Augmented Reality Meets Deep Learning for Car Instance Segmentation in Urban Scenes

Hassan Abu Alhaija[1,2]
hassan.abu_alhaija@tu-dresden.de
Siva Karthik Mustikovela[1,2]
siva_karthik.mustikovela@tu-dresden.de
Lars Mescheder[3]
lmescheder@tuebingen.mpg.de
Andreas Geiger[3,4]
andreas.geiger@tue.mpg.de
Carsten Rother[1,2]
carsten.rother@tu-dresden.de

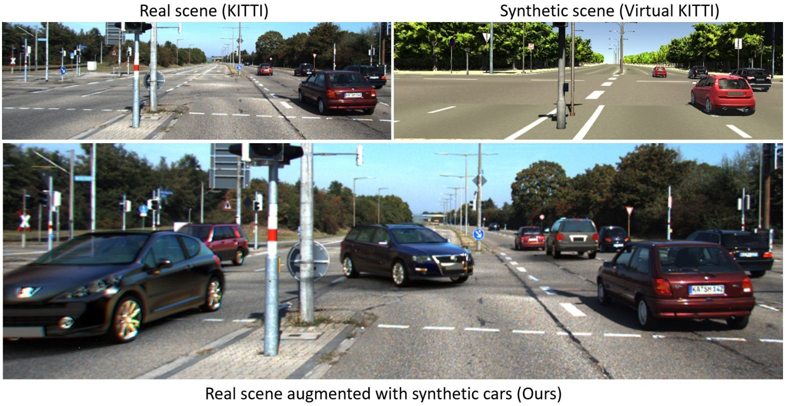
[1] Computer Vision Lab
TU Dresden, Germany
[2] Visual Learning Lab
Heidelberg University, Germany
[3] Autonomous Vision Group
MPI-IS Tübingen, Germany
[4] Computer Vision and Geometry Group
ETH Zürich, Switzerland

Abstract

The success of deep learning in computer vision is based on the availability of large annotated datasets. To lower the need for hand labeled images, virtually rendered 3D worlds have recently gained popularity. Unfortunately, creating realistic 3D content is challenging on its own and requires significant human effort. In this work, we propose an alternative paradigm which combines real and synthetic data for learning semantic instance segmentation models. Exploiting the fact that not all aspects of the scene are equally important for this task, we propose to augment real-world imagery with virtual objects of the target category. Capturing real-world images at large scale is easy and cheap, and directly provides real background appearances without the need for creating complex 3D models of the environment. We present an efficient procedure to augment these images with virtual objects. This allows us to create realistic composite images which exhibit both realistic background appearance as well as a large number of complex object arrangements. In contrast to modeling complete 3D environments, our data augmentation approach requires only a few user interactions in combination with 3D shapes of the target object category. We demonstrate the utility of the proposed approach for training a state-of-the-art high-capacity deep model for semantic instance segmentation. In particular, we consider the task of segmenting car instances on the KITTI dataset which we have annotated with pixel-accurate ground truth. Our experiments demonstrate that models trained on augmented imagery generalize better than those trained on synthetic data or models trained on limited amounts of annotated real data.

1 Introduction

In recent years, deep learning has revolutionized the field of computer vision. Many tasks that seemed elusive in the past can now be solved efficiently and with high accuracy using deep neural networks, sometimes even exceeding human performance [27].



Real scene (KITTI)

Synthetic scene (Virtual KITTI)

Real scene augmented with synthetic cars (Ours)

Figure 1: Obtaining synthetic training data usually requires building large virtual worlds (top right)[8]. We propose a new way to extend datasets by augmenting real training images (top left) with realistically rendered cars (bottom), keeping the resulting images close to real while expanding the diversity of training data.

However, it is well-known that training high capacity models such as deep neural networks requires huge amounts of labeled training data. This is particularly problematic for tasks where annotating even a single image requires significant human effort, *e.g.*, for semantic or instance segmentation. A common strategy to circumvent the need for human labels is to train neural networks on synthetic data obtained from a 3D renderer for which ground truth labels can be automatically obtained [8, 11, 16, 20, 21, 23, 29, 31]. While photo-realistic rendering engines exist [13], the level of realism is often lacking as fine details in the 3D world, *e.g.*, leaves of trees can only be modeled approximately.

In this paper, we demonstrate that state-of-the-art photo-realistic rendering can be utilized to augment real-world images and obtain virtually unlimited amounts of training data for a specific task such as semantic instance segmentation. Towards this goal, we consider real images with additional side information, such as camera calibration and environment maps, and augment these images using novel object instances. This allows us to keep the full realism of the background while being able to generate arbitrary amounts of object configurations.

Figure 1 shows a real image before and after augmentation. While our rendered objects rival the realism of the input data, they provide the variations (*e.g.*, pose, shape, appearance) needed for training deep neural networks for instance aware semantic segmentation of cars. By doing so, we are able to considerably improve the accuracy of a state-of-the-art deep neural network trained on real data.

While the level of realism is an important factor when synthesizing new data, there are two other important aspects to consider - data diversity and human labor. Manually assigning a class or instance label to every pixel in an image is possible but tedious, requiring up to one hour per image [4]. Thus, existing real-world datasets are limited to a few hundred [2] or thousand [4] annotated examples, thereby severely limiting the diversity of the data. In contrast, the creation of virtual 3D environments allows for arbitrary variations of the data and virtually infinite number of training samples. However, the creation of 3D content requires professional artists and the most realistic 3D models (designed for modern computer games or movies) are not publicly available due to the enormous effort involved in creating them. While Richter *et al.* [20] have recently demonstrated how content from commercial games can be accessed through manipulating low-level GPU instructions, legal problems are likely to arise and often the full flexibility of the data generation process is no longer given.

In this work, we demonstrate that the creation of an augmented dataset which combines real with synthetic data requires only moderate human effort while yielding the variety of data necessary for improving the accuracy of a state-of-the-art instance segmentation system [5]. In particular, we show that a model trained using our augmented dataset generalizes better than models trained purely on synthetic data as well as models which use a smaller number of manually annotated real images. Since our data augmentation approach requires only minimal manual effort, we believe that it constitutes an important milestone towards the ultimate task of creating virtually infinite, diverse and realistic datasets with ground truth annotations. In summary, our contributions are as follows:

- We propose an efficient solution for augmenting real images with photo-realistic synthetic object instances which can be arranged in a flexible manner.
- We provide an in-depth analysis of the importance of various factors of the data augmentation process, including the number augmentations per real image, the realism of the background and the realism of the foreground regions.
- We find that models trained on augmented data generalize better than models trained on purely synthetic data or small amounts of labeled real data.
- For conducting the experiments in this paper, we introduce two newly labeled instance segmentation datasets, named KITTI-15 and KITTI-360, with a total of 400 images.

2 Related Work

Due to the scarcity of real-world data for training deep neural networks, several researchers have proposed to use synthetic data created with the help of a 3D rendering engine. Indeed, it was shown [16, 20, 23] that deep neural networks can achieve state-of-the-art results when trained on synthetic data and that the accuracy can be further improved by fine tuning on real data [20]. Moreover, it was shown that the realism of synthetic data is important to obtain good performance [16].

Making use of this observation, several synthetic datasets have been released which we will briefly review in the following. Hattori *et al.* [12] present a scene-specific pedestrian detector using only synthetic data. Varol *et al.* [29] present a synthetic dataset of human bodies and use it for human depth estimation and part segmentation from RGB images. In a similar effort, Chen *et al.* [3] use synthetic data for 3D human pose estimation. In [6], synthetic videos are used for human action recognition with deep networks. Zhang *et al.* [32] present a synthetic dataset for indoor scene understanding. Similarly, Handa *et al.* [11] use synthetic data to train a depth-based pixelwise semantic segmentation method. In [31], a synthetic dataset for stereo vision is presented which has been obtained from the UNREAL rendering engine. Zhu *et al.* [33] present the AI2-THOR framework, a 3D environment and physics engine which they leverage to train an actor-critic model using deep reinforcement learning. Peng *et al.* [17] investigate how missing low-level cues in 3D CAD models affect the performance of deep CNNs trained on such models. Stark *et al.* [25] use 3D CAD models for learning a multi-view object class detector.

In the context of autonomous driving, the SYNTHIA dataset [21] contains a collection of diverse urban scenes and dense class annotations. In [8], Gaidon *et al.* introduce a synthetic

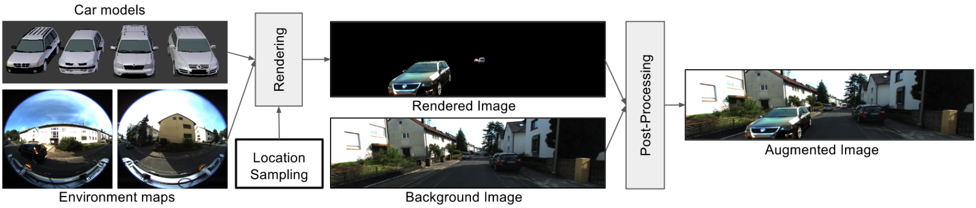


Figure 2: Overview of our augmentation pipeline. Given a set of 3D car models, locations and environment maps, we render high quality cars and overlay them on top of real images. The final post-processing step ensures better visual matching between the rendered and real parts of the resulting image.

video dataset (V-KITTI) which was obtained from the KITTI dataset [9] alongside with dense class annotations, optical flow and depth. Su *et al.* [26] use a dataset of rendered 3D models on random real images for training a CNN on viewpoint estimation. While all aforementioned methods require labor intensive 3D models of the environment, we focus on exploiting the synergies of real and synthetic data using augmented reality. In contrast to purely synthetic datasets, we obtain a large variety of realistic data in an efficient manner. Furthermore, as evidenced by our experiments, combining real and synthetic data within the same image results in models with better generalization performance.

While most works use either real or synthetic data, only few papers consider the problem of training deep models with mixed reality data. Rozantsev *et al.* [22] estimate the parameters of a rendering pipeline from a small set of real images for training an object detector. Gupta *et al.* [10] use synthetic data for text detection in images. Pishchulin *et al.* [19] use synthetic human bodies rendered on random backgrounds for training a pedestrian detector. Dosovitskiy *et al.* [7] render flying chairs on top of random Flickr backgrounds to train a deep neural network for optical flow. Unlike existing mixed reality approaches, which are either simplistic, consider single objects or augment objects in front of random backgrounds, our goal is to create high fidelity augmentations of complex multi-object scenes at high resolution. In particular, our approach takes the geometric layout of the scene, environment maps as well as artifacts stemming from the image capturing device into account. We experimentally evaluate which of these factors are important for training good models.

3 Data Augmentation Pipeline

In this section, we describe our approach to data augmentation through photo-realistic rendering of 3D models on top of real scenes. To achieve this, three essential components are required: (i) detailed high quality 3D models of cars, (ii) a set of 3D locations and poses used to place the car models in the scene, and (iii) the environment map of the scene that can be used to produce realistic reflections and lighting on the models that matches the scene.

We use 28 high quality 3D car models covering 6 categories (SUV, sedan, hatchback, station wagon, mini-van and van) obtained from online model repositories[1]. The car color is chosen randomly during rendering to increase the variety in the data. To achieve high quality augmentation, it is essential to correctly place virtual objects in the scene, matching the distribution of poses and occlusion in the real data. Knowing the intrinsic parameters of the capturing camera and its exact pose, it is possible to estimate the ground plane in

[1] http://www.dmi-3d.net

the scene. This reduces the problem of sampling the pose from 6D to 3D, namely the 2D position on the ground plane and one rotation angle around the model's vertical axis.

We explore two methods to estimate a good set of model poses. Our first method leverages the homography between the ground plane and the image plane, transforming the perspective image into a birdseye view of the scene. Based on this birdseye view, we used in-house annotators to mark possible car trajectories and sample car locations and orientations using those paths. Our second, more automatic method, uses the algorithm proposed by Teichmann *et al.* [28] which segments the image into road and non-road areas with high accuracy. We back-project those road pixels and compute their location on the ground plane to obtain possible car locations, using a random rotation around the vertical axis of the vehicle. While the latter strategy is simpler, it can lead to visually less realistic augmentations. In addition, we empirically found it to perform slightly worse than the former strategy as described in Sec. 4. We use manual labeling in all our experiments, unless stated otherwise.

We leverage the 360 degree panoramas of the environment from the dataset of [30] as realistic environment map proxies for realistic rendering of cars in street scenes. Using the 3D models, locations and environment maps, we render cars using the Cycle renderer implemented in Blender [1]. Figure 2 illustrates our rendering approach. However, the renderings obtained from Blender lack typical artifacts of the image formation process such as motion blur, lens blur, chromatic aberrations, etc. To better match the image statistics of the background, we thus design a post-processing workflow in Blender's compositing editor which applies a sequence of 2D effects and transformations to simulate those effects, resulting in renderings that are more visually similar to the background. More specifically, we apply color shifts to simulate chromatic aberrations in the camera lens as well as depth-blur to match the camera depth-of-field. Finally, we use several color curve and gamma transformations to better match the color statistics and contrast of the real data. The parameters of these operations have been estimated empirically and some results are shown in Figure 1.

4 Evaluation

In this section, we study the performance of our data augmentation method on the challenging task of instance segmentation. Using different setups of our augmentation method, we investigate how the quality and quantity of augmented data affects the performance of a state-of-the-art instance segmentation model. We compare our results to training on real and fully synthetic data, as well as a combination of the two (*i.e.*, training on synthetic data and fine-tuning on real data). In particular, we explore how the number of augmentations improves the quality of the learned models and how it compares to training on purely synthetic data. We also experiment with different aspects of realism such as environment maps, photo-realistic rendering and car placement.

4.1 Evaluation Protocol

KITTI-360 For our experiments, we created a new dataset which contains 200 images from the dataset presented in [30]. We labeled all car instances at pixel level using our inhouse annotators to create high quality semantic instance segmentation ground truth. KITTI-360 is unique compared to KITTI [14] or CityScapes [4] in that each frame comes with two 180° images taken by two fish-eye cameras on top of the recording platform. Using an equirectangular projection, the two images are warped and combined to create a full 360° omni-directional image that we use as an environment map during the rendering process.

Environment maps are key to creating photo-realistic augmented images and are used frequently in Virtual Reality and Cinematic special effects applications. The set of 200 images form the basis for augmentation in all our experiments, *i.e.*, we reuse each image n times with differently rendered car configurations to obtain an n-fold augmented dataset.

KITTI-15 To demonstrate the advantage of data augmentation for training robust models, we create a new benchmark dataset different from the training set using the popular KITTI 2015 dataset [15]. More specifically, we annotated the 200 images of the KITTI 2015 dataset with pixel-accurate semantic instance labels using our in-house annotators. While the statistics of the KITTI 2015 dataset are similar to the KITTI-360 dataset, it has been recorded in a different year and at a different location/suburb. Thus it allows us to assess the generalization performance of instance segmentation methods trained on the KITTI-360 and Virtual KITTI dataset.

VKITTI To compare our augmentation method to fully synthetic data, we use the Virtual KITTI dataset [8] which has been designed as a virtual proxy for the KITTI 2015 dataset. Thus, the statistics of Virtual KITTI (*e.g.*, semantic class distribution, car poses and environment types) closely resembles those of KITTI-15 which we use as a test bed for evaluation. The dataset comprises $\sim$12,000 images divided into 5 sequences with 6 different weather and lighting conditions for each sequence.

Evaluation We train the state-of-the-art Multi-task Network Cascade (MNC) [5] for instance-aware semantic segmentation. In particular, we focus on the task of car instance segmentation in outdoor driving scenes. We initialize the model using the VGG weights [24] trained on ImageNet then train it for car instance segmentation using variants of real, augmented or virtual training data. For each variant, we train the model for 30K iterations and average the best performing 5 snapshots on the KITTI-15 test set using the standard average precision metric. We report this metric using an intersection-over-union threshold of 50% (AP50) and 70% (AP70), respectively. While the former primarily measures the detection capability of the model, the latter is more sensitive to the accuracy of the estimated instance shape.

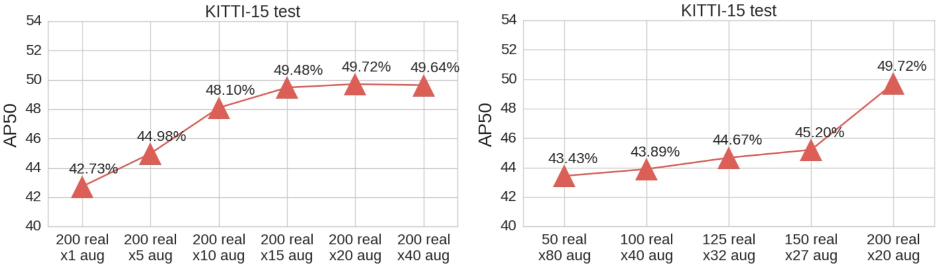
4.2 Dataset Variability and Size

In this section, we show how augmenting driving scenes with synthetic cars is an effective way to expand a dataset and increase its quality and variance. In particular, we investigate two aspects. First, introducing new synthetic cars in each image with detailed ground truth labeling makes the model less likely to overfit to the small amount of real data and exposes it to a large variety of car poses, colors and models that might not exist or be rare in real training data. Second, our augmented cars introduce realistic occlusions of real cars which makes the learned model more robust to occlusions since it is trained to detect the same real car each time with a different occlusion configuration. This second aspect also protects the model from over-fitting to the relatively small amount of annotated real car instances.

In Figure 3a we demonstrate how increasing the number of augmented images per real image improves the performance of the trained model through the increased diversity of the target class, but then saturates beyond 20 augmentations per real image.

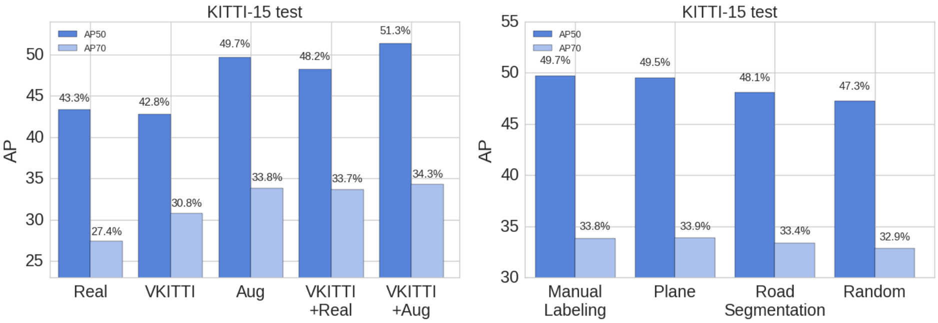
4.3 Comparing Real, Synthetic and Augmented Data

Synthetic data generation for autonomous driving has shown promising results in recent years. Nevertheless, it comes with several drawbacks:



(a) (b)

Figure 3: Instance segmentation performance using augmented data. (a) We fix the number of real images to 200 but vary the number of augmentations per real image. (b) We vary the number of real images while keeping the resulting augmented dataset size fixed to 4000 images by changing the number of augmentations accordingly.

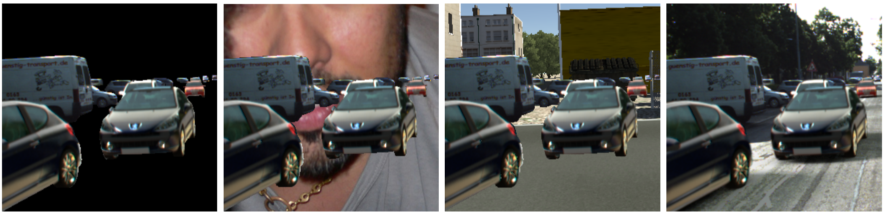


(a) (b)

Figure 4: (a) Using our augmented dataset, we can achieve better performance on the KITTI-15 test set compared to using synthetic data or real data separately. We also outperform models trained on synthetic data and fine-tuned with real data (VKITTI+Real) while significantly reducing manual effort. Additionally, fine-tuning the model trained on VKITTI using our Augmented data (VKITTI+Aug) further improves the performance. (b) Results using different techniques for sampling car poses.

- The time and effort needed to create a realistic and detailed 3D world and populate it with agents that can move and interact.
- The difference in data distribution and pixel-value statistics between the real and virtual data prevents it from being a direct replacement to real training data. Instead, it is often used in combination with a two stage training procedure where the model is first pre-trained on large amounts of virtual data and then fine-tuned on real data to better match the test data distribution.

Using our data augmentation method, we hope to overcome these two limitations. First, by using real images as background, we limit the manual effort of modeling high quality 3D cars compared to designing full 3D scenes. A large variety of 3D cars is available through online 3D model warehouses and can be easily customized. Second, by limiting the modification of the images to the foreground objects and compositing them with the real backgrounds, we keep the difference in appearance and image artifacts at a minimum. As a result, we are able



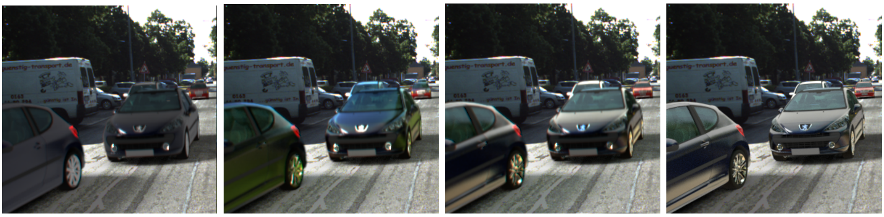
(a) Black BG
AP50 = 21.5%

(b) Flickr BG
AP50 = 40.3%

(c) Virtual KITTI BG
AP50 = 47.7.3%

(d) Real BG
AP50 = 49.7%

Figure 5: Comparison of performance of models trained on augmented foreground cars (real and synthetic) over different kinds of background.



(a) No env. map
AP50 = 49.1%

(b) Random env. map
AP50 = 49.2%

(c) True env. map
AP50 = 49.7%

(d) No postprocessing
AP50 = 43.8%

Figure 6: Comparison of the effect of post-processing and environment maps for rendering.

to boost the performance of the model directly trained on the augmented data without the need for a two stage pre-training/refinement procedure.

To compare our augmented data to fully synthetic data, we train a model using Virtual KITTI and fine-tune it with the real KITTI-360 training set. Figure 4a shows our results. While fine-tuning a model trained on Virtual KITTI with real data improves the results from 42.8% to 48.2%, our augmented dataset achieves a performance of 49.7% in a single step. Additionally, using our augmented data for fine-tuning the Virtual KITTI model significantly improves the results (51.3%). This demonstrates that our augmented dataset is closer in nature to real data than to synthetic data. While the flexibility of synthetic data can provide important variability in the training data, it fails to provide the expected boost over real data due to differences in appearance. On the other hand, augmented data complements this by providing high visual similarity to the real data, yet preventing over-fitting.

While virtual data captures the semantics of the real world, at the low level real and synthetic data statistics can differ significantly. Thus training with purely synthetic data leads to biased models that under-perform on real data. Similarly, training or fine-tuning on a small dataset of real images restricts the generalization performance of the model. In contrast, the composition of real images and synthetic cars into a single frame can help the model to learn shared features between the two data distributions without over-fitting on the synthetic ones. Note that our augmented dataset alone performs slightly better than the models trained on virtual KITTI and fine-tuned on the real dataset only. This demonstrates that state-of-the-art performance can be obtained without designing complete 3D models of the environment.

Even though our task is mainly concerned with segmenting foreground car instances, having a realistic background is very important for learning good models. Here, we analyze the effect of realism of the background for our task. In Figure 5, we compare models trained on the same foreground objects consisting of a mix of real and synthetic cars, while changing the background using the following four variations: (i) black background, (ii) random Flickr images [18], (iii) Virtual KITTI images, (iv) real background images. The result clearly shows the importance of real background imagery and its impact even when using the same foreground instance. Furthermore, to demonstrate the effect of more foreground objects vs. more diverse backgrounds we train several models with the same number of augmented images on top of varying number of real backgrounds. The results in Figure 3b show that having more diverse real backgrounds is important for training better models.

Finally, we take a closer look at the importance of realism in the augmented data. In particular, we focus on three key aspects of realism *i.e.* accurate reflections, post-processing and object positioning. Reflections are extremely important for visual quality when rendering photo-realistic car models (see Figure 6) but are they of the same importance for learning instance-level segmentation? In Figure 6, we compare augmented data using the true environment map to that using a random environment map chosen from the same car driving sequence or using no environment map at all. The results demonstrate that the choice of environment map during data augmentation affects the performance of the instance segmentation model only minimally. This finding means that it is possible to use our data augmentation method even on datasets that do not provide spherical views for the creation of accurate environment maps. On the other hand, comparing the results with and without post-processing (Figure 6c+6d) reveals the importance of realism in low-level appearance.
Another important aspect which can bias the distribution of the augmented dataset is the placement of the synthetic cars. We experiment with 4 variants: (i) randomly placing the cars in the 3D scene while only the rotation of the vehicle is modified around its up axis, (ii) randomly placing the cars on the ground plane with a random rotation around the up axis, (iii) using semantic segmentation to find road pixels and projecting them onto the 3D ground plane while setting the rotation around the up axis at random, (iv) using manually annotated tracks from bird's eye views. Figure 4b shows our results. Randomly placing the cars in 3D performs noticeably worse than placing them on the ground plane. This is not surprising as cars can be placed at physically implausible locations which do not appear in our validation data. The road segmentation method tends to place more synthetic cars in the clear road areas closer to the camera which occludes the majority of the smaller (real) cars in the background, leading to slightly worse results. The other two location sampling protocols don't show significant differences. This indicates that manual annotations are not necessary for placing the augmented cars as long as the ground plane and camera parameters are known.

5 Conclusion

In this paper, we have proposed a new paradigm for efficiently enlarging existing data distributions using augmented reality. The realism of our augmented images rivals the realism of the input data, thereby enabling us to create highly realistic data sets which are suitable for training deep neural networks. In the future we plan to expand our method to other data sets and training tasks. We also plan to improve the realism of our method by making use of additional labels such as depth and optical flow or by training a generative-adversarial method which allows for further fine-tuning the low-level image statistics to match the distribution of real-world imagery.

References

[1] Blender Online Community. *Blender - a 3D modelling and rendering package*. Blender Foundation, Blender Institute, Amsterdam, 206. URL http://www.blender.org.

[2] Gabriel J. Brostow, Julien Fauqueur, and Roberto Cipolla. Semantic object classes in video: A high-definition ground truth database. *Pattern Recognition Letters*, 30(2):88–97, 1 2009.

[3] Wenzheng Chen, Huan Wang, Yangyan Li, Hao Su, Zhenhua Wang, Changhe Tu, Dani Lischinski, Daniel Cohen-Or, and Baoquan Chen. Synthesizing training images for boosting human 3d pose estimation. In *Proc. of the International Conf. on 3D Vision (3DV)*, 2016.

[4] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[5] Jifeng Dai, Kaiming He, and Jian Sun. Instance-aware semantic segmentation via multi-task network cascades. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[6] César Roberto de Souza, Adrien Gaidon, Yohann Cabon, and Antonio Manuel López Peña. Procedural generation of videos to train deep action recognition networks. *arXiv.org*, 1612.00881, 2016.

[7] A. Dosovitskiy, P. Fischer, E. Ilg, P. Haeusser, C. Hazirbas, V. Golkov, P. v.d. Smagt, D. Cremers, and T. Brox. Flownet: Learning optical flow with convolutional networks. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV)*, 2015.

[8] Adrien Gaidon, Qiao Wang, Yohann Cabon, and Eleonora Vig. Virtual worlds as proxy for multi-object tracking analysis. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[9] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The KITTI dataset. *International Journal of Robotics Research (IJRR)*, 32(11):1231–1237, 2013.

[10] Ankush Gupta, Andrea Vedaldi, and Andrew Zisserman. Synthetic data for text localisation in natural images. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[11] Ankur Handa, Viorica Patraucean, Vijay Badrinarayanan, Simon Stent, and Roberto Cipolla. Understanding real world indoor scenes with synthetic data. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[12] Hironori Hattori, Vishnu Naresh Boddeti, Kris M. Kitani, and Takeo Kanade. Learning scene-specific pedestrian detectors without real data. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[13] Wenzel Jakob. Mitsuba renderer, 2010. http://www.mitsuba-renderer.org.

[14] Bernd Kitt, Andreas Geiger, and Henning Lategahn. Visual odometry based on stereo image sequences with ransac-based outlier rejection scheme. In *Proc. IEEE Intelligent Vehicles Symposium (IV)*, 2010.

[15] Moritz Menze and Andreas Geiger. Object scene flow for autonomous vehicles. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[16] Yair Movshovitz-Attias, Takeo Kanade, and Yaser Sheikh. How useful is photo-realistic rendering for visual learning? In *Proc. of the European Conf. on Computer Vision (ECCV) Workshops*, 2016.

[17] Xingchao Peng, Baochen Sun, Karim Ali, and Kate Saenko. Learning deep object detectors from 3d models. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV)*, 2015.

[18] James Philbin, Ondrej Chum, Michael Isard, Josef Sivic, and Andrew Zisserman. Object retrieval with large vocabularies and fast spatial matching. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2007.

[19] Leonid Pishchulin, Arjun Jain, Christian Wojek, Mykhaylo Andriluka, Thorsten Thormählen, and Bernt Schiele. Learning people detection models from few training samples. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2011.

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

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

[22] Artem Rozantsev, Vincent Lepetit, and Pascal Fua. On rendering synthetic images for training an object detector. *Computer Vision and Image Understanding (CVIU)*, 137:24–37, 2015.

[23] Alireza Shafaei, James J. Little, and Mark Schmidt. Play and learn: Using video games to train computer vision models. *arXiv.org*, 1608.01745, 2016.

[24] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *Proc. of the International Conf. on Learning Representations (ICLR)*, 2015.

[25] Michael Stark, Michael Goesele, and Bernt Schiele. Back to the future: Learning shape models from 3d cad data. In *Proc. of the British Machine Vision Conf. (BMVC)*, 2010.

[26] Hao Su, Charles Ruizhongtai Qi, Yangyan Li, and Leonidas J. Guibas. Render for CNN: viewpoint estimation in images using cnns trained with rendered 3d model views. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV)*, 2015.

[27] Yaniv Taigman, Ming Yang, Marc'Aurelio Ranzato, and Lior Wolf. Deepface: Closing the gap to human-level performance in face verification. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[28] Marvin Teichmann, Michael Weber, J. Marius Zöllner, Roberto Cipolla, and Raquel Urtasun. Multinet: Real-time joint semantic reasoning for autonomous driving. *arXiv.org*, 1612.07695, 2016.

[29] Gül Varol, Javier Romero, Xavier Martin, Naureen Mahmood, Michael J. Black, Ivan Laptev, and Cordelia Schmid. Learning from synthetic humans. *arXiv.org*, 1701.01370, 2017.

[30] Jun Xie, Martin Kiefel, Ming-Ting Sun, and Andreas Geiger. Semantic instance annotation of street scenes by 3d to 2d label transfer. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[31] Yi Zhang, Weichao Qiu, Qi Chen, Xiaolin Hu, and Alan L. Yuille. Unrealstereo: A synthetic dataset for analyzing stereo vision. *arXiv.org*, 1612.04647, 2016.

[32] Yinda Zhang, Shuran Song, Ersin Yumer, Manolis Savva, Joon-Young Lee, Hailin Jin, and Thomas A. Funkhouser. Physically-based rendering for indoor scene understanding using convolutional neural networks. *arXiv.org*, 1612.07429, 2016.

[33] Yuke Zhu, Roozbeh Mottaghi, Eric Kolve, Joseph J. Lim, Abhinav Gupta, Li Fei-Fei, and Ali Farhadi. Target-driven visual navigation in indoor scenes using deep reinforcement learning. *arXiv.org*, 1609.05143, 2016.