EVALUATION OF A TRAFFIC SIGN DETECTOR BY SYNTHETIC IMAGE DATA FOR ADVANCED DRIVER ASSISTANCE SYSTEMS

Alexander Hanel[1,*], David Kreuzpaintner[1], Uwe Stilla[1]

[1] Photogrammetry and Remote Sensing, Technical University of Munich, 80333 Munich, Germany
(alexander.hanel, david.kreuzpaintner, stilla)@tum.de

Commission II, ICWG II/III

KEY WORDS: Scene Understanding, Traffic Sign Detection, Machine Learning, Neural Network, Synthetic Images

ABSTRACT:

Recently, several synthetic image datasets of street scenes have been published. These datasets contain various traffic signs and can therefore be used to train and test machine learning-based traffic sign detectors. In this contribution, selected datasets are compared regarding ther applicability for traffic sign detection. The comparison covers the process to produce the synthetic images and addresses the virtual worlds, needed to produce the synthetic images, and their environmental conditions. The comparison covers variations in the appearance of traffic signs and the labeling strategies used for the datasets, as well. A deep learning traffic sign detector is trained with multiple training datasets with different ratios between synthetic and real training samples to evaluate the synthetic SYNTHIA dataset. A test of the detector on real samples only has shown that an overall accuracy and ROC AUC of more than 95% can be achieved for both a small rate of synthetic samples and a large rate of synthetic samples in the training dataset.

1. INTRODUCTION

Supervised machine learning classifiers require a high number of training samples with known class label to be able to predict the unknown class of a new sample with a high certainty. In automotive applications, machine learning is used for example in an advanced driver assistance system informing the driver about important traffic signs. In this system, the function of machine learning is to evaluate whether street scene images provided by a vehicle camera show traffic signs and which meaning the traffic signs have. Hereby, the term *traffic sign detection* refers to distinguish whether a part of an image contains a traffic sign or other objects, like buildings or vegetation. In contrast, the term *traffic sign recognition* refers to distinguish between different meanings of traffic signs, like *give way* or *do not enter* (Zhu et al., 2016).

Figure 1. Blanding of a synthetic RGB street scene image and the corresponding semantic ground truth image (Richter et al., 2017). Traffic signs are labelled in yellow.

Both methods for semantic segmentation (e.g. (Long et al., 2015)) and object detection (e.g. (Redmon et al., 2016)) can be used for traffic sign detection in street scene images. The methods from both authors rely on machine learning by evaluating a so-called *full image*, the street scene image (figure 1) in the above mentioned system, in one processing step. In other methods for object detection, image patches (example see figure 2) are extracted from a full image and each patch is classified in a separate processing step (e.g. (Wu et al., 2013)).

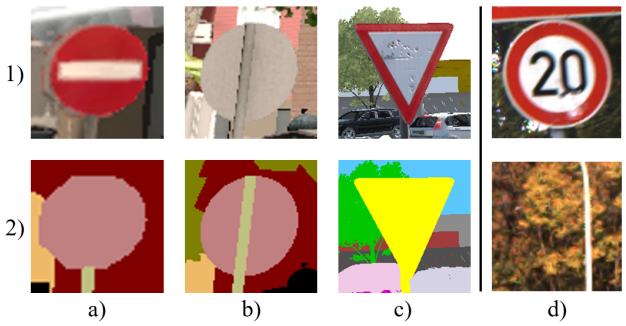

Figure 2. Synthetic image patches of traffic signs: 1a), 1b) SYNTHIA, 1c) vKITTI. Real image patches: 1d) GTSRB, 2d) GTSDB. Semantic ground truth corresponding to the synthetic patches: 2a), 2b) Front side and back side of a traffic sign having the same semantic class, 2c) Traffic sign and pole having the same semantic class. (Ros et al., 2016), (Gaidon et al., 2016), (Stallkamp et al., 2011), (Houben et al., 2013)

There is a wide range of published datasets available providing real images with known semantic class labels allowing to train machine learning models for traffic sign detection or traffic sign recognition. Some datasets (e.g. the German GTSRB dataset (Stallkamp et al., 2011)) provide images patches of traffic signs (figure 2, 1d)). Some datasets provide full street scene images together with enclosing rectangles around traffic signs as labels (e.g. the Swedish traffic sign dataset (Larsson and Felsberg, 2011), the US-American LISA US Traffic Sign Dataset (Mogel-

*Corresponding author

mose et al., 2012) or the German GTSDB dataset (Houben et al., 2013)), while other ones provide pixel-wise labels for traffic signs in street scene images (e.g. the Chinese Tsinghua-Tencent 100K dataset (Zhu et al., 2016)). All mentioned datasets have in common that their labels provide information about the meaning of the traffic signs to road users, in addition.

While the afore-mentioned datasets have been designed for traffic sign detection or traffic sign recognition, training data for traffic sign detection can also be derived from datasets designed for semantic scene understanding, like the Cityscapes dataset (Cordts et al., 2016), which provides semantic class labels for all pixels in the street scene images by semantic ground truth images (see figure 1). Cityscapes and other datasets of this kind do not provide information about the meaning of traffic signs, though.

A common approach to create labels for real images is to manually annotate the desired objects in the images by enclosing rectangles or pixel-wise and to enter additional information, like the sign meaning. This process is costly both in time and money (e.g. creating and quality-checking one high-quality semantic ground truth image for Cityscapes required more than 90 minutes (Cordts et al., 2016)) and prone to errors made by the lablers (e.g. the Cityscapes dataset has in average an annotation density of 97.1% of all pixels of the high-quality semantic ground truth images (Richter et al., 2016)).

For use in an advanced driver assistance system, a traffic sign detector needs - in addition to the high classification certainty - to ensure functional safety of the system (e.g. ISO26262 - (Organización Internacional de Normalización, 2011)) in a wide variety of situations. A challenging situation for a traffic sign detector could be for example, if the sunlight falls from behind a traffic sign on the camera, which changes the appearance of the traffic sign in the image remarkably compared to its typical appearance. For a high degree of functional safety, the training samples need to cover a wide variety of appearances of traffic signs.

In addition to the aspects mentioned above about labeling, real images of challenging situations might not be available, as they only occur under special and rare circumstances. Another way than annotating real images to acquire training samples for machine learning is to render synthetic RGB images of 3D models of a virtual world, for which the semantic ground truth images can be derived from the rendering pipeline as well. Images of challenging situations as described above can be generated and a wide variety of the appearance of traffic signs in the images achieved by modeling the virtual world as desired.

Recently, several datasets providing synthetic images of street scenes have been published for research in the field of advanced driver assistance systems. The main contribution of this paper is to analyze selected synthetic image datasets:

- First, selected synthetic image datasets are compared regarding their use for traffic sign detection using machine learning.
- Second, a selected synthetic image dataset is evaluated for traffic sign detection by training a neural network-based detector with real and synthetic training samples and by testing the detector on real samples only.

2. COMPARISON OF SYNTHETIC IMAGE DATASETS WITH REGARD TO TRAFFIC SIGN DETECTION

In this section, several publicly available synthetic image datasets are compared with regard to traffic sign detection using machine learning. Basis of the comparison is the evaluation of published information about the datasets: Complete evaluation of the dataset websites and the papers, in which the datasets have been introduced, and the evaluation of the datasets themselves. The evaluation of the datasets is done on a sample basis because of the high number of several hundred thousands of images in all datasets in total. The evaluation has been performed in March of 2018.

As motivated in the beginning of this paper, the comparison focus on the following two requirements of synthetic image datasets for traffic sign detection using machine learning:

- A **high number of traffic signs** in the synthetic images to ensure a high overall detection certainty
- A **high number of variations in the appearance of the traffic signs** in the synthetic images to ensure robustness of the detector against a wide variety of appearances of traffic signs

Derived from the requirements above, the elements of the comparison cover:

- the approaches for synthetic image derivation,
- the virtual scenes and the environmental conditions in the datasets,
- the traffic signs in the synthetic RGB images,
- the semantic labels assigned to the synthetic RGB images.

2.1 Overview over the datasets

Five datasets (overview in table 1) have been preselected by the criterion that they provide synthetic RGB images of street scenes with traffic signs and semantic ground truth images. The datasets are, ordered by the date of publication of the corresponding paper: the Virtual KITTI (further abbreviated as vKITTI) dataset from (Gaidon et al., 2016), the SYNTHetic collection of Imagery and Annotations (SYNTHIA) dataset from (Ros et al., 2016), the unnamed dataset from (Richter et al., 2016), inspired by the title of the paper, *Playing for Data*, further called PfD, the unnamed dataset from (Johnson-Roberson et al., 2016), further called DitM, also inspired by the title of the paper, *Driving in the Matrix*, and the VIsual PERception (VIPER) dataset from (Richter et al., 2017).

Dataset	#Images	Creation method	TS labels
vKITTI	21,260	Self-rendering	✓
SYNTHIA	>200,000	Self-rendering	✓
PfD	24,966	Game export	✓
DitM	>260,000	Game export	✗
VIPER	>200,000	Game export	✓

Table 1. Preselected datasets with synthetic images of street scenes with traffic signs (TS).

Beyond the pre-selected datasets, there are some other image synthetic datasets of street scenes available, but which don't contain traffic signs at all, e.g. a virtual pedestrian dataset (Marin et al., 2010), an *urban canyon* dataset to study the influence of the field-of-view of a camera on visual odometry (Zhang et al., 2016) or a multi-object and multi-camera tracking dataset (Bochinski et al., 2016). Scientific papers (e.g. (Tsirikoglou et al., 2017)) and websites (e.g. (7D Labs, 2018), (BIT Technology Solutions, 2018)) addressing synthetic image data indicate that there might be other synthetic image datasets existing, which have not been published yet.

The DitM dataset will not be considered for further comparison, as it provides no traffic sign labels and is therefore outside of the focus of this paper, though the synthetic RGB images are similar to the ones in the PfD and VIPER dataset because of the similar production process.

2.2 Synthetic image production process

The synthetic data in the datasets in our comparison is produced by one of two approaches being described by the following part (both pipelines see figure 3).

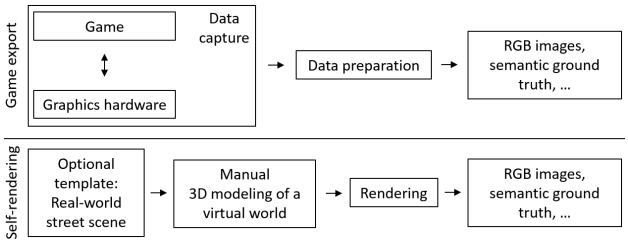

Figure 3. Approaches used to produce the synthetic data in datasets in our comparison: Top: Synthetic RGB images and buffer data are exported from video games playing on streets of a virtual world, buffer data is processed to get the semantic ground truth images. Bottom: A virtual world with streets is created by 3D modeling artificially, the synthetic RGB images and semantic ground truth images are rendered from the models.

The PfD and VIPER datasets are produced by data export from the open-world game GTA V, whose game play takes place on the streets of a virtual world. Resources (like geometry meshes, texture maps or shaders) communicated during a game session from the game to the graphics hardware to display the game on a computer monitor are captured. Lack of access to the source code and the virtual world inside GTA V requires this effort. The captured images showing the player's view on the virtual street scenes are saved as synthetic RGB images. Identifier images containing permanent identifiers for all resources corresponding with a pixel in the RGB image are created. Pixels sharing the same resources are clustered to a patch automatically. The semantic ground truth image for each RGB image is created by manually annotating each patch with a semantic class. The manual effort is reduced by propagating the class of a patch for other RGB images after its manual annotation. The VIPER dataset provides in addition semantic ground truth on the so called ”instance-level”, identifying individual objects in multiple images, by exploiting information from additional resources.

The vKITTI and the SYNTHIA datasets are produced by the self-rendering approach. A 3D model of a virtual world with streets is created as first step. In the case of vKITTI, the virtual world is created based on real scenes shown in the image sequences in the KITTI dataset (Geiger et al., 2012). The 3D models for objects in the virtual world are taken from a publicly available database of 3D models. In the case of SYNTHIA, there are no links of the virtual world to real scenes. Aside from the 3D models of the objects, the illumination in the virtual world and the position of the camera can be set as desired. Finally, the RGB images and the semantic ground truth images are created by rendering, i.e. by taking virtual images of different scenes in the virtual world. The Unity game engine is used to render the vKITTI and SYNTHIA images.

One can assume that the appearance of traffic signs in synthetic images should be as close as possible to the appearance in real images to apply machine learning models trained on synthetic data on real data, for example in the afore-mentioned advanced driver assistance system. Therefore, the question of the degree of realism in the synthetic datasets is addressed. The authors of the PfD and VIPER datasets motivate the game-based approach by the higher degree of realism of commercial games compared to open-source virtual worlds (e.g. from a driving simulator). For example, they state that ”realism” is achieved by ”the high fidelity of material appearance and light transport simulation” and ”the content of the virtual worlds”, which includes ”the layout of objects and environments, the realistic textures, the motion of vehicles and autonomous characters, the presence of small objects that add detail, and the interaction between the player and the environment (Richter et al., 2016). More details, for example which criteria define a high-fidelity material, are not given and subsequently it is also not proven whether GTA V fulfills these criteria. The authors of SYNTHIA state to have used ”realistic models of cars, vans, pedestrians and cyclists” (Ros et al., 2016), but do not further discuss the degree of realism. Advantageous of their self-rendering approach is the full access to the virtual world allowing to change every part of it, like the geometry and the material of objects, the illumination and the world composition. Both the GTA engine and the Unity engine are game engines relying on a technique called *deferred shading*. This technique allows real-time rendering for a fluent game session with computers available on the market today at the cost of some drawbacks. Mainly, the support of anti-aliasing and the use of semi-transparent materials are limited (Unity Technologies, 2017).

2.3 Virtual scenes and environmental conditions

Variations in the scenes and the environmental conditions can increase the variations in the appearance of traffic signs (cf. figure 6) and their placement, for example.

Each virtual world contains different scene types, like downtown, suburb, countryside or highway with typical traffic sign population. The scenes in the PfD and VIPER dataset are defined by the game scenes of GTA, but can be changed with third party modificators for the game (e.g. Map Builder, (OmegaKingMods, 2016)). The game scenes of GTA are controlled by the player, who moves a virtual person within the virtual world. Though the area of the land in the virtual world in GTA V is more than 48 km^2 (KeWiS, 2015) and is larger than the virtual world in many other games, the limited influence of a third person on the virtual world of the game makes it challenging to create challenging situations as described in the motivation. The full access to the virtual world in the approach used for SYNTHIA and vKITTI is more flexible for creating challenging situations.

The synthetic RGB images of the different datasets show variations in the environmental conditions, whereby the seasons of

Figure 4. Synthetic images (bottom) in the vKITTI dataset are based on real images (top) in the KITTI dataset. Objects like buildings, vegetation, streets and cars, can be identified in both the virtual and real scene based on their placement, but the geometry and material of the virtual object seem to be simplified.

the year, the daytimes and weather conditions are meant. Images showing summer season, daylight and sunny weather are the most frequent ones. Other environmental conditions, like winter or sunrise, sunset, night or overcast, rain are only partially covered by each dataset.

Assessing virtual worlds with regard to their degree of realism is especially interesting for the vKITTI dataset, where the images of the real KITTI dataset are available as reference. Visual comparison between sample image pairs form the real KITTI and the synthetic vKITTI dataset shows that dominant objects, like cars, buildings, vegetation, of the synthetic images can be identified in the corresponding real image (example see figure 4). According to the authors of the vKITTI dataset, the cars in the virtual world are placed and oriented using information about the position and orientation of the cars provided in the real KITTI dataset. In contrast, the geometry and the material of the virtual objects are different from the real objects, what can be drawn back to the use of 3D models from a public database instead of creating own models. Comparing the shape of shadows of a corresponding virtual and real object in an image pair, the statement of the authors of the vKITTI dataset can be confirmed that the illumination is set manually. The described differences in the geometry and material of objects between vKITTI and real KITTI images indicate that the appearance of objects in synthetic and real images is different as well. The influence of those differences in the appearances on an object detector trained on one dataset and tested on the other one would have to be investigated.

vKITTI inheres also some unrealistic effects like cars disappearing from one frame to the next one without any reason, like being occluded behind a larger object. While the lack of real data as reference makes it hard to discuss the degree of realism of the virtual worlds of the other datasets, they show inconsistencies as well. For example, the SYNTHIA dataset contains a scene in which vehicles are driving on a road, which is completely blocked by construction barriers and another scene, in which pedestrians and cars intersect each other. The player of GTA game sessions can cause or avoid inconsistencies as well, especially in the interaction with artificially controlled road users. In contrast to the appearance addressed above, the inconsistencies might have a larger influence on other learning tasks, e.g. learning the behavior of road users, than on traffic sign detection.

2.4 Traffic sign variations

Variations in the training data can also be achieved by variations in the position of traffic signs in the street scene images, by variations in their meanings or by modifications of the 3d models of the signs themselves, for example.

Varying positions of traffic signs in the street scene images can help to avoid that a machine learning model gets trained to ”expect” a traffic sign only in certain parts of a street scene image. That could be a drawback with regard to the robustness of the detector. The distribution of traffic signs in the street scene images of the datasets is visualized by heatmaps in figure 5 created from the semantic ground truth image. The ”hot spots” (red parts in the heatmaps showing a high density of traffic signs in that part of the synthetic RGB image) of traffic signs in the synthetic images are in the image center, while the hot spot for the real image dataset Cityscapes, which covers data from various cities, is in the right upper part of the image. Further, the hot spot for vKITTI is closer to the bottom border of the image then the other hot spots are. Probably, because vKITTI is the only dataset, in which the traffic signs and their poles (typically shown below the sign in street scene images) have the same semantic class. The heatmaps also show that the distribution of traffic signs varies between the synthetic datasets. The distribution is wider in the datasets produced with GTA V than in the datasets produced with a self-created virtual world. Approximately, the distribution in the Cityscapes dataset is as wide as in the datasets produced with GTA. Note also the colorbars of the heatmaps showing that there is a large difference in the number of traffic signs between the datasets.

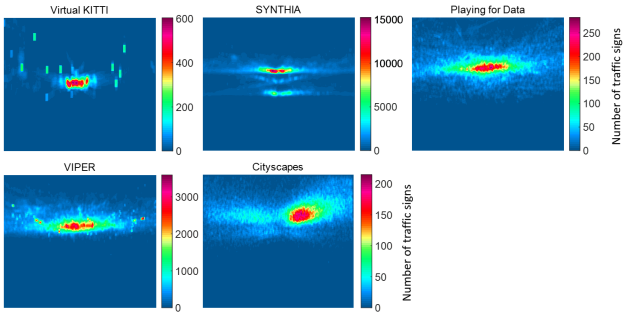

Figure 5. Heatmaps showing the distribution of traffic signs in the synthetic images of different datasets. The distributions are derived from the semantic ground truth images. The heatmap of the real-world dataset Cityscapes is given as reference.

Not very much information can be given about the meanings of traffic signs; all datasets contain traffic signs of different meanings. The datasets produced with the self-created virtual worlds contain a smaller number of different meanings then the datasets produced with GTA. But as all traffic signs in the datasets share the same semantic class, the distribution of traffic sign meanings can’t be derived directly from the semantic ground truth, though this information would be interesting in regard to variations in training data. Even for traffic sign detectors, it might be advantageous to train different detector models for different groups of traffic signs separated by their color or by their shape. Meaningless traffic signs (e.g. figure 6 d)) can be found in some synthetic images; they should not be used for training detector models.

Several modifications are applied to the basic 3D models and materials of signs to make them looking more realistic. Modifications can be for example dirt or reflections on the sign or aging

a) b) c) d)

Figure 6. Examples of synthetic images of traffic signs from different datasets: With modifications applied: a) painting chipping off from the sign, b) dirt and reflections on the sign, c) during rain. Without modifications: d) meaningless sign.

effects, like paint peeling off from signs (see examples in figure 6). As the same modifications appear in different images in a dataset, one can assume that the modifications are applied manually when creating the virtual worlds. To give an example, all signs with the same meaning (figure 6 b), c)) in the GTA-based datasets show the same dirt pattern. The same observation can be made for paint chipping off from the signs in the vKITTI dataset. Applying the same modifications to several traffic signs in the virtual world increases the number of variations in the training data only a bit. Different combinations of modifications could be applied to the signs to get more variations.

2.5 Labeling strategy and inconsistencies

Semantic labels are needed together with RGB images to create training datasets for machine learning-based object detectors. Especially for large (synthetic) datasets, training data has to be created automatically from the datasets. Labeling strategies or inconsistencies in the labels can be a cost factor, if correction or removal of undesired data from training datasets require manual action.

All analyzed datasets provide semantic ground truth images (example see figure 1) containing pixel-wise semantic labels for the objects in the corresponding synthetic RGB images. Regarding traffic signs, different labeling strategies are used for different datasets (table 2).

Dataset	vKITTI	SYNTHIA	PfD	VIPER
TS meanings distinguished	✗	✗	✗	✗
Pixel-wixe labels for TS	✓	✓	✓	✓
Instance labels for TS	✗	✗	✗	✗
Front / back side distinguished	✗	✗	✓	✓
TS and pole distinguished	✗	✓	✓	✓

Table 2. Comparison of labeling strategies used for the analyzed synthetic image datasets with regard to traffic signs (TS).

The datasets only provide semantic labels on a ”class-level” for traffic signs. The term ”class” is used to distinguish different kinds of objects in street scenes. The semantic class definition of the analyzed datasets is identical or similar to the class definition proposed by Cityscapes. In both Cityscapes and the analyzed datasets, all traffic signs are assigned to a single class for traffic signs. None of the analyzed datasets provides labels distinguishing different meanings of traffic signs. VIPER, SYNTHIA and vKITTI provide labels on a ”instance-level” for some kinds of objects, but not for traffic signs. The term ”instance” is used to distinguish between different individual objects.

With regard to traffic sign detection, the labeling strategies mentioned in table 2 can influence the trained models. In vKITTI, the pole of a traffic sign has the same semantic label as the sign itself. In this case, the trained detector model might ”remember” the combination of sign and pole, which could lead to a decrease in the detector performance, if the pole is occluded behind other objects. In vKITTI and SYNTHIA, the front side and back side of a traffic sign have the same label, which has to be questioned, because the different appearance of front side (different colors, text, ...) and back side (typically metal-gray) might lead to a worse detector performance of the front side of the signs, which is typically the side of interest in traffic sign detection.

a) b) c) d)

Figure 7. Examples of inconsistencies in the *traffic sign* labels of different datasets. Top row: Semantic ground truth images (yellow: *traffic sign*, gray: *pole*), bottom row: Corresponding synthetic RGB images: a) Ladder labelled as *traffic sign*, b) Varying class labels for one sign, c) and d) Both cases show the same traffic sing; pole one time labelled as *pole*, the other time as *traffic sign*; backside of the traffic sign labelled as *traffic sign*.

Inconsistencies in the semantic labels, like a wrong label, occur in the analyzed datasets, but should be avoided, as the labels are used as ground truth for the class of an object in the corresponding RGB image. In the semantic labels of traffic signs of the analyzed datasets, the following inconsistencies have been found: Other objects are labelled as traffic signs (figure 7 a)), the same sign has varying class labels (figure 7 b)). In one dataset, the backside of a traffic sign is sometimes labelled as traffic sign, other times as different object (figure 7 c)). In one dataset, the pole of a traffic sign is sometimes labelled as traffic sign, sometimes as pole (figure 7 d)).

3. EVALUATION OF A SYNTHETIC DATASET FOR TRAFFIC SIGN DETECTION

The second part of this paper aims at the evaluation of a synthetic dataset for machine learning-based traffic sign detection. The authors of the analyzed datasets use machine learning to evaluate their datasets, as well. Their applications are multi-object car tracking (vKITTI) and semantic segmentation (SYNTHIA, PfD, VIPER).

As the purpose is to evaluate synthetic data independent of side effects, the experimental setup is kept as easy as possible (pipeline see figure 8). Especially, that means using a detector, which evaluates an image patch instead of a full image (example patches see figures 2, 9). To keep in mind from the motivation of this paper,

a street scene image is considered as full image, while an image showing a traffic sign with a small border around it showing background (examples see figure 9) is considered as image patch. Side effects can be for example the influence of the position or size of traffic signs in street scene images on the detector, or the different shape of image patches if traffic signs and their poles have the same semantic class. Both examples are not typical for synthetic image datasets in particular, as they can theoretically occur also in real image datasets. The evaluation should be independent of them, therefore.

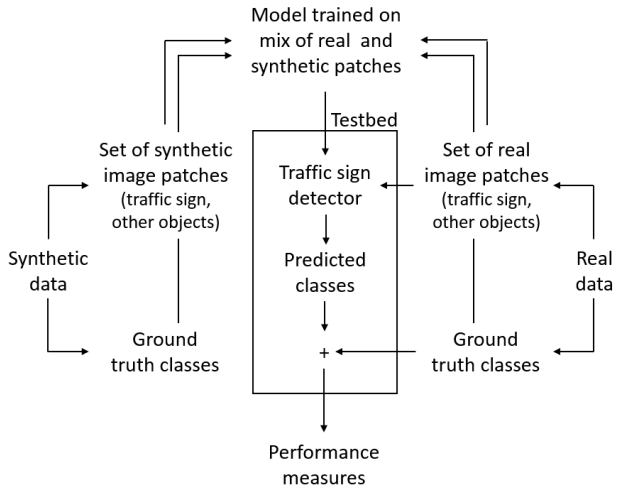

Figure 8. Pipeline to evaluate the performance of traffic sign detection on synthetic images. Performance measures are ROC AUC, OA, P and R. The testbed could be extended to be able to control different variations in the virtual world, for example regarding the incidence of sunlight.

The detector bases on deep learning, following recent work of other authors in the field of traffic sign detection and recognition in images (e.g. (Wu et al., 2013), (Zhu et al., 2016)). The architecture of the deep learning network is a modification of LeNet, which has been used in other work for object detection (Sermanet and LeCun, 2011). The detector has been already trained and tested on real image data of the GTSDB and GTSRB dataset (Hanel and Stilla, 2018). The overall accuracy achieved in that work is 96%. The training and test datasets for that detector need to contain samples of image patches, as described in the previous paragraph, and their semantic class labels, i.e. either *traffic sign* or *other objects*. The image patches have to have a geometric resolution of 32 x 32 px and have to be RGB. Either *traffic sign* or *other objects* is the output of the detector for each test sample.

For the evaluation, multiple detector models are trained with a mixture of synthetic samples and real samples, as shown in figure 8. The detector model is tested only with real samples. Thinking of a traffic sign detector as part of an advanced driver assistance system, the detector training can be performed offline and can rely on synthetic and real samples therefore. In contrast, the detector is applied online, i.e. on public roads, to images of a vehicle camera, where no synthetic images will be available. Therefore, the test dataset contains only real samples. The performance measures for evaluation are the common ones for binary classifiers: the overall accuracy (OA), the area under the curve (AUC) of the receiver operating characteristic (ROC) curve, precision (P) and recall (R).

4. EXPERIMENT WITH THE SYNTHIA DATASET

The real samples for training and testing have been derived from the GTSRB and the GTSDB dataset. The real image patches of *traffic signs* are taken from the GTSRB dataset (Stallkamp et al., 2011), while the real image patches of *other objects* are randomly sampled from street scene images in the GTSDB dataset (Houben et al., 2013). Both datasets have been already used as reference by other authors working in this field (e.g. (Wu et al., 2013)) and are the basis for benchmarks on traffic sign detection and recognition (Ruhr-University Bochum, Institute for Neuroscience, 2010).

The synthetic samples for training have been derived from the SYNTHIA dataset. SYNTHIA contains traffic signs with the same shape and texture as the signs in the GTSRB and GTSDB datasets. Though, the number of different traffic sign meanings overlapping between these real datasets and that synthetic dataset is only around ten (examples see figure 9), depending on how strictly the shape and texture of the signs should match between the datasets. Therefore, the absolute number of synthetic samples is small (details see following paragraphs). Performing the experiments with vKITTI was discarded because of the expected high effort to separate the signs from the poles when creating the image patches of traffic signs from the street scene images, as pole and sign share the same semantic class (see figure 2). Performing the experiments with a GTA-based synthetic dataset would have resulted in a high number of synthetic samples from the US, but, to the knowledge of the authors, there is no suitable real image dataset of traffic signs from the US available. Most of the street scene images in the afore-mentioned LISA dataset (Mogelmose et al., 2012) have a low quality, e.g. due to compression effects or are blurred.

Figure 9. Traffic signs with only a few different meanings can be used to train the traffic sign detector, as there is no larger overlap between a published real and a synthetic dataset (Ros et al., 2016).

The synthetic RGB images of street scenes and the semantic ground truth images are processed as follows to extract image patches of traffic signs: Tight enclosing rectangles around traffic signs are calculated using the pixel-wise semantic labels. The rectangles are enlarged to include a border showing background objects in the image patches, as the real image patches in the GTSRB dataset contain a border as well. This process is done automatically. For creating the training data, suitable image patches have to be selected: Real image patches showing traffic signs with meanings not provided by the SYNTHIA dataset are removed, the same for synthetic image patches. Synthetic patches showing the backside of traffic signs are removed (see labeling strategy of SYNTHIA in table 2), which has to be done manually, but is an easy task. Image patches of traffic signs, which can be mirrored

like the *do not enter* sign, are mirrored by horizontal flipping to increase the number of synthetic training samples. Synthetic image patches of *other objects* are randomly sampled from the synthetic RGB street scene images in SYNTHIA, excluding traffic signs in these images, of course.

The total number synthetic image patches of traffic signs derived from SYNTHIA, as described above, is around 1,750. The ratio of patches of traffic signs to other objects is 1:5 to consider that the most parts of a street scene image do not contain traffic signs. The experiment covers multiple training steps of the detector model with different ratios of synthetic samples to real samples. The ratio is varied from 1:10 to 10:1. The total number of samples is kept constant for all trainings. The training dataset is split into a 80% part for the training itself and a 20% part for validation. The ratio between training and test dataset is 80%:20%. Training is performed until the loss asymptotes. The hyperparameter values have been found by hand-tuning: The batch size is 64, the learning rate 0.001, the drop-out probability 80%.

5. DETECTOR PERFORMANCE IN THE EXPERIMENT

The traffic sign detector achieves for all ratios between the number of synthetic training samples and real training samples an overall accuracy and ROC AUC of more than 95% (see figure 10). The shapes of the OA and AUC curves reveals a periodic increase and decrease of the OA and AUC values, respectively, over the stepwise change of the ratio between the samples. A clear statement about the reason for these shapes can’t be given. Note, that for training a drop-out probability of 80% has been used, which can be a source for indeterministic performance of a classifier for multiple trainings. Figure 10 further shows that the difference between OA and AUC increases with an increasing rate of synthetic samples in the training data; the OA curve shows a slightly negative linear trend, especially for ratios larger than 10:5. Though, the overall very high values of more than 95% let us draw the conclusion that synthetic data from SYNTHIA can be used to train a traffic sign detector.

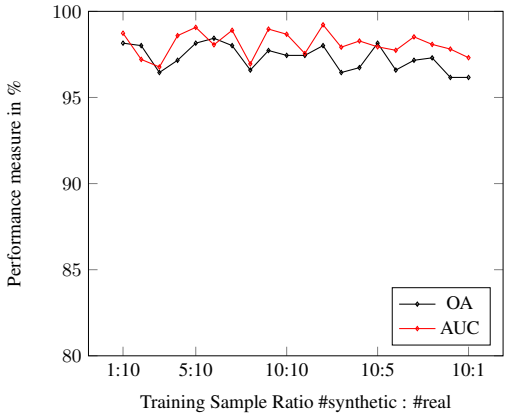

Figure 10. Overall accuracy and ROC AUC of the traffic sign detector trained with different ratios between synthetic and real samples. Tests are done on real samples only.

The precision and recall curves (figure 11) show the same periodic increase and decrease of the values for a changing ratio between synthetic and real training samples. The precision and recall for the class *other objects* are always above 95%. The precision and recall for the class *traffic sign* ranges between 80% and 100%; the periodic increase and decrease is much larger than for the other class. As the recall curve for *traffic sign* shows lower values than the precision curve for *traffic sign*, the rate of false negative detections will be higher than the rate of false positives detections for this class.

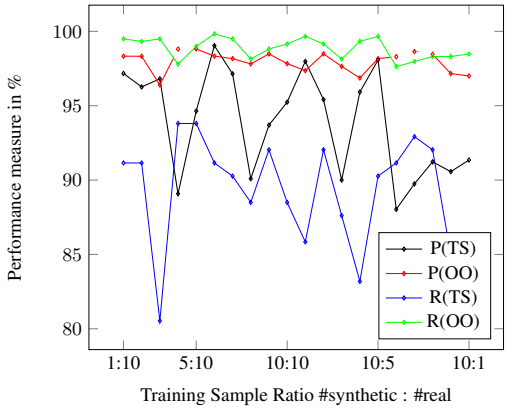

Figure 11. Precision and recall curves for both classes *traffic sign* (TS) and *other objects* (OO).

When discussing the detector performance, one might question the small number of just a few thousand samples (see section 4) for training of a deep learning network. Therefore, the same network has been trained on all image patches of traffic signs from the GTSRB dataset and a corresponding amount of image patches of other objects sampled from the GTSDB dataset, in total more than 100,000 image patches. Test on the same (small set of) real samples as before results in an overall accuracy of 98.7% and a precision of 96.4% and a recall of 95.5% for the class *traffic sign*. Compared with the values (OA = 98.2%, P(TS) = 97.2%, R(TS) = 91.2%) obtained for training on a real-only set, but with the same number of samples as used for figures 10, 11, the overall accuracy and precision obtained for the large real sample training set differ by less than 1 percentage point. Only the difference between the recall values is a little bit larger, around 4 percentage points. This comparison can be interpreted as that the small real and synthetic mixed training set is not in general problematic, especially as the network used for the experiments has around 25,000 parameters only. In addition, (Mayer et al., 2018) write in a recent work, that a small training set of 1,000 image pairs in their application of optical flow estimation can be sufficient to train a deep network.

6. CONCLUSION

In this contribution, selected synthetic datasets providing RGB images of street scenes and corresponding semantic ground truth images have been compared with regard to traffic sign detection using machine learning. The comparison has shown that the datasets contain images with traffic signs in different types of street scenes, under different environmental conditions and with different modifications applied to the signs. The semantic labels are provided pixel-wise. The labeling strategies might cause manual effort when creating training datasets for a traffic sign detector, for example, as the front side and back side of traffic signs has not been assigned to different semantic classes in all datasets. The degree of realism of the datas is addressed by the authors of the dataset mainly by evaluating the performance of machine learning tasks, e.g. semantic segmentation, on their dataset. Own experiments with the synthetic SYNTHIA dataset and a LeNet-based network for traffic sign detection have shown a high overall

accuracy and ROC AUC of more than 95% for training data with different ratios between synthetic and real data. Though, it was only possible to derive around 1,750 synthetic training samples of traffic signs because of the low overlap of signs with the same meaning with a suitable real image dataset. Future work can focus on assessing the quality of synthetic images without using a package of machine learning and the data.

REFERENCES

7D Labs, 2018. 7D Labs - Synthetic data for visual machine learning. Website. https://7dlabs.com, accessed 2018-04-01.

BIT Technology Solutions, 2018. BIT Technology Solutions - Synthetic training data with extremly high degree of reality. Website. http://www.bit-ts.com/de/, accessed 2018-04-01.

Bochinski, E., Eiselein, V. and Sikora, T., 2016. Training a convolutional neural network for multi-class object detection using solely virtual world data. In: *IEEE International Conference on Advanced Video and Signal-Based Surveillance*, Colorado Springs, CO, USA, pp. 278–285.

Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S. and Schiele, B., 2016. The cityscapes dataset for semantic urban scene understanding. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Gaidon, A., Wang, Q., Cabon, Y. and Vig, E., 2016. Virtual worlds as proxy for multi-object tracking analysis. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4340–4349.

Geiger, A., Lenz, P. and Urtasun, R., 2012. Are we ready for autonomous driving? the kitti vision benchmark suite. In: *Conference on Computer Vision and Pattern Recognition (CVPR)*.

Hanel, A. and Stilla, U., 2018. Iterative calibration of a vehicle camera using traffic signs detected by a convolutional neural network. In: *Proceedings of the 4th International Conference on Vehicle Technology and Intelligent Transport Systems*, Vol. 1, pp. 187–195.

Houben, S., Stallkamp, J., Salmen, J., Schlipsing, M. and Igel, C., 2013. Detection of traffic signs in real-world images: The German Traffic Sign Detection Benchmark. In: *International Joint Conference on Neural Networks*, pp. 1–8.

Johnson-Roberson, M., Barto, C., Mehta, R., Sridhar, S. N. and Vasudevan, R., 2016. Driving in the matrix: Can virtual worlds replace human-generated annotations for real world tasks? *CoRR*.

KeWiS, 2015. The most accurate GTA map comparison ever! - Grand Theft Auto Series - GTAForums. Website. http://gtaforums.com/topic/821812-the-most-accurate-gta-map-comparison-ever/, accessed 2018-04-05.

Larsson, F. and Felsberg, M., 2011. Using fourier descriptors and spatial models for traffic sign recognition. In: A. Heyden and F. Kahl (eds), *Image Analysis*, Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 238–249.

Long, J., Shelhamer, E. and Darrell, T., 2015. Fully convolutional networks for semantic segmentation. In: *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3431–3440.

Marin, J., Vzquez, D., Gernimo, D. and Lpez, A. M., 2010. Learning appearance in virtual scenarios for pedestrian detection. In: *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 137–144.

Mayer, N., Ilg, E., Fischer, P., Hazirbas, C., Cremers, D., Dosovitskiy, A. and Brox, T., 2018. What makes good synthetic training data for learning disparity and optical flow estimation? *CoRR*.

Mogelmose, A., Trivedi, M. M. and Moeslund, T. B., 2012. Vision-based traffic sign detection and analysis for intelligent driver assistance systems: Perspectives and survey. *IEEE Transactions on Intelligent Transportation Systems* 13(4), pp. 1484–1497.

OmegaKingMods, 2016. Map Builder - GTA5-Mods.com. Website. https://de.gta5-mods.com/tools/map-builder, accessed 2018-04-05.

Organización Internacional de Normalización, 2011. ISO 26262: Road vehicles – Functional safety.

Redmon, J., Divvala, S. K., Girshick, R. B. and Farhadi, A., 2016. You only look once: Unified, real-time object detection. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pp. 779–788.

Richter, S. R., Hayder, Z. and Koltun, V., 2017. Playing for benchmarks. In: *International Conference on Computer Vision (ICCV)*.

Richter, S. R., Vineet, V., Roth, S. and Koltun, V., 2016. Playing for data: Ground truth from computer games. In: B. Leibe, J. Matas, N. Sebe and M. Welling (eds), *European Conference on Computer Vision (ECCV)*, LNCS, Vol. 9906, Springer International Publishing, pp. 102–118.

Ros, G., Sellart, L., Materzynska, J., Vazquez, D. and Lopez, A. M., 2016. The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3234–3243.

Ruhr-University Bochum, Institute for Neuroscience, 2010. German Traffic Sign Benchmarks. Website. http://benchmark.ini.rub.de/?section=gtsdb&subsection=dataset, accessed 2018-04-05.

Sermanet, P. and LeCun, Y., 2011. Traffic sign recognition with multi-scale convolutional networks. In: *The 2011 International Joint Conference on Neural Networks*, pp. 2809–2813.

Stallkamp, J., Schlipsing, M., Salmen, J. and Igel, C., 2011. The German Traffic Sign Recognition Benchmark: A multi-class classification competition. In: *IEEE International Joint Conference on Neural Networks*, pp. 1453–1460.

Tsirikoglou, A., Kronander, J., Wrenninge, M. and Unger, J., 2017. Procedural modeling and physically based rendering for synthetic data generation in automotive applications. *CoRR*.

Unity Technologies, 2017. Unity - Manual: Deferred shading rendering path. Website. https://docs.unity3d.com/Manual/RenderTech-DeferredShading.html, accessed 2018-04-05.

Wu, Y., Liu, Y., Li, J., Liu, H. and Hu, X., 2013. Traffic sign detection based on convolutional neural networks. In: *The 2013 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–7.

Zhang, Z., Rebecq, H., Forster, C. and Scaramuzza, D., 2016. Benefit of large field-of-view cameras for visual odometry. In: *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 801–808.

Zhu, Z., Liang, D., Zhang, S., Huang, X., Li, B. and Hu, S., 2016. Traffic-sign detection and classification in the wild. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2110–2118.