Enhancing Photorealism in CARLA Autonomous Driving Simulator

Stefanos Pasios, Nikos Nikolaidis
School of Informatics, Aristotle University of Thessaloniki
Thessaloniki, Greece
{pstefanos,nnik}@csd.auth.gr

***Abstract*—Simulators are essential in autonomous systems research, providing a controlled test environment for self-driving vehicles, autonomous robots, and unmanned aerial vehicles. Despite recent significant improvements in simulation realism, the noticeable gap between the simulation and the real-world complexities persists, hindering the direct applicability of algorithms trained on simulated data to real-world scenarios. In this work, we extend the applicability of CARLA2Real[1]—a previously developed publicly available tool for enhancing CARLA's visual realism—to the latest Unreal Engine 5 version of the simulator, which already features improved rendering capabilities. Our approach utilizes a state-of-the-art image-to-image translation method to generate photorealism-enhanced autonomous driving-related visual data that target the characteristics of the real-world datasets, Cityscapes and KITTI. Based on this, we generated synthetic datasets from both the simulator and the photorealism enhancement model outputs, including their corresponding ground truth annotations for semantic segmentation. Subsequently, by employing the photorealism-improved synthetic data as training data, we conducted experiments to assess how the proposed approach affected the accuracy of a semantic segmentation approach. The findings illustrated that, although Unreal Engine 5 improves the baseline realism of CARLA, a sim2real gap still persists in various aspects of the scenes. However, our method significantly reduces this gap, leading to improved segmentation performance on real-world data.**

***Index Terms*—Sim2real gap, CARLA, Semantic Segmentation, Autonomous Driving, Image-to-image Translation, Photorealism Enhancement**

I. Introduction

SIMULATORS are of utmost importance in research for autonomous driving and other related areas, such as robotics, offering a cost-effective approach for rapid prototyping. Additionally, they eliminate real-world risks, reduce physical test time, and enable the generation of large-scale synthetic datasets that are essential for overcoming the real-world data sparsity and thus enhancing the robustness of the models that are to be trained on these data.

Although there has been significant progress in simulation and game engine technology, the gap that exists between the simulated and real-world data, usually referred to as the sim2real gap, remains a critical challenge. This gap is present in aspects such as vehicle dynamics, physics, and the general realism of the synthetic scenes, often resulting from the utilization of low-cost assets and old rendering techniques. NVIDIA [1] classifies the sim2real gap into two main categories: the appearance gap, namely the difference between synthetic and real-world images at the pixel level, and the content gap, related to differences in objects position, distribution, and the general semantic layout structure.

To reduce this gap, a significant amount of research has been conducted on approaches that employ deep learning-based image translation methods such as Generative Adversarial Networks (GANs) [2] to translate synthetic images towards real-world ones. These approaches are often subject to visual artifacts that limit the applicability of these data in models that are to be applied in the real world.

In this work, we employ a state-of-the-art (SotA) image-to-image translation method [3] that utilizes Geometry Buffers (G-Buffers) extracted from the CARLA simulator [4] to enhance the photorealism of the simulated data towards the characteristics of real-world datasets such as Cityscapes [5] and KITTI [6]. Based on this, we updated the data generation functionality of a pre-existing photorealism enhancement tool for the Unreal Engine 4 (UE4) version of the CARLA simulator in the latest Unreal Engine 5 (UE5) version, which introduces novel computer graphics techniques that aim to mitigate this visual gap. To evaluate the impact of the photorealism enhancement method in this visually upgraded version of the simulator, a semantic segmentation dataset was generated. This dataset was utilized to assess the effect on the accuracy of a semantic segmentation model when employing the photorealism-enhanced data compared to the original UE5 rendered images. Our contributions can be summarized as follows:

1) To the best of our knowledge, we are the first to investigate the sim2real gap that exists in the latest graphically improved version of the CARLA simulator, which is based on UE5, and illustrate the importance of a photorealism enhancement approach on an autonomous driving-related task.
2) We propose a dataset generation pipeline for the latest UE5 version of the CARLA simulator that is capable of producing temporally and spatially consistent photorealistic visual data.

The rest of this work is structured as follows: in Section II, we provide an overview of the related works that focus on mitigating the sim2real gap that exists in the previous UE4 versions of the CARLA simulator. Next in Section III, we describe the methodologies and tools that were utilized for this work. In Section IV, the generated dataset is introduced. Section V presents and discusses our experimentation with an autonomous driving-related model for semantic segmentation. Finally, Section VI concludes the paper.

[1]https://github.com/stefanos50/CARLA2Real

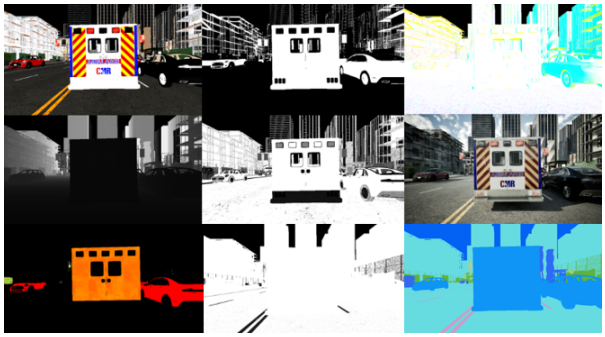

Fig. 1: G-Buffers generated from the UE5 [7] deferred rendering pipeline. **1st Row:** Base Color, Metallic, World Normals. **2nd Row:** Scene Depth, Roughness, Scene Color. **3rd Row:** Subsurface Colors, Specular, Semantic Segmentation (Stenctil).

II. Related Work

Only a few works have explored the effectiveness of synthetic-to-real image-to-image translation methods in improving the performance of deep learning models on a variety of autonomous driving-related tasks using visual data from the CARLA simulator. In [8], Pix2Pix [9] was employed to synthesize photorealistic frames as described by the semantic label maps provided by the CARLA simulator. The translated images showed marginal or no improvement compared to the raw rendered frames when utilized by a Deep Deterministic Policy Gradient Reinforcement Learning Algorithm [10] since Pix2Pix was found to be sensitive to a significant number of visual artifacts.

To evaluate the similarity of features extracted from translated and synthetic datasets with features extracted from a real-world dataset, the authors of [11] carried out a comparison between features derived from GTA-V, CARLA, and the Virtual KITTI [12] datasets and features extracted from Cityscapes [5]. The study employed VGG-19 [13] and ResNet-152 [14] and used PCA to calculate the distance between the centroids of the formed feature clusters. The results illustrated that the features derived from the photorealism-enhanced data were closer to ones extracted from Cityscapes. Another recent work [15] employed Dual Contrastive Learning Adversarial Generative Network (DCLGAN) [16] for a Lane Keeping Assist System in CARLA. DCLGAN translated frames generated by CARLA to mimic the characteristics of real images captured from the Korea Intelligent Automotive Parts Promotion Institute (KIAPI). The translated frames achieved a lower Frechet Inception Distance (FID) and improved the lane detection accuracy of ENet-SAD [17] when applied to the real-world images. Finally, the ability of the system for lane restoration was proved to improve when employing the translated images.

III. CARLA Enhancer

In this section, we discuss the methodologies and tools involved in our research, which focuses on the employment of a SotA image-to-image translation method with the ultimate goal of reducing the sim2real gap that exists in the latest UE5 version of the CARLA simulator.

A. Enhancing Photorealism Enhancement (EPE)

Enhancing Photorealism Enhancement (EPE) [3] is a SotA image-to-image translation method that enhances the photorealism of synthetic data. The novelty of the approach is based on the utilization of additional information (G-Buffers, Figure 1) generated by game engines to further describe the lighting, geometry, and materials of the virtual scene. The G-Buffers, along with the semantic segmentation of the scene, are employed by a G-Buffer encoder, which learns to encode and treat in a different way each unique object described in the semantic segmentation label map. Additionally, to mitigate the distribution difference that exists between the objects depicted in real-world and synthetic images, EPE follows a patch-matching approach that matches regions of the images that contain similar objects. This approach prevents the discriminator from learning to classify whether a generated image is real or fake based on that distribution difference, thus reducing the probability of visual artifacts.

B. CARLA Simulator

CARLA [4] is an open-source autonomous driving simulator, initially developed on UE4. The simulator bridges the gap between simulation and machine learning tools, namely PyTorch and TensorFlow, by providing a Python API for manipulating the environment from a Python client.

CARLA offers realistic physics, several drivable vehicles, AI pedestrians, city configurations, and weather presets and supports sensors such as Lidar, IMU, radar, and cameras and is thus well-suited for deploying and testing perception algorithms in autonomous driving research. The latest version, 0.10.0, of the simulator is currently under development and is released as a work in progress. This version utilizes the SotA technologies Lumen and Ninite of UE5, thus significantly improving the visual appearance and details of the environment. While CARLA 0.10.0 includes a variety of upgraded assets from the previous UE4 versions, it is still limited to two environments (Town10 and Mine) and a single weather and time configuration. A subset of the functionalities, including the ability to extract the G-Buffers that are generated during the rendering of the camera sensor, is currently not supported.

C. CARLA2Real

CARLA2Real [18] is a tool that focuses on reducing the sim2real gap that exists in autonomous driving simulators. In detail, it enhances the photorealism of CARLA (UE4) by employing the EPE method targeting the characteristics of real-world datasets, including Cityscapes and KITTI. Additionally, CARLA2Real allows the easy configuration of various functionalities such as autonomous driving tasks and data generation processes through a YAML parameterization file. It enables the generation of a wide range of datasets with various annotations and modalities, including semantic

Fig. 2: Translation results towards the KITTI characteristics, demonstrating temporal and spatial consistency: multiple views of the same vehicle (1st row), time sequenced frames (2nd row).

segmentation, depth maps, and object detection annotations, and provides a framework for simulating complex driving scenarios in (near) real-time by employing multithreading and SotA compilers such as ONNX Runtime [19] and TensorRT [20].

Considering that the latest UE5 version of the CARLA simulator currently does not support the extraction of the required G-Buffers from the Python API, which is required from CARLA2Real, in this work we extend the data generation functionality of the tool to the latest version of the simulator. To achieve this, the G-Buffers were exposed directly from the engine using the HighResScreenshot functionality, which is not part of the Python API. The proposed data extraction pipeline maintains a synchronization between the frames and G-buffers exported from HighResScreenshot and the rest of the CARLA camera sensors, including the semantic segmentation sensor, which are accessible through the Python API.

D. Translation Results

Fig. 3: Results of CARLA's UE5 version of Town10HD (1st column) translations towards Cityscapes (2nd column), and KITTI (3rd column).

The results of the pre-trained photorealism enhancement models targeting the characteristics of Cityscapes and KITTI show (Figure 3) that the models can effectively improve the photorealism of the UE5 version of the Town10HD environment. The models primarily enhance the objects materials, making them more realistic and glossy. The sky clouds are also adjusted to match those in the real dataset, with color distributions aligned with the specific dataset, such as brighter skies for KITTI. The general lighting of the scene is also significantly improved. The latter indicates that while technologies such as Lumen can contribute to improving the lighting of synthetic scenes, they still struggle to capture the complexities that exist in the real world. Additionally, since CARLA 0.10.0 appears to have an issue with pixelated images, potentially due to anti-aliasing, the enhancement model remedies this issue, further improving the realism of the scene. Finally, as illustrated in Figure 2, the utilization of additional information of the game engine (G-Buffers) results in translations that maintain consistency over time and across viewpoint variations. The overall colors, structure, and general appearance of objects remain unchanged across different camera angles and frame sequences.

IV. Experimental Dataset

In order to evaluate the impact of the EPE approach on the reduction of the sim2real gap that exists in the latest CARLA simulator version output, a number of experiments were conducted, targeting an autonomous driving-related model. To achieve this, a dataset was exported by employing the updated CARLA2Real dataset generation functionality. The experimental dataset was produced directly through the CARLA simulator in synchronous mode by preserving 1 in 60 frames of the simulation. To avoid any potential bias during the experiments, the original CARLA and the enhanced data shared the same informational content and, thus, identical semantic segmentation annotations. Considering that CARLA 0.10.0 is a work in progress and the diversity is significantly limited, a training set of just 500 images was extracted, including the rendered frames, the G-Buffers, and the semantic segmentation maps from the single integrated CARLA Town10HD environment and the fixed daylight setting (currently there is no support for weather and time modifications). The extracted dataset was processed with both KITTI and Cityscapes EPE-pre-trained models to derive enhanced data targeting the characteristics of KITTI and Cityscapes, respectively. These two subsets included 500 frames each.

V. Semantic Segmentation

Semantic segmentation is a computer vision task that is closely related to autonomous driving as it provides a pixel-wise understanding of the environment for the autonomous vehicle. This detailed level of scene understanding enables

	KITTI Validation Set			Cityscapes Validation Set		
Classes	CARLA	Enh. CARLA (KITTI)	KITTI	CARLA	Enh. CARLA (Cityscapes)	Cityscapes
All	0.2678	**0.3522**	0.6376	0.3139	0.3468	0.4887
Background	0.2158	**0.2988**	0.5799	0.2286	0.2244	0.2938
Road	0.2645	**0.4511**	0.8730	0.1318	**0.1741**	0.4896
Sidewalk	0.0046	0.0123	0.2162	0.0910	**0.1401**	0.1822
Building	0.1061	**0.1802**	0.2988	0.4056	**0.4994**	0.6289
Vegetation	0.5038	0.5084	0.8293	0.5696	0.5371	0.6814
Sky	0.4582	**0.7102**	0.7976	0.5001	**0.5897**	0.5607
Car	0.3043	0.2821	0.8122	0.2875	0.26	0.5914

Table I: Evaluation of DeepLabV3 models trained on (a) CARLA original, CARLA enhanced (targeting KITTI), and KITTI, tested on the KITTI validation set (columns 1–3); and (b) CARLA original, CARLA enhanced (targeting Cityscapes), and Cityscapes, tested on the Cityscapes validation set (columns 4–6). Higher IoU values indicate better performance. Bold values indicate the most significant increase in accuracy after applying the photorealism enhancement approach.

the vehicle to avoid collisions and react to objects in the environment, such as traffic lights, signs, and pedestrians. This section investigates the performance of the visually improved CARLA version 0.10.0 and the influence on the accuracy of a semantic segmentation model after applying the photorealism enhancement method.

A. Experimental Setup

To conduct the semantic segmentation experiments, we employed Google's DeeplabV3 [21] using a ResNet-50 [14] backbone architecture, since it is integrated into the PyTorch framework [22]. Standard DeeplabV3 preprocessing procedures were followed, involving image loading within the [0, 1] range, followed by normalization using mean values of [0.485, 0.456, 0.406] and standard deviations of [0.229, 0.224, 0.225]. The training lasted 15 epochs, utilizing the AdamW optimizer with a learning rate of 0.00005, a cross-entropy loss function, and a batch size of eight. The training phase included two parts, each utilizing three different types of data. The first part deals with Cityscapes and involves the training of DeepLabV3 models on a) the original CARLA frames, b) enhanced CARLA frames targeting the Cityscapes characteristics, and c) Cityscapes frames. The second part deals with KITTI. In this part, DeepLabV3 was trained on a) the original CARLA frames, b) enhanced CARLA targeting the KITTI characteristics, and c) KITTI. Only a subset of seven fundamental semantic classes (Table I) was preserved to provide compatibility between the different annotation schemes of these datasets (KITTI, Cityscapes, and CARLA). The remainning classes were grouped in a background class.

For the evaluation phase, the validation set of Cityscapes was used for the first part, while for the second part, the remaining 50 real-world frames from the KITTI semantic segmentation benchmark, which were not utilized during training, were employed. The evaluation was performed using the Intersection over Union (IoU) metric, computed both per class and for all classes combined, after iterating through the respective validation sets with a batch size of one.

B. Results and Discussion

From the results presented in Table I, it is evident that the models trained on the enhanced CARLA frames targeting Cityscapes or KITTI demonstrated a significant increase in performance compared to the models trained on the original CARLA data when both are evaluated on the real-world Cityscapes (columns CARLA, Enh. CARLA-Cityscapes) and KITTI (columns CARLA, Enh. CARLA-KITTI) validation sets. However, the significantly improved quality introduced in the UE5 version of CARLA, in combination with the photorealism improvements introduced by CARLA2Real, still cannot reach the performance achieved when training solely on the real-world datasets (columns Cityscapes and KITTI in the same table).

To further investigate these observations, Table I also provides the per-class accuracy of the seven classes that were preserved to enable compatibility. It is apparent that the observations made above for the overall (all classes) accuracy hold also for most of the available classes. Indeed, when training on enhanced simulation data targeting the characteristics of Cityscapes and KITTI, per-class accuracy increases when evaluating on the validation sets of these real-world datasets. This again indicates that the enhanced data can indeed reduce the sim2real appearance gap that exists even on the more sophisticated rendering pipeline of UE5 and lead to higher performance when utilized by semantic segmentation models that are to be deployed in the real world.

Particularly, significant increases in accuracy were observed in the road, sidewalk, building, and sky classes. Interestingly enough, roads and sidewalks were also noted in [3] as scene elements that EPE manages to improve significantly. This finding demonstrates that CARLA2Real could be particularly effective in lane detection tasks, which are typically used in autonomous vehicles. The significant improvement in the sky class was also an expected finding since, as depicted in Figure 3, for the KITTI dataset, the model removes all clouds and enhances the light intensity of the sky similarly to the real-world dataset. The accuracy increase related to the road, sidewalk, and sky classes is also obvious in Figure 4, which depicts the semantic segmentation results on a sample frame from the KITTI validation set generated by DeepLabV3 trained on CARLA and enhanced CARLA data. The car class was the single class that did not seem to benefit from the approach. The notable quality increase in CARLA

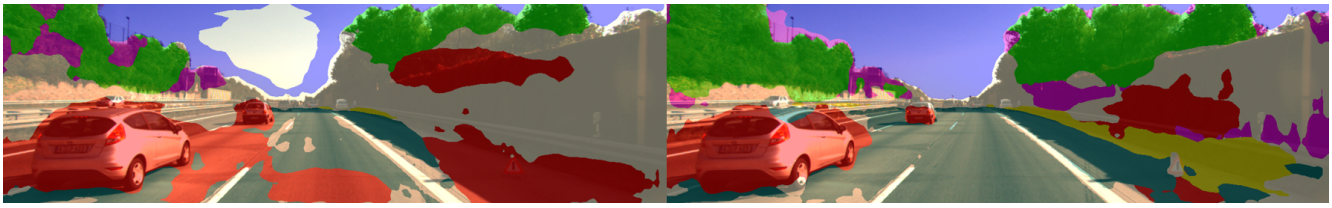

Fig. 4: Semantic segmentation results obtained from a CARLA-trained model (left) and an Enhanced CARLA-trained model (right) on a KITTI validation frame.

0.10.0 of these particular assets (cars) is a possible explanation for this fact. In general, we observed that the EPE-based CARLA2Real tool provided more visually appealing and photorealistic translation results on UE5-powered CARLA when targeting KITTI, compared to Cityscapes, with results that visually surpass or come very close to SotA rendering technologies such as ray and path tracing. This fact is also supported by the segmentation accuracy results presented in Table I. Indeed, the model that utilized the enhanced dataset targeting KITTI achieved a more significant improvement with respect to the one that utilized the original CARLA frames, compared to the accuracy improvement observed in the Cityscapes experiments.

VI. Conclusions

In this paper, a robust photorealism enhancement approach was employed to enhance the photorealism of the latest UE5 version of the CARLA simulator with the goal of investigating and further closing the sim2real gap that exists in this quality-improved version of the simulator. To achieve this we employed CARLA2Real tool and extended the data generation functionality to the new version of the simulator in order to extract a semantic segmentation dataset with improved photorealism based on the characteristics of two real-world datasets, namely Cityscapes and KITTI. The results illustrated that a significant sim2real gap still persists, however the proposed approach is capable of improving the performance of a semantic segmentation model (DeepLabV3) when evaluated on the respective real-world validation sets despite the pre-existing visual improvements of UE5. As future work, we plan to upgrade and integrate the whole functionality of the CARLA2Real tool in the latest CARLA 0.10.0 version and perform experiments on additional autonomous driving-related computer vision tasks.

Acknowledgement

The work presented here has been partially supported by the RoboSAPIENS project funded by the European Union Horizon Europe programme under grant agreement number 101133807. This publication reflects the authors' views only. The European Commission is not responsible for any use that may be made of the information it contains.

References

[1] K. Gupta and N. Worker. (2022) Closing the Sim2Real gap with NVIDIA Isaac Sim and NVIDIA Isaac Replicator. https://developer.nvidia.com/blog/closing-the-sim2real-gap-with-nvidia-isaac-sim-and-nvidia-isaac-replicator.

[2] 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.

[3] S. R. Richter, H. A. AlHaija, and V. Koltun, "Enhancing photorealism enhancement," arXiv:2105.04619, 2021.

[4] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun, "CARLA: An open urban driving simulator," in *Proceedings of the 1st Annual Conference on Robot Learning*, 2017, pp. 1–16.

[5] 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 *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[6] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The KITTI dataset," *International Journal of Robotics Research (IJRR)*, 2013.

[7] Epic Games, "Unreal Engine," https://www.unrealengine.com/en-US, 2025, accessed: 2025-05-20.

[8] J. Ram, E. Bakker, and M. S. Lew, "Sim-to-real autonomous driving in CARLA using image translation and deep deterministic policy gradient," https://theses.liacs.nl/pdf/2021-2022-RamJ.pdf, Netherlands, 2022.

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

[10] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, "Continuous control with deep reinforcement learning," arXiv:1509.02971, 2019.

[11] N. Gadipudi, I. Elamvazuthi, M. Sanmugam, L. I. Izhar, T. Prasetyo, R. Jegadeeshwaran, and S. S. A. Ali, "Synthetic to real gap estimation of autonomous driving datasets using feature embedding," in *2022 IEEE 5th International Symposium in Robotics and Manufacturing Automation (ROMA)*, 2022, pp. 1–5.

[12] A. Gaidon, Q. Wang, Y. Cabon, and E. Vig, "Virtualworlds as proxy for multi-object tracking analysis," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 4340–4349.

[13] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *International Conference on Learning Representations*, 2015.

[14] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.

[15] J. Pahk, J. Shim, M. Baek, Y. Lim, and G. Choi, "Effects of Sim2Real image translation via DCLGAN on lane keeping assist system in CARLA simulator," *IEEE Access*, vol. 11, pp. 33 915–33 927, 2023.

[16] J. Han, M. Shoeiby, L. Petersson, and M. A. Armin, "Dual contrastive learning for unsupervised image-to-image translation," in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2021, pp. 746–755.

[17] Y. Hou, Z. Ma, C. Liu, and C. C. Loy, "Learning lightweight lane detection CNNs by self attention distillation," in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019, pp. 1013–1021.

[18] S. Pasios and N. Nikolaidis, "CARLA2Real: a tool for reducing the sim2real gap in CARLA simulator," *arXiv preprint arXiv:2410.18238*, 2024.

[19] O. R. developers. (2021) ONNX Runtime. https://onnxruntime.ai/.

[20] NVIDIA Corporation. (2016) NVIDIA TensorRT. https://developer.nvidia.com/tensorrt/.

[21] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam, "Rethinking atrous convolution for semantic image segmentation," arXiv:1706.05587, 2017.

[22] A. Paszke et al., "Pytorch: An imperative style, high-performance deep learning library," in *Advances in Neural Information Processing Systems 32*. Curran Associates, Inc., 2019, pp. 8024–8035.