jiang, Xinyu Gong, Ding Liu, Yu Cheng, Chen Fang, Xiaohui Shen, Jianchao Yang, Pan Zhou, and Zhangyang Wang. Enlightengan: Deep light enhancement without paired supervision. *IEEE TIP*, 2021. 5, 8, 9

[14] Andrei Kapishnikov, Tolga Bolukbasi, Fernanda Viegas, and Michael Terry. Xrai: Better attributions through regions. In *ICCV*, 2019. 7

[15] Attila Lengyel, Sourav Garg, Michael Milford, and Jan C. van Gemert. Zero-shot day-night domain adaptation with a physics prior. In *ICCV*, 2021. 5, 6, 7, 8, 9, 10, 11

[16] Chongyi Li, Chunle Guo Guo, and Chen Change Loy. Learning to enhance low-light image via zero-reference deep curve estimation. *IEEE TPAMI*, 2021. 5, 8, 9, 10, 11

[17] Yanghao Li, Naiyan Wang, Jianping Shi, Jiaying Liu, and Xiaodi Hou. Revisiting batch normalization for practical domain adaptation. In *ICLRW*, 2017. 8, 9

[18] Guosheng Lin, Anton Milan, Chunhua Shen, and Ian Reid. Refinenet: Multi-path refinement networks for high-resolution semantic segmentation. In *CVPR*, 2017. 3, 7, 8, 9, 10, 11

[19] Risheng Liu, Long Ma, Jiaao Zhang, Xin Fan, and Zhongxuan Luo. Retinex-inspired unrolling with cooperative prior architecture search for low-light image enhancement. In *CVPR*, 2021. 5, 8, 9

[20] Long Ma, Tengyu Ma, Risheng Liu, Xin Fan, and Zhongxuan Luo. Toward fast, flexible, and robust low-light image enhancement. In *CVPR*, 2022. 5, 8, 9, 10, 11

[21] Duo Peng, Yinjie Lei, Munawar Hayat, Yulan Guo, and Wen Li. Semantic-aware domain generalized segmentation. In *CVPR*, 2022. 5, 8, 9

[22] Filip Radenović, Giorgos Tolias, and Ondřej Chum. Fine-tuning cnn image retrieval with no human annotation. *IEEE TPAMI*, 2019. 3, 5, 8, 12

[23] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*, 2015. 1

[24] Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Guided curriculum model adaptation and uncertainty-aware evaluation for semantic nighttime image segmentation. In *ICCV*, 2019. 6, 7, 8, 9, 11

[25] Xin Tan, Ke Xu, Ying Cao, Yiheng Zhang, Lizhuang Ma, and Rynson W. H. Lau. Night-time scene parsing with a large real dataset. *IEEE TIP*, 2021. 7

[26] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *JMLR*, 2008. 6

[27] Wenjing Wang, Xinhao Wang, Wenhan Yang, and Jiaying Liu. Unsupervised face detection in the dark. *IEEE TPAMI*, 2022. 5

[28] Wenhui Wu, Jian Weng, Pingping Zhang, Xu Wang, Wenhan Yang, and Jianmin Jiang. Uretinex-net: Retinex-based deep unfolding network for low-light image enhancement. In *CVPR*, 2022. 5, 8, 9, 10, 11

[29] Yuecong Xu, Jianfei Yang, Haozhi Cao, Kezhi Mao, Jianxiong Yin, and Simon See. Arid: A new dataset for recognizing action in the dark. In *Deep Learning for Human Activity Recognition*, 2021. 3, 5

[30] Fan Zhang, Yu Li, Shaodi You, and Ying Fu. Learning temporal consistency for low light video enhancement from single images. In *CVPR*, 2021. 5, 13, 14

[31] Shen Zheng and Gaurav Gupta. Semantic-guided zero-shot learning for low-light image/video enhancement. In *WACVW*, 2022. 5, 13, 14

[32] Kaiyang Zhou, Yongxin Yang, Yu Qiao, and Tao Xiang. Domain generalization with mixstyle. In *ICLR*, 2021. 5

[33] Shangchen Zhou, Chongyi Li, and Chen Change Loy. Lednet: Joint low-light enhancement and deblurring in the dark. In *ECCV*, 2022. 5, 8, 9