Scene Understanding for Parking Space Management

Daniele Di Mauro
Department of Mathematics and Informatics, University of Catania, Catania, Italy

1 RESEARCH PROBLEM

Smart cities is one of the new frontier of the Computer Vision community. The major part of world-wide population moved to urban areas, after such process many issues of major cities have worsened, e.g. air pollution, traffic, security. The increase of security cameras and the improvements of Computer Vision algorithm can be a good solution for many of those problems.

Park Smart s.r.l., a company located in Catania, believes that Computer Vision can be the answer for parking space management. Their aim is to help private companies and public administrations to manage free entry parking areas, as well closed ones, in order to offer better services to the final customer i.e. the drivers and to increase the revenue per stall for public administrations.

The architecture of the system follow the Edge Computing design which brings the Computer Vision computation close to the parking area.

The main problem the company has to face is to find a fast way to deploy working solutions, lowering the labeling effort to the minimum, across different scene, cities, parking areas.

During the three years of doctoral studies we have tried to solve the problem through the use of various methods such as Semi-Supervised Learning, Counting and Scene Adaptation through Image Classification, Object Detection and Semantic Segmentation.

2 OUTLINE OF OBJECTIVES

The rest of the paper is organized as follows. This Section introduces the objectives of the work. Section 3 presents the state of the art. Section 4 explains the formulation of our methods. Section 5 reports the results. Section 6 concludes the paper.

Classification is the task where the computer vision community has obtained great results since the introduction of deep CNNs. Thus we decided to tackle the problem to decide if a parking space is empty or not as a classification task over patches corresponding to parking lots.

Object detection is the task which deals with detecting instances of semantic objects of a certain class (i.e. pedestrians, cars, building, etc.) in digital images and videos. We used object detection as a method for counting cars present in a scene.

Segmentation is a partition of an image into coherent parts, but without any attempt at understanding what these parts represent. Coherence is defined in terms of low-level cues such as color, texture and smoothness of boundary. Semantic Segmentation attempts to partition the image into semantically meaningful parts, and to classify each part into one of the pre-determined classes or in other word semantically understanding the role of each pixel in the image. Semantic segmentation is the starting point to have full knowledge of a parking area.

3 STATE OF THE ART

The state of the art is divided considering different topics: in Section 3.1 we introduce related works on parking areas. Counting approaches are reported in Section 3.2. Studies related to Semantic Segmentation are discussed in Section 3.3. Works about Generative Adversarial Networks and Domain Adaptation are reported in Section 3.4 and in Section 3.5 respectively.

3.1 Parking Spaces and Computer Vision

Wu et al. (Wu et al., 2007) proposed a simple pipeline to detect empty vs non-empty parking spaces. The color distribution on rectangular patches is computed and used to feed a Multi-class Support Vector Machine (SVM) for classification purposes. The results of the classification are processed using a Markov Random Field (MRF) to refine potential conflicts between two neighboring patches.

A method using image processing techniques was proposed by Yusnita et al. in (Yusnita et al., 2012).

Mauro, D.
Scene Understanding for Parking Space Management.
In *Doctoral Consortium (DCETE 2018)*, pages 3-11

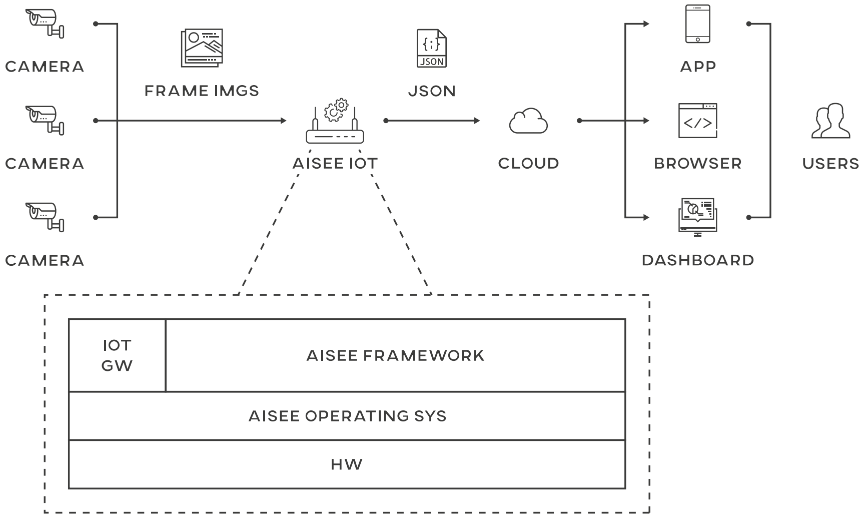


Figure 1: This diagram shows the current Park Smart system: images and videos are captured by cameras which send them to the AISEE embedded where the computation is performed. From there the information about the parking status is send to the cloud in order to be viewed by users.

The authors mark the real scene painting each stall with a circle in the center. Using morphological operators the system looks for the circles that are still visible, using an eccentricity based measure to check if the detected blobs are roughly circular. The system applies a threshold and counts the remaining spots, giving in output the number of free stalls.

Lin et al. (Ng and Chua, 2012) makes use of trajectories or events to separate empty stalls from non-empty ones. Motion trajectories are the feature vectors used in an adaptive Gaussian Mixture Model (GMM) and connected component analysis for background modeling and objects tracking.

In (De Almeida et al., 2015), authors built a dataset in order to test and assess both old and new algorithms to solve the free parking slots classification problem. The pictures were taken in different climatic conditions to provide a large variability. In order to validate the “goodness” of the dataset, the authors performed three kind of tests using hand-crafted features such as Local Binary Patterns and Local Phase Quantization.

In (Amato et al., 2017) they present a deep learning approach which make use of a modified *AlexNet* CNN is employed to obtain a reduced-size model in order to make inference possible in real-time on low-cost embedded devices, also a new dataset (*CNRPark-EXT*) has been introduced.

3.2 Counting in Computer Vision

Object counting is a challenging Computer Vision problem that needs a fine-grained understanding of the scene. The task has been typically studied considering specific contexts. For instance, some methods tackle the problem of counting people in crowded scenes (Chan et al., 2008; Chen et al., 2015; Li et al., 2008; Lempitsky and Zisserman, 2010; Zhang et al., 2015), cells in biological images (Lempitsky and Zisserman, 2010), bacterial colonies (Ferrari et al., 2017), penguins (Arteta et al., 2016), etc.

According to (Loy et al., 2013), counting methods can be divided into three groups:

- *counting by detection*, which uses object detection methods and count extensively (Chen et al., 2015);
- *counting by clustering*, which assumes the presence of individual entities presenting unique yet coherent patterns which can be clustered to approximate the final number of instances (Rabaud and Belongie, 2006);
- *counting by regression*, which counts entities by learning a direct mapping from low-level imagery to numbers (Chan et al., 2008; Lempitsky and Zisserman, 2010; Arteta et al., 2014; Fiaschi et al., 2012).

3.3 Semantic Segmentation

On the semantic segmentation side, one of the networks, which is currently state of the art, is presented in (Zhao et al., 2017). In their work the author exploit global context information by different-region-based context aggregation through a pyramid pooling module (PSPNet). Their global prior representation is effective to produce good quality results on the scene parsing task, and from obtained results the work provides a superior framework for pixel-level prediction. The proposed approach achieves state-of-the-art performance on various datasets. It came first in ImageNet scene parsing challenge 2016 (Russakovsky et al., 2015), PASCAL VOC 2012 benchmark (Everingham et al., 2010) and Cityscapes benchmark (Cordts et al., 2016). A single PSPNet yields the new record of mIoU accuracy 85.4% on PASCAL VOC 2012 and accuracy 80.2% on Cityscapes.

3.4 Generative Adversarial Networks

In 2014 the work of Goodfellow et al. (Goodfellow et al., 2014) introduced the *Generative Adversarial Networks*. The idea behind this work is to build a generative model using two networks, a generative network and a discriminative one, which work one against the other. The goal of the generative network is to create a sample similar to elements of training set. The goal of the discriminative network, on the other end, is to learn to recognize fake images from the real belonging to the training set.

There are several application of GANs since 2014, among all we can remember super resolution (Ledig et al., 2016), next video frame prediction (Lotter et al., 2016), generative visual manipulation (Zhu et al., 2016), image-translation (Isola et al., 2017).

3.5 Domain Adaptation for Semantic Segmentation

Domain Adaptation for classification problems has many years of application, but it has been less investigated in the world of Semantic Segmentation, to the best of our knowledge the first work in this direction is from Hoffman et al. (Hoffman et al., 2016). This method consists of a global domain alignment performed using a novel semantic segmentation network with fully convolutional domain adversarial learning. This initially adapted space then enables category specific adaptation through a generalization of constrained weak learning, with explicit transfer of the spatial layout from the source to the target domains. In (Isola et al., 2017) authors introduce the problem of image-translation through the use of Generative Adversarial Networks. There are two defined domains X and Y, their goal is to model a function $F : X \to Y$. They further devoloped the idea using unpaired image trasformation in (Zhu et al., 2017). In this work they translate from one domain to another and try to reconstructs the original element enforcing the transformation to have a cycle consistency.

The approach in (Hoffman et al., 2017) is to adapt representations at both the pixel-level and feature-level, through cycle-consistency without requiring aligned pairs. The model has been applied to a variety of visual recognition and prediction settings, also in the semantic segmentation task for road scenes demonstrating transfer from synthetic to real world domains.

The authors of (Sankaranarayanan et al., 2017) use a model with 4 networks: an embedding network, a pixel-wise classifier, a generator network which takes as input the learned embedding and reconstructs the RGB image and finally the discriminator network which performs two different tasks given an input: it classifies the input as real or fake in a domain consistent manner and it performs a pixel-wise labeling task similar to the pixel-wise network. The output of the pixel-wise classifier is a label map up-sampled to the same size as the input of the embedding network.

4 METHODOLOGY

4.1 The System at Glance

As explained in our previous work (Di Mauro et al., 2017), the system depicted in Figure 1 has four main components:

Cameras. We use wide angle cameras to optimize the number of parking spaces monitored. Our approach is not vendor locked. To have best results the resolution needed is at least 50px per side for each parking space.

AISEE IoT. We analyze the video stream as closest as possible to the camera. It is an embedded system capable of elevated computing power, enough to do inference using deep learning models. Once inference is done the results are sent to the cloud platform. The embedded operating system has been developed with security, privacy and resilience in mind. We can deploy several AISEE IoT boxes depending on the number of cameras and the dimension of the installation.

Cloud. We collect all the information from several installed embedded systems through a cloud platform which is scalable by design.

Presentation Layer. The system is accessible through different kind of appliances:

- The *dashboard* is the business and administration front-end which allows all the operations and to manage the installations (e.g. to add new cameras, configure cameras, add embedded, remove embedded and upgrade them, etc.).
- The *mobile app* or *browser* are the ending point for the people who are looking for a free spot where to park.

4.2 Images

Semi-supervised Dataset. The proposed PSD dataset was acquired from August 2015 to November 2015 in a parking lot of the University of Catania. The monitored parking lot is composed by 46 parking spaces. To cover the whole parking lot the data have been acquired by four cameras with a resolution of 1920×1080 extracted from motion jpeg registration. For each image the different parking spaces have been manually labeled as free or occupied. For experimental purpose the final set of parking spaces is composed by 270796 crops. We extracted a subset called PSD* which has 21000 non-empty parking spaces and 21000 empty parking spaces.

Counting Dataset. The dataset has been acquired using three Full-HD cameras looking at different parking spaces. The three cameras are referred to as “Camera 1”, “Camera 2” and “Camera 3”. “Camera 1” observes 12 parking spaces (Figure 2), “Camera 2” monitors 14 parking spaces (Figure 3), and “Camera 3” acquires images of 12 parking spaces (Figure 4). Given the different viewpoints of the cameras, the acquired scenes are characterized by different scene geometries. We recorded two long videos per camera at $1fps$. The two videos have been acquired in different days.

We propose two different ways of splitting data into training and testing sets. The first split assumes that training and testing data have been acquired using a single camera. This gives rise to 6 different data subsets (one for each camera), where one of the two videos is used for training. The second data split assumes that both training and test data have been acquired using the three cameras.

Synthetic Images. Another dataset used to perform experiments was created using the CARLA Urban Driving Simulator (Dosovitskiy et al., 2017). The simulator was developed to build dataset for autonomous driving situations. The systems does not permit to place cameras in fixed positions, but with a trick, i.e. placing a car in one of the predefined position in the map and placing the cameras at different altitudes and pitch, yaw and roll angles relative to the car is possible to overcome this limitation. We created the dataset using 3 views and 3 scenes.

4.3 Image Classification

The main idea is to divide each frame captured by the camera in several crops, where every crop is a square image corresponding to a parking space. This approach considers the problem as an image-based binary classification task. For each stall, we first extract the smallest square image patch containing it. Each image patch is labeled as “empty” or “full” depending on the occupancy status of the related stall. A classifier is hence trained to discriminate between “empty” and “full” stalls. At inference time, the trained classifier is used to determine the status of each stall in order to obtain the number of non-empty parking spaces.

4.4 Semi-supervised Learning

In (Lee, 2013) the author proposed a network trained in a semi-supervised fashion through the use, at the same time, of labeled and unlabeled data. To the unlabeled data is assigned the label that the network computed on the forward pass. The loss function is calculated on both labeled data and pseudo-labeled using the following formula:

$$L = \frac{1}{n}\sum_{m=1}^{n}\sum_{i=1}^{C} L(y_i^m, f_i^m) + \alpha(t)\frac{1}{\tilde{n}}\sum_{m=1}^{\tilde{n}}\sum_{i=1}^{C} L(\tilde{y}_i^m, \tilde{f}_i^m) \tag{1}$$

where n is the number of labeled data, $\tilde{n}$ the number of unlabeled data, C the classes, t is the number of iterations, y and f are the labels and network result for labeled data, $\tilde{y}$ $\tilde{f}$ are pseudo-labels and network result for unlabeled data and $\alpha(t)$ is defined as

$$\alpha(t) = \begin{cases} 0 & \text{if } t < T_1 \\ 1 & \text{if } T_1 \leq t < T_2 \\ a_f & \text{if } T_2 \leq t \end{cases} \tag{2}$$

where $a_f = 3$ and $T_1 = 100$, $T_2 = 600$. We adapted an AlexNet to perform pseudo-label training.

4.5 Object Detection for Counting

This approach employs a car detector to localize all the cars present in the image. All bounding boxes

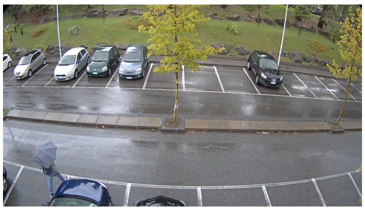


Figure 2: Camera 1 observes 12 parking spaces.

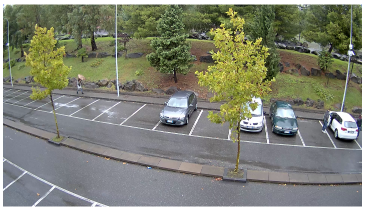


Figure 3: Camera 2 observes 14 parking spaces.

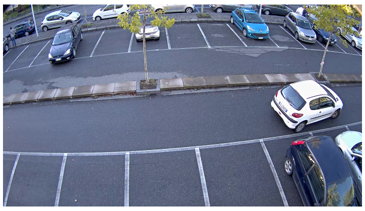
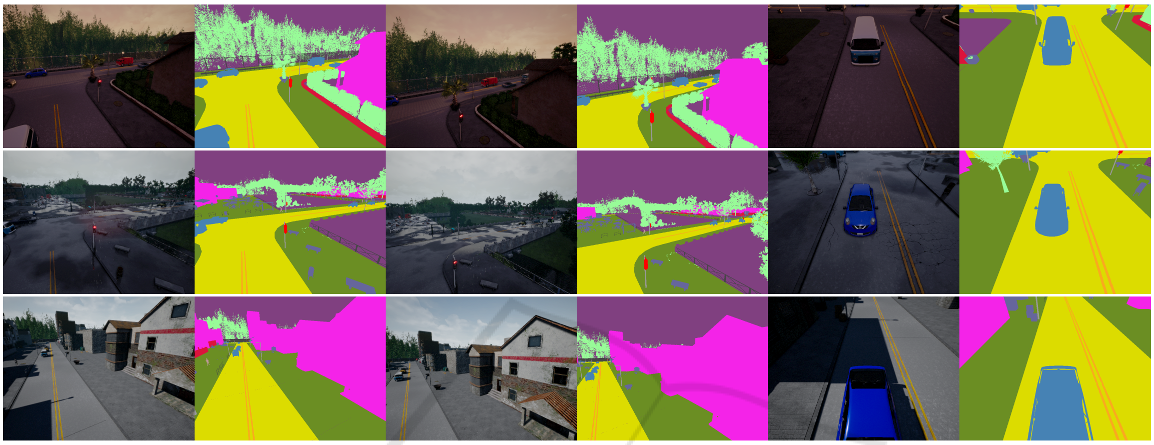


Figure 4: Camera 3 observes 12 parking spaces.



(m) View B (n) View B labels (o) View B (p) View B labels (q) View C (r) View C labels

Figure 5: Pictures of the three scenes and three points of view created.

detected with a score lower than a given threshold d_1 are discarded. The Intersection Over Union (IoU) measure between each stall and each retained bounding box is hence computed, a stall is deemed to be occupied if the IoU with at least one detected car is higher than a given threshold d_2. The method allows to count the number of non-empty parking spaces by determining the status of each stall. This approach allows to obtain also information about cars which are parked on non-marked spaces. Such information can be useful to allow for better management of parking areas, e.g. detecting mis-parked cars.

4.6 Scene Adaptation

Our goal is to learn a semantic segmentation from which we are able to reconstruct correctly images similar to target, where semantic labels are not given and source domain were semantic labels are given. In order to do so we use 3 distinct networks trained jointly. The loss function to minimize is the sum of a semantic loss, an adversarial loss and a reconstruction loss.

A network generate a semantic map from the image and we calculate a loss over it. We use a classical cross-entropy function between the inferred label and the ground truth. A second network try to reconstruct the original image starting from the semantic map inferred. We use a L_1 loss to measure the reconstruction quality.

Finally, to force to have better reconstruction we add an adversarial loss for the mapping function $G : Y \rightarrow X$ which is defined from the semantic label space Y to the image space X. The third network is a discriminator which use an adversarial loss.

5 EXPECTED OUTCOME

During the first year of research we built a strong background over the domain and over the deep learning methods suitable to be used to solve the problem using supervised classification. Through an analysis of the domain it was easy to understand that, to build a proper dataset for training, there are several variabilities which have to be considered such as: camera view, shapes of the parking spaces, and other classic variabilities of standard image classification problem such as background, light, deformation, weather.

Semi-Supervised classification was the first approach used to decrease labeling effort for fast deployment as showed in Section 6.2.

During the second year we tried a different method which is based on counting objects, like cars and parking spots, as solution for fast deployment, further details can be found in Section 6.3.

Currently we are moving our attention to a full knowledge of the scene through Semantic Segmentation and the use of Generative Adversarial Networks in order to find a viable way to reach good Scene Adaptation results further details can be found in Section 6.4.

6 STAGE OF THE RESEARCH

6.1 Park Smart

In (Di Mauro et al., 2017) we introduced the Park Smart system, an end-to-end pipeline for smart parking assistance and management. The infrastructure implements the Edge Computing paradigm (sometimes referred as Fog Computing) through a set of IoT devices which allows to perform the computation on the edge of cloud. The system relies on computer vision algorithms able to classify parking spaces, given their spatial configuration. To investigate the approach we used PKLot dataset (De Almeida et al., 2015), it has 12417 images with resolution of 1280 × 720 pixels. We sampled three datasets, one for each parking area, and fine-tuned AlexNet, results are reported in Table 1. We also tested this approach using our data, creating 3 subsets: DS1 has 17688 train images, 3924 in val and 21612 in test; DS2 has 20636 train images, 4578 in val and 31374 in test; DS3 has 13032 train images, 2820 in val and 25212 in test, results on this second experiment are reported in Table 2.

6.2 Learning Approaches for Parking Lots Classification

In (Di Mauro et al., 2016) we analyzed supervised vs semi-supervised approaches on the problem of parking lots classification. Results shown that the supervised approach using a classical AlexNet with fine-tuning outperforms a semi-supervised method which use pseudo-labels. Moreover the pseudo-label suffers when the dataset to be classified is composed by samples unbalanced with respect to the classes. The experiments (see Table 3 and Table 4) pointed out that the supervised method (AlexNet plus fine-tuning) outperforms the semi-supervised one (Pseudo-label) in all cases, obtaining very high accuracy (over 96% with few images as training). Moreover good results can be obtained with Pseudo-label only when the dataset to be classified is balanced in terms of samples per classes, which is a prior knowledge too difficult have in real applications, the trade-off discourage, in this case, to use this approach.

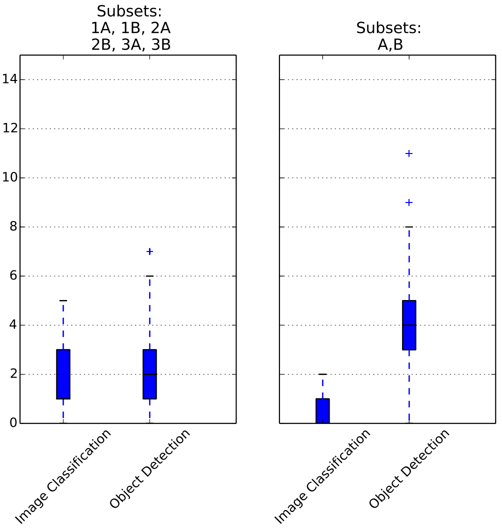


Figure 6: Box plots for counting non-empty spaces. We plot the mean absolute error for counting non-empty spaces in the single camera experiment and in the multiple camera. Higher is the worst.

6.3 A Comparison of Techniques based on Image Classification and Object Detection to Count Cars and Non-empty Stalls in Parking Spaces

In (Di Mauro et al., 2018) we investigated and compared two different approaches to count non-empty spaces and cars in parking areas. To perform the analysis, a dataset of videos has been collected in a real scenario and each frame has been labeled according to the position of parking stalls, the number of occupied stalls, and the number of cars in the frame. Results show that, when the geometry of the scene is known (i.e., stalls are marked), the system can take advantage of binary classification methods to obtain competitive results.

In Figure 6 we reports box plots with the Absolute Error values on counting non-empty spaces using classification and object detection. Figure 6 we reports box plots with the Absolute Error values values on counting cars using classification and object detection. In both figures we denote with subsets 1A, 2A, 3A, 1B, 2B, 3B a subset where training set and test set are composed with images from one camera. We denote with A and B subset with training set and test set with images from all cameras.

Table 1: Results using a fine-tuned AlexNet on PKLot.

Sample	Train	Val	Test	Accuracy
UFPR05	19281	4820	24101	99.93%
UFPR04	20000	5000	25000	99,96%
PUC	20000	5000	25000	99,92%

Table 2: Results obtained considering different CNN models and three dataset created from our data.

CNN Models	DS1	DS2	DS3	Avg. Accuracy	Footprint
AlexNet (Krizhevsky et al., 2012)	98,80%	99.20%	93.82%	97,27%	217M
GoogLeNet (Szegedy et al., 2015)	99.72%	99.58%	99.26%	99.52%	40M
VGG16 (Simonyan and Zisserman, 2014)	99.13%	98.70%	94.91%	97.58%	528M

Table 3: Results with training balanced per camera and class, *PKLot* has 72000 images, *PSD* has 144000 images, *PSD** has 42000 balanced between empty and non-empty.

Dataset	*Method*	*Loss*	Training Size 0.17%	Training Size 1%	Training Size 1.7%	Training Size 5%
PKLot	*finetuning*	*crossentropy*	97.35% ± 2.17	99.40% ± 0.04	99.54% ± 0.04	99.76% ± 0.02
	pseudolabel	*crossentropy*	94.85% ± 1.81	98.90% ± 0.13	99.35% ± 0.17	99.77% ± 0.04
	finetuning	*softmax*	97.35% ± 2.17	99.40% ± 0.04	99.54% ± 0.04	99.76% ± 0.02
	pseudolabel	*softmax*	97.03% ± 0.79	99.07% ± 0.17	99.32% ± 0.37	99.81% ± 0.06
PSD	*finetuning*	*crossentropy*	99.02% ± 0.14	99.46% ± 0.15	99.52% ± 0.01	99.73% ± 0.01
	pseudolabel	*crossentropy*	95.76% ± 1.60	99.25% ± 0.04	99.38% ± 0.13	99.81% ± 0.02
	finetuning	*softmax*	99.02% ± 0.14	99.46% ± 0.15	99.52% ± 0.01	99.73% ± 0.01
	pseudolabel	*softmax*	96.89% ± 0.94	99.34% ± 0.07	99.35% ± 0.13	99.81% ± 0.04
*PSD**	*pseudolabel*	*crossentropy*	98.24% ± 0.13	99.06% ± 0.02	97.24% ± 0.56	97.86% ± 0.02
	pseudolabel	*softmax*	97.55% ± 0.56	98.82% ± 0.11	97.45% ± 0.24	97.93% ± 0.22

Table 4: Results with training balanced per class, *PKLot* has 72000 images, *PSD* has 144000 images, *PSD** has 42000 balanced between empty and non-empty.

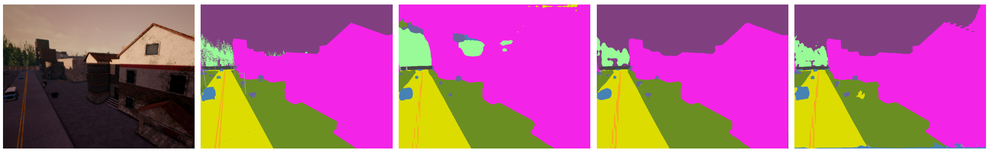
Dataset	*Method*	*Loss*	Training Size 0.17%	Training Size 1%	Training Size 1.7%	Training Size 5%
PKLot	*finetuning*	*crossentropy*	96.46% ± 0.49	98.36% ± 0.33	98.70% ± 0.01	99.02% ± 0.04
	pseudolabel	*crossentropy*	15.24% ± 0.67	17.13% ± 0.91	20.65% ± 6.00	14.65% ± 0.00
	finetuning	*softmax*	96.39% ± 0.26	98.25% ± 0.33	98.47% ± 0.08	99.00% ± 0.15
	pseudolabel	*softmax*	15.24% ± 0.67	17.13% ± 0.91	20.65% ± 6.00	14.65% ± 0.00
PSD	*finetuning*	*crossentropy*	96.92% ± 0.13	98.24% ± 0.05	98.59% ± 0.08	99.05% ± 0.06
	pseudolabel	*crossentropy*	15.14% ± 0.55	51.69% ± 26.55	61.78% ± 33.33	38.22% ± 33.33
	finetuning	*softmax*	96.83% ± 0.55	98.10% ± 0.39	98.68% ± 0.17	99.12% ± 0.11
	pseudolabel	*softmax*	15.14% ± 0.55	51.69% ± 26.55	61.78% ± 33.33	38.22% ± 33.33
*PSD**	*pseudolabel*	*crossentropy*	98.50% ± 0.12	98.99% ± 0.05	97.80% ± 0.21	98.19% ± 0.28
	pseudolabel	*softmax*	98.23% ± 0.29	98.99% ± 0.17	97.89% ± 0.30	98.01% ± 0.47

6.4 Scene Adaptation through Generative Adversarial Networks

Our current attention is focused on using Generative Adversarial Network in conjunction with a state of the art Semantic Segmentation Network. This approach is proving to be a very promising way to perform semantic adaptation. The classical training, on fixed camera, tend to overfit the network in order to increase the accuracy for background pixels, while with the GAN approach we make it easier for the network to generalize better. We tested the trained network also on the source domain, and in many cases, the resulting network is better also on those. In Figure 8 we can see initial qualitative results of our method.

ACKNOWLEDGEMENTS

I would like to thank my Ph.D. supervisor Prof. Giovanni Maria Farinella (University of Catania) for his advice, as well as Prof. Sebastiano Battiato (University of Catania), Dr. Antonino Furnari (University of Catania) and Dott. Giuseppe Patané (Park Smart) for their support and suggestions.



(a) Image (b) Label (c) Baseline (d) Upper (e) Our Method

Figure 8: Some results of current work with Generative Adversarial Networks.

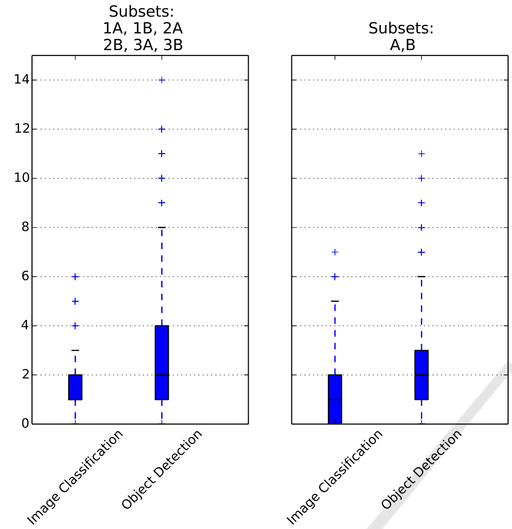


Figure 7: Box plots for counting cars. We plot the mean absolute error for counting cars in the single camera experiment and in the multiple camera. Higher is the worst.

REFERENCES

Amato, G., Carrara, F., Falchi, F., Gennaro, C., Meghini, C., and Vairo, C. (2017). Deep learning for decentralized parking lot occupancy detection. *Expert Systems with Applications*, 72(15):327–334.

Arteta, C., Lempitsky, V., Noble, J. A., and Zisserman, A. (2014). Interactive Object Counting. In *European Conference on Computer Vision*, pages 1–15.

Arteta, C., Lempitsky, V., and Zisserman, A. (2016). Counting in the wild. In *European Conference on Computer Vision*, pages 483–498.

Chan, A. B., Liang, Z. S.-J., and Vasconcelos, N. (2008). Privacy preserving crowd monitoring: Counting people without people models or tracking. In *Conference on Computer Vision and Pattern Recognition*, pages 1–7. IEEE.

Chen, S., Fern, A., and Todorovic, S. (2015). Person count localization in videos from noisy foreground and detections. In *Conference on Computer Vision and Pattern Recognition*, pages 1364–1372. IEEE.

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 *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

De Almeida, P. R., Oliveira, L. S., Britto, A. S., Silva, E. J., and Koerich, A. L. (2015). PKLot-A robust dataset for parking lot classification. *Expert Systems with Applications*, 42(11):4937–4949.

Di Mauro, D., Battiato, S., Patanè, G., Leotta, M., Maio, D., and Farinella, G. M. (2016). Learning approaches for parking lots classification. In *International Conference on Advanced Concepts for Intelligent Vision Systems*, pages 410–418. Springer.

Di Mauro, D., Furnari, A., Patanè, G., Battiato, S., and Farinella, G. M. (2018). A comparison of techniques based on image classification and object detection to count cars and non-empty stalls in parking spaces. In *Proceedings of the 15th International Joint Conference on e-Business and Telecommunications - SIGMAP, (ICETE 2018)*. INSTICC, SciTePress.

Di Mauro, D., Moltisanti, M., Patanè, G., Battiato, S., and Farinella, G. M. (2017). Park smart. In *International Workshop on Traffic and Street Surveillance for Safety and Security*. IEEE.

Dosovitskiy, A., Ros, G., Codevilla, F., Lopez, A., and Koltun, V. (2017). CARLA: An open urban driving simulator. In *Proceedings of the 1st Annual Conference on Robot Learning*, pages 1–16.

Everingham, M., Van Gool, L., Williams, C. K. I., Winn, J., and Zisserman, A. (2010). The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 88(2):303–338.

Ferrari, A., Lombardi, S., and Signoroni, A. (2017). Bacterial colony counting with convolutional neural networks in digital microbiology imaging. *Pattern Recognition*, 61:629–640.

Fiaschi, L., Koethe, U., Nair, R., and Hamprecht, F. A. (2012). Learning to count with regression forest and structured labels. In *International Conference on Pattern Recognition*, pages 2685–2688.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680.

Hoffman, J., Tzeng, E., Park, T., Zhu, J.-Y., Isola, P., Saenko, K., Efros, A. A., and Darrell, T. (2017). Cycada: Cycle-consistent adversarial domain adaptation. *arXiv preprint arXiv:1711.03213*.

Hoffman, J., Wang, D., Yu, F., and Darrell, T. (2016). Fcns in the wild: Pixel-level adversarial and constraint-based adaptation. *arXiv preprint arXiv:1612.02649*.

Isola, P., Zhu, J.-Y., Zhou, T., and Efros, A. A. (2017). Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1125–1134.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105.

Ledig, C., Theis, L., Huszár, F., Caballero, J., Cunningham, A., Acosta, A., Aitken, A., Tejani, A., Totz, J., Wang, Z., et al. (2016). Photo-realistic single image super-resolution using a generative adversarial network. *arXiv preprint*.

Lee, D.-H. (2013). Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on Challenges in Representation Learning, ICML*, volume 3.

Lempitsky, V. and Zisserman, A. (2010). Learning to Count Objects in Images. In *Advances in Neural Information Processing Systems*, pages 1324–1332.

Li, M., Zhang, Z., Huang, K., and Tan, T. (2008). Estimating the number of people in crowded scenes by MID based foreground segmentation and head-shoulder detection. In *International Conference on Pattern Recognition*, pages 1–4.

Lotter, W., Kreiman, G., and Cox, D. (2016). Deep predictive coding networks for video prediction and unsupervised learning. *arXiv preprint arXiv:1605.08104*.

Loy, C. C., Chen, K., Gong, S., and Xiang, T. (2013). *Crowd Counting and Profiling: Methodology and Evaluation*, pages 347–382. Springer, New York, NY.

Ng, L. L. and Chua, H. S. (2012). Vision-based activities recognition by trajectory analysis for parking lot surveillance. In *International Conference on Circuits and Systems*, pages 137–142.

Rabaud, V. and Belongie, S. (2006). Counting Crowded Moving Objects. In *Conference on Computer Vision and Pattern Recognition*, pages 705–711. IEEE.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A. C., and Fei-Fei, L. (2015). ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252.

Sankaranarayanan, S., Balaji, Y., Jain, A., Lim, S., and Chellappa, R. (2017). Unsupervised domain adaptation for semantic segmentation with gans. *CoRR*, abs/1711.06969.

Simonyan, K. and Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., and Rabinovich, A. (2015). Going deeper with convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–9.

Wu, Q., Huang, C., Wang, S.-y., Chiu, W.-c., and Chen, T. (2007). Robust parking space detection considering inter-space correlation. In *International Conference on Multimedia and Expo*, pages 659–662. IEEE.

Yusnita, R., Norbaya, F., and Basharuddin, N. (2012). Intelligent parking space detection system based on image processing. *International Journal of Innovation, Management and Technology*, 3(3):232.

Zhang, C., Li, H., Wang, X., and Yang, X. (2015). Cross-scene crowd counting via deep convolutional neural networks. In *Conference on Computer Vision and Pattern Recognition*, pages 833–841. IEEE.

Zhao, H., Shi, J., Qi, X., Wang, X., and Jia, J. (2017). Pyramid scene parsing network. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 2881–2890.

Zhu, J.-Y., Krähenbühl, P., Shechtman, E., and Efros, A. A. (2016). Generative visual manipulation on the natural image manifold. In *Proceedings of European Conference on Computer Vision (ECCV)*.

Zhu, J.-Y., Park, T., Isola, P., and Efros, A. A. (2017). Unpaired image-to-image translation using cycle-consistent adversarial networks. *arXiv preprint arXiv:1703.10593*.