sensors

MDPI

Article

Virtual to Real Adaptation of Pedestrian Detectors

Luca Ciampi *, Nicola Messina *, Fabrizio Falchi, Claudio Gennaro and Giuseppe Amato

Institute of Information Science and Technologies, National Research Council, Via G. Moruzzi 1, 56124 Pisa, Italy; fabrizio.falchi@isti.cnr.it (F.F.); claudio.gennaro@isti.cnr.it (C.G.); giuseppe.amato@isti.cnr.it (G.A.)

* Correspondence: luca.ciampi@isti.cnr.it (L.C.); nicola.messina@isti.cnr.it (N.M.)

Received: 12 August 2020; Accepted: 9 September 2020; Published: 14 September 2020

Abstract: Pedestrian detection through Computer Vision is a building block for a multitude of applications. Recently, there has been an increasing interest in convolutional neural network-based architectures to execute such a task. One of these supervised networks' critical goals is to generalize the knowledge learned during the training phase to new scenarios with different characteristics. A suitably labeled dataset is essential to achieve this purpose. The main problem is that manually annotating a dataset usually requires a lot of human effort, and it is costly. To this end, we introduce ViPeD (Virtual Pedestrian Dataset), a new synthetically generated set of images collected with the highly photo-realistic graphical engine of the video game GTA V (Grand Theft Auto V), where annotations are automatically acquired. However, when training solely on the synthetic dataset, the model experiences a Synthetic2Real domain shift leading to a performance drop when applied to real-world images. To mitigate this gap, we propose two different domain adaptation techniques suitable for the pedestrian detection task, but possibly applicable to general object detection. Experiments show that the network trained with ViPeD can generalize over unseen real-world scenarios better than the detector trained over real-world data, exploiting the variety of our synthetic dataset. Furthermore, we demonstrate that with our domain adaptation techniques, we can reduce the Synthetic2Real domain shift, making the two domains closer and obtaining a performance improvement when testing the network over the real-world images.

Keywords: pedestrian detection; domain adaptation; synthetic datasets; convolutional neural networks; deep learning

1. Introduction

A key task in many intelligent video surveillance systems is pedestrian detection, as it provides essential information for semantic understanding of video. Accurate detection of individual instances of pedestrians in images plays a vital role in a myriad of applications that can positively impact the quality of human life. They range from video surveillance [1,2], robotics, automotive [3,4] and assistive technologies to people with visual disabilities [5], just to name a few.

Convolutional neural networks-based methods [6] have recently demonstrated their superiority compared to the approaches relying on hand-crafted features. However, despite the recent advances, the pedestrian detection task remains a challenging active research area in Computer Vision. While there exist some large annotated generic datasets suitable for training these supervised learning networks, such as ImageNet [7] and MS COCO [8], in many real-world situations they are not enough. Hence, as a consequence, a model trained using these data usually experiences a drastic drop in performance when applied to another scenario at inference time.

The crux of Convolutional Neural Networks (CNNs) is that, to generalize well at inference time, they require a huge amount of diverse labeled data during the training phase, covering the widest

number of different scenarios. Since manually annotating new collections of images is expensive and requires a great human effort, a recently promising approach is to gather data from virtual world environments that mimics as much as possible all the characteristics of the real-world scenarios, and where the annotations can be acquired with a partially automated process. To this end, in this work, we provide ViPeD (Virtual Pedestrian Dataset), a new synthetic dataset generated with the highly photo-realistic graphical engine of the video game GTA V (Grand Theft Auto V) by Rockstar North, that extends the JTA (Joint Track Auto) dataset presented in [9].

The use of synthetic datasets based on 3D rendering to tackle the annotation problem is not new. Some notable examples are GTA5 [10] and SYNTHIA [11] for semantic segmentation. However, to the best of our knowledge, ViPeD is the first synthetic dataset suitable for the pedestrian detection task, which is annotated with bounding boxes locating the people's instances present in the scenes.

While synthetic data collections are very appealing, usually, when training solely on a synthetic dataset, the model does not generalize well to real-world data. This performance gap is due to the fact that the network learns from one domain (named training or source domain) and is then applied on another different domain (test or target domain), and is commonly referred as Domain Shift [12]. In this particular case, the source and the target domains are the synthetic and the real-world ones, respectively. Hence, we call this Domain Shift as Synthetic2Real.

In this paper, we propose two different Domain Adaptation (DA) methods to mitigate this Synthetic2Real Domain Shift, suitable for the pedestrian detection task, but possibly applicable to general object detection. The first one consists of training the model exploiting the synthetic data, and then, in a second step, fine-tuning it using the real-world images. Instead, the second one consists of an end-to-end training procedure in which we employ mixed batches containing both synthetic and real data.

First, we test the generalization capabilities of the detector over unseen scenarios. We show that we can obtain better or comparable results when training exploiting the synthetic data than when using the same model trained using only real-world images, just taking advantage of the variety of ViPeD. Secondly, we experiment with the two proposed domain adaptation techniques to boost the performance over specific real-world scenarios. We demonstrate that we can reduce the Synthetic2Real Domain Shift by bringing the two domains closer together, thus achieving better results.

Summarizing, the main contributions of this work are the followings:

- We introduce and make publicly available ViPeD, a new vast synthetic dataset suitable for the pedestrian detection task, generating the images using photo-realistic video game GTA V (Grand Theft Auto V), that extends the JTA (Joint Track Auto) dataset presented in [9].
- We present two supervised Domain Adaptation techniques to mitigate the Synthetic2Real Domain Shift existing between the synthetic and the real images.
- We conduct extensive experimentation on various real-world pedestrian detection datasets present in the literature. First, we test the detector's generalization capabilities, demonstrating that we achieve comparable or better results using synthetic data during the training phase rather than relying solely on the real-world images. Second, we experiment with the two proposed DA solutions to boost the performance over specific real-world scenarios, bringing the synthetic and the real domains closer, achieving better results.

Specifically, in this work, we extend our previous paper [13]. Compared to [13], we obtain better results, employing a new state-of-the-art detector that exhibits higher performance and introducing a new domain adaptation strategy. Furthermore, we carry out extensive experimentation over additional publicly available datasets, demonstrating the robustness of our approach over different real-world scenarios. The code, the models, and the dataset are made freely available at https://ciampluca.github.io/viped/.

2. Related Work

In this section, we review some relevant works about the object and pedestrian detection task. We also analyze some previous studies on DA, focusing on the Synthetic2Real domain shift.

2.1. Pedestrian Detection

Pedestrian detection is highly related to object detection. It deals with locating and recognizing instances of pedestrians' specific class, usually in images of urban environments, without taking into account group dynamics. We can subdivide approaches for the pedestrian detection task into two main research areas. The first class of detectors is based on handcrafted features, such as ICF (Integral Channel Features) [14–18]. These methods can usually rely on higher computational efficiency, at the cost of lower accuracy. On the other hand, more recently, Deep Neural Network approaches have been explored. For example, references [19–22] proposed some solutions based on the CNN networks [6] to detect pedestrians, even accounting for different scales.

Recent advances using CNNs were also possible thanks to the availability of many new datasets. Some of the most used in literature are Caltech [23], INRIA [24], MOT17Det [25], MOT19Det [26], and CityPersons [27]. In this work, we considered the latter three ones since they describe very heterogeneous video-surveillance scenarios, and they have proved to be enough challenging due to their high variability outlining most of the real-world problematic situations. The Caltech and the INRIA datasets are instead specifically collected for detecting pedestrians in self-driving contexts, a different scenario not considered in this paper.

2.2. Synthetic2Real Domain Adaptation

With the need for huge amounts of labeled data, synthetically-generated datasets have recently gained considerable interest. Some notable examples are GTA5 [10] and SYNTHIA [11] for semantic segmentation.

However, as already mentioned in the introduction, there is a non-negligible domain gap between the synthetic and the real worlds. Many techniques try to fill this gap, using both supervised and unsupervised approaches. An exhaustive survey about deep learning DA techniques is provided in [28]. For example, authors in [29] and in [30] proposed two unsupervised domain adaptation solutions for the counting task and the segmentation task, respectively, taking advantage of the output space. Authors in [9] created JTA (Joint Track Auto), a synthetic dataset taken from the highly photo-realistic video game GTA V. They demonstrated that it is possible to reach excellent results on tasks such as people tracking and pose estimation when validating on real data. In this work, we extend this dataset, making it suitable for the pedestrian detection task.

Authors in [31,32] have also focused on learning features from synthetic data for the pedestrian detection task. Still, they did not take into account deep learning approaches, exploring only traditional detection techniques. In [33], instead, the authors employed a synthetic dataset to train a CNN able to detect objects belonging to different classes in a video. This CNN is responsible only for the classification of the objects, while the detection of them relied on a background subtraction algorithm based on Gaussian Mixture Models (GMMs). This approach's performance over real-world scenarios was evaluated employing two pedestrian detection datasets, one of which, the 2D MOT 2015 [34], is an older version of the dataset we used to carry out our experiments. To the best of our knowledge, References [33,35] are the closest works to our. In particular, they also used GTA V as the source for the acquisition of the synthetic data, but they focus their efforts on the vehicle detection task.

3. ViPeD (Virtual Pedestrian Dataset)

In this section, we illustrate the motivations for using synthetic data, pointing out the main benefits and drawbacks. Then, we introduce and describe the construction of ViPeD, our synthetic collection of images exploited for training the pedestrian detector.

3.1. Training with Synthetic Datasets

As already pointed out in Section 1, the main drawback of CNN-based methods is that they hinge on large quantities of annotated data. Since they require ground truth labels for supervised learning, they may not generalize well to unseen images, especially when there is a large domain gap between the training (source) and the test (target) sets, such as different perspectives, illuminations, and object scales. This gap often severely hampers the application of CNN-based solutions to very large scale scenarios since annotating images for all the possible cases is an expensive operation, implying a considerable human-effort.

A possible solution to this problem is to create a vast and suitable dataset by collecting images from virtual world environments that resemble, as closely as possible, all the characteristics of the target real-world scenarios. Here, the main advantage is that the labels of the images can be acquired with a partially automated process, and so the data collection is significantly less costly. Consequently, it is possible to record a considerable amount of images covering a large number of different scenarios.

However, besides these positive aspects, there are some drawbacks to be considered. In particular, synthetic images' appearance is still significantly different from that observed in real-world images, even using current rendering techniques. Thus, the model trained solely on the synthetic dataset does not generalize to real-world data as one might expect due to the Synthetic2Real Domain Shift.

With the purpose of reducing the described domain shift, domain adaptation techniques can be exploited during the CNN-based networks' training phase. These methods try to make more similar the two data distributions, i.e., the distribution of the features belonging to the synthetic world and the one belonging to the real-world environment. Thus, it is possible to take advantage of the synthetic dataset's diversity, mitigating the underlying differences between the two domains.

3.2. ViPeD

ViPeD (Virtual Pedestrian Dataset) is a new synthetic dataset generated with the highly photo-realistic graphical engine of the video game GTA V (Grand Theft Auto V) by Rockstar North. It extends JTA (Joint Track Auto) dataset, presented in [9]. The dataset includes a total of about 500K images, extracted from 512 full-HD videos of different urban scenarios. These videos are organized into a training set (256 videos) and a test set (the remaining 256 videos).

While we can reuse the already existing JTA images, we need to generate suitable annotations for the pedestrian detection task. Indeed, the JTA dataset provides only skeletal information useful in the pose estimation and tracking tasks. In our scenario, instead, we are required to annotate each pedestrian with the four coordinates (x, y, w, h) delimiting its minimum enclosing bounding box. Hence, we employed the already available JTA images producing a new set of labels suitable for our task.

Estimating the precise bounding box surrounding each pedestrian instance can be tricky, as we do not have access to the underlying GTA game engine. Other works tried to overcome this problem by using some interesting work-around. For example, reference [35] extracted the semantic masks around each object in the scene and separated the instances by exploiting the depth information available through the depth buffer.

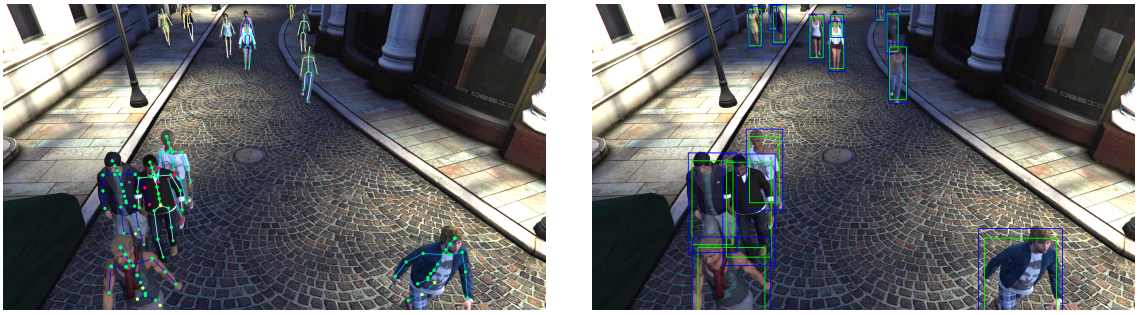
Our solution relies instead on the skeletal information already provided by the JTA annotations. Indeed, differently from [35], we deal with multiple instances of pedestrians in possibly highly crowded scenarios. In these cases, the depth information may be insufficient for distinguishing two different pedestrians, leading to possible severe bounding box estimation errors.

As a first approximation, we exploited the skeleton joints' positions in screen coordinates, directly available from the JTA annotations, for drawing the minimum bounding box enclosing all the skeleton joints (green boxes in Figure 1b). However, it can be noticed that the bounding boxes produced using this simple procedure are undersized compared to the full-sized pedestrian instance, as the skeleton always lays below the skin surface. We solved this issue by constructing a bigger bounding box (blue boxes in Figure 1b), obtained by estimating an amount of padding through a

simple heuristic. In particular, we estimated the height of a pedestrian mesh, denoted as h_m, from the height h_s of its skeleton, through the formula:

$$h_m = h_s + \frac{\alpha}{z} \tag{1}$$

where z is the distance of the pedestrian center of mass from the camera, and α is a parameter that depends on the camera projection matrix.



(**a**) (**b**)

Figure 1. (**a**) Pedestrians in the JTA (Joint Track Auto) dataset with their skeletons. (**b**) Examples of annotations in the ViPeD (Virtual Pedestrian Dataset) dataset; original bounding boxes are in green, while the sanitized ones are in blue.

The z value for each pedestrian is already included in the JTA annotations, while α is unknown since we can not access the camera parameters. Then, we evaluated α from Equation (1), estimating h_m for a small representative population of pedestrians. To this end, we isolated 50 random pedestrians from different scenarios, and we manually annotated them with their height in pixels units. At this point, it has been possible to recover the value of α from Equation (1) performing a simple linear regression to find the best fit.

The height padding depends basically only on the distance of the pedestrian from the camera. Instead, the width is also linked to the specific pedestrian pose. However, we found that we can ignore these pose-dependent effects while still obtaining an excellent estimate by deriving the pedestrian width w_m assuming no changes in the original bounding box aspect ratio. For this reason, we simply derived w_m from the computed h_m as follows:

$$w_m = h_m \frac{w_s}{h_s} = h_m r \tag{2}$$

where r is the aspect ratio of the bounding box enclosing the skeleton. Some examples of final estimated bounding boxes are shown in blue in Figure 1b.

We then assessed the quality of the produced bounding boxes. In Figure 2, we report a histogram depicting the distribution of the distances of the pedestrians from the camera. We observed that human annotators tend not to annotate pedestrians far than a certain amount from the camera in real-world datasets. We compute this distance limit by finding the minimum bounding box height, in pixels, occurring in human annotations of the MOT17Det [25] dataset, and seeing at what distance from the camera we reach this bounding box limit size on the JTA annotations. We concluded that human annotators do not include bounding boxes for pedestrians farther than 30–40 m from the camera. Then, to be consistent with real-world datasets on which we will validate our approach, we cleaned the produced bounding boxes by pruning all the ones enclosing pedestrians farther than 40 m.

In Figure 3, we report some examples of images of the ViPeD dataset together with the sanitized bounding boxes.

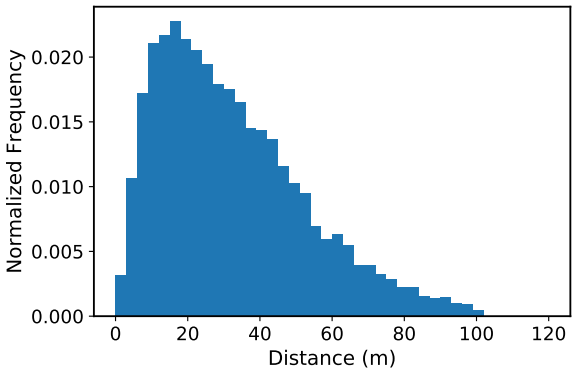


Figure 2. Histogram of distances between pedestrians and cameras.

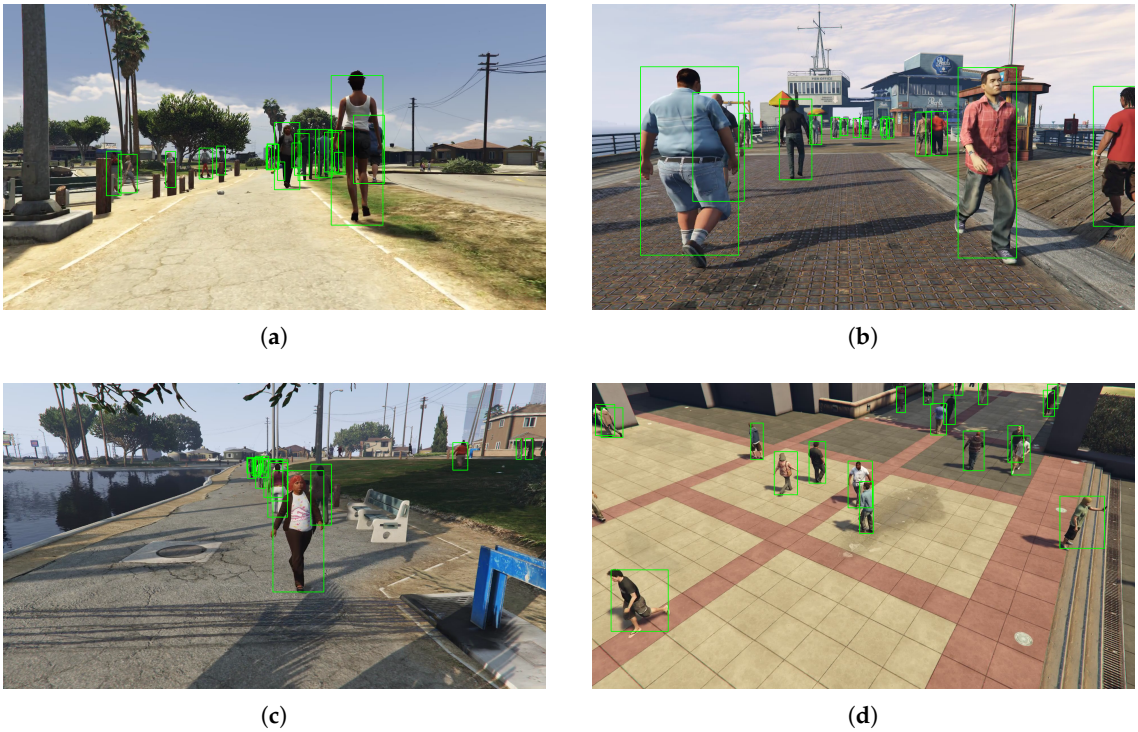


Figure 3. Examples of images of the ViPeD dataset together with the sanitized bounding boxes.

4. Domain Adaptation for Synthetic2Real Pedestrian Detection

In this section, we describe the object detector and the domain adaptation strategies we employed in this work. We exploited Faster R-CNN [36], a widely used state-of-the-art object detector that we briefly review in Section 4.1. We train this CNN using ViPeD, our collection of synthetic images automatically annotated, already outlined in Section 3.2. To mitigate the existing domain shift between these data and the real-world ones, we propose two domain adaptation techniques. The first one, described in Section 4.2, consists of training the detector with the synthetic data and then fine-tuning it exploiting the real-world images. In the second approach, described in Section 4.3, we employ instead another supervised technique, called Balanced Gradient Contribution (BGC) [11,37], where we mix the synthetic and the real-world data during the training phase. Figures 4 and 5 show an overview of the two solutions.

4.1. Faster R-CNN Object Detector

We exploit Faster-RCNN [36] as object detector architecture. In our previous work [13], we employed YOLOv3 [38], another state-of-the-art object detector. Here, our choice fell on

Faster R-CNN since it provides better performance. Furthermore, we do not consider pedestrian detection-specific solutions since the two proposed domain adaptation techniques can also be applied to other tasks, accounting for another class of objects different from the pedestrian one.

Faster R-CNN is a two-stage CNN-based algorithm composed of different networks: The backbone, the Region Proposal Network (RPN), and the Evaluation Network (EN). In the first stage, a CNN acts as a backbone, extracting the input image features. Starting from these features space, the RPN is in charge of generating region proposals that might contain objects. Briefly, RPN slices pre-defined region boxes (called anchors) over this space and ranks them, suggesting the ones most likely containing objects. Once RPN produces the Regions Of Interests (ROIs), they might be of different sizes. Since it is hard to work on features having different sizes, RPN reduces them into the same dimension using the Region of Interest Pooling algorithm. These fixed-size proposals are finally processed by the EN, responsible for classifying and locating the objects inside them. Then, given an input image, the EN network final outputs are class scores and bounding boxes coordinates.

Faster R-CNN is then a versatile and modular network in which it is possible to change the building blocks. Regarding the backbone, our choice fell on the ResNet-50 network, a lighter version of the very popular ResNet-101 network [39]. Indeed, Faster R-CNN with ResNet-50 can produce satisfactory detection results compared to the low computational resources and the time required during the training and test phases.

4.2. Domain Adaptation Using Real-World Fine-Tuning

The first proposed DA solution relies on a Transfer Learning (TL) strategy. As pointed out in [28], DA is a particular TL case that employs labeled data in one or more relevant source domains to execute the task in a target domain. In particular, the crucial point in this methodology consists of fine-tuning a previously trained model with the target-domain data.

We divide our fine-tuning methodology into two different steps.

In the first step, we consider as the baseline the Faster R-CNN detector described in the above section, having a ResNet-50 backbone pre-trained on the COCO dataset [8], a large collection of images depicting complex everyday scenes of ordinary objects in their natural context, divided into 80 different categories. Since this network is a generic object detector that can distinguish between many different classes of objects, we modify the EN building block to adapt the model to our purposes. In particular, we reduce the last fully connected layers of the detector to recognize and locate object instances belonging only to a specific category, i.e., the pedestrian category. Then, we train this modified Faster R-CNN-based network exploiting our synthetic images of the ViPeD, leaving all the model weights unfrozen during this phase so that the back-propagation algorithm can tune them.

Then, in the second step, we fine-tune this pre-trained model using real-world images as the target domain. So, in the end, the network will have processed both source and target images, memorizing in its weights information from both the domains. Figure 4 shows an overview of this approach. This method, looking at real images in this last step, is particularly useful for boosting the detector's performance on a specific real-world target scenario.

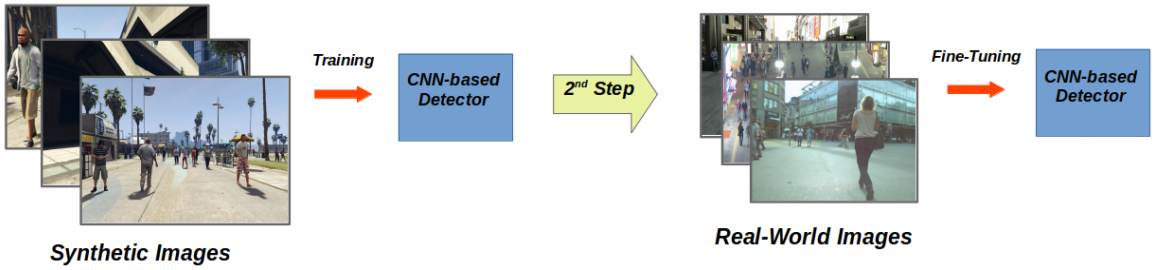


Figure 4. Overview of the first domain adaptation technique. In a first step, we train the detector using ViPeD, our synthetic collection of images. Then, in a second step, we fine-tune the network using real-world images.

4.3. Domain Adaptation using Balanced Gradient Contribution

The second DA approach is an end-to-end training, so it benefits from not relying on a two-step process like the previous one.

As in the previous solution, we start with the modified Faster R-CNN detector having the ResNet-50 backbone pre-trained on the COCO dataset. This time, we train the network using mixed batches, i.e., we employ batches containing synthetic and real-world images simultaneously, given a fixed mixing ratio. As explained in [37], the real-world data acts as a regularization term over the synthetic data training loss. In particular, we exploit batches composed of 2/3 of synthetic images and of 1/3 of real-world data. Thus, statistics from both domains are considered throughout the entire procedure, creating a more accurate model for both.

Again, during this phase, we leave all the network weights unfrozen so that the back-propagation algorithm can modify the network parameters accordingly. Consequently, we mitigate the Synthetic2Real Domain Shift straight in a single-step training. Figure 5 shows an overview of this approach.

In the experiments, we employed this technique for boosting the detector's performance on a precise target scenario, using batches composed of the synthetic data and the real-world images specific for the particular considered scenes. Besides, we exploited this solution also for achieving wide generalization capabilities, considering batches containing synthetic images and generic real-world images containing pedestrians.

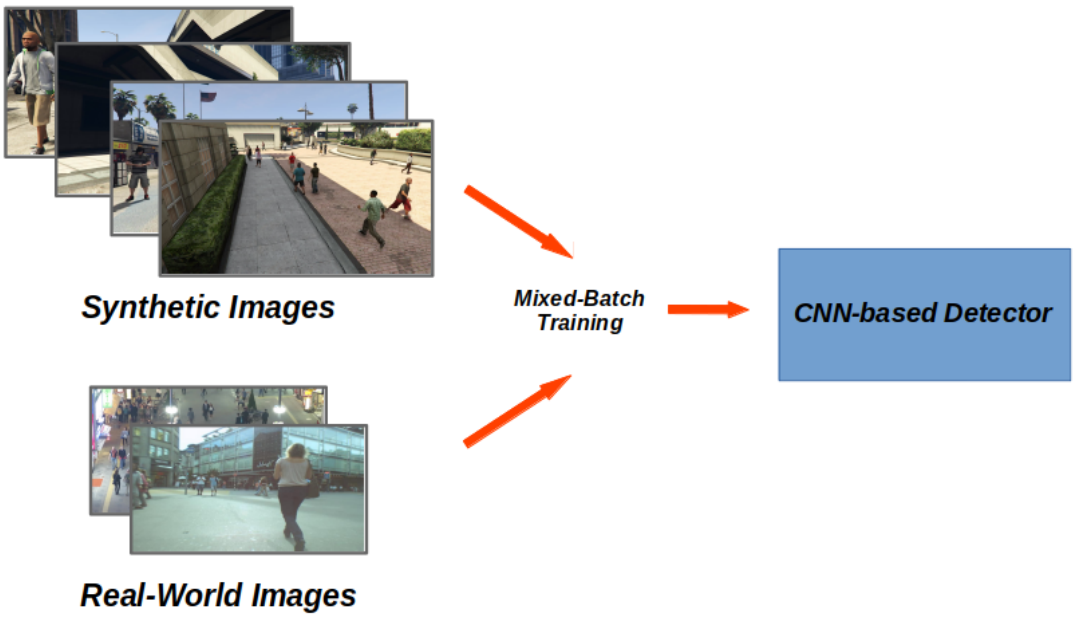


Figure 5. Overview of the second domain adaptation technique. We mitigate the Synthetic2Real domain shift in a single-step training procedure, employing mixed batches containing both synthetic and real images at the same time.

5. Experimental Evaluation

In this section, we briefly report some details about the real-world datasets exploited for the experiments. Then, we show and discuss the results concerning the generalization capabilities of our detector trained using ViPeD. Finally, we illustrate the performance of the two domain adaptation techniques over specific real-world scenarios.

5.1. Real-World Datasets

MOT17Det [25] and MOT19Det [26] are newly introduced datasets with manual annotations for pedestrian detection that are particularly suitable for surveillance applications. They comprise a

collection of challenging images (5316 and 8931, respectively) taken from multiple sequences with various crowded scenarios having different viewpoints, weather conditions, and camera motions. The authors provided training and test subsets, but they released only the ground-truth annotations belonging to the former. The performance metrics concerning the test subsets are instead available submitting results to their MOT Challenge website (https://motchallenge.net). The main peculiarity of MOT19Det compared to MOT17Det is the massive crowding of the collected scenarios.

CityPersons dataset [27] consists of a large and diverse set of stereo video sequences recorded in streets from different cities in Germany and neighboring countries. In particular, the authors provide 5000 images from 27 cities labeled with bounding boxes and divided across train/validation/test subsets. This dataset is more focused on self-driving applications, and images are collected from a moving car.

COCOPersons dataset is a split of the popular COCO dataset [8] comprising images collected in general contexts belonging to 80 categories. We filter these images considering only the ones belonging to the persons category. Hence, we obtain a new dataset of about 66,000 images containing at least one pedestrian instance.

5.2. Experiments

We evaluate the detection performances using the standard mean average precision (mAP) metric. In particular, we consider the detection proposals having a score confidence greater than 0.05. Then, we employ the COCO mAP [8] and the MOT AP metrics [25], fixing the IoU threshold to 0.5 and varying only the detection confidence threshold.

5.2.1. Testing Generalization Capabilities

To test the generalization capabilities, we train the detector on a source domain, and then we validate it on a different target domain. In particular, we train the model using a dataset, and then we test it on another one. In this way, we guarantee that the two distributions are different and not related.

In particular, we train the modified Faster R-CNN-based detector described in Section 4.1 using ViPeD. This procedure corresponds to the first step of the previously described domain adaptation solution (see Section 4.2). We evaluate this model testing it on the real-world datasets MOT17Det, MOT19Det and CityPersons, defining three validation subsets containing images not present in the training subset.

To form a solid baseline for this experiment, we train the same detector using every one of the three real-world datasets, and then we test them over the remaining two. We also report a further baseline considering the detector trained only on the real-world general-purpose COCO dataset, considering only the detections belonging to the person category.

We also experiment with the mixed-batch domain-adaptation approach explained in Section 4.3, using the same evaluation protocol as before. We exploit batches composed of 2/3 of ViPeD and by the remaining 1/3 of COCOPersons. We choose the latter as the real-world dataset since it depicts humans in highly heterogeneous scenarios, and it is not biased towards a specific application (e.g., autonomous driving). Again, we evaluate this model testing on all the three remaining real-world datasets.

We report the results in Table 1. Note that we omit results concerning a specific dataset if employed during the training phase, for a fair evaluation of the overall generalization capabilities.

Table 1. Evaluation of the generalization capabilities. The first section of the table reports results obtained training the detector with real-world data, while the latter is related to the model trained over synthetic images. ViPeD + Real refer to the mixed batch experiments with 2/3 ViPeD and 1/3 of COCOPersons. Results are evaluated using the COCO mAP. We report in bold the best results.

	Test Dataset		
Training Dataset	MOT17Det	MOT19Det	CityPersons
COCO	0.636	0.466	0.546
MOT17Det	-	0.605	**0.571**
MOT19Det	0.618	-	0.419
CityPersons	0.710	0.488	-
ViPeD	0.721	**0.629**	0.516
ViPeD + Real	**0.733**	0.582	0.546

In most cases, as we can see, our network performs better than those trained using only the manually annotated real-world datasets, taking advantage of the high variability and size of the ViPeD dataset. In particular, concerning the MOT17Det dataset, all our solutions trained with synthetic data outperform those trained with real ones. We obtain the best results using the mixed-batch approach. Considering the MOT19Det dataset, we achieve the best result in training the detector with ViPeD. CityPersons is the only dataset on which the algorithm maintains higher performances when trained with real-world data. In particular, the highest mAP on CityPersons is obtained when the detector is trained with the MOT17Det dataset. However, the mixed-batch approach achieves, in this case, results comparable with the baselines.

5.2.2. Testing Domain Adaptation Techniques over Specific Real-World Scenarios

To test how the two proposed domain adaptation techniques behave when considering specific target real-world scenarios, we consider the MOT17Det and MOT19Det real-world datasets.

Regarding the fine-tuning DA approach, we consider as training sets those proposed by the authors of [25,26], and we obtain the evaluation of our results over the test sets submitting them to the Mot Challenge website. For the mixed-batch DA solution, during the training phase, we inject in the same batch 2/3 of synthetic images from the ViPeD dataset and 1/3 of real-world images from the training subsets of the MOTDet17 or the MOT19Det dataset. Again, we validate our results by submitting them to the Mot Challenge website.

Tables 2 and 3 report the results for the two considered scenarios. We report our results together with the state-of-the-art approaches publicly released in the MOT Challenges (at the time of writing).

Table 2. Evaluation of the two Domain Adaptation (DA) techniques on the MOT17Det dataset. FT-DA (Fine Tuning DA) is the first proposed solution, while MB-DA (Mixed Batch DA) is the second one. Results are evaluated using the MOT mean average precision (mAP). We report in bold the best results.

Method	**MOT AP**
YTLAB [21]	0.89
KDNT [40]	0.89
ViPeD FT-DA (our)	**0.89**
ViPeD MB-DA (our)	0.87
ZIZOM [41]	0.81
SDP [20]	0.81
FRCNN [36]	0.72

Table 3. Evaluation of the two DA techniques on the MOT19Det dataset. FT-DA (Fine Tuning DA) is the first proposed solution, while MB-DA (Mixed Batch DA) is the second one. Results are evaluated using the MOT mAP. We report in bold the best results.

Method	MOT AP
SRK_ODESA	**0.81**
CVPR19_det	0.80
Aaron	0.79
PSdetect19	0.74
ViPeD FT-DA [13] (our)	0.80
ViPeD MB-DA [13] (our)	0.80

As we can see, the two DA approaches can mitigate the Synthetic2Real Domain Shift. In both datasets, we obtain an improvement in performance compared to the results in Table 1. It is also worth noting that we achieve competitive results in both scenarios compared to the state-of-the-art, reaching the first and the second places in the leader boards of the MOT17Det and MOT19Det challenges, respectively.

6. Conclusions

In this work, we addressed the pedestrian detection task by proposing a CNN-based solution trained using synthetically generated data. The choice of training a CNN using synthetic data is motivated by the fact that the network, to generalize well, requires a considerable amount of manually annotated images representing different scenarios. This procedure usually requires a significant human effort, and it is error-prone.

To this end, we introduced a synthetic dataset named ViPeD, containing a massive collection of images rendered from the highly photo-realistic video game GTA V developed by Rockstar North and a full set of precise bounding boxes annotations around all the visible pedestrians. To the best of our knowledge, it is the first synthetic dataset suitable for the pedestrian detection task.

Furthermore, we proposed two different Domain Adaptation techniques to mitigate the Synthetic2Real Domain Shift, which are suitable for the pedestrian detection task and possibly applicable to more general object detection tasks.

The experiments showed that, in most cases, the detector trained with the synthetic data can generalize better on unseen scenarios than the same algorithm trained using only the manually annotated real-world datasets. Moreover, the two proposed DA approaches can mitigate the underlying differences between the two worlds, obtaining a performance improvement on specific real-world scenarios.

In our opinion, the result of this work opens new perspectives to address the scalability of pedestrian and object detection methods for large physical systems with limited supervisory resources. Using our freely available model trained using ViPeD, future researchers will have at their disposal a detector able to localize instances of people over images belonging to a multitude of different scenarios and, therefore, a system robust to newly added sources of data. On the other hand, they will also have the possibility of further specializing the detector to work over new added real-world scenarios using our two domain adaptation techniques, obtaining an additional performance boost.

Author Contributions: Conceptualization, L.C., N.M., G.A., F.F. and C.G.; methodology, L.C., N.M., G.A., F.F. and C.G.; software, L.C. and N.M.; investigation, L.C., N.M., G.A., F.F. and C.G.; data curation, L.C. and N.M.; writing–original draft preparation, L.C., N.M.; writing–review and editing, L.C., N.M., G.A., F.F. and C.G.; visualization, L.C. and N.M.; supervision, G.A., F.F. and C.G.; project administration, G.A.; funding acquisition, G.A., F.F. and C.G. All authors have read and agreed to the published version of the manuscript.

Funding: This work was partially supported by H2020 project AI4EU under GA 825619 and by Automatic Data and documents Analysis to enhance human-based processes (ADA), CUP CIPE D55F17000290009.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Bilal, M.; Khan, A.; Khan, M.U.K.; Kyung, C.M. A low-complexity pedestrian detection framework for smart video surveillance systems. *IEEE Trans. Circuits Syst. Video Technol.* **2016**, *27*, 2260–2273. [CrossRef]
2. Varga, D.; Szirányi, T. Robust real-time pedestrian detection in surveillance videos. *J. Ambient. Intell. Humaniz. Comput.* **2017**, *8*, 79–85. [CrossRef]
3. Gavrila, D.M.; Munder, S. Multi-cue pedestrian detection and tracking from a moving vehicle. *Int. J. Comput. Vis.* **2007**, *73*, 41–59. [CrossRef]
4. Shashua, A.; Gdalyahu, Y.; Hayun, G. Pedestrian detection for driving assistance systems: Single-frame classification and system level performance. In Proceedings of the IEEE Intelligent Vehicles Symposium, Parma, Italy, 14–17 June 2004; pp. 1–6.
5. Tian, Y. RGB-D sensor-based computer vision assistive technology for visually impaired persons. *Computer Vision and Machine Learning with RGB-D Sensors*; Shao, L., Han, J., Kohli, P., Zhang, Z., Eds.; Springer: Heidelberg, Germany, 2014; Volume 20, pp. 173–194. [CrossRef]
6. LeCun, Y.; Bottou, L.; Bengio, Y.; Haffner, P. Gradient-based learning applied to document recognition. *Proc. IEEE* **1998**, *86*, 2278–2324. [CrossRef]
7. Deng, J.; Dong, W.; Socher, R.; Li, L.; Li, K.; Fei-Fei, L. ImageNet: A large-scale hierarchical image database. In Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition, Miami Beach, FL, USA, 22–24 June 2009; pp. 248–255. [CrossRef]
8. Lin, T.Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; Zitnick, C.L. Microsoft coco: Common objects in context. In Proceedings of the 13th European Conference on Computer Vision (ECCV) Zurich, Switzerland, 6–12 September 2014; pp. 740–755.
9. Fabbri, M.; Lanzi, F.; Calderara, S.; Palazzi, A.; Vezzani, R.; Cucchiara, R. Learning to Detect and Track Visible and Occluded Body Joints in a Virtual World. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 430–446.
10. Richter, S.R.; Vineet, V.; Roth, S.; Koltun, V. Playing for data: Ground truth from computer games. In Proceedings of the European Conference on Computer Vision (ECCV), Amsterdam, The Netherlands, 8–16 October 2016, pp. 102–118.
11. Ros, G.; Sellart, L.; Materzynska, J.; Vazquez, D.; Lopez, A.M. The SYNTHIA Dataset: A Large Collection of Synthetic Images for Semantic Segmentation of Urban Scenes. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 26 June–1 July 2016; pp. 3234–3243.
12. Torralba, A.; Efros, A.A. Unbiased look at dataset bias. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Colorado Springs, CO, USA, 20–25 June 2011; pp. 1521–1528.
13. Amato, G.; Ciampi, L.; Falchi, F.; Gennaro, C.; Messina, N. Learning Pedestrian Detection from Virtual Worlds. In Proceedings of the 20th International Conference of Image Analysis and Processing (ICIAP), Trento, Italy, 09–13 September 2019; pp. 302–312.
14. Benenson, R.; Omran, M.; Hosang, J.; Schiele, B. Ten Years of Pedestrian Detection, What Have We Learned? In Proceedings of the 13th European Conference on Computer Vision (ECCV) Workshops, Zurich, Switzerland, 06–12 September 2014; pp. 613–627.
15. Zhang, S.; Bauckhage, C.; Cremers, A.B. Informed Haar-like Features Improve Pedestrian Detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Columbus, OH, USA, 24–27 June 2014; pp. 947–954.
16. Zhang, S.; Benenson, R.; Schiele, B. Filtered channel features for pedestrian detection. In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015; pp. 1751–1760. [CrossRef]
17. Zhang, S.; Benenson, R.; Omran, M.; Hosang, J.; Schiele, B. How Far Are We From Solving Pedestrian Detection? In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 26 June–1 July; pp. 1259–1267.
18. Nam, W.; Dollar, P.; Han, J.H. Local Decorrelation For Improved Pedestrian Detection. In Proceedings of the 2014 Neural Information Processing Systems Conference (NIPS), Quebec, Canada, 8–13 December 2014; pp. 424–432.

19. Tian, Y.; Luo, P.; Wang, X.; Tang, X. Deep Learning Strong Parts for Pedestrian Detection. In Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 11–18 December 2015; pp. 1904–1912. [CrossRef]
20. Yang, F.; Choi, W.; Lin, Y. Exploit All the Layers: Fast and Accurate CNN Object Detector with Scale Dependent Pooling and Cascaded Rejection Classifiers. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 26 June–1 July 2016; pp. 2129–2137. [CrossRef]
21. Cai, Z.; Fan, Q.; Feris, R.S.; Vasconcelos, N. A Unified Multi-scale Deep Convolutional Neural Network for Fast Object Detection. In Proceedings of the European Conference on Computer Vision (ECCV), Amsterdam, The Netherlands, 8–16 October 2016; pp. 354–370.
22. Sermanet, P.; Kavukcuoglu, K.; Chintala, S.; Lecun, Y. Pedestrian Detection with Unsupervised Multi-stage Feature Learning. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Portland, OR, USA, 23–28 June 2013; pp. 3626–3633.
23. Dollar, P.; Wojek, C.; Schiele, B.; Perona, P. Pedestrian Detection: An Evaluation of the State of the Art. *IEEE Trans. Pattern Anal. Mach. Intell.* **2012**, *34*, 743–761. [CrossRef] [PubMed]
24. Dalal, N.; Triggs, B. Histograms of oriented gradients for human detection. In Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), San Diego, CA, USA, 20–26 June 2005; Volume 1, pp. 886–893.
25. Milan, A.; Leal-Taixé, L.; Reid, I.; Roth, S.; Schindler, K. MOT16: A benchmark for multi-object tracking. *arXiv* **2016**, *arXiv:1603.00831*.
26. Dendorfer, P.; Rezatofighi, H.; Milan, A.; Shi, J.; Cremers, D.; Reid, I.; Roth, S.; Schindler, K.; Leal-Taixe, L. CVPR19 Tracking and Detection Challenge: How crowded can it get? *arXiv* **2019**, *arXiv:1906.04567*.
27. Zhang, S.; Benenson, R.; Schiele, B. CityPersons: A Diverse Dataset for Pedestrian Detection. *arXiv* **2017**, *arXiv:1702.05693*.
28. Wang, M.; Deng, W. Deep visual domain adaptation: A survey. *Neurocomputing* **2018**, *312*, 135–153. [CrossRef]
29. Ciampi, L.; Santiago, C.; Costeira, J.P.; Gennaro, C.; Amato, G. Unsupervised Vehicle Counting via Multiple Camera Domain Adaptation. *arXiv* **2020**, *arXiv:2004.09251*.
30. Tsai, Y.H.; Hung, W.C.; Schulter, S.; Sohn, K.; Yang, M.H.; Chandraker, M. Learning to adapt structured output space for semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 7472–7481.
31. Kaneva, B.; Torralba, A.; Freeman, W.T. Evaluation of image features using a photorealistic virtual world. In Proceedings of the 2011 International Conference on Computer Vision, Barcelona, Spain, 6–13 November 2011; pp. 2282–2289. [CrossRef]
32. Marín, J.; Vázquez, D.; Gerónimo, D.; López, A.M. Learning appearance in virtual scenarios for pedestrian detection. In Proceedings of the 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, San Francisco, CA, USA, 13–18 June 2010; pp. 137–144. [CrossRef]
33. Bochinski, E.; Eiselein, V.; Sikora, T. Training a convolutional neural network for multi-class object detection using solely virtual world data. In Proceedings of the 13th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS 2016), Colorado Springs, CO, USA, 23–26 August 2016; pp. 278–285.
34. Leal-Taixé, L.; Milan, A.; Reid, I.; Roth, S.; Schindler, K. Motchallenge 2015: Towards a benchmark for multi-target tracking. *arXiv* **2015**, *arXiv:1504.01942*.
35. Johnson-Roberson, M.; Barto, C.; Mehta, R.; Sridhar, S.N.; Rosaen, K.; Vasudevan, R. Driving in the matrix: Can virtual worlds replace human-generated annotations for real world tasks? *arXiv* **2016**, *arXiv:1610.01983*.
36. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. In Proceedings of the 28th International Conference on Neural Information Processing Systems (NIPS), Montreal, QC, Canada, 7–12 December 2015; pp. 91–99.
37. Ros, G.; Stent, S.; Alcantarilla, P.F.; Watanabe, T. Training constrained deconvolutional networks for road scene semantic segmentation. *arXiv* **2016**, *arXiv:1604.01545*.
38. Redmon, J.; Farhadi, A. Yolov3: An incremental improvement. *arXiv* **2018**, *arXiv:1804.02767*.
39. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition, Las Vegas, NV, USA, 26 June–1 July 2016; pp. 770–778.

40. Yu, F.; Li, W.; Li, Q.; Liu, Y.; Shi, X.; Yan, J. Poi: Multiple object tracking with high performance detection and appearance feature. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 8–16 October 2016; pp. 36–42.
41. Lin, C.; Lu, J.; Wang, G.; Zhou, J. Graininess-Aware Deep Feature Learning for Pedestrian Detection. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 732–747.