NightLab: A Dual-level Architecture with Hardness Detection for Segmentation at Night

Xueqing Deng[1,2]*, Peng Wang[2], Xiaochen Lian[2], Shawn Newsam[1]
[1]EECS, University of California at Merced, [2]ByteDance Inc.
{xdeng7, snewsam}@ucmerced.edu, {peng.wang, xiaochen.lian}@bytedance.com

Abstract

The semantic segmentation of nighttime scenes is a challenging problem that is key to impactful applications like self-driving cars. Yet, it has received little attention compared to its daytime counterpart. In this paper, we propose `NightLab`*, a novel nighttime segmentation framework that leverages multiple deep learning models imbued with night-aware features to yield State-of-The-Art (SoTA) performance on multiple night segmentation benchmarks. Notably,* `NightLab` *contains models at two levels of granularity, i.e. image and regional, and each level is composed of light adaptation and segmentation modules. Given a nighttime image, the image level model provides an initial segmentation estimate while, in parallel, a hardness detection module identifies regions and their surrounding context that need further analysis. A regional level model focuses on these difficult regions to provide a significantly improved segmentation. All the models in* `NightLab` *are trained end-to-end using a set of proposed night-aware losses without handcrafted heuristics. Extensive experiments on the NightCity [44] and BDD100K [59] datasets show* `NightLab` *achieves SoTA performance compared to concurrent methods. Code and dataset are available at https://github.com/xdeng7/NightLab.*

1. Introduction

Semantic segmentation is a fundamental task in computer vision on which there has been much progress recently with the introduction of deep semantic parsing methods, *e.g.*, DeepLab [4, 6] and Transformers [13, 26]. However, the focus has been almost entirely limited to daytime benchmarks such as CityScapes [9] and ADE20k [62]. Much less progress has been made on the nighttime problem such as establishing strong benchmarks and designing effective architectures. Yet, success on the nighttime scene segmentation is crucial for a number of impactful applications such as autonomous driving [29], robotic vision [11], etc.

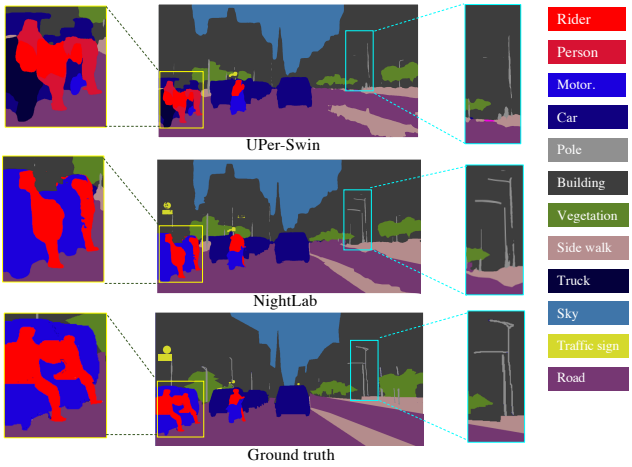

Figure 1. Visual comparisons of segmentation results from UPer-Swin [26] and our proposed `NightLab`. `NightLab` shows improvements on the parts of motorcycle and rider, where UPer-Swin predicts rider as person, and motorcycle as car, and poles are missing. `NightLab` is able to provide details for small objects.

There are far fewer open-source labelled nighttime images than daytime ones. Most nighttime image collections contain only unlabeled images and so there has been a lot of work [54] on unsupervised domain adaptation between daytime and nighttime for segmentation. Our experience, based on experiments, is that these adaptation frameworks perform poorly in practice due to the large domain gap between daytime and nighttime scenes.

Recently, Tan *et al.* [44] proposed NightCity which makes progress on two key challenges in nighttime segmentation: the lack of a large realistic dataset and the large illumination variation that results from over or under exposure in night scenes. The NightCity effort resulted a large dataset of densely labelled images and a segmentation model that contains an exposure-guided layer designed for light changes. The model is shown to outperform unsupervised methods.

Our work in this paper takes additional steps in this di-

*This work was done when Xueqing was interned at ByteDance.

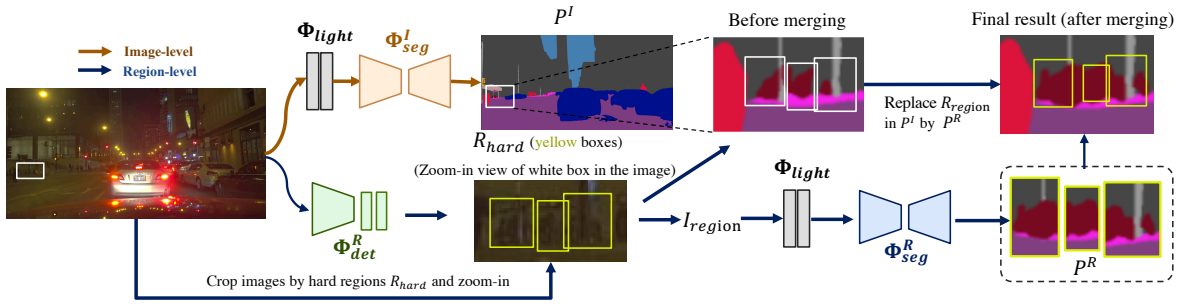

Figure 2. **NightLab overview and inference**. With the input images, there is a duel-level architecture to produce the final output. Note the hard contexts (boxes) here are automatically discovered without ground truth. 1) The image-level networks $\Phi^I_{light}, \Phi^I_{seg}$ is used to create predictions P^I for the whole images. Most of the easy regions $\mathcal{R}_{easy}$ can be accurately predicted by Φ^I_{seg}. 2) Then, hard regions $\mathcal{R}_{hard}$ will be detected by HDM Φ^R_{det} with the input images. Once the regions are discovered, they will be zoom-in and processed by Φ^R_{light} and Φ^R_{seg} to obtain local prediction P^R of $\mathcal{R}_{hard}$. At last, P^R will be merged back to P^I to generate the final segmentation output.

rection and proposes NightLab, a nighttime segmentation framework focused on architecture optimization using real labelled night images which results in a significant, *i.e.*, $\sim$10%, absolute improvement over the original NightCity baseline [44]. Specifically, we employ effective model architecture design to achieve two goals related to the large illumination variation in nighttime images. First, is to reduce the amount of light variation. Rather than performing simple exposure enhancement, we propose a regularized light adaptation module (ReLAM) based on a large amount of day and night images. Different from image translation approaches [1,21,64] that can significantly alter image appearance, ReLAM preserves night image texture which helps avoid large domain shifts during adaptation, yielding better generalization for night images. Second, due to the low-light levels and blurry texture, small objects are often not distinguishable based on their appearance alone. Therefore, as illustrated in early work on object understanding [30], context is crucial for helping resolve potential ambiguity in certain nighttime image regions. While most deep networks contain multi-scale structures by enumerating scales such as HRNet [49], night images have objects with substantial scale variation, *e.g.*, road light and bicycle as illustrated in Fig. 1. Such variation is often beyond the scope of the enumerated scales in modern networks. To tackle this issue, we propose a hardness detection module (HDM), which adopts the idea of regional proposal network (RPN) from objection detection. Our HDM identifies regions, along with their context, that need additional attention and analysis. Finally, we adopt the SoTA architecture of Swin-Transformer [26] as our segmentation encoder and embed DeformConv [10] as the decoder. This provides improved architecture capacity and context modeling ability.

In summary, as illustrated in Fig. 2, inference in NightLab works as follows. Given a night image, the image level model first adapts the image light through ReLAM (Φ^I_{light}) and sends it to an image-level segmentation model (Φ^I_{seg}), producing an overall segmentation. In tandem, HDM is used to detect hard regions that need further analysis. These regions are cropped, batched, and sent to a regional level model which adapts and segments these regional patches similar to the image level. Here, our regional model is not trained over the full set of classes but is limited to a subset of automatically identified difficult classes such as bicycle and road light to better mine the context information needed to distinguish their semantics. The segmented results from the region level are then merged with the image level parsing results, yielding the final segmentation.

Finally, since we found many mislabelled pixels in the NightCity validation images as shown in Sec. 4, we manually relabel the dataset so that the evaluation of our and other methods is more meaningful. Extensive experimentation shows NightLab outperforms concurrent methods and ablative studies demonstrate the contribution of each of the proposed modules

In summary, our contribution includes:

1. We propose NightLab, a dual-level architecture with novel modules including ReLAM and HDM specifically designed for night scene segmentation. The framework achieves SoTA performance on multiple nighttime benchmarks.
2. We propose an effective training pipeline for the architecture whose modularity provides good interpretability of our improvements.
3. We derive a more accurate benchmark dataset from NightCity and conduct extensive experiments that investigate a variety of night scene segmentation strategies. Our benchmark dataset and strong baseline serve as a good starting point for future researchers.

2. Related Works

Semantic segmentation. This task has been actively studied in past few decades, and turns to be practical in many real-world applications with the rising of deep learning with

convolutions [51, 56, 63, 66, 67]. In general, two principles are followed when designing the architecture,*i.e*., discover multi-scale context and design high-resolution representation. Some representative works includes the initial Fully Convolutional Network (FCN) based methods [27], the series of DeepLab networks [4–6], multi-scale aware networks like HRNet [49], PSPNet [60], and models with attention module such as cross attention [20]. To better model object context, Deformable convolution [10] is proposed to be embedded in these network architecture for performance enhancement.

Most recently, Transformer [12] shows advanced performance due to its multi-layer full attention mechanism in language processing. DPT [33] first shows a full transformer based network which outperforms convolutional based architectures. Later, transformers have raised the attention in the computer vision community. Vision Transformers [3,13,23,45,52,58] have been widely studied for various vision tasks. Most recently, to reduce the computational complexity inside transformer, Swin-Transformer [26] proposes a shifted window operation, which provides SoTA performance over various benchmarks. In our work, we adopt Swin-Transformer as our backbone, and show it significantly improves over our night segmentation benchmarks from NightCity [44]. However, there still remain issues brought by light variation in nighttime motivating us to design various modules to enhance its performance.

Domain adaptation (DA) for segmentation. DA is designed for transferring knowledge from source to target domain, where usually there are rich labeled data in source domain while unlabeled data in target domain. For example, lots of works [28,31,40,46–48,65] try to adapt segmentation model trained from synthetic images, *e.g*., GTA5 [35] or SYNTHIA [37], to real images, *e.g*., Cityscapes. Instead of adopting models, some works [15,17,18,24,32,55,61] try to adapt images by applying style transfer [64] that transforms images in target domain to source domain. The former is trying to obtain a domain-invariant representations which has to be retrained when a new domain is added. The latter does not need to change the segmentation network but only need to train a new adaptor. Specifically, when targeting at daytime and nighttime adaptation, Song *et al*. [42] follows the former strategy, which proposes to transfer unlabeled day-time and night-time images into a shared latent feature space. Sun *et al*. [43] follows the latter by proposing to translate the day-time and night-time images by CycleGAN, and training the segmentation model on synthetic night-time images. Similar approaches using CyclanGAN such as using a curriculum framework for adaptation [38, 39] is explored across different time (daytime, twilight, and nighttime). However, existing domain adaptation methods for segmentation are mostly in a unsupervised manner. The improvement of adaptation will be significantly marginalized when supervised label is available. In `NightLab`, our architecture falls in the strategy of adaptation then segmentation since it has better explain ability, and we carefully design a regularized module, *i.e*., ReLAM, to make it useful in supervised manner.

Vision tasks in the dark. Meanwhile, there are rising interests in analyzing images in the dark, *e.g*., localization [2], depth estimation [50], object detection [41], person reidentification [7, 36], etc. Instead of using synthesized images, the major approaches are working on various *enhancement*, which try to lighten the low-illumination areas and scenes for easier feature extraction. For example, [50] leverages mapping consistent image enhancement module to enhance image visibility for depth estimation. `NightLab` set up a new benchmark in the field of segmentation, and we hope our approach could benefit mutually with other tasks such as depth or video understanding.

3. NightLab

As shown in Fig. 2, `NightLab` consists of two levels: image level and region level. At each level, there is a segmentation module with a network Φ_{seg}. To improve the generalization of the segmentation models for night images, a ReLight Adaptation Module (ReLAM), Φ_{light}, is used at each level. Region-level module works solely on hard regions in night images, which are detected by a region proposal network Φ^R_{det}, *i.e*., region detection network (RDN) or Hardness Detection Module (HDM). Segmentation results from the two levels are merged to create the final results.

In this section, we elaborate the proposed `NightLab` architecture as follows: In section 3.1 we describe the core modules at each level, *i.e*., ReLAM and segmentation networks. We then introduce two region proposal networks which propose hard regions to the region-level segmentation module during inference: RDN as the baseline method in section 3.2, and its improvement, HDM, in section 3.3.

3.1. Core modules at image and region levels

Regularized Light Adaptation Module (ReLAM). ReLAM contains a generator to adapt the image light, and a discriminator for training. For generator we adopt the style transfer network proposed in [22], which contains 6 level of resnet layers as designed [1]. We denote the generator as Φ_{light} which tries to align the lighting of nighttime to daytime images. It takes an RGB night image I as input and output a RGB light shift $L = \Phi_{light}(I)$. The final enhanced image can be denoted as $I' = I + L$.

To train ReLAM generator Φ_{light}, we induce two objectives based on our collected day time image set $\mathcal{I}_d$ and night time image set $\mathcal{I}_n$. The first objective is structural similarity loss (SSIM) [50,53,54] which penalize a dramatic change of

[1] https://github.com/jcjohnson/fast-neural-style

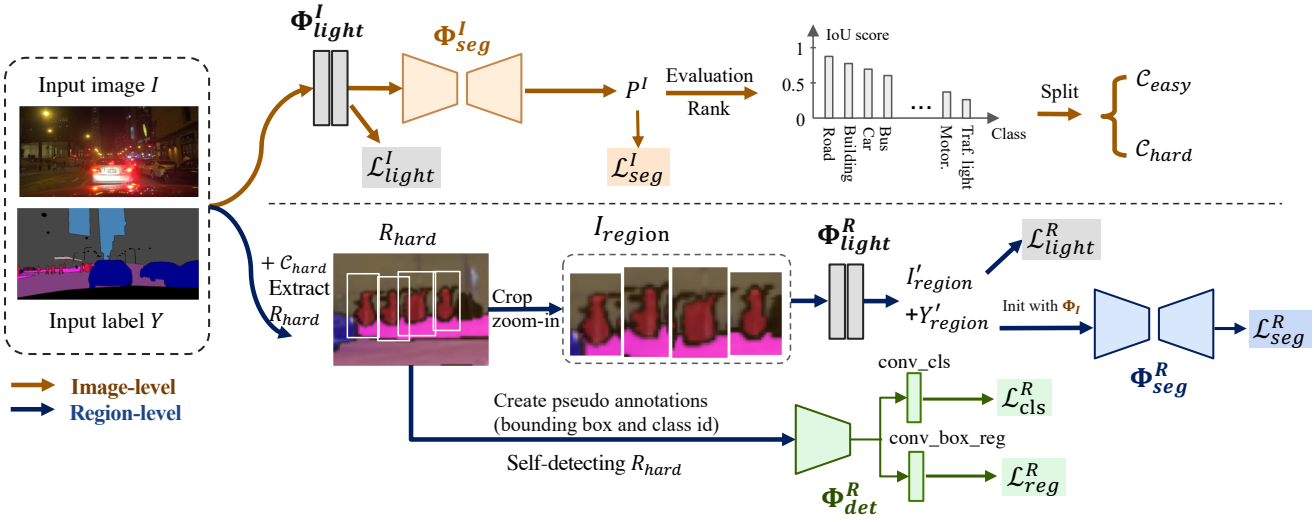

Figure 3. **Training pipleline**. `NightLab` training framework. The image-level modules (Upper) $\Phi^I_{light}, \Phi^I_{seg}$ will be first trained by their corresponding losses $\mathcal{L}^I_{light}, \mathcal{L}^I_{seg}$ (Sec. 3.1). After prediction over a split validation set, evaluation and ranking will be performed to split the classes into easy and hard categories $\mathcal{C}_{easy}, \mathcal{C}_{hard}$. Then, we adopt the semantic ground truth from $\mathcal{C}_{hard}$ to extract the hard regions (white boxes) by finding connected components (red regions). They can be first used to crop and zoom-in our training images to form a regional train set ($\mathcal{I}_R$ and$\mathcal{Y}_R$), which is utilized to optimize region-level modules (Lower) $\Phi^R_{light}, \Phi^R_{seg}$ with regional losses $\mathcal{L}^R_{light}, \mathcal{L}^R_{seg}$. Finally, based on hard regions, we derive pseudo ground truth for Φ^R_{det} (RDN or HDM) which can be trained with regression and classification losses, i.e. $\mathcal{L}^R_{reg}, \mathcal{L}^R_{cls}$. (Sec. 3.2 and Sec. 3.3)

internal texture. The second is GAN loss [16] which transfer the image for easy appearance distinguishing. Formally, for SSIM, we have the loss defined as,

$$\mathcal{L}_S = \sum_{I_i \in \mathcal{I}_d, \mathcal{I}_n} 1 - \text{SSIM}(I_i, I'_i), \tag{1}$$

where I'_i is the adapted image of I_i.

For GAN loss, ReLAM uses a discriminator D to distinguish if the adapted image is close to daytime or nighttime following the GAN training pipeline, which can be formulated as,

$$\mathcal{L}_P(D) = \sum_{I_i \in \mathcal{I}_d, I'_i \in \mathcal{I}'} \log(D(I_i)) + \log(1 - D(I'_i)) \tag{2}$$

and our final loss for ReLAM is $\mathcal{L}_{light} = \mathcal{L}_S + \mathcal{L}_P(D)$.

ReLAM will be trained at both image (Φ^I_{light}) and region level (Φ^R_{light}) with their corresponding training set. In our experiments, directly using a CycleGAN [64] image transfer could harm the segmentation performance since it generates images with lots of texture distortion, while ReLAM works better thanks to the regularization inside the architecture and losses.

Image-level segmentation module. The architecture of Φ^I_{seg} is a network composed of a encoder based on Swin-Transformer [26] and a decoder based on UperNet [57]. Additionally, to increase the context modelling ability, we replace convolutional layers in UperNet with DeformConv [10]. We name this as "NightLab-Baseline", which is the best baseline we could obtain as a single network. To train such an architecture, we use the enhanced images and their corresponding ground truths. Formally, we adopt 2D cross entropy loss $\mathcal{L}^I_{seg}$ to compare the segmentation prediction P^I and the ground truth annotation Y^I, with the same training setting as Swin-Transformer.

Region-level segmentation module. As we explained earlier, due to variations like lighting and scales, using only image-level segmentation module is not sufficient as some objects require different context (*e.g.* smaller objects and low-light regions will benefit from zoomed-in views). To solve this issue, we use a region-level segmentation module that focuses on hard regions R_{hard}, *i.e.*, regions at which the image-level module fails. We adopt the same architecture for region-level segmentation network Φ^R_{seg} as Φ^I_{seg}, with the exception of the number of classes, which is dependent on the number of hard classes determined by the auto selection process described below.

We extract the hard regions R_{hard} via a simple yet efficient auto selection process which utilizes the segmentation masks predicted by the image-level module Φ^I_{seg}: Based on the initial segmentation prediction P^I from Φ^I_{seg}, the semantic classes $\mathcal{C}$ are split into two sets, the easy set $\mathcal{C}_{easy}$ and the hard set $\mathcal{C}_{hard}$. Specifically, as shown in Fig. 3, the per-class IoU of P^I are first computed, and then the classes with low IoU scores (< 0.5) are selected as $\mathcal{C}_{hard}$ and the rest classes consists of $\mathcal{C}_{easy}$. Next the instances of $\mathcal{R}_{hard}$ are generated based on the label masks. Since we do not have the instance-level segmentation in the ground truth annotations, we approximately consider every con-

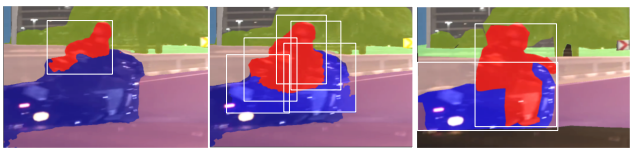

Figure 4. Comparison of hard regions from different strategies (white boxes). Left: using prediction from Φ^I_{seg}; Middle: using region detection network (RDN); Right: using ground truth. (red: rider, blue: motorcycle, dark blue: car).

nected component from a class as an instance of that class. For each instance, we crop an image using a bounding box around it, which serves as the context of the instance. These cropped images are "zoomed-in" to predefined sizes before being fed to the region-level network Φ^R_{seg} for training. With the dataset, Φ^R_{seg} can be learnt using cross entropy loss $\mathcal{L}^R_{seg} = \sum_{P^R_j \in \mathcal{P}^R, Y^R_j \in \mathcal{Y}_R} CE(P^R_j, Y^R_j)$, where $P_j = \Phi_R(I^R_j), I^R_j \in \mathcal{I}_R$, and $\mathcal{I}_R$ and $\mathcal{Y}_R$ represent the cropped images and their semantic label masks.

3.2. Self-detecting hard regions at night

Region-level module cannot be directly used at inference when ground truth annotations are not available for cropping hard regions. One solution is to use prediction from Φ^I_{seg} to create $\mathcal{R}_{hard}$. The issue of this approach is that the prediction of Φ^I_{seg} is not always accurate. Instead we train a region detection network (RDN) Φ^R_{det} to detect instances in $\mathcal{R}_{hard}$. Inspired by Faster RCNN [34], we adopt idea of the region proposal network (RPN) in [34] to detect the hard regions: we first produce the pseudo annotation boxes and label them with hard or easy based on quality of the prediction of Φ^I_{seg}, and then RDN is learned by optimizing the objective $\mathcal{L}^R_{det}$, which is the sum of the bounding box regression loss and the classification loss:

$$\mathcal{L}^R_{det} = \mathcal{L}^R_{reg} + \mathcal{L}^R_{cls} \tag{3}$$

$$\mathcal{L}^R_{reg} = \sum\nolimits_{r_k \in \mathcal{R}_{hard}} \text{smooth\_l1}(t(r_k), t(\hat{r}_k))$$

$$\mathcal{L}^R_{cls} = \sum\nolimits_{r_k \in \mathcal{R}_{hard}} CE(p_{r_k}, y_{r_k}) \tag{4}$$

where $t(.)$ is a tuple that represents a bounding box, *i.e.*, (x, y, dw, dh), and y_{r_k} denotes the classification label for the box derived from semantic label mask. We keep 10 proposals for each image using non maximum suppression.

As shown in Fig. 4, our proposed RDN address the concerns. The results cover most area of rider and motorcycle even though each proposal covers only part of the object. Once $\mathcal{R}_{hard}$ is learned, images can be cropped and zoomin, later fed through Φ^R_{seg} to produce regional estimation for $\mathcal{C}_{hard}$. At last, we can merge results from $\mathcal{P}^R$ to $\mathcal{P}^I$ to create a new mixture prediction of $\mathcal{R}_{hard}$ and $\mathcal{R}_{easy}$.

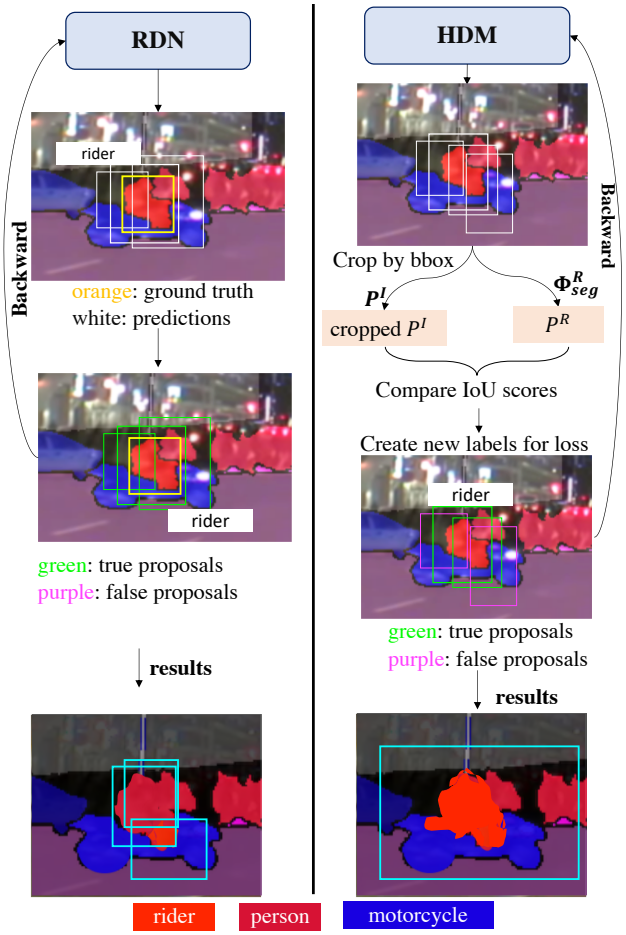

Figure 5. Region proposal networks. Left: region detection network (RDN) results of hard regions using ground truth solely based on segmentation mask. Right: hardness detection module (HDM) results using generated ground truth guided by region-level models.

3.3. Detecting hard regions with contexts

The problem of RDN is that RDN tends to generate a lot of proposals most of which do not contain the contexts that the region-level network Φ^R_{seg} needs. For example, as shown at the bottom left of Fig. 5, most of the proposals of "rider" from RDN only cover the rider but not motorcycle. Images cropped according to these proposals will be misclassified as "person" by Φ^R_{seg}. In this section, we propose our improvement over RDN, Hardness Detection Module (HDM), which is learned to propose regions with contexts that favor the prediction of Φ^R_{seg}.

Before jumping into the details of HDM, let's first explain what a "good proposal" is: A proposal is good if Φ^R_{seg}'s prediction to its region is better than the image-level model Φ^I_{seg}'s prediction to the same region; in other words, a good proposal should help Φ^R_{seg} improve Φ^I_{seg}'s prediction to its region.

Follow the above intuition, we modify the learning rou-

tine of RPN in RDN as follows: for each proposal, we crop the image according to the proposal and obtain its segmentation prediction P^R from Φ^R_{seg}. The same proposal is used to crop the prediction from P^I by Φ^I_{seg}. The proposal is considered positive if the IoU score of P^I is better than that of P^R, and vice versa. The RPN is then trained with the new labels. As a result, HDM tends to propose regions that have better context.

As shown at the bottom right of Fig. 5, unlike RDN, HDM manages to propose a region with both the motorcycle and the person in it and Φ^R_{seg} correctly recognizes the person as "rider" thanks to the correct context in the region.

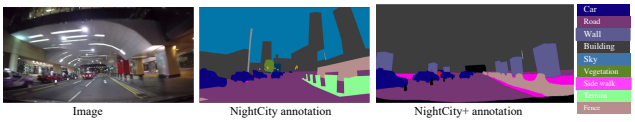

Figure 6. Example corrected label of NightCity by NightCity+.

4. Experiments

4.1. Experimental setups

Datasets. We consider two nighttime segmentation datasets to evaluate `NightLab`. First, **NightCity** [44], which is a large dataset with urban driving scenes at nighttime designed for supervised semantic segmentation. It consists of 2998/ 1299 train/val images with full pixel-wise annotations. The labels are compatible with Cityscapes [9] where there are 19 classes of interests in total. From the dataset, we found there are some mis-labelled validation images (Fig. 6), especially for some slim objects, which is difficult to reveal the true improvements. Therefore, we asked human labellers to relabel part of the "validation" set for more accurate evaluation. We call this **NightCity+**, and report all our experimental results on the new val set. More labelled details can be found in supplementary materials. Similar with NightCity [44], we also experimented with Cityscapes as assist training data to help improve the performance. Second, **BDD100K** [59], which is a high-resolution autonomous driving dataset with 100,000 video clips in multiple cities and under various conditions.

We pick the night images with their label inside to build a new dataset, namely **BDD100K-Night**, which consists of 343/58 images in train/val sets with 18 classes of interests. The amount of data is much less than NightCity+, to augment the training, we adopt the whole BDD100K dataset including 7,000 images and the corresponding annotations to jointly train, and then evaluate on BDD100K-Night val set. Last, we also explore many other datasets for setting benchmarks, such as Zurich-Dark [38], ApolloScape [19], but found them containing few or no night training images with labels, which is not suitable in our situation.

Implementation Details. Since both datasets do not include official test sets, we treat their val set as test, and randomly split the original train set to train and val set with a portion of 3:1 for hyperparameter tuning, hard class selection and model selection. After the tuning, the full train set is used to optimize the final model, and test over the test set. For training and inference, the images of NightCity+ will be rescaled to 1024×2048. For both datasets, we adopt training augmentations of random scale with ratio sampled in range of $(0.5, 2.0)$, random flip, photonmetric distortion and normalization. Afterwards, the image will be cropped into a shape of 512×1024 before feeding into the model. For evaluation, we apply a multi-scale augmentation strategy with ratios of [0.25, 0.5, 0.75, 1.0, 1.25]. During training, we select hard classes with mIoU less than 0.5. We use CityScapes and BDD100K day split for day image set, NightCity+ and BDD100K night split for night image set to train image-level ReLAM. For region-level ReLAM, we crop out corresponding hard classes in day and night images to compose the train set.

We run our experiments based on mmsegmentation [8]. Our experiments are performed on 8 V100 GPUs, with 2 samples per GPU. Sync BatchNorm is turned on for all experiments. We produce the baseline results for NightCity+

(a) NightCity+

Network	Backbone	Resolution	NightCity+	w/Citys
*NightCity [44]	Res101	512x1024	51.5	53.9
PSPNet [60]	Res101	512x1024	54.75	56.89
PSPNet [60]	Res101	1024x2048	55.64	57.52
DeeplabV3+ [6]	Res101	512x1024	54.21	58.29
DeeplabV3+ [6]	Res101	1024x2048	54.47	59.03
UPerNet [26]	Swin-Base	512x1024	57.71	59.35
HRNetV2 [49]	HRNet-W48	1024x2048	55.89	58.49
DANet [14]	Res101	1024x2048	55.98	57.72
UPer-Swin [57]	Res101	1024x2048	55.81	56.98
UPer-ViT [13]	ViT	1024x2048	57.13	58.07
UPer-Swin [26]	Swin-Base	1024x2048	58.25	59.67
*`NightLab-HDM`	Swin-Base	512x1024	59.84	61.07
`NightLab` (DeeplabV3+)	Res101	1024x2048	56.21	60.41
`NightLab-Baseline`	Swin-Base	1024x2048	59.25	60.37
`NightLab-RDN`	Swin-Base	1024x2048	60.27	62.11
`NightLab-HDM`	Swin-Base	1024x2048	**60.73**	**62.82**

(b) BDD100K-Night

Network	Backbone	Resolution	Night	w/100K
PSPNet [60]	Res101	720x1280	29.96	46.24
HRNetV2 [49]	HRNet-W48	720x1280	29.86	44.32
DANet [14]	Res101	720x1280	29.46	42.64
DeeplabV3+ [6]	Res101	720x1280	30.11	43.44
UPerNet [57]	Res101	720x1280	30.88	47.68
UPer-ViT [13]	ViT	720x1280	30.74	47.81
UPer-Swin [26]	Swin-Base	720x1280	31.74	48.04
`NightLab` (DeeplabV3+)	Res101	720x1280	31.27	45.11
`NightLab-Baseline`	Swin-Base	720x1280	32.37	48.52
`NightLab-RDN`	Swin-Base	720x1280	34.13	49.81
`NightLab-HDM`	Swin-Base	720x1280	**35.41**	**50.42**

Table 1. Comparisons to SoTA semantic segmentation networks on NightCity+ and BDD-Night with metric of mIoU. The results of first column after "Resolution" are models train with only night images, and the results of the column after trained with daytime data augmentation, i.e. with Cityscapes to NightCity, and BDD100K day images to BDD100K-Night. Here, for NightCity, lines with $*$ denotes evaluation is done over the original NightCity val set since we do not have models in NightCity [44].

Method	Adaptation Approach	Network	Nightcity+	Nightcity+ and Citys	BDD100K-Night	BDD100K
NightCity [44]	Exposure-Aware	Res101	51.8	53.9	-	-
UPerNet [26]	Segmentation	UPerNet-Swin	57.71	59.35	31.74	48.52
Pix2PixHD [21]	Image Translation	UPerNet-Swin	-	43.38	-	38.67
CycleGAN [64]	Image Translation	UPerNet-Swin	-	44.07	-	39.64
SingleHDR [25]	Image Enhancement	UPerNet-Swin	57.07	58.88	31.64	48.32
DANNet [54]	Network Adaptation	UPerNet-Swin	-	58.69	-	48.25
AdaptSeg [46]	Network Adaptation	UPerNet-Swin	-	58.29	-	48.32
NightLab-B	Segmentation	UPerNet-Swin-DeformConv	59.25	60.37	32.37	48.52
NightLab-RDN	Dual-level segmentation	UPerNet-Swin-DeformConv	60.27	62.11	34.13	49.81
NightLab-HDM	Dual-level segmentation	UPerNet-Swin-DeformConv	**60.73**	**62.82**	**35.41**	**50.24**

Table 2. Comparison study of adaptation approaches. mIoU(%) are reported.

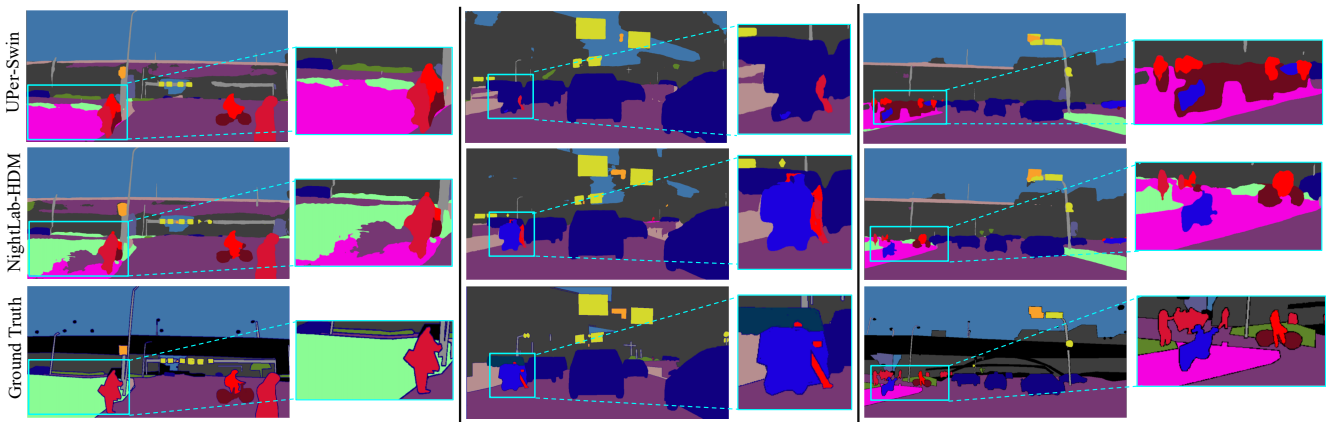

Figure 7. Qualitative results on NightCity+ val set jointly trained with Cityscapes.

and BDD100K in Tab. 1 with default hyperparameters in mmsegmentation where the models are trained with 80k iterations. NightLab follows the same training configurations of UPer-Swin in [26] for training segmentation modules. While for ReLAM, we adopt the configurations as in [46], and for HDM, we follow the setting in FastR-CNN [34]. Since the two level model can be run in parallel, our approach runs in the same speed as Swin-Transformer.

4.2. Experimental results

Compare to SoTA methods. In Tab. 1, we compare NightLab to SoTA semantic segmentation methods. For each baseline, we create the experimental result with the same training configurations as ours, and obtain the result by training with night data only or training with additional day data as discussed in experimental setup in Sec. 4.1.

From the table, we can see our constructed baseline, "NightLab-Baseline", simplified as NightLab-B latter, contains only image-level segmentation module already outperforms the best of concurrent SoTA networks, *i.e.*, "Uper-Swin" based on Swin-transformer [26], on both datasets with gains of ∼1% for both training configurations. Specifically, for NightCity+, "NightLab-RDN" represents adding RDN module inside the network, and it outperforms UPer-Swin with a margin of 2.44% when jointly trained with Cityscapes. After replacing RDN with HDM, "NightLab-HDM" is able to perform better, which achieved mIoU scores of 60.73% and 62.82% in single train and joint train, yielding 2.48% and 3.15% improvements over UPer-Swin. Similar gain is observed for BDD100K-Night. "NightLab-HDM" achieves the best performances with 35.41% and 50.42% under single train and joint train settings. To further verify the effectiveness of our proposed modules, we switch the base network of NightLab-HDM to Deeplab V3+, as shown in lines of "NightLab (DeeplabV3+)", we observed sufficient gain over baseline "DeeplabV3+" with improvements of 1.74% and 1.38% in NightCity+, and 1.16% and 1.67% in BDD100K-Night.

Compare to adaptation for segmentation methods. Since most existing methods for segmentation through adaption are unsupervised, directly comparing with them on our benchmarks is not fair. Therefore, we consider to train an adaptor to adapt night images to day images, then use the adapted night dataset plus real day dataset to supervise a segmentation model based on UPerNet-Swin. We hope the adaptation can help the segmentation network learn better. In Tab. 2, we explore various SoTA adaptors to adapt night images to day images for training. Specifically, we first use "Pix2PixHD/CycleGAN" to transfer the appearance of the nighttime images into daytime. However, under supervised setting, we found such adaptation actually performs worse than a vanilla baseline. This is because the adapted appearance of night images are dramatically changed, which can hardly be consistent in training and testing time. Then, we additionally explore other DA methods with less modification on image contents by using a pretrained SingleHDR [25] for image enhancement, although it does not harm the results much, we do not see any improvements.

Method	road	side.	build.	wall	fence	pole	light	sign	vege	terr.	sky	pers.	rider	car	truck	bus	train	moto.	bicy.	mIoU
UPerNet-Swin	92.1	**55.3**	84.4	59.1	56.1	38.9	34.0	60.9	63.1	**29.9**	89.0	60.9	32.7	85.7	66.5	73.5	60.1	39.2	45.7	59.35
SingleHDR	90.8	51.7	83.1	59.0	53.4	34.9	34.2	57.1	60.4	27.5	86.5	55.4	34.0	80.9	66.5	73.2	57.9	39.0	38.7	57.07
Pix2PixHD	85.9	33.5	68.8	50.0	42.9	27.0	13.8	34.6	47.9	20.1	82.8	35.5	12.1	72.7	53.5	58.3	42.6	17.3	25.0	43.38
DANNet	91.5	53.8	85.4	59.9	54.9	38.9	34.7	60.0	62.2	28.7	88.2	58.1	35.4	83.1	66.9	72.1	58.1	40.6	42.0	58.69
NightLab-B	92.4	54.2	85.3	59.3	57.3	38.2	28.3	61.8	62.3	24.0	89.1	62.4	43.2	86.0	68.1	78.9	61.4	48.6	46.2	60.37
NightLab-RDN	92.5	53.6	85.2	59.9	58.0	42.2	37.7	62.9	**63.4**	26.6	**89.7**	63.3	45.2	86.4	70.1	80.5	62.6	50.2	50.6	62.11
NightLab-HDM	**92.6**	54.9	**85.8**	59.1	**58.4**	**43.1**	**38.1**	**63.3**	63.0	26.6	89.3	63.3	**47.1**	**86.7**	**71.9**	**81.0**	**63.7**	**52.3**	**54.8**	**62.82**

Table 3. Per class iou scores. Model are jointly trained with NightCity+ and Cityscapes, evaluated on NightCity+ val set.

Finally, we explore whether unsupervised adapted networks can provide better pre-trained feature for night images since the weight itself should contain adaptation ability. Specifically, we adopt DANNet [54] and AdaptSeg [46] to first train an adapted segmentation network with unlabelled day/night images, and then finetune it with our full labelled day/night dataset. Unfortunately, it also does not help with the accuracy as shown in lines of "DANNet" and "AdaptSeg". It seems the adapted feature could be biased, yielding a slightly worse optimized weights than vanilla training in our experiments. More details can be found in supplementary materials.

Class performance We further analyze the model contributions for each class shown in Tab. 3. We can see NightLab makes improvements on almost all classes. Especially, hard classes detected such as pole, rider, motorcycle, and bicycle are improved mostly thanks to HDM, ReLAM and region-level modules. For example, the score of class *pole* is increased from 38.2% to 42.2% after applied HDM and region-level model, which are within our expectation. Corresponding qualitative results are shown in Fig. 7. However, there are some hard classes detected have not been improved such as *terrain*. This is due to the fact that terrain is more likely to be background rather than object, which can be easily confused by *vegetation* in particular in the dark environment. We found the approach is more effective for object-like classes.

Method	Backbone	FPN	Seg head	Nightcity+	+Citys
UPerNet [57]	Res101	Conv2D	Conv2D	55.81	59.03
NightLab-B	Res101	Conv2D	DefConv [10]	56.31	59.33
NightLab-B	Res101	DefConv [10]	DefConv [10]	56.54	59.85
UPer-Swin [26]	Swin-Base	Conv2D	Conv2D	58.25	59.67
NightLab-B	Swin-Base	Conv2D	DefConv [10]	58.68	59.99
NightLab-B	Swin-Base	DefConv [10]	DefConv [10]	**59.25**	**60.37**

Table 4. Ablation study on our proposed baseline architectures adding deformable convolution (DefConv [10]) to enrich contextual features for multiscale objects. Results are reported on NightCity+ val set.

4.2.1 Ablation study

NightLab-B architecture We present the ablation study of the baseline architecture shown in Tab. 4. We make modification of Conv2D of the decode head (UPerHead from UPerNet [57]) to build our baseline method. UPerHead is composed of a feature fusion module FPN and a segmentation conv head. Deformable conv can be a substitute of regular conv to produce feature with better context. The proposed decode head can be combined with any backbone. We verify the effectiveness by performing experiments on model with a backbone of Swin-Transformer and ResNet101. As shown in the table, Deformable conv results in better performances with both backbones. When replacing all the conv layers with deformable conv, the model achieves the best performance.

Method	mIoU(%)
UPerNet-Swin[[26]]	59.67
NightLab-B	60.37
NightLab-B + Φ^R_{seg} w/ RDN proposed hard regions	61.51
NightLab-B + Φ^R_{seg} w/ HDM proposed hard regions	62.31
NightLab-B + Φ^I_{light}	60.74
NightLab-B + Φ^I_{light} + Φ^R_{seg} w/ RDN proposed hard regions	61.87
NightLab-B + Φ^I_{light} + Φ^R_{seg} w/ HDM proposed hard regions	62.51
NightLab-B + Φ^I_{light} + Φ^R_{seg} w/ RDN proposed hard regions + Φ^R_{light}	62.11
NightLab-B + Φ^I_{light} + Φ^R_{seg} w/ HDM proposed hard regions + Φ^R_{light}	62.82

Table 5. Ablation study on NightLab model variants. Models are trained jointly with NightCity+ and Cityscapes, evaluated on NightCity+ val set. NightLab-B represents our proposed baseline segmentation architecture.

NightLab modules We present the ablation study for each module of NightLab in Tab. 5. We first show the effectiveness of the dual-level architecture without lighting adaptation. Utilizing Φ^R_{seg} with hard regions detected by RDN or HDM results can raise the performance from 60.37% to 61.51% and 62.01% respectively. HDM provides the best result. Next, we demonstrate the lighting adaptation module of ReLAM. We can see both levels of ReLAM raise the performance for all modules. We observe that adding the lighting adaptation module towards the whole image can increase the mIoU by ∼0.5%. A further region based lighting adaptation module can raise slight improvement of ∼0.3%.

5. Conclusion

This paper presents NightLab, an architecture that is suitable for night scene segmentation. It contains dual-level models, which segments images with proper context and lights in a supervised setting, yielding SoTA performance. However, the overall performance of segmentation accuracy at night is still far behind that from daytime. This work takes a few steps in effectively mining context and reduce light variations in challenging visual situation, and we hope it may motivate other researchers to discover other crucial properties toward closing the performance gap at night.

References

[1] Ivan Anokhin, Pavel Solovev, Denis Korzhenkov, Alexey Kharlamov, Taras Khakhulin, Aleksei Silvestrov, Sergey Nikolenko, Victor Lempitsky, and Gleb Sterkin. High-resolution daytime translation without domain labels. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 2

[2] Asha Anoosheh, Torsten Sattler, Radu Timofte, Marc Pollefeys, and Luc Van Gool. Night-to-day image translation for retrieval-based localization. In *2019 International Conference on Robotics and Automation (ICRA)*, 2019. 3

[3] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020. 3

[4] Liang-Chieh Chen, George Papandreou, IIasonas Kokkinos, Kevin Murphy, and Allan Yuille. DeepLab: semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2017. 1, 3

[5] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*, 2017. 3

[6] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018. 1, 3, 6

[7] Yanbei Chen, Xiatian Zhu, and Shaogang Gong. Instance-guided context rendering for cross-domain person re-identification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019. 3

[8] MMSegmentation Contributors. MMSegmentation: Openmmlab semantic segmentation toolbox and benchmark. https://github.com/open-mmlab/mmsegmentation, 2020. 6

[9] 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 *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 1, 6

[10] Jifeng Dai, Haozhi Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu, and Yichen Wei. Deformable convolutional networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2017. 2, 3, 4, 8

[11] Guilherme N DeSouza and Avinash C Kak. Vision for mobile robot navigation: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2002. 1

[12] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (ACL)*, 2018. 3

[13] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations (ICLR)*, 2021. 1, 3, 6

[14] Jun Fu, Jing Liu, Haijie Tian, Yong Li, Yongjun Bao, Zhiwei Fang, and Hanqing Lu. Dual Attention Network for Scene Segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 6

[15] Rui Gong, Wen Li, Yuhua Chen, and Luc Van Gool. DLOW: Domain flow for for adaptation and generalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 3

[16] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2014. 4

[17] Judy Hoffman, Eric Tzeng, Taesung Park, Jun-Yan Zhu, Phillip Isola, Kate Saenko, Alexei Efros, and Trevor Darrell. Cycada: Cycle-consistent adversarial domain adaptation. In *Proceedings of the 35th International Conference on Machine Learning (ICML)*, 2018. 3

[18] Sheng-Wei Huang, Che-Tsung Lin, Shu-Ping Chen, Yen-Yi Wu, Po-Hao Hsu, and Shang-Hong Lai. Auggan: Cross domain adaptation with gan-based data augmentation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018. 3

[19] Xinyu Huang, Xinjing Cheng, Qichuan Geng, Binbin Cao, Dingfu Zhou, Peng Wang, Yuanqing Lin, and Ruigang Yang. The apolloscape dataset for autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2018. 6

[20] Zilong Huang, Xinggang Wang, Lichao Huang, Chang Huang, Yunchao Wei, and Wenyu Liu. Ccnet: Criss-cross attention for semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019. 3

[21] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 2, 7

[22] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2016. 3

[23] Kwonjoon Lee, Huiwen Chang, Lu Jiang, Han Zhang, Zhuowen Tu, and Ce Liu. Vitgan: Training gans with vision transformers. *arXiv preprint arXiv:2107.04589*, 2021. 3

[24] Che-Tsung Lin, Yen-Yi Wu, Po-Hao Hsu, and Shang-Hong Lai. Multimodal structure-consistent image-to-image translation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020. 3

[25] Yu-Lun Liu, Wei-Sheng Lai, Yu-Sheng Chen, Yi-Lung Kao, Ming-Hsuan Yang, Yung-Yu Chuang, and Jia-Bin Huang. Single-image HDR reconstruction by learning to reverse the camera pipeline. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 7

[26] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021. 1, 2, 3, 4, 6, 7, 8

[27] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015. 3

[28] Yawei Luo, Liang Zheng, Tao Guan, Junqing Yu, and Yi Yang. Taking a closer look at domain shift: Category-level adversaries for semantics consistent domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition(CVPR)*, 2019. 3

[29] Moritz Menze and Andreas Geiger. Object scene flow for autonomous vehicles. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015. 1

[30] Aude Oliva and Antonio Torralba. The role of context in object recognition. *Trends in cognitive sciences*, 11, 2007. 2

[31] Fei Pan, Inkyu Shin, Francois Rameau, Seokju Lee, and In So Kweon. Unsupervised intra-domain adaptation for semantic segmentation through self-supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition(CVPR)*, 2020. 3

[32] Fabio Pizzati, Pietro Cerri, and Raoul de Charette. Comogan: continuous model-guided image-to-image translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 3

[33] René Ranftl, Katrin Lasinger, David Hafner, Konrad Schindler, and Vladlen Koltun. Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2020. 3

[34] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 2015. 5, 7

[35] Stephan R. Richter, Vibhav Vineet, Stefan Roth, and Vladlen Koltun. Playing for data: Ground truth from computer games. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2016. 3

[36] Eduardo Romera, Luis M Bergasa, Kailun Yang, Jose M Alvarez, and Rafael Barea. Bridging the day and night domain gap for semantic segmentation. In *2019 IEEE Intelligent Vehicles Symposium (IV)*, 2019. 3

[37] German Ros, Laura Sellart, Joanna Materzynska, David Vazquez, and Antonio M Lopez. The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 3

[38] Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Guided curriculum model adaptation and uncertainty-aware evaluation for semantic nighttime image segmentation. In *The IEEE International Conference on Computer Vision (ICCV)*, 2019. 3, 6

[39] Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Map-guided curriculum domain adaptation and uncertainty-aware evaluation for semantic nighttime image segmentation. *arXiv preprint arXiv:2005.14553*, 2020. 3

[40] Swami Sankaranarayanan, Yogesh Balaji, Arpit Jain, Ser Nam Lim, and Rama Chellappa. Learning from synthetic data: Addressing domain shift for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 3

[41] Mark Schutera, Mostafa Hussein, Jochen Abhau, Ralf Mikut, and Markus Reischl. Night-to-day: Online image-to-image translation for object detection within autonomous driving by night. *IEEE Transactions on Intelligent Vehicles*, 2020. 3

[42] Can Song, Jin Wu, Lei Zhu, Mei Zhang, and Haibin Ling. Nighttime road scene parsing by unsupervised domain adaptation. *IEEE Transactions on Intelligent Transportation Systems*, 2020. 3

[43] Lei Sun, Kaiwei Wang, Kailun Yang, and Kaite Xiang. See clearer at night: towards robust nighttime semantic segmentation through day-night image conversion. In *Artificial Intelligence and Machine Learning in Defense Applications*, 2019. 3

[44] Xin Tan, Yiheng Zhang, Ying Cao, Lizhuang Ma, and Rynson WH Lau. Night-time scene parsing with a large real dataset. *IEEE Transactions on Image Processing (TIP)*, 2021. 1, 2, 3, 6, 7

[45] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2021. 3

[46] Yi-Hsuan Tsai, Wei-Chih Hung, Samuel Schulter, Kihyuk Sohn, Ming-Hsuan Yang, and Manmohan Chandraker. Learning to adapt structured output space for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 3, 7, 8

[47] Yi-Hsuan Tsai, Kihyuk Sohn, Samuel Schulter, and Manmohan Chandraker. Domain adaptation for structured output via discriminative patch representations. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019. 3

[48] Tuan-Hung Vu, Himalaya Jain, Maxime Bucher, Matthieu Cord, and Patrick Pérez. Advent: Adversarial entropy minimization for domain adaptation in semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition(CVPR)*, 2019. 3

[49] Jingdong Wang, Ke Sun, Tianheng Cheng, Borui Jiang, Chaorui Deng, Yang Zhao, Dong Liu, Yadong Mu, Mingkui Tan, Xinggang Wang, et al. Deep high-resolution representation learning for visual recognition. *IEEE Transactions on*

Pattern Analysis and Machine Intelligence (TPAMI), 2017. 2, 3, 6

[50] Kun Wang, Zhenyu Zhang, Zhiqiang Yan, Xiang Li, Baobei Xu, Jun Li, and Jian Yang. Regularizing nighttime weirdness: Efficient self-supervised monocular depth estimation in the dark. *arXiv preprint arXiv:2108.03830*, 2021. 3

[51] Peng Wang, Xiaohui Shen, Zhe Lin, Scott Cohen, Brian Price, and Alan Yuille. Towards unified depth and semantic prediction from a single image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015. 3

[52] Wenxiao Wang, Lu Yao, Long Chen, Deng Cai, Xiaofei He, and Wei Liu. Crossformer: A versatile vision transformer based on cross-scale attention. *arXiv e-prints*, 2021. 3

[53] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing (TIP)*, 2004. 3

[54] Xinyi Wu, Zhenyao Wu, Hao Guo, Lili Ju, and Song Wang. Dannet: A one-stage domain adaptation network for unsupervised nighttime semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 1, 3, 7, 8

[55] Zuxuan Wu, Xintong Han, Yen-Liang Lin, Mustafa Gokhan Uzunbas, Tom Goldstein, Ser Nam Lim, and Larry S Davis. Dcan: Dual channel-wise alignment networks for unsupervised scene adaptation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018. 3

[56] Fangting Xia, Peng Wang, Liang-Chieh Chen, and Alan L Yuille. Zoom better to see clearer: Human and object parsing with hierarchical auto-zoom net. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2016. 3

[57] Tete Xiao, Yingcheng Liu, Bolei Zhou, Yuning Jiang, and Jian Sun. Unified perceptual parsing for scene understanding. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018. 4, 6, 8

[58] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M Alvarez, and Ping Luo. Segformer: Simple and efficient design for semantic segmentation with transformers. *arXiv preprint arXiv:2105.15203*, 2021. 3

[59] Fisher Yu, Wenqi Xian, Yingying Chen, Fangchen Liu, Mike Liao, Vashisht Madhavan, and Trevor Darrell. Bdd100k: A diverse driving video database with scalable annotation tooling. *arXiv preprint arXiv:1805.04687*, 2018. 1, 6

[60] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid Scene Parsing Network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 3, 6

[61] Ziqiang Zheng, Yang Wu, Xinran Han, and Jianbo Shi. Forkgan: Seeing into the rainy night. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020. 3

[62] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 1

[63] Fangrui Zhu, Yi Zhu, Li Zhang, Chongruo Wu, Yanwei Fu, and Mu Li. A unified efficient pyramid transformer for semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*, 2021. 3

[64] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017. 2, 3, 4, 7

[65] Xinge Zhu, Hui Zhou, Ceyuan Yang, Jianping Shi, and Dahua Lin. Penalizing top performers: Conservative loss for semantic segmentation adaptation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018. 3

[66] Yi Zhu, Karan Sapra, Fitsum A. Reda, Kevin J. Shih, Shawn Newsam, Andrew Tao, and Bryan Catanzaro. Improving semantic segmentation via video propagation and label relaxation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 3

[67] Yi Zhu, Zhongyue Zhang, Chongruo Wu, Zhi Zhang, Tong He, Hang Zhang, R Manmatha, Mu Li, and Alexander J Smola. Improving semantic segmentation via efficient self-training. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2021. 3