Bridging the Day and Night Domain Gap for Semantic Segmentation

Eduardo Romera[1], Luis M. Bergasa[1], Kailun Yang[2], Jose M. Alvarez[3] and Rafael Barea[1]

Abstract— Perception in autonomous vehicles has progressed exponentially in the last years thanks to the advances of vision-based methods such as Convolutional Neural Networks (CNNs). Current deep networks are both efficient and reliable, at least in standard conditions, standing as a suitable solution for the perception tasks of autonomous vehicles. However, there is a large accuracy downgrade when these methods are taken to adverse conditions such as nighttime. In this paper, we study methods to alleviate this accuracy gap by using recent techniques such as Generative Adversarial Networks (GANs). We explore diverse options such as enlarging the dataset to cover these domains in unsupervised training or adapting the images on-the-fly during inference to a comfortable domain such as sunny daylight in a pre-processing step. The results show some interesting insights and demonstrate that both proposed approaches considerably reduce the domain gap, allowing IV perception systems to work reliably also at night.

I. INTRODUCTION

Convolutional Neural Networks (CNNs) have become an exceptional ally to Intelligent Vehicles (IV) thanks to the ground-breaking advances that they have produced in the Computer Vision (CV) field in the recent years. Vision tasks like object detection (e.g. pedestrians, cars) or semantic segmentation (i.e. pixel-wise scene classification) can be easily addressed with these methods in an accurate and efficient way, which have displaced other expensive sensors like RADAR and LiDAR to a second place, normally meant for redundancy and security. Specifically, semantic segmentation methods can solve most perception tasks in an unified way, yielding diverse benefits over previous existing techniques that needed to be modeled in separate complex ways [1].

Some CNN methods like PSPNet [3] or DeepLab [4] perform semantic segmentation with very high accuracies, but these architectures are also extremely inefficient computationally. For these reasons, in a previous work we proposed ERFNet [5][6], a CNN that produces semantic segmentation both efficiently and accurately, specially designed for IV environments which have computational constraints. In a follow-up work, we explored how robustness in semantic segmentation methods could be addressed with effective techniques to make a model work reasonably well in any
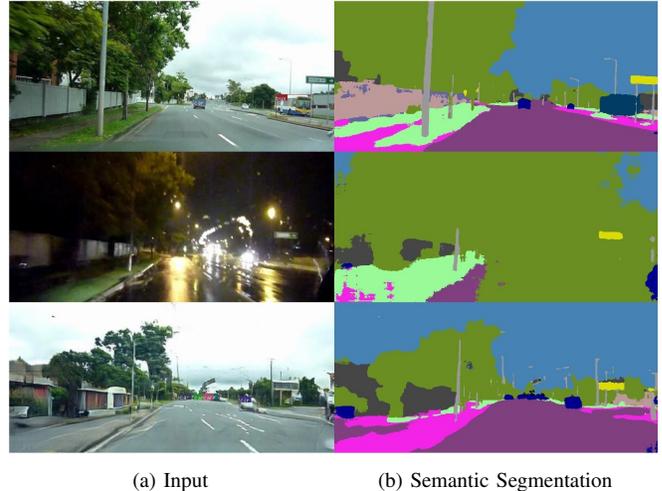
(a) Input (b) Semantic Segmentation

Fig. 1. Comparative example from Alderley [2] dataset. It can be seen that the night image is rainy and has very low visibility, which causes a large accuracy gap between day (top row) and night (mid row) performance. The converted-to-day image (bottom row) allows the segmentation model to acquire an accurate understanding of the scene with minimal error.

environment regardless of the training domain [7]. However, there are diverse non-ideal environments that remain challenging for CV methods and have not yet been extensively studied, such as the night time. The difference in the image properties between day and night domains produces a large accuracy downgrade (see Fig. 1) for all models, which are normally trained in daylight datasets [8].

In this paper, we propose two methods to bridge the domain gap between night and day images and extensively analyze their performance in diverse datasets. We leverage recent techniques such as Generative Adversarial Networks (GANs) and explore diverse options such as converting the dataset to cover the night domain in an unsupervised training, or adapting the images on-the-fly during inference to a comfortable domain such as sunny daylight in a pre-processing step. With these experiments, the ultimate goal is to make current CNN-based perception methods work robustly in any domain and lighting condition.

Additionally, we collect four datasets in diverse environments with both day and night images and GPS information, which will be made publicly available. We test a known semantic segmentation model in our datasets by comparing day, night and generated day images (converted from night) to effectively measure the domain gap and the suitability of our proposal. Additionally, we extend the publicly available Cityscapes dataset [8] to night images in order to train a segmentation model that works well in that domain, performing qualitative and quantitative evaluations. Our experiments re-

(a) UAH   (b) ZJU   (c) Alderley   (d) Milford

Fig. 2. Day (top row) and Night (bottom row) examples for each of the 4 collected datasets. UAH and ZJU were collected with our instrumented vehicles at our universities, while Alderley and Milford were collected from the internet. These four datasets are used to train our stylization GANs.

flect interesting findings about the performance of stylization GANs and their capability to help in solving the domain gap between perception in day and night.

II. RELATED WORKS

A. Domain Adaptation

CNNs learn features in the training phase that are specific to the training domain, so a common problem in these networks is that they are dependant on training images and they usually do not work that well on other domains. In a recent work, we explored how to adapt segmentation networks like ERFNet with effective techniques to improve robustness to unseen domains [7]. What we found is that simple techniques like data augmentation played a big role in improving robustness to any domain. Other works like Ros et al. [9] proposed more complex domain adaptation techniques based on an unsupervised image transformation method which follows a global color transfer strategy to convert the image colors into a suitable domain where the models can perform much better. More recently, works like [10] and [11] proposed to leverage synthetic data to improve the flexibility of the training domains and create models that would be more prepared to unseen data during inference.

B. Generative models

Very recent works aim to use generative models to generate artificial images that can help to train deep models in a better way. The main idea of Generative Adversarial Networks[12] (GANs) is to train two networks simultaneously. The first network is a generator that produces a (realistic) image from an input seed, and the second is a discriminator which is trained to take the generator's output and evaluate how real it looks compared to a set of desired target images (Ground Truth). Both networks evolve together during training, resulting in the generator becoming better at generating real images for tricking the discriminator, and the discriminator becoming better at detecting if the generated image is real or fake. Despite their impressive results, GANs are extremely hard to train and authors must resort to several tricks to make discriminator and generator converge to a

good solution during training. The reader may refer to [13] for learning more about architectural features and training procedures that can be applied to GANs in the specific context of semi-supervised image classification.

C. Image Stylization

GANs are specially useful in the domain adaptation field since it is possible to train one to perform style conversion out of a set of input images. This way, practically infinite variations of the input can be generated with different styles and features to train a model with diverse domains. A known network is Pix2Pix [14], which uses a conditional GAN to learn a mapping from input to output images. An example use would be converting black and white images into color. The generator in this case tries to learn how to colorize a black and white image and the discriminator looks at the generators colorization attempts and learns to tell the difference between the colorizations the generator provides and the true colorized target image provided in the dataset.

While Pix2Pix produces impressive results, feeding training data is challenging since the two image spaces that need to be learned have to be pre-formatted into a single X/Y image that holds both tightly-correlated images. Instead, CycleGAN [15] proposed to perform a full translation cycle, which allows to train the model using two discrete and unpaired collections of images, resulting in a model that excels at performing style conversion without Pix2Pix difficulties.

A very recent work presented UNsupervised Image-to-image Translation Networks (UNIT) [16], which learns a joint distribution of images in different domains by using images from the marginal distributions in each domain. Authors proposed this image-to-image translation framework based on Coupled GANs which is able to learn the shared-latent space between training image sets. Additionally, their experiments were specially aimed at tasks like winter to spring conversion or day to night conversion, which is closer to the IV field than other style conversions in the literature. In the next section, we leverage the technique proposed in UNIT for designing a framework to make semantic segmentation models work robustly at night.
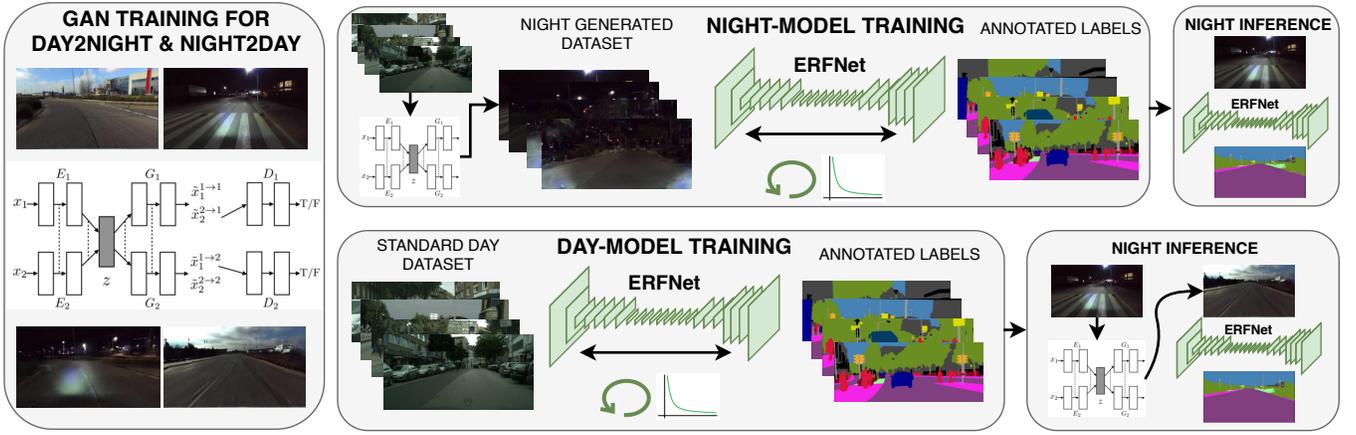
Fig. 3. Diagram showing the proposed methods for reducing the day-night performance gap. On the left, a stylization GAN is trained for converting images between day and night. On the right, the two proposed methods for training and deployment of semantic segmentation models at night. The top path involves converting a labeled dataset into night to train a model that performs well directly at night. The bottom path involves training a standard day model and performing night2day conversion at night inference prior to using the model.

III. METHOD

In this paper, the main objective is to reduce the large gap that is present in IV perception systems between day and night images (see Fig. 1). A full diagram of the proposed framework is depicted in Fig. 3. We study two different solutions, both based on training a style conversion GAN to perform day to night and vice-versa (as described in Section III-A). As a first option, we propose to convert day images present in fully labeled datasets like Cityscapes [8] to night to take advantage of their labels for training a model that performs well during night-time operation (Section III-B). As a second option, we propose to perform real-time night to day conversion during night inference to convert the input into a suitable domain (daytime) in which already trained models like ERFNet perform well (Section III-C).

A. Training GAN for Day-Night and Day-Night Conversions

Since both solutions depend on a trained GAN for conversion between domains, we describe how this process is carried out in the present subsection. We use the previously mentioned UNIT framework [16] as the stylization converter to use in our experiments. UNIT framework models each image in a domain by using GANs and Variational AutoEncoders (VAEs). The adversarial training objective interacts with a weight-sharing constraint, which enforces a shared latent space, to generate corresponding images in two domains, while the variational autoencoders relate translated images with input images in the respective domains. We refer the reader to [16] for specific details.

In the experiments section, we experiment with 4 different datasets which are divided in two training subsets: Day (TrainA) and Night (TrainB). Some examples can be seen in Fig. 2. During training with a dataset, two auto-encoder generators and two discriminators are trained for each domain (day and night). At every iteration, each generator picks a batch of random images in each domain and encodes them into a shared latent space. This latent space is then decoded by the opposite generator into an image of the other domain, which is enforced to be realistic by the adversarial discriminator. Both generators and both discriminators are trained in parallel, becoming better in each iteration at their respective tasks as their loss converges.

The resulting model contains a generator for day-to-night conversion and another one for night-to-day conversion, each one being an encoder-decoder network. For obtaining the conversion of any image after training, the process just consists of performing a forward pass in a generator's encoder and a forward pass in the opposite generator's decoder. The result is an image of the same resolution as the input but with its style converted into the other domain.

B. Generating a night dataset for training

Our first approach consists of using the trained GAN to convert all the day images from a dataset like Cityscapes [8] into night images in order to leverage their precise semantic segmentation labels. After converting the whole Train set to night, we feed these as inputs to ERFNet together with the segmentation labels to train end-to-end using a cross-entropy loss as usual (training described in [6]). The result is a segmentation model that performs well in night images and is ready to be deployed in IVs operating at night. Below are summarized the pros and cons of following this path (as opposed to the one described in Section III-C):

Benefits: This option is more efficient at inference time since the new night model is trained in an off-line way, so its architecture (and efficiency) is exactly the same as the original day model. Additionally, following this path allows putting additional care in how the new model trains by making its parameters more specific to the night domain or by cleaning the generated training set for the images that do not have artifacts created in the generation.

Disadvantages: One of the main disadvantages of this option is that the perception models require retraining, which can be time-consuming and not always feasible in the same conditions. For example, the images used for training rely on

the generator's resolution, which is normally smaller since the GAN memory requirements only allowed to train with 720x360 images, which is smaller than the usual 1024x512 or 2048x1024 used for training in Cityscapes. Additionally, since we do not have night images from Cityscapes to train a new GAN, we must rely on one trained in other dataset, which will not work as well due to the domain differences.

C. Adapting images to day during inference

Our second approach consists of using the trained GAN as well but leveraging the opposite conversion (night to day) to convert the images directly seen by the camera during night operation into synthetic day images where our models will perform better. This solution is simpler since it doesn't require retraining any of the detection or segmentation models but it adds an additional computational cost to the inferencing process since images have to be converted in real-time prior to segmentation. The trained generator will just encode every frame into the latent space and decode into a day frame, which will be fed into ERFNet to perform inference. Below are summarized the pros and cons of following this path (as opposed to the one described in Section III-B):

Benefits: In this option, the original perception models such as ERFNet can be conserved with their original weights. This means that there is no need to retrain models that already work well and have their performances extensively evaluated and assured. This also means that it is not needed to collect a training dataset like Cityscapes and convert it. Only day and night images of a domain are needed to train the stylization GAN, which can then be used to convert night images to day during inference in a single step.

Disadvantages: This option is less efficient because it requires performing a forward pass in the stylization GAN previous to the forward pass in the perception system. Additionally, the generator sometimes creates artifacts, which are harder to handle at on-line inference since we must trust on the generator's output directly (there is no possibility to filter them like in the off-line training case).

IV. EXPERIMENTS

A. Datasets and Setup

For training our chosen stylization GAN (UNIT) we collected 4 datasets that have day and night images. We captured data in our two universities (UAH and ZJU) with our instrumented vehicles, each car recorded data in two routes during day and two routes during night. Additionally we gathered data from the internet from two datasets that contain day and night sets: Alderley [2] and Milford [17]. However, the data from these two datasets is not ideal since the perspectives and quality of the images is extremely different from day to night. For example, Milford day images contain ego-vehicle's bonnet while nigh images do not, or Alderley's night images are rainy and are full of rain-drops in the windshield while day images are sunny. This messes up the GAN stylization training since it forces the model to believe that the latent space of "day" includes generating a bonnet in the bottom of the image, or the latent space of

TABLE I
MAIN INFORMATION OF THE FOUR USED DATASETS.

Dataset	Resolution	Num. Images		Comments
		Day	Night	
UAH	1280x720	9177	15257	Few cars/persons at night
ZJU	1920x1080	6848	6282	High contrast at night
Alderley	640x260	14607	16960	Night images rainy/noisy
Milford	1920x1080	83684	26624	Day contains ego-vehicle

"night" includes adding rain drops. For these reasons, in our two controlled datasets in UAH and ZJU we forced night and day to be as similar as possible, with the same perspectives, similar routes and weather conditions.

Table I shows the main info of the four collected datasets. Additionally some image examples can be seen in Fig. 2. Their main properties are also described below:

- **University of Alcala (UAH)** dataset was collected in university campus in Alcala de Henares (Spain) with our fully electric car "SmartElderlyCar". The dataset contains few pedestrians due to the location in the campus and has more cars during day images than night.
- **Zhejiang University (ZJU)** dataset captured in Yuquan campus in Hangzhou (China) with our multi-modal stereo vision sensor. Its illumination is remarkable for areas near to the camera but very dark for far areas, causing a lot of shadows. In general it has more trees and pedestrians than the others.
- **Alderley**: It was captured in two different conditions for the same routein the suburb of Alderley (Australia). The abrupt differences between domains, one having clear images (sunny day) and the other having very washed-out images with low visibility (rainy night), makes the data problematic for training the stylization GAN.
- **Milford**: The day route was recorded with a GoPro (wide FOV) for 10 km in Brisbane (Australia). The night drive is 13 km recorded with Sony A7s camera mounted on the roof. This dataset is the largest but its perspective and camera differences between day and night causes some undesired effects in our GAN.

For each of the four presented datasets we trained an UNIT model as described in Section III-A. This results in a day2night and night2day converter in each of the datasets, which will be the base of our tests. We use every converter in the corresponding dataset, which is essential to keep consistency in the transformations. We additionally trained a GAN mixing all datasets, which resulted in the model getting a very wrong idea of the latent space behind day and night, generating properties from the most frequent images like the perspective and bonnet from Milford dataset. Therefore, in all our tests we use a GAN trained in each specific dataset.

For testing semantic segmentation, we use our known Efficient Residual Factorized Network (ERFNet) [5][6], developed in previous publications and which holds a suitable trade-off for efficient and accurate performance. For obtaining quantitative evaluation, we train and evaluate the model in Cityscapes [8], which is a widely adopted semantic segmentation dataset due to its highly varied set of scenarios

(a) Input   (b) Semantic Segmentation   (c) Input   (d) Semantic Segmentation
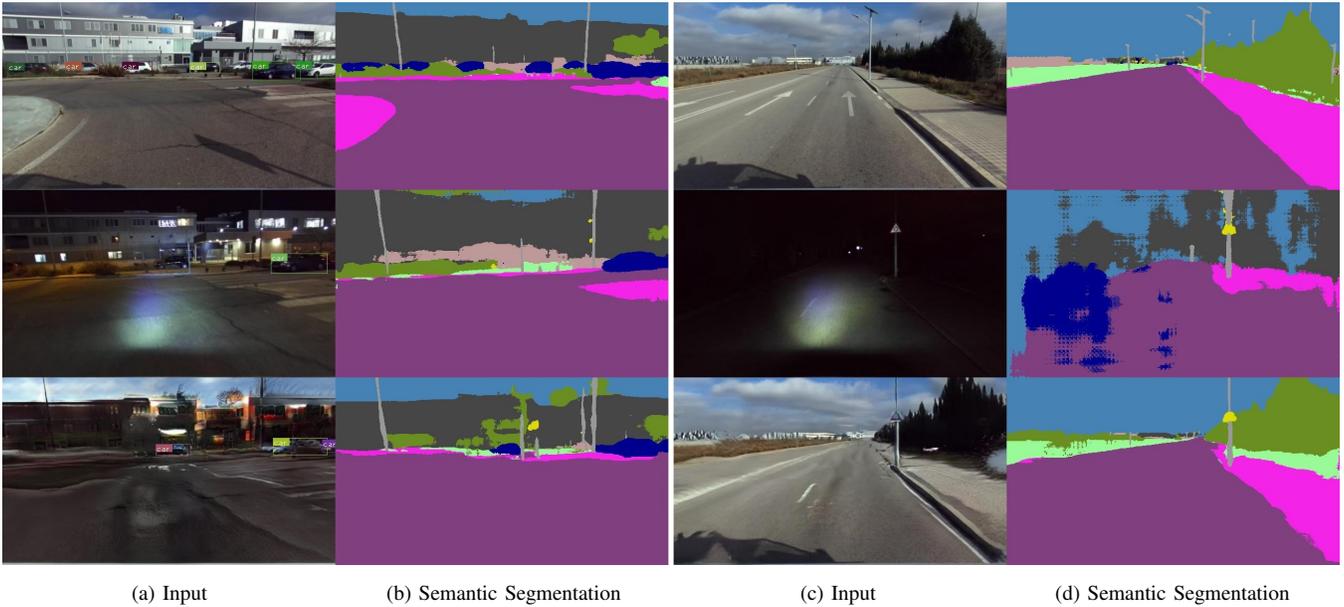
Fig. 4. Examples from UAH dataset (Top: day input, Mid: night input, Bottom: night-to-day conversion). Left image is better illuminated than the right one, causing the standard model to work decently in that frame. However, it is missing some cars and a good understanding of the ahead constructions. In the right example, the model is not capable of detecting anything correctly at night. The night-to-day conversion helps ERFNet detect everything accurately.

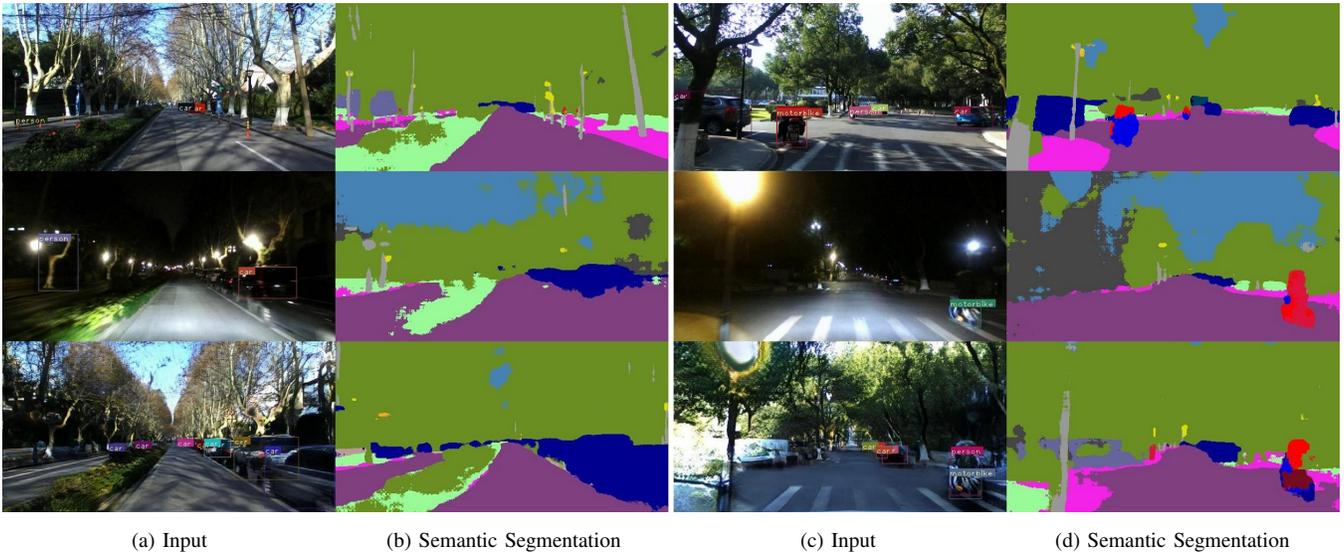(a) Input   (b) Semantic Segmentation   (c) Input   (d) Semantic Segmentation

Fig. 5. Examples from ZJU dataset (Top: day input, Mid: night input, Bottom: night-to-day conversion). In the left case, the models are missing some cars at the right and left sides of the road. In the right case, they are missing the cyclist and some sidewalk. The synthetic day image alleviates these issues.

and challenging set of 19 labeled classes. It contains a train set of 2975 images and a validation set of 500 images, all with fully labeled images at the pixel level. The model is trained with the same hyperparameters as specified in [6].

For enhancing visualization and comparison, we also added qualitative object detection results using the known single-shot detector Yolo-V3 [18], which detects traffic elements of interest and has a good precision-recall trade-off.

B. Qualitative Results in our Four Datasets

We use the night-to-day conversion to generate the synthetic day images from all night samples in all datasets. Afterwards, we compare qualitatively in each dataset both

perception systems (object detection and semantic segmentation) in each day image, night image, and synthetic day image (converted from night). We present results in different figures. Shown day and night frames (top and mid rows) might be from a different point of view or location, while both night and synthetic images (mid and bottom rows) should be from exactly the same input. For every example, we display the input image with drawn object detection boxes (left) and the semantic segmentation output (right).

Results for UAH dataset can be seen in Fig. 4. This dataset has high illumination in some areas (left example) while some others are very poorly illuminated (right example). In the highly illuminated areas, it can be seen that the night
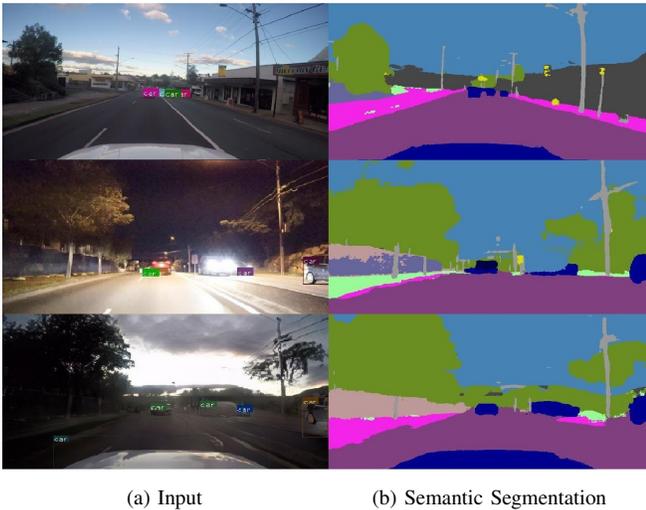
| | (a) Input | | (b) Semantic Segmentation |

Fig. 6. Example from Milford [17]. The night image (mid) has already good illumination so the model works decently in that frame. However, a car's headlight causes a glare which makes the model detect it as sky, while the to-day conversion (bottom) allows ERFNet to detect that car as well.

model is missing some cars in both the object detection and segmentation systems, while the transformation to day helps these models detect the missing cars. It also improves the segmentation of buildings and the sky. However, the models work already decently here given the high illumination in the campus. In the lowly illuminated areas, in the night image the models are not capable of detecting anything correctly, while the synthetic day image helps the segmentation model detect the road and its main parts accurately.

Results for ZJU dataset can be seen in Fig. 5. It can be seen in both examples that the segmentation performed at night is missing some important objects like cars and the bicyclist, while the converted-to-day image helps both segmentation and object detection detect these accurately.

Fig. 1 shows an example from Alderley dataset. The night image is very rainy, causing lightning effects which ruin the whole segmentation. The GAN creates a day image that is a suitable input for the segmentation model, which segments the scene accurately. Fig. 6 shows an example from Milford dataset, where the night image has already good illumination so the difference is smaller. However, the synthetic day image allows the model to prevent some glares which allows for better detection of frontal cars.

C. Semantic Segmentation Comparison

For evaluating our proposals quantitatively, we test them in Cityscapes Validation Set by converting its images to night and using the day ground-truth. We tested both UAH and ZJU-trained GANs for performing the conversions. Alderley and Milford ones were discarded due to their poor conversion results in the Cityscapes domain. Fig. 7 shows some qualitative results for the UAH case and Fig. 8 for the ZJU case. In all images, the top row displays the input image with the standard day segmentation model, in second row the night image with the standard day model, and in third row the night image with the night-trained model. This way,

TABLE II
RESULTS OF OUR PROPOSALS IN CITYSCAPES VAL SET.

Test Domain	Model Train Domain	UAH-GAN		ZJU-GAN	
		IoU	Acc	IoU	Acc
Day	Day	65.8%	94.1%	65.8%	94.1%
Night	Day	10.8%	64.1%	14.0%	66.2%
Night	Night	53.6%	90.1%	55.4%	90.9%
Night2Day	Day	33.7%	82.2%	45.0%	87.3%

the benefits of using a night-trained model versus using a standard model in the night domain can be clearly seen.

Additionally, Table II shows the main quantitative results in Intersection over Union (IoU) and Pixel Accuracy (Acc), widely used metrics in semantic segmentation. It can be seen that in both UAH and ZJU cases, the standard segmentation model works very poorly in the night domain, while the night-trained model reduces the accuracy gap between Day and Night from over 52% to roughly ~10% in IoU loss, and from over 28% to roughly ~3% in pixel accuracy loss. In the last table row, we also added results for the night image reconstructed back to day, performing segmentation with the standard day model. In this case, the day-night gap is significantly reduced compared to the performance of the day-model in night, but the accuracy is also significantly lower compared to the night-trained model.

V. CONCLUSIONS

In this paper, we analyzed the domain gap between Day and Night images for the task of semantic segmentation. We leverage novel techniques such as stylization GANs to propose two methods for bridging the gap between the two domains. On one hand, we successfully trained a segmentation model to perform inference directly from night images, obtaining a significant performance boost compared to directly running a standard day-trained model at night. On the other hand, we perform night to day conversion during night inference to transform the input data into a more suitable domain for models that were already trained in daylight imagery. In both cases, we use a novel GAN that we train in 4 datasets, 2 collected by our instrumented vehicles and 2 collected from the internet. Our qualitative and quantitative experiments demonstrate that both proposed approaches considerably reduce the domain gap, allowing state-of-the-art semantic segmentation methods like ERFNet to work reliably also at night.

REFERENCES

[1] E. Romera, L. M. Bergasa, and R. Arroyo, "Can we unify monocular detectors for autonomous driving by using the pixel-wise semantic segmentation of cnns?" *arXiv preprint arXiv:1607.00971*, 2016.
[2] M. J. Milford and G. F. Wyeth, "Seqslam: Visual route-based navigation for sunny summer days and stormy winter nights," in *IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2012, pp. 1643–1649.
[3] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," *arXiv preprint arXiv:1612.01105*, 2016.
[4] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam, "Rethinking atrous convolution for semantic image segmentation," *arXiv preprint arXiv:1706.05587*, 2017.
[5] E. Romera, J. M. Alvarez, L. M. Bergasa, and R. Arroyo, "Efficient convnet for real-time semantic segmentation," in *IEEE Intelligent Vehicles Symp. (IV)*, 2017, pp. 1789–1794.

(a) Input   (b) Semantic Segmentation   (c) Input   (d) Semantic Segmentation
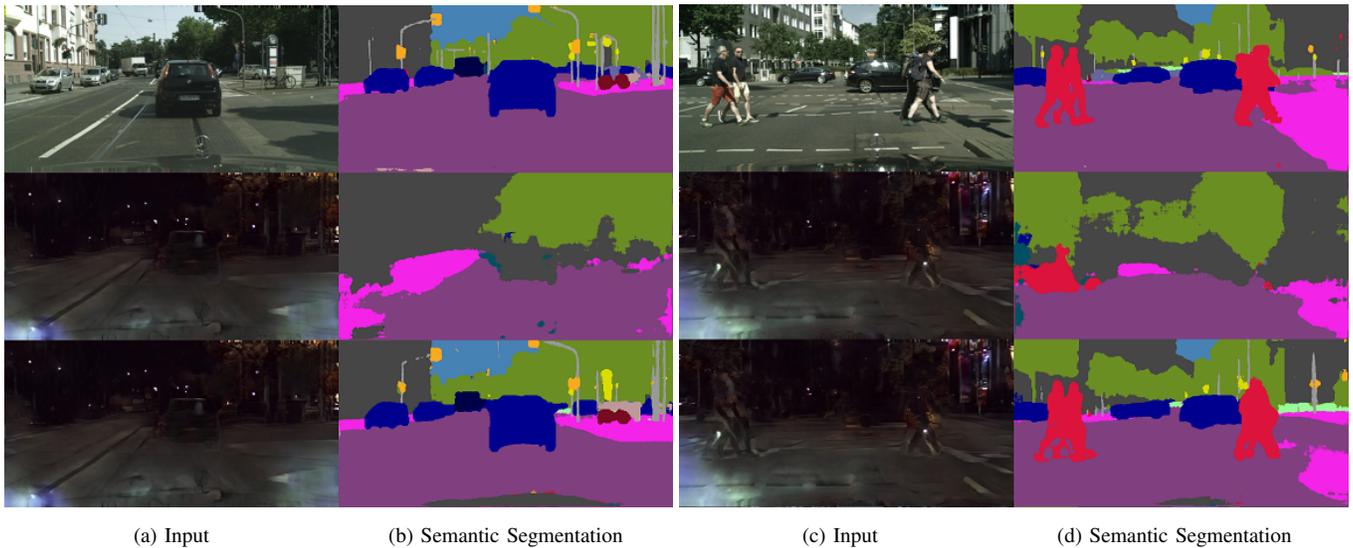
Fig. 7. Cityscapes Validation set examples at day (top row) and at night converted by the UAH-trained GAN (mid and bottom rows). In both examples, it can be seen how the day-model semantic segmentation performance at night images (mid row) is significantly lower than for the night-model (bottom).

(a) Input   (b) Semantic Segmentation   (c) Input   (d) Semantic Segmentation
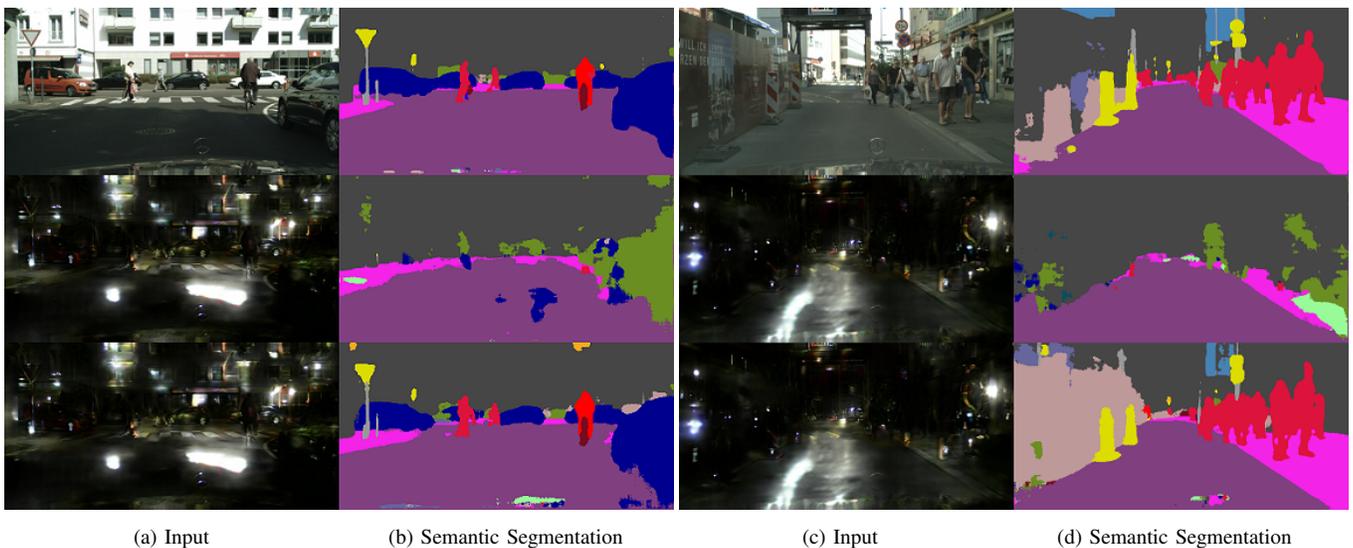
Fig. 8. Cityscapes Validation set examples at day (top row) and at night converted by the ZJU-trained GAN (mid and bottom rows). In both examples, it can be seen how the day-model semantic segmentation performance at night images (mid row) is significantly lower than for the night-model (bottom).

[6] E. Romera, J. M. Alvarez, L. M. Bergasa, and R. Arroyo, "Erfnet: Efficient residual factorized convnet for real-time semantic segmentation," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 1, pp. 263–272, 2018.

[7] E. Romera, L. M. Bergasa, J. M. Alvarez, and M. Trivedi, "Train here, deploy there: Robust segmentation in unseen domains," in *2018 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2018, pp. 1828–1833.

[8] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset for semantic urban scene understanding," in *IEEE Conf. on Computer Vision and Pattern Recog. (CVPR)*, 2016, pp. 3213–3223.

[9] G. Ros and J. M. Alvarez, "Unsupervised image transformation for outdoor semantic labelling," in *IEEE Intelligent Vehicles Symposium (IV)*, 2015, pp. 537–542.

[10] S. Sankaranarayanan, Y. Balaji, A. Jain, S. N. Lim, and R. Chellappa, "Learning from synthetic data: Addressing domain shift for semantic segmentation," *arXiv preprint arXiv:1711.06969*, 2017.

[11] L. Deng, M. Yang, H. Li, T. Li, B. Hu, and C. Wang, "Restricted deformable convolution based road scene semantic segmentation using surround view cameras," *arXiv preprint arXiv:1801.00708*, 2018.

[12] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley,

S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in neural information processing systems*, 2014, pp. 2672–2680.

[13] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, "Improved techniques for training gans," in *Advances in Neural Information Processing Systems*, 2016, pp. 2234–2242.

[14] P. Isola, J.-Y. Zhu, T. Zhou, and A. Efros, "Image-to-image translation with conditional adversarial networks," *arXiv preprint*, 2017.

[15] J.-Y. Zhu, T. Park, P. Isola, and A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," *arXiv preprint*, 2017.

[16] M.-Y. Liu, T. Breuel, and J. Kautz, "Unsupervised image-to-image translation networks," in *Advances in Neural Information Processing Systems*, 2017, pp. 700–708.

[17] M. Milford, C. Shen, S. Lowry, N. Suenderhauf, S. Shirazi, G. Lin, F. Liu, E. Pepperell, C. Lerma, B. Upcroft, *et al.*, "Sequence searching with deep-learnt depth for condition- and viewpoint-invariant route-based place recognition," in *IEEE Conf. on Computer Vision and Pattern Recog. (CVPR) Workshops*, 2015, pp. 18–25.

[18] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," *arXiv preprint arXiv:1804.02767*, 2018.