

Chapter 12

Semantic Segmentation of Urban Scenes via Domain Adaptation of SYNTHIA

German Ros, Laura Sellart, Gabriel Villalonga, Elias Maidanik, Francisco Molero, Marc Garcia, Adriana Cedeño, Francisco Perez, Didier Ramirez, Eduardo Escobar, Jose Luis Gomez, David Vazquez and Antonio M. Lopez

Abstract Vision-based semantic segmentation in urban scenarios is a key functionality for autonomous driving. Recent revolutionary results of deep convolutional neural networks (CNNs) foreshadow the advent of reliable classifiers to perform such visual tasks. However, CNNs require learning of many parameters from raw images; thus, having a sufficient amount of diverse images with class annotations is needed. These annotations are obtained via cumbersome, human labor which is particularly challenging for semantic segmentation since pixel-level annotations are required. In this chapter, we propose to use a combination of a virtual world to automatically generate realistic synthetic images with pixel-level annotations, and domain adaptation to transfer the models learned to correctly operate in real scenarios. We address the question of how useful synthetic data can be for semantic segmentation—in particular, when using a CNN paradigm. In order to answer this question we have generated a synthetic collection of diverse urban images, named SYNTHIA, with automatically generated class annotations and object identifiers. We use SYNTHIA in combination with publicly available real-world urban images with manually provided annotations. Then, we conduct experiments with CNNs that show that combining SYNTHIA with simple domain adaptation techniques in the training stage significantly improves performance on semantic segmentation.

12.1 Introduction

In practice, even the best visual descriptors, class models, feature encoding methods and discriminative machine learning techniques are not sufficient to produce reliable classifiers if properly annotated datasets with sufficient diversity are not

G. Ros (✉) · L. Sellart · G. Villalonga · E. Maidanik · F. Molero · M. Garcia · A. Cedeño · F. Perez · D. Ramirez · E. Escobar · J.L. Gomez · D. Vazquez · A.M. Lopez
Computer Vision Center, Campus UAB, Barcelona, Spain
e-mail: gros@cvc.uab.es

© Springer International Publishing AG 2017
G. Csurka (ed.), *Domain Adaptation in Computer Vision Applications*,
Advances in Computer Vision and Pattern Recognition,
DOI 10.1007/978-3-319-58347-1_12

227

available. Indeed, this is not a minor issue since data annotation remains a cumbersome, human-based labor prone to error; even exploiting crowd-sourcing for annotation is a nontrivial task [34]. For instance, for some ADAS and for AD, semantic segmentation is a key issue [281, 395, 413] and it requires pixel-level annotations (i.e. obtained by delineating the silhouette of the different classes in urban scenarios, namely pedestrian, vehicle, road, sidewalk, vegetation, building, etc.).

Autonomous driving (AD) will be one of the most revolutionary technologies in the near future in terms of the impact on the lives of citizens of the industrialized countries [538]. Nowadays, advanced driver assistance systems (ADAS) are already improving traffic safety. The computer vision community, among others, is contributing to the development of ADAS and AD due to the rapidly increasing performance of vision-based tools such as object detection, recognition of traffic signs, road segmentation, etc.

Roughly until the end of the first decade of this century, the design of classifiers for recognizing visual phenomena was viewed as a two-fold problem. First, enormous effort was invested in research of discriminative visual descriptors to be fed as features to classifiers; as a result, descriptors such as Haar wavelets, SIFT, LBP, or HOG, were born and became widely used. Second, different machine learning methods were developed, with discriminative algorithms such as SVM, AdaBoost, or Random Forests usually reporting the best classification accuracy due to their inherent focus on searching for reliable class boundaries in feature space. Complementarily, in order to make easier the search for accurate class boundaries, methods for transforming feature space (e.g., PCA, BoW encoding, Kernel mappings) as well as more elaborate class models (e.g., DPM, superpixels) were proposed.

In order to ameliorate this problem there are paradigms such as unsupervised learning (no annotations assumed), semi-supervised learning (only a few annotated data), and active learning (to focus on annotating informative data), under the assumption that having annotated data (e.g., images) is problematic but data collection is cheap. However, for ADAS and AD such data collection is also an expensive activity since many kilometers must be traveled to obtain sufficient diversity. Moreover, it is well known that, in general terms, supervised learning (annotations assumed) tends to provide the most accurate classifiers.

Recently, the need for large amounts of accurately annotated data has become even more crucial with the massive adoption of deep convolutional neural networks (CNNs) by the computer vision community. CNNs have yielded a significant performance boost for many visual tasks [192, 275, 442, 487]. Overall, CNNs are based on highly nonlinear, end-to-end training (i.e. from the raw annotated data to the class labels) which implies the learning of millions of parameters and, accordingly, they require a relatively larger amount of annotated data than methods based on hand-crafted visual descriptors.

As we will review in Sect. 12.2, the use of visually realistic synthetic images is gaining attention in recent years (e.g., training in virtual worlds [19, 175, 222, 229, 325, 328, 358, 386, 435], synthesizing images with real-world backgrounds and inserted virtual objects [364, 370]) due to the possibility of having diversified samples with automatically generated annotations. In this spirit, in this chapter we address

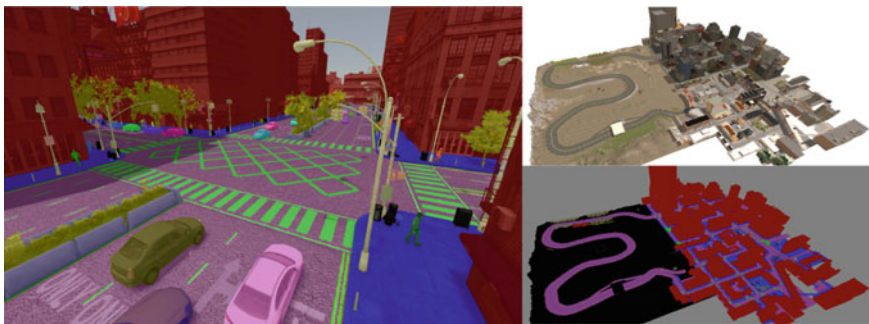


Fig. 12.1 The SYNTHIA Dataset. A sample frame with overlaid semantic labels (*Left*) and a general view of the city (*right*)

the question of *how useful can the use of realistic synthetic images of virtual-world urban scenarios be for the task of semantic segmentation—in particular, when using a CNN paradigm*. To the best of our knowledge, this analysis has not been done so far. Note that, the synthetic training data can not only come with automatically generated class annotations from multiple viewpoints and simulated lighting conditions (providing diversity), but also with ground truth for depth (e.g., simulating stereo rigs and LIDAR), optical flow, object tracks, etc.

Moreover, in the context of ADAS/AD the interest in using virtual scenarios is already increasing for the task of validating functionalities in the Lab, i.e. to perform validation in the real world (which is very expensive) only once after extensive and well-designed simulations are passed. Therefore, these virtual worlds can be used for generating synthetic images to train the classifiers involved in environmental perception. In addition, the realism of these virtual worlds is constantly increasing thanks to the continuously growing videogames industry.

To address the above-mentioned question, we have generated *SYNTHIA*¹: a *SYNTHetic* collection of *Imagery* and *Annotations* of urban scenarios (see Figs. 12.1, 12.2 and 12.3). In Sect. 12.3 we highlight its diversity and how we can automatically obtain a large number of images with annotations. As virtual and real-world cameras are different *sensors*, classifiers trained only with virtual images may require domain adaptation to work on real images [357, 467, 503, 545]; this is not surprising as domain adaptation is also often required when training images and testing images come from different real-world camera sensors [489, 503].

In Sect. 12.4 first we describe the CNN used to perform semantic segmentation. Then we present our domain adaptation strategy which consists of training with the synthetic data and a smaller number of real-world data simultaneously, i.e. in the same spirit than [503] for a HOG-LBP/SVM setting. In our case, the data combination is done in the generation of batches during CNN training. The experiments conducted in Sect. 12.5 show how SYNTHIA successfully complements different

¹SYNTHIA is available at <http://synthia-dataset.net>.



Fig. 12.2 An example of the different areas/environments available in SYNTIA: city center, *top-left*; town, *top-right*; highway, *bottom-left*; tunnel, *bottom-right*.)



Fig. 12.3 Dynamic objects in SYNTIA. (*Top*) vehicles; (*middle*) cyclists; (*bottom*) pedestrians

datasets (Camvid, GeigerIJRR13Vision, RussellIJCV08labelme, Bileschi07CBCL) for the task of semantic segmentation based on CNNs, i.e. the use of the combined data significantly boosts the performance obtained when using the real-world data alone. The future work that we foresee given these results is pointed out in Sect. 12.6, together with the conclusions of the chapter.

12.2 Related Work

The generation of semantic segmentation datasets with pixel-level annotations is costly in terms of effort and money, factors that are currently slowing down the development of new large-scale collections like ImageNet [265]. Despite these factors, the community has invested great effort to create datasets such as the NYU-Depth V2

[441] (more than 1,449 images densely labeled), the PASCAL-Context Dataset [336] (10,103 images densely labeled over 540 categories), and MS COCO [303] (more than 300,000 images with annotations for 80 object categories). These datasets have definitely contributed to boost research on semantic segmentation of indoor scenes and also on common objects, but they are not suitable for more specific tasks such as those involved in autonomous navigation scenarios.

When semantic segmentation is seen in the context of autonomous vehicles, we find that the amount and variety of annotated images of urban scenarios is much lower in terms of total number of labeled pixels, number of classes and instances. A good example is the CamVid [53] dataset, which consists of a set of monocular images taken in Cambridge, UK. However, only 701 images contain pixel-level annotations over a total of 32 categories (combining objects and architectural scenes), although usually only the 11 largest categories are used. Similarly, Daimler Urban Segmentation dataset [413] contains 500 fully labeled monochrome frames for five categories. The more recent GeigerIJRR13Vision benchmark suite [186] has provided a large amount of images of urban scenes from Karlsruhe, Germany, with ground truth data for several tasks. However, it only contains a total of 430 labeled images for semantic segmentation. A common limitation of the aforementioned datasets is the bias introduced by the acquisition of images in a specific city. The RussellIJCV08labelme project [406], later refined by [397], corrects this by offering around 1,000 fully annotated images of urban environments around the world and more than 3,000 images with partial (noisy) annotations.

A larger dataset is the Bileschi07CBCL StreetScenes [37], containing 3,547 images of the streets of Chicago over nine classes with noisy annotations. This dataset has been enhanced in [397], by improving the quality of the annotations and adding extra classes. To date, the largest dataset for semantic segmentation is CordtsFDV15CityScapes [98], which consists of a collection of images acquired in 50 cities around Germany, Switzerland, and France in different seasons, and having 5,000 images with fine annotations and 20,000 with coarse annotations over a total of 30 classes. However, the cost of scaling this sort of project would require a prohibitive economic investment in order to capture images from a larger variety of countries, in different seasons and traffic conditions. For these reasons, a promising alternative proposed in this work is to use synthetic imagery that simulate real urban scenes in a vast variety of conditions and produce the appropriate annotations.

The use of synthetic data has increased considerably in recent years within the computer vision community for several problems. For instance, in [262], the authors used a virtual world to evaluate the performance of image features under certain types of changes. In the area of object detection, similar approaches have been proposed by different groups [229, 325, 364, 467], making use of CAD models, virtual worlds, and studying topics such as the impact of a realistic world on the final accuracy of detectors and the importance of domain adaptation. Synthetic data has also been used for pose estimation [19, 357] to compensate for the lack of precise pose annotations of objects. The problem of semantic segmentation has also begun to benefit from this trend, with the creation of virtual scenes to perform segmentation of indoor environments [221, 222, 358]. Recently, virtual worlds have also debuted in outdoor

scenes for semantic segmentation [175, 386, 396] and related scene understanding problems, such as optical flow, scene flow, and depth estimation [328].

In this chapter, we show how state-of-the-art virtual worlds can be exploited in combination of simple DA methods to produce neural network models to operate in real environments for the task of semantic segmentation of driving scenes. We show that after adaptation these models are able to outperform state-of-the-art approaches trained on real data. To this end, we introduce a new and powerful synthetic datasets of urban scenes, which we call SYNTHIA. This dataset is a large collection of images with high variability due to simulated seasonal changes, realistic variations in illumination, textures, pose of dynamic objects, and camera viewpoints.

12.3 The SYNTHIA Dataset

The SYNTHetic collection of Imagery and Annotations (SYNTHIA) has been generated with the purpose of aiding scene understanding related problems in the context of driving scenarios, with special emphasis in semantic segmentation and instance semantic segmentation. It contains enough information to be useful in additional ADAS and AD-related tasks, such as object recognition, place identification and change detection, among others.

SYNTHIA consists of realistic frames (high fidelity with reality), rendered from a virtual city and comes with precise pixel-level semantic annotations at the level of object instances, i.e. individual object can be uniquely identified. The categories included in SYNTHIA are increasing over time due to the dynamic nature of the project, which is in continuous improvement, but the initial core contained 13 classes: sky, building, road, sidewalk, fence, vegetation, lane-marking, pole, car, traffic signs, pedestrians, cyclists, and miscellaneous (see Fig. 12.1). Frames are acquired from multiple viewpoints, and each of the frames also contains an associated depth map and information about the camera pose in global coordinates.

The Virtual World Generator. SYNTHIA has been generated by rendering a virtual city created with the Unity development platform [478]. This city includes the most important elements present on driving environments, such as street blocks, highways, rural areas, shops, parks and gardens, general vegetation, variety of pavements, lane markings, traffic signs, lamp poles, and people, among others. The virtual environment allows us to freely place any of these elements in the scene and to generate its semantic annotations without additional effort. This enables the creation of new and diverse cities as a simple combination of basic blocks. The basic properties of these blocks, such as textures, colors, and shapes can be easily changed to produce new looks and to enhance the visual variety of the data.

The city is populated with realistic models of cars, trucks, trains, vans, pedestrians, cyclists, etc. (see Fig. 12.3). In order to extend visual variability, some of these models are modified to generate new and distinctive versions.



Fig. 12.4 The same area captured in different seasons and light conditions. *Top-left, fall; top-right, winter; bottom-left, spring; bottom-right, summer*

We have defined suitable material coefficients for each of the surfaces of the city in order to produce realistic outcomes that look as similar as possible to real data. However, our intention is to maintain a balance between the cost of creating realistic images and the gain that this element brings to the learning process. The main point behind this work is to highlight that in terms of learning models for semantic segmentation, a cost-effective approach seems to be the one that uses just realistic enough images in combination with DA techniques. This philosophy leaves the generation of photo-realistic images as a less effective approach in terms of costs and learning gains.

Our virtual world includes four different seasons with drastic change of appearance, with snow during winter, blooming flowers during spring, etc., (see Fig. 12.4). Moreover, a dynamic illumination engine serves to produce different illumination conditions, to simulate different moments of the day, including sunny and cloudy days, rain, fog, and dusk. Shadows caused by clouds and other objects are dynamically cast on the scene, adding additional realism.

We would like to highlight the potential of this virtual world in terms of extension capabilities. New parts of the cities are constantly being extended, by adding existing blocks in different setups and additional ground truth can be produced almost effortlessly. Extending the number of classes of the city is also a simple task which consists of assigning a new id to objects. In this way, we can generate a broad variety of urban scenarios and situations, which we believe is very useful to help modern classifiers based on deep learning.

From our virtual city we have generated two complementary sets of images, referred to as SYNTHIA-RAND² and SYNTHIA-SEQS.

SYNTHIA-RAND. It consists of random images with no temporal consistency, created to maximize the visual variability of each image including a high density of objects. It contains 20,000 frames of the city taken from a virtual array of cameras moving randomly through the city, with its height limited to the range [1.5, 10 m] from the ground. In each of the camera poses, several frames are acquired changing the type of dynamic objects present in that part of the scene along with the illumination, season, weather conditions of the scene, and the textures of road and sidewalks. We enforce that the separation between camera positions is at least of 10 m in order to improve visual variability. This collection is oriented to serve as training data for semantic segmentation methods based on CNNs.

SYNTHIA-SEQS. It consists of video sequences, i.e. with temporal consistency, simulating different traffic situations such as traffic jams, joining a roundabout, going through a tunnel, highway traffic, etc. It is currently composed by more than eight different sequences, also referred to as “scripts.” Each script captures one or various traffic situations, from regular driving to traffic jams, including very critical situations such as joining a roundabout, dealing with objects blocking transit, traffic in highways, and many others. Each script takes place in one or several of the environments present in SYNTHIA, which are highway, town, and city center. Each traffic situation is simulated under different weather and seasonal conditions, from spring to winter; from sunny days to heavy rain. Each of these frames is captured by eight cameras, forming an omni-directional stereo-rig. In total, the combination of all these factors leads to more than 300,000 frames that are available for training.

Our virtual acquisition platform consists of two omni-cameras separated by a baseline $B = 0.8$ m in the x-axis. Each of these omni-cameras consists of four monocular cameras with a common center and orientations varying every 90° , as depicted in Fig. 12.5. Since all cameras have a field of view of 100° the visual overlapping serves to create an omni-directional view on demand, as shown in Fig. 12.5. Each of these cameras also has a virtual depth sensor associated, which works in a range from 1.5 to 600 m and is perfectly aligned with the camera center, resolution and field of view (Fig. 12.5, bottom). The virtual vehicle moves through the city interacting with dynamic objects such as pedestrians and cyclists that present dynamic behavior. This interaction produces changes in the trajectory and speed of the vehicle and leads to variations of each of the individual video sequences providing data to exploit spatio-temporal constraints.

²Here we refer to the new SYNTHIA-RAND subset, extended after the CVPR version.

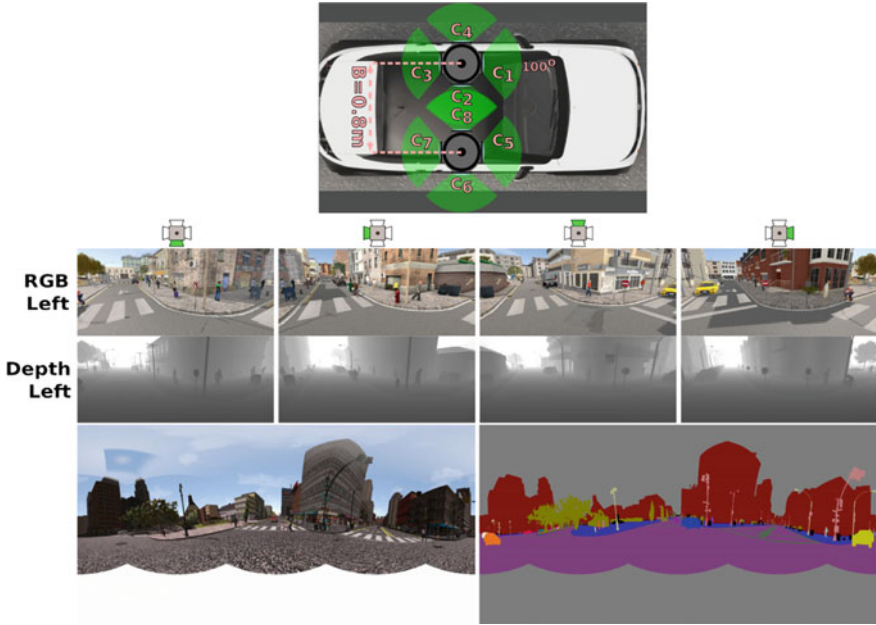


Fig. 12.5 *Top*, virtual car setup used for acquisition. Two multi-cameras with four monocular cameras are used. The baseline between the cameras is 0.8m and the FOV of the cameras is 100 deg. *Bottom*, One shot example: the four views from the *left* multi-camera with its associated depth maps and the resulting 360 deg panorama with its semantic ground truth

12.4 Semantic Segmentation and Synthetic Images

We first define a simple but competitive deep Convolutional Neural Network (CNN) for the task of semantic segmentation of urban scenes, following the description of [397]. This architecture, referred as Target-Net (T-Net) is more suitable for its application to urban scenes due to its reduced number of parameters. As a reference, we also consider the Fully convolutional networks (FCN) [307], a state-of-the-art architecture for general semantic segmentation. Finally, we describe the strategy used to deal with the synthetic (virtual data) and the real domain during the training stage.

T-Net [397] architecture. along with its associated training procedure is drawn from [397], due to its good performance and ease of training. Similar architectures have proven to be very effective in terms of accuracy and efficiency for segmentation of general objects [346] and urban scenes [24]. Figure 12.6 shows a graphical schema of T-Net. The architecture is based on a combination of contraction, expansion blocks and a soft-max classifier. Contraction blocks consist of convolutions, batch normalization, ReLU, and max-pooling with indices storage. Expansion blocks consist of an

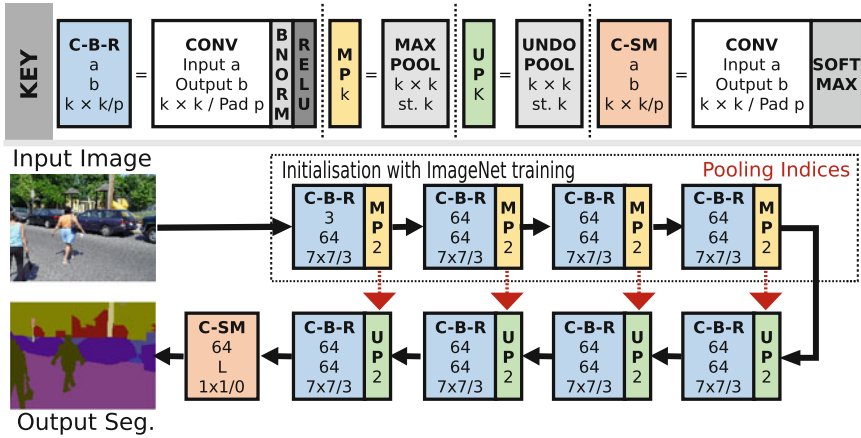


Fig. 12.6 Graphical schema of the semantic segmentation network consisting of a set of contraction (yellow) expansion (green) blocks, where a and b stand for the input and output number of channels, k is the kernel size and p the padding size, respectively. For pooling, $st.$ stands for stride

unpooling of the blob using the pre-stored indices, convolution, batch normalization and ReLU.

FCN [307] architecture is an extension of VGG-16 [443] with deconvolution modules. Different from T-Net, FCN does not use batch normalization and its up-sampling scheme is based on deconvolutions and mixing information across layers.

We use weighted cross-entropy (WCE) as a loss function for both architectures. WCE re-scales the importance of each class, $l \in [1, \dots, L]$, according to its inverse frequency $f^l(\mathcal{X})^{-1}$ in the training data set \mathcal{X} , i.e.,

$$\text{Loss}_{\text{WCE}}(x^n, y^n) = - \sum_{ijl}^{HWL} \omega(y_{ijl}^n) y_{ijl}^n \log(\mathcal{F}(x^n, \theta))_{ijl}, \quad (12.1)$$

where \mathcal{F} refers to the network, x^n , y^n stand for the n -th training image and ground truth image, respectively. This helps prevent problems due to class imbalance. During training both contraction and expansion blocks are randomly initialized following the method of He et al. [231]. Input data is normalized to produce zero-mean images re-scaled in the range $[-1, 1]$ for faster convergence. Networks are trained end-to-end using KingmaICLR15Adam [270] since the learning rates are automatically adjusted. Using KingmaICLR15Adam leads the network to converge in a couple of hundred iterations, speeding up the training procedure considerably.

Training on Real and Synthetic Data. The aim of this work is to show that the use of synthetic data helps to improve semantic segmentation results on real imagery. There exist several ways to exploit synthetic data for this purpose. A trivial option would be to use the synthetic data alone for training a model and then apply it on

real images. However, due to domain shift [467, 503] this approach does not usually perform well. An alternative is to train a model on the vast amount of synthetic images and afterwards fine-tuning it on a reduced set of real images. As shown later, this leads to better results, since the statistics of the real domain are considered during the second stage of training [364].

However, here we employ a more effective approach referred to as Balanced Gradient Contribution (BGC), which was first introduced in [397]. During the training stage data from both, virtual and real domain is exploited in a controlled fashion in order to maximize accuracy and generalization capabilities. The severe statistical difference between the domains usually induces a large variance in gradients for a sequence of mini-batches. Data from the virtual domain is more stable and suitable for dynamic objects, but less informative for the architectural classes. Data from the real domain is highly informative for the architectural classes, but usually insufficient. To deal with these aspects we propose to compute search directions in a controlled fashion, using the directions proposed by the virtual domain under a controlled perturbation given by the real domain as shown in Eq. (12.2).

$$\text{Loss}_{\text{BGC}}(\mathcal{X}, \mathcal{Y}) = \text{Loss}_{\text{WCE}}(\mathcal{X}^V, \mathcal{Y}^V) + \lambda \text{Loss}_{\text{WCE}}(\mathcal{X}^R, \mathcal{Y}^R), \quad (12.2)$$

where \mathcal{X}, \mathcal{Y} stand for a subset of samples and their associated labels, drawn from the virtual (V) or real (R) domains. This procedure can be seen as the addition of a very informative regularizer controlled by the parameter λ , but an analogous effect can be achieved by generating mini-batches containing a carefully chosen proportion of images from each domain, such that $|\mathcal{X}^V| \gg |\mathcal{X}^R|$. In Sect. 12.5 we show that extending real data with virtual images using this technique leads to a systematic boost in segmentation accuracy.

12.5 Experimental Evaluation












We present the evaluation of the CNNs for semantic segmentation described in Sect. 12.4, training and evaluating on several state-of-the-art datasets of driving scenes. We test how the new SYNTHIA dataset can be useful both on its own and along with real images to produce accurate segmentation results. For the following experiments, we have made use of the 20,000 images of the new SYNTHIA-RAND collection to favor visual variability while using a moderate number of images.

Validation Datasets. We selected publicly available urban datasets to study the benefits of SYNTHIA. Table 12.1 shows the different datasets along with the number of training and test images used in our experiments. It is worth highlighting the differences between these datasets. Each of them has been acquired in a different city or cities. CamVid and GeigerIJRR13Vision datasets have high quality labels and low complexity in terms of variations and atypical scenes. RussellIJCV08labelme is very challenging, since it contains images from different cities with several viewpoints. It

Table 12.1 Driving scenes sets for semantic segmentation

Dataset	# Frames	# Training	# Test
CamVid [53, 54]	701	300	401
GeigerIJRR13Vision [186, 281, 395, 397]	547	200	347
RussellIJCV08labelme [397, 406]	942	200	742
Bileschi07CBCL StreetScenes [37, 397]	3547	200	3347
SYNTHIA-RAND	20,000	20,000	0

Table 12.2 Results of training a T-Net and a FCN on SYNTHIA- RAND and evaluating it on state-of-the-art datasets of driving scenes

Method	Training	Validation												Per-class	Global
T-Net [397]	SYNTHIA- RAND	CamVid	96.2	77.8	88.4	78.2	0.4	81.8	23.4	87.0	23.1	85.2	51.1	63.0	81.6
	SYNTHIA- RAND	GeigerIJRR13Vision	89.5	72.8	49.8	56.5	0.0	83.8	35.6	46.9	14.7	66.5	24.6	49.2	66.9
	SYNTHIA- RAND	RussellIJCV08labelme	73.9	74.1	56.4	33.8	0.4	78.4	31.5	86.1	15.1	66.8	44.9	51.0	68.1
	SYNTHIA- RAND	Bileschi07CBCL	68.5	74.0	72.7	29.9	0.4	74.0	40.3	82.5	22.5	52.5	56.8	52.2	71.0
	SYNTHIA- RAND	CamVid	92.9	86.4	79.3	86.1	0.1	84.4	21.3	97.6	17.8	78.2	39.0	62.1	81.3
FCN [307]	SYNTHIA- RAND	CamVid	92.9	86.4	79.3	86.1	0.1	84.4	21.3	97.6	17.8	78.2	39.0	62.1	81.3
	SYNTHIA- RAND	GeigerIJRR13Vision	84.3	79.0	25.2	60.5	0.0	78.2	41.9	68.7	9.5	88.3	20.1	50.5	63.9
	SYNTHIA- RAND	RussellIJCV08labelme	64.7	82.5	37.3	63.6	0.0	90.3	30.9	79.9	15.1	74.9	24.8	51.3	69.5
	SYNTHIA- RAND	Bileschi07CBCL	71.9	72.3	71.4	39.2	0.0	86.3	45.6	79.7	19.8	57.2	56.3	54.5	72.9












has been annotated by several users and contains images with partial or noisy annotations. Bileschi07CBCL is also challenging, containing many noisy, semi-supervised annotations [397]. Each dataset is split to include a large number of validation images, keeping enough images for the training.

Analysis of Results. The following experiments have been carried out by training two types of CNNs, prioritizing a good average per-class accuracy in order to maximize recognition capabilities. All images are resized to a common resolution of 512×256 . This is done to speed-up the training process and save memory. However, it has the disadvantage of decreasing the recognition of certain textures and could make harder to recognize small categories such as traffic signs and poles.

In our first experiment, we evaluate the capability of SYNTHIA- RAND in terms of the generalization of the trained models on state-of-the-art datasets. To this end, we report in Table 12.2 the accuracy (%) of T-Net and FCN for each of the 11 classes along with their average per-class and global accuracies for each of the testing sets.

The networks trained on just synthetic data produce good results recognizing roads, buildings, cars, and pedestrians in the presented datasets. Moreover, sidewalks and vegetation classes are fairly well recognized in CamVid and GeigerIJRR13Vision, probably due to their homogeneity. The high accuracy at segmenting buildings, cars, and pedestrians in RussellIJCV08labelme—one of the most challenging datasets due to the large variety of viewpoints—is a proof of the high quality of SYNTHIA. Notice also that FCN performs better than T-Net for many of the classes due to the higher capacity of the model, although in practice FCN has the

Table 12.3 Evaluation of training a T-Net and FCN on different configurations, using real and virtual images, BGC and fine-tuning. We take as references the results of each architecture when trained just with real data of a given domain (**green**). Then accuracy improvements are shown in **blue** if they are positive or in **red**, otherwise

Method	Training	Testing												per-class	global
T-Net [397]	Camvid	CamVid	98.1	67.0	87.8	27.3	11.3	90.8	23.7	92.1	20.6	42.8	24.5	53.3	76.0
T-Net [397]	SYNTHIA-RAND	CamVid	96.2	77.8	88.4	78.2	0.4	81.8	23.4	87.0	23.1	85.2	51.1	63.0 (9.7)	81.6 (5.6)
T-Net FT	Camvid	CamVid	91.8	67.5	79.4	64.8	0.2	85.3	25.7	89.9	27.7	93.4	45.5	61.0 (7.7)	74.6 (-1.4)
T-Net BGC	Camvid + SYNTHIA-RAND	CamVid	98.9	77.7	95.2	73.0	8.3	95.1	34.8	94.0	35.1	79.6	41.4	66.7 (13.4)	85.7 (9.7)
FCN [307]	Camvid	CamVid	99.1	80.1	97.4	64.2	18.2	95.7	49.6	94.3	41.0	72.0	35.4	67.9	86.9
FCN [307]	SYNTHIA-RAND	CamVid	92.9	86.4	79.3	86.1	0.1	84.4	21.3	97.6	17.8	78.2	39.0	62.1 (-5.8)	81.3 (-5.6)
FCN FT	Camvid	CamVid	97.9	72.9	97.0	61.2	5.2	97.1	30.6	96.2	47.7	63.1	79.9	68.1 (0.2)	84.3 (-2.6)
FCN BGC	Camvid + SYNTHIA-RAND	CamVid	97.6	86.4	96.7	85.8	19.1	98.0	35.8	96.0	45.3	76.4	90.9	75.3 (7.4)	90.2 (3.3)
T-Net [397]	GeigerJRR13Vision	GeigerJRR13Vision	84.1	83.0	84.1	57.4	43.8	87.1	17.1	83.3	4.0	2.3	0.8	49.7	80.2
T-Net [397]	SYNTHIA-RAND	GeigerJRR13Vision	89.5	72.8	49.8	56.5	0.0	83.8	35.6	46.9	14.7	66.5	24.6	49.2 (-0.6)	66.9 (-13.3)
T-Net FT	GeigerJRR13Vision	GeigerJRR13Vision	81.4	75.6	78.6	43.8	0.7	88.1	23.2	90.6	11.4	48.5	18.2	50.9 (1.2)	76.4 (-3.8)
T-Net BGC	GeigerJRR13Vision + SYNTHIA-RAND	GeigerJRR13Vision	90.7	81.4	85.5	66.1	33.3	90.3	34.3	88.0	15.7	33.1	26.5	58.6 (8.9)	82.6 (2.4)
FCN [307]	GeigerJRR13Vision	GeigerJRR13Vision	82.7	88.6	89.1	75.0	50.5	92.7	25.9	87.8	10.2	3.3	3.4	55.4	86.0
FCN [307]	SYNTHIA-RAND	GeigerJRR13Vision	84.3	79.0	25.2	60.5	0.0	78.2	41.9	68.7	9.5	88.3	20.1	50.5 (-4.9)	63.9 (-22.1)
FCN FT	GeigerJRR13Vision	GeigerJRR13Vision	86.3	79.7	86.3	42.0	5.4	85.3	20.2	89.1	8.5	52.6	18.8	52.2 (-3.2)	78.0 (-8.0)
FCN BGC	GeigerJRR13Vision + SYNTHIA-RAND	GeigerJRR13Vision	87.7	84.3	87.6	79.1	41.7	94.8	52.5	88.1	21.1	30.9	28.4	63.3 (7.9)	86.2 (0.2)
T-Net [397]	RussellIJCV08labelme	RussellIJCV08labelme	88.7	86.6	76.4	34.2	0.1	63.3	6.1	60.9	2.7	48.1	25.7	44.8	75.6
T-Net [397]	SYNTHIA-RAND	RussellIJCV08labelme	73.9	74.1	56.4	33.8	0.4	78.4	31.5	86.1	15.1	66.8	44.9	51.0 (6.2)	68.1 (-7.5)
T-Net FT	RussellIJCV08labelme	RussellIJCV08labelme	79.6	84.4	65.5	50.6	0.0	80.2	9.5	75.8	6.6	82.8	49.1	53.1 (8.3)	75.6 (0.0)
T-Net BGC	RussellIJCV08labelme + SYNTHIA-RAND	RussellIJCV08labelme	86.6	83.4	71.4	36.6	0.0	79.3	13.9	81.7	8.4	71.2	57.4	53.6 (8.8)	76.5 (0.9)
FCN [307]	RussellIJCV08labelme	RussellIJCV08labelme	93.6	85.6	63.2	65.9	0.2	72.8	16.9	63.0	22.3	67.5	48.4	54.5	77.2
FCN [307]	SYNTHIA-RAND	RussellIJCV08labelme	64.7	82.5	37.3	63.6	0.0	90.3	30.9	79.9	15.1	74.9	24.8	51.3 (-3.2)	69.5 (-7.7)
FCN FT	RussellIJCV08labelme	RussellIJCV08labelme	83.0	90.0	69.4	61.1	0.0	83.5	20.8	76.4	10.3	77.5	33.1	55.0 (0.5)	80.1 (2.9)
FCN BGC	RussellIJCV08labelme + SYNTHIA-RAND	RussellIJCV08labelme	87.2	90.5	67.2	69.6	0.3	86.4	21.7	82.8	10.9	82.1	38.1	57.9 (3.4)	81.9 (4.7)
T-Net [397]	Bileschi07CBCL	Bileschi07CBCL	88.5	72.7	87.9	34.3	4.9	82.4	45.5	77.9	8.5	8.5	23.2	48.6	77.3
T-Net [397]	SYNTHIA-RAND	Bileschi07CBCL	68.5	74.0	72.7	29.9	0.4	74.0	40.3	82.5	22.5	52.5	56.8	52.2 (3.6)	71.0 (-6.2)
T-Net FT	Bileschi07CBCL	Bileschi07CBCL	81.4	69.8	80.4	47.6	0.4	85.0	53.1	86.8	15.2	45.1	52.0	56.1 (7.5)	76.4 (-0.9)
T-Net BGC	Bileschi07CBCL + SYNTHIA-RAND	Bileschi07CBCL	81.6	76.5	81.7	49.4	1.4	81.5	47.4	85.2	22.8	56.9	55.7	58.2 (9.6)	78.1 (0.8)
FCN [307]	Bileschi07CBCL	Bileschi07CBCL	86.4	78.5	83.5	71.6	3.4	86.6	48.7	74.7	20.2	23.5	29.3	55.1	80.3
FCN [307]	SYNTHIA-RAND	Bileschi07CBCL	71.9	72.3	71.4	39.2	0.0	86.3	45.6	79.7	19.8	57.2	56.3	54.5 (-0.6)	72.9 (-7.4)
FCN FT	Bileschi07CBCL	Bileschi07CBCL	73.9	73.0	74.3	54.6	0.1	83.7	55.2	79.3	18.1	45.2	40.6	54.4 (-0.7)	74.5 (-5.8)
FCN BGC	Bileschi07CBCL + SYNTHIA-RAND	Bileschi07CBCL	78.7	80.2	85.8	59.0	1.1	85.0	53.1	89.1	46.4	59.3	43.7	62.0 (6.8)	81.9 (1.5)

disadvantage of being too large for embedded context such as autonomous driving. It is worth highlighting that the average per-class accuracy of the models trained with SYNTHIA is close or sometimes even higher than of the models trained on real data (see Table 12.3).

Our second experiment evaluates the true potential of SYNTHIA to boost CNN models trained on real data. To this end, we perform several tests combining data from SYNTHIA- RAND along with individual real datasets, as described in Sect. 12.4. We set our GBC method to use 10 images per batch, containing six images from the real domain and four from the synthetic one. These results are compared against using just real data coming from each respective training split and also against a fine-tuning model (i.e. taking models originally trained on SYNTHIA and fine-tune them to individual real domains). The outcome of this experiment is shown in Table 12.3. We show baselines results (training only with real data) highlighted in green. Im-

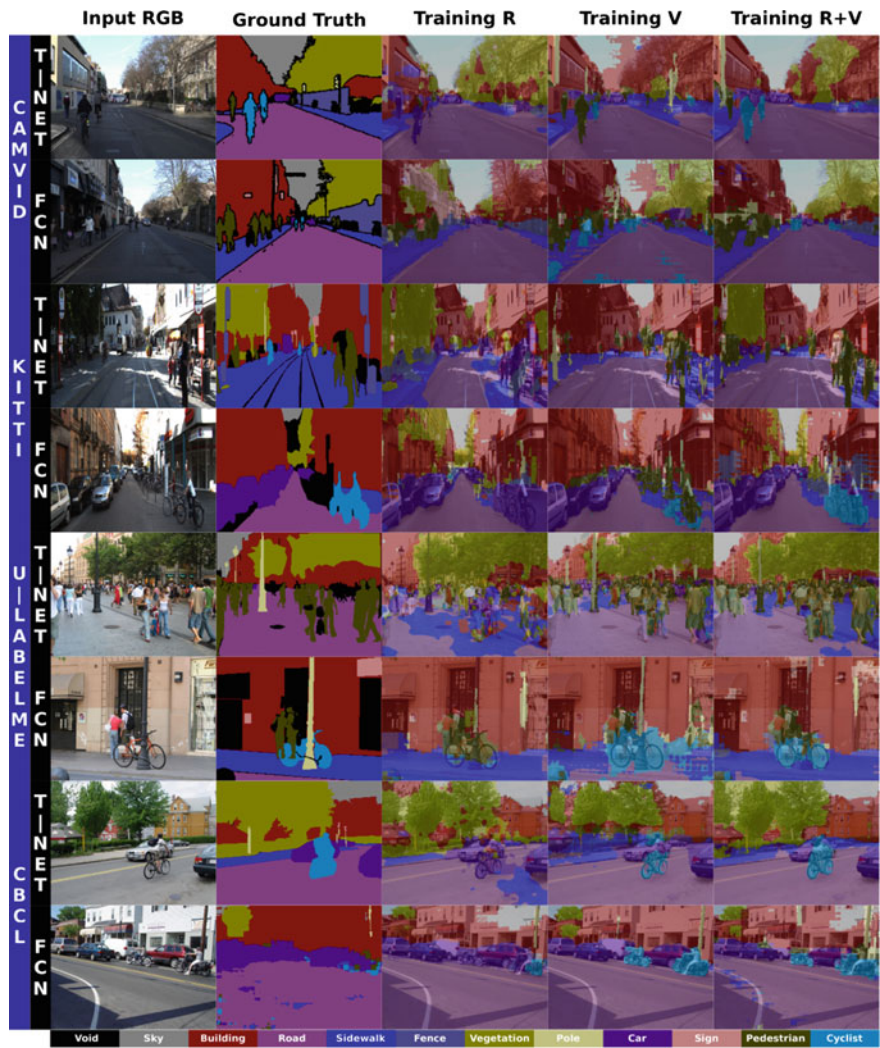


Fig. 12.7 Qualitative results for T-Net and FCN on different datasets. First column shows the **RGB** test images; second column is the **ground truth**; **Training R**, **Training V**, and **Training R+V** are results when training with the real dataset, with SYNTHIA and with the real and SYNTHIA- RAND collection, respectively. Including SYNTHIA for training considerably improves the results

provements with respect to the baselines are highlighted in blue if they are positive or in red if they are negative. Observe that, for all the datasets and architectures, the use of SYNTHIA and the proposed BGC domain adaptation method systematically outperform training just on real data and model fine-tuning for both average per-class and overall accuracy. There are improvements of more than 10 points (up to 13.4 points) in per-class accuracy. The classes that most benefit from the addition of syn-

thetic data are pedestrian, car, and cyclist (dynamic objects), which is due to the lack of enough instances of these classes in the original datasets. On the other hand, signs and poles are very hard to segment as a consequence of the low-resolution images.

Figure 12.7 shows qualitative results of the previous experiments. Observe how the training on synthetic data is good enough to recognize pedestrians, roads, cars, and some cyclists. Then the combination of real and synthetic data (right column) produces smooth and very accurate results for both objects and architectural elements, even predicting thin objects like poles. We consider the results of these experiments an important milestone for the use of synthetic data as the main information source for semantic segmentation.

12.6 Conclusions

We empirically showed how the combination of nonphoto-realistic synthetic data and simple domain adaptation can boost a critical scene understanding problem in driving scenarios, as semantic segmentation. To this end, we presented SYNTHIA, a new dataset for scene understanding related tasks, with major focus on semantic segmentation and instance segmentation. SYNTHIA is actively growing, and currently contains more than 320,000 synthetic images, when counting both, random snapshots, and video sequences of a virtual city. Images are generated simulating different seasons, weather, and illumination conditions from multiple viewpoints. Frames include pixel-level semantic annotations and depth. Our experiments in driving semantic segmentation showed that SYNTHIA is good enough to produce good segmentations by itself on state-of-the-art real datasets, and that its combination with real data dramatically boosts the accuracy of deep learning models. We believe that further research in SYNTHIA-like approaches is a must in order to bring new advances in scene understanding for autonomous vehicles.

Acknowledgements Authors want to thank Andrew Bagdanov for his help and proofreading and the next funding bodies: the Spanish MEC Project TRA2014-57088-C2-1-R, the Spanish DGT Project SPIP2014-01352, the People Programme (Marie Curie Actions) FP7/2007-2013 REA grant agreement no. 600388, and by the Agency of Competitiveness for Companies of the Government of Catalonia, ACCIO, the Generalitat de Catalunya Project 2014-SGR-1506, and to all the members of the SYNTHIA team.