UAB

Universitat Autònoma de Barcelona

Pedestrian Detection based on Local Experts

A dissertation submitted by **Fco. Javier Marín Tur** at Universitat Autònoma de Barcelona to fulfil the degree of **Doctor of Philosophy**.

Bellaterra, June 2013

Director	**Dr. Antonio M. López Penã** Dept. Ciències de la Computació & Centre de Visió per Computador Universitat Autònoma de Barcelona
Co-director	**Dr. Jaume Amores Llopis** Centre de Visió per Computador Universitat Autònoma de Barcelona

This document was typeset by the author using LaTeX 2_{ε}.

The research described in this book was carried out at the Computer Vision Center, Universitat Autònoma de Barcelona.

ISBN:

Printed by

A mis padres, Concepción y Ginés, y a Tia

Je donnerais tout ce que je sais, pour la moitié de ce que je ne sais pas.
René Descartes (1596-1650).

Acknowledgements

Without any doubt, making a PhD thesis has been the most difficult intellectual challenge I have done in my entire life. Besides, needless to say that it would have been impossible to complete this work without the aid of the people I have been working with and the support of my loved ones. During these years, I have also been able to spend some of my time in two different countries, which has giving me the opportunity of to meet people around the world with different mentalities and personalities. From most of these people I will keep a beautiful memory for the rest of my life. It would be a crime to forget the time I have also been spending in Barcelona which has enriched me as a person, and in which I have had the chance of meeting what I consider today some of my best friends and loved ones. I can consider myself a lucky person. Next, I would like to add some lines to the people and institutions which have supported me along my PhD.

First of all, I would like to express my gratitude to my supervisors Dr. Antonio M. López and Dr. Jaume Amores. I want to thank them for their advice and contributions to this work. I would also remark their continued effort on teaching me how to become a qualified researcher. And of course, my most sincere gratitude goes to Antonio for transmitting me always his optimism even when an objective seemed impossible to reach.

I am particular grateful to Dr. Ludmila I. Kuncheva for granting me the occasion of spending my first internship during my PhD in her group in Bangor, and also, for her guidance, great patience and encouragement along this fantastic time I spent in her group, and to introduce Tia and I to her splendid family. I am also indebted to Dr. Bastian Leibe who gave me the opportunity to spend my second internship in one of the best research centres in Germany, and from whom I learned in a incommensurable way.

To all the researchers and friends I met during my first internship, Thomas Christy, Haider Easa, Catrin Plumpton, Jamie Whitaker, Nicholas Musembi, and Juan J. Rodriguez. And my second stage, Tobias Weyand, Torsten Sattler, Dennis Mitzel, Patrick Sudowe, Esther Horbert, Georgios Floros, Wolfgang Mehner, Oscar and Charlie Puñal, and Pau Panareda.

To all the friends in the Computer Vision Center, including all my colleagues and friends in the ADAS group who deserve my sincerest thanks. In particular, David Vázquez, with whom I have shared most of my working and good times with. To José C. Rubio, Pep Gonfaus, Jorge Bernal, with whom I have travelled abroad and spend memorable times. To Dr. Angel Sappa, Dr. Joan Serrat, Dr. Daniel Ponsa, Dr. David Gerónimo, Dr. José M. Álvarez, Dr. Xavier Roca, and Dr. Jordi Poal.

To my parents, Concepción and Ginés, for supporting and encouraging me in this adventure, and for their effort on providing me the best education they could afford. Also to my closest family, Cristina, Paquita, and Fermín, which have been like my second parents. And

to Mar, Sergi, and Daniel which I consider the brothers and sister I never had. My friends and professors from Ibiza and Mallorca, which have contributed to built the person I have become.

Finally, but not least important, I want to express my gratitude to my loved Tia. Thanks for supporting me during these years in which both have suffered the same problems and shared the same joys.

Resum

Al llarg dels darrers anys, els sistemes de detecció humana basats en visió per computador han començat a exercir un paper clau en diverses aplicacions lligades a l'assisténcia a la conducció, la videovigilància, la robòtica i la domòtica. Detectar persones és, sens cap dubte, una de les tasques més difícils en el camp de la Visió per Computador. Això es deu principalment al grau de variabilitat en l'aparença humana associada a la roba, postura, forma i grandária. A més, altres factors com escenaris amb molts elements, oclusions parcials o condicions ambientals poden fer que la tasca de detecció sigui encara més difícil.

Els mètodes més prometedors a l'estat de la qüestió es basen en models d'aprenentatge discriminatius que són entrenats amb exemples positius (vianants) i negatius (no vianants). El conjunt d'entrenament és un dels elements més rellevants a l'hora de construir un detector que faci front a la citada gran variabilitat. Per tal de crear el conjunt d'entrenament es requereix supervisió humana. L'inconvenient en aquest punt és el gran esforç que suposa haver d'anotar, així com la tasca de cercar l'esmentada variabilitat.

En aquesta tesi abordem dos problemes recurrents a l'estat de la qüestió. En la primera etapa, es pretén reduir l'esforç d'anotar mitjançant l'ús de gràfics per computador. Més concretament, desenvolupem un escenari urbà per més endavant generar un conjunt d'entrenament. Tot seguit, entrenem un detector usant aquest conjunt, i finalment, avaluem si aquest detector pot ser aplicat amb èxit en un escenari real.

En la segona etapa, ens centrem en millorar la robustesa dels nostres detectors en el cas en que els vianants es trobin parcialment ocluids. Més concretament, presentem un nou mètode de tractament d'oclusions que consisteix en millorar la detecció de sistemes holístics en cas de trobar un vianant parcialment ocluid. Per dur a terme aquesta millora, fem ús de classificadors (experts) locals a través d'un mètode anomenat *random subspace method* (RSM). Si el sistema holístic infereix que hi ha un vianant parcialment ocluid, aleshores s'aplica el RSM, el qual ha estat entrenat prèviament amb un conjunt que contenia vianants parcialment ocluids. L'últim objectiu d'aquesta tesi és proposar un detector de vianants fiable basat en un conjunt d'experts locals. Per aconseguir aquest objectiu, utilitzem el mètode anomenat *random forest*, a on els arbres es combinen per classificar i cada node és un expert local. En particular, cada expert local es centra en realitzar una classificació robusta de zones del cos. Cal remarcar, a més, que el nostre mètode presenta molta menys complexitat a nivell de disseny que altres mètodes de l'estat de la qüestió, alhora que ofereix una eficiència computacional raonable i una major precisió.

Abstract

During the last decade vision-based human detection systems have started to play a key role in multiple applications linked to driver assistance, surveillance, robot sensing and home automation. Detecting humans is by far one of the most challenging tasks in Computer Vision. This is mainly due to the high degree of variability in the human appearance associated to the clothing, pose, shape and size. Besides, other factors such as cluttered scenarios, partial occlusions, or environmental conditions can make the detection task even harder.

Most promising methods of the state-of-the-art rely on discriminative learning paradigms which are fed with positive and negative examples. The training data is one of the most relevant elements in order to build a robust detector as it has to cope the large variability of the target. In order to create this dataset human supervision is required. The drawback at this point is the arduous effort of annotating as well as looking for such claimed variability.

In this PhD thesis we address two recurrent problems in the literature. In the first stage, we aim to reduce the consuming task of annotating, namely, by using computer graphics. More concretely, we develop a virtual urban scenario for later generating a pedestrian dataset. Then, we train a detector using this dataset, and finally we assess if this detector can be successfully applied in a real scenario.

In the second stage, we focus on increasing the robustness of our pedestrian detectors under partial occlusions. In particular, we present a novel occlusion handling approach to increase the performance of block-based holistic methods under partial occlusions. For this purpose, we make use of local experts via a Random Subspace Method (RSM) to handle these cases. If the method infers a possible partial occlusion, then the RSM, based on performance statistics obtained from partially occluded data, is applied. The last objective of this thesis is to propose a robust pedestrian detector based on an ensemble of local experts. To achieve this goal, we use the random forest paradigm, where the trees act as ensembles an their nodes are the local experts. In particular, each expert focus on performing a robust classification of a pedestrian body patch. This approach offers computational efficiency and far less design complexity when compared to other state-of-the-art methods, while reaching better accuracy.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Over the past decades the total population around the world, which is mostly found in urban scenarios, have significantly increased. Figure 1.1 shows a daily scene in one of the most populated cities in the world. In 2010, the world population reached the figure of 6.89 billion people [74]. Being India and China the two countries with the highest increase. At the same time, the number of vehicles around the world grew noticeably. In 2010, a research study reported an estimated ownership rate of 148 vehicles (in which trucks were not included) per 1000 people in the whole world [67]. These two growths during the last years have led consequently to a greater interaction between both vehicles and pedestrians, which sadly has resulted in a large number of accidents with injuries and deaths.

Meanwhile, governments, institutions and automotive companies around the world have been incrementing their efforts to reduce the number of casualties caused by traffic accidents. Education, awareness through different media[1], and making vehicles and roads more safe have been the main instruments to decrease the traffic casualties. However, as reported by the World Health Organization (WHO) [53] in their last 2013 publication, 1.24 million people are still being killed every year on the world's roads, which continues to be unacceptable. Moreover, WHO predicts that traffic injuries will become the fifth cause of death by 2030.

Although great efforts are being made, as mentioned before, reducing the number of traffic casualties has recently become a crusade. In order to improve the traffic safety the automotive industry and scientific research community have been developing and integrating new systems into the vehicle. These systems, commonly known as *advanced driver assistance systems* (ADAS) are designed to help the driver through warnings and, in certain cases, automatically taking active decisions, *e.g.*braking the vehicle. For instance, three examples of successfully commercialised ADAS are lane departure warning, intelligent headlights control, and adaptive cruise control (ACC), which are intended to keep the vehicle on the lane, to avoid dazzling the other drivers, and maintain a safe distance from the preceding vehicle, respectively. In Figure 1.2, we show an example of each system.

[1] In the most dangerous pedestrian crossings of Barcelona we can read 'One in three deaths in traffic accidents were pedestrians' to alert the citizens.

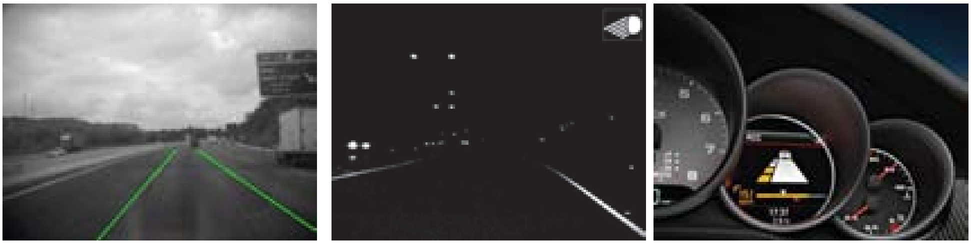
Figure 1.1: A typical crowded scene in the Shibuya intersection, Tokyo. Photo credit: www.tokyoluv.com

1.1 Pedestrian Protection Systems

One of the most complex safety systems in ADAS are *pedestrian protection systems* (PPSs) [3, 19, 27, 30], which are specialised in avoiding vehicle-to-pedestrian collisions. In fact, this kind of accidents results in approximately 150000 injuries and 7000 killed pedestrians every year just in the European Union [73]. Similar statistics apply to the United States, while underdeveloped countries are increasing theirs year after year. Actually, over a third of road traffic deaths in low- and middle-income countries are among pedestrians and cyclists, in which less than 35% of these countries have policies in place to protect these road users [53].

In the case of PPSs, the most promising approaches make use of images as main source of information, as can be seen in the large amount of proposals exploiting them [30]. Hence, the core of a PPS is a forward facing camera that acquires images and processes them using Computer Vision techniques. In fact, the Computer Vision community has traditionally maintained a special interest in detecting humans, given the challenging topic it represents. Pedestrians are one of the most complex objects to analyse mainly due to their variability in pose, clothing, and size. Moreover, other factors such as heavily cluttered backgrounds, partial occlusions, or environmental conditions may influence the detector accuracy. In addition, the images are acquired from a mobile platform, which makes human detection algorithms such as background subtraction, key in fields like video-surveillance, hardly applicable in a straightforward manner for PPSs. Finally, the task has to be carried out in real-time.

Most successful vision-based pedestrian detection systems rely on discriminative learning paradigms. Along this line, researchers have been mostly working on two different is-



(a) (b) (c)

Figure 1.2: Three different ADAS examples. (a) the lane departure warning is showing the lane markings location. (b) the intelligent headlights control adapts the headlights mode (*e.g.*high and low beans) according to the traffic situation. (c), the ACC of a porsche car. Image credits: (a) and (b), cvc.uab.es/adas; and (c) www.porsche.com.

sues: designing features [10, 56, 79, 81, 82], and classification through machine learning algorithms [10, 25, 45, 79, 88, 89]. State-of-the-art approaches can be divided into two groups: *holistic*, which rely on detecting the target as a whole, and *part-based*, which usually combine the detection of different parts of the body (head, torso, arms, legs, etc.) with a deformable structure among the parts. Holistic methods offer robustness with respect to illumination, background and texture changes, whereas part-based methods have an advantage for different poses and a claimed robustness to partial occlusions [18]. The current learning process in both groups consists on feeding the algorithm with examples (pedestrians) and counterexamples (background). Then, once the algorithm ’learns’ the differences in the form of a pedestrian classifier, this can be fed into a pedestrian detector to operate in the traffic environment. In most cases the positive examples consist of cropped images, termed windows, in which pedestrians are placed in the center with a certain margin around them, and for the negative examples, windows without[2] any pedestrian. Note that some part-based approaches require a set of positive and negative annotated parts [8] or patches [26, 39, 68], which considerably increases the amount of work involved in the data acquisition phase.

1.2 Objectives

In this thesis, we address two recurrent subjects in the literature. First, we aim to mitigate the annotation labour required to train a pedestrian classifier, by making use of last advances in computer graphics. Second, to increase the accuracy of state-of-the-art pedestrian detection. In particular, we focus on both detection under partial occlusion as well as capturing pedestrian appearance variability using an ensemble of local experts. Accordingly, we describe the main objectives of this dissertation are:

[2] Some authors also introduce as negative samples windows containing non centered pedestrians [84].

O1 Demonstrating the viability of using virtual data for pedestrian detection. We first create a virtual dataset, then train a detector, and finally evaluate its performance in a real scenario.

O2 Presenting a novel occlusion handling approach to improve the performance of block-based holistic methods. Here, we make use of local experts to handle partial occlusions. If the holistic confidence is not high enough when classifying a candidate window and the window is inferred as a partially occluded pedestrian, then an ensemble of local experts, based on performance statistics obtained from partially occluded data, is applied.

O3 Proposing a robust detector. To achieve this goal we present a novel pedestrian detection approach that combines the strength of a local expert at each node using rich block-based features with the advantages of a Random Forest ensemble, such as computational efficiency and far less design complexity when compared to other state-of-the-art methods [18].

1.3 Thesis outline

State-of-the-art

Chapter 2 summarizes the state-of-the-art. This is done while describing the different stages of a vision-based pedestrian detection system. In this chapter, we emphasize the classification stage.

Exploring virtual worlds

To build a robust detector every specific aspect in the training process can be crucial. One of the most important elements in this process is the training dataset. In fact, a rich dataset in terms of different shapes, poses, clothes, illuminations, and backgrounds is always desirable to obtain a reliable detector. To create such rich dataset, human supervision is required. For instance, once a video sequence or an image has been obtained, it is usually[3] necessary to annotate the bounding box of each pedestrian for training purposes. Moreover, although negative examples can be extracted from negative images without any kind of supervision, the negative images from which these examples come from also need to be labelled. The drawback at this point is the required effort of annotating as well as looking for such claimed variability (unless a large amount of data is collected to cope the necessary variability, which does not help anything to reduce the work). Besides, while one might think that this labour needs to be done only once, a new camera settings, a change of environment or other issues may imply the requirement of collecting more examples or an entire new dataset. Therefore, removing or mitigating the consuming task of manually annotating examples is a substantial contribution.

In the last decade, thanks to the last advances in computer graphics the video game scenarios and models have reached a high realism. This fact brought us the idea of developing a virtual world in order to generate a virtual pedestrian dataset. In particular, we used a public

[3] In some cases, it is even necessary to provide the parts or patches of the pedestrians.

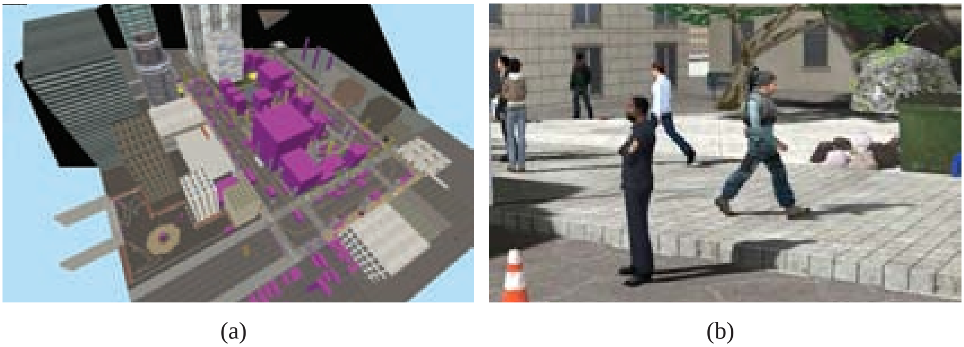


Figure 1.3: Our virtual city. (a) a global view from the map editor of our virtual city. (b) an image shot of our scenario.

available modification, called Object Virtual Video Tool (OVVT) published in [69], of the Half Life 2 engine game, one of the most popular games in the gamers community[4]. We developed and created from the scratch a virtual city with four different variants (four different type of illuminations with building and ground texture changes) in which human models and cars were moving around (see Fig. 1.3). Then, using some additional libraries provided by the same company which developed the OVVT, we recorded several virtual video sequences in which the information related to the target such as the location, the pixel by pixel groundtruth, and the level of occlusion were automatically generated. Next, using the collected video sequences and the stored information we generated a pedestrian virtual dataset to finally training a pedestrian detector. The main concern of this work was if such a virtual detector could be successfully used in real scenarios. To clarify this question, in Chap. 3, we carried out several experiments by comparing the virtual detector versus a large real one pedestrian dataset, the Daimler pedestrian dataset [19]. Additionally, we explored the impact in the accuracy with respect to the different number of virtual models, the total number of examples and the pose distribution. Thus, in Chapter 3 we address objective O1.

So far, the results shown in Chap. 3 demonstrated that a pedestrian classifier trained with virtual data could be successfully applied in a real-world scenario. More importantly, our contribution motivated other researchers to start working on the use of synthetic data [37, 61, 62]. Besides, once finished this work, we detected new issues. In fact, the performance of the virtual detector was not performing as expected in other datasets. After analysing the possible causes, we finally found that these problems were related to the domain adaptation context. This finding gave rise to a new thesis carried out by one of my group colleagues [76]. Another contribution was that after the publication of this first work, we made publicly available our virtual training dataset for benchmarking purposes.

[4] In 2004, Half Life 2 was the winner of over 35 ’Game of the Year’ awards.

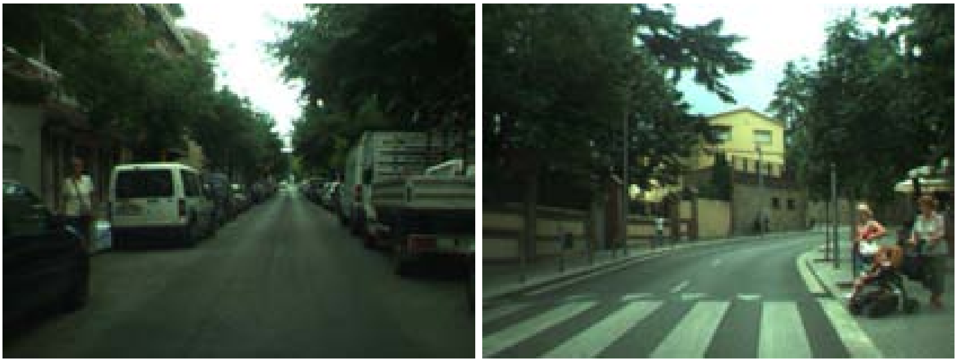


Figure 1.4: Some examples of partial occlusions. On the left, a pedestrian stepping out from behind a vehicle. On the right, a pedestrian partially occluded by a baby trolley.

An occlusion handling approach

Detecting partially occluded pedestrians can be determinant in certain situations. For instance, a parent pushing a baby trolley or a pedestrian stepping out from behind a car are two specific cases which can be frequently seen in urban scenarios (see Fig. 1.4). While detecting non occluded pedestrians, as mentioned before, deserves a special effort, detecting partially occluded pedestrians is an even harder labour. If parts such as the legs, the arms or even half body are occluded, pedestrian classifiers can interpret such regions of the window as noise or background and consequently misclassifying the candidate window. Therefore, a robust detector able to increase its accuracy against partial occlusions is vital.

In Chapter 4, we describe a general framework for occlusion handling. This proposal consists in a block-based holistic classifier supported by a random subspace method (RSM) to handle the candidates inferred as partially occluded. More concretely, in case the holistic method fails due to a possible partial occlusion, the RSM is applied in the region of the window less likely to be occluded. This is possible thanks to the training procedure, in which a validation dataset of partially occluded pedestrians (generated using our previous virtual framework) is used through a selection strategy to choose the best combination of classifiers. Therefore, in Chapter 4 we are addressing objective O2.

To conduct a full study, we evaluated and compared our method with the current state-of-the-art. Motivated by the lack of a public pedestrian dataset with a large number of partially occluded targets, we additionally created a dataset with hundreds of partially occluded pedestrians named PobleSec (the largest dataset in the literature had at most around 100 partially occluded pedestrians as reported in [81], in our case we have 1117 partially occluded annotated pedestrians). Our dataset was used to evaluate at the detection level the accuracy of the current methods against partial occlusions. Evaluations at the classification and detection level are also reported for non-occluded data. For evaluating the classification accuracy, we used the publicly available Daimler [44] dataset, which is divided into two different subsets, non occluded and partially occluded pedestrians, and the well-known INRIA person dataset [10] for the evaluation on non-occluded pedestrians.

The results obtained when compared with the current state-of-the-art for holistic approaches using occlusion handling [82] confirmed the advantages of using our framework. Besides our method presented several benefits: it could be extended to other class of objects; no extra computation in terms of features was required when applying the RSM; it did not need stereo or motion information to handle partial occlusion as other methods [44]; and, for training our method, it only required non- and partially-occluded positive examples, while other methods required the exact annotation of the occluded region of the target. Both, the real and the virtual (used in the training procedure) datasets created during this work were also publicly released for comparison purposes. Only a recently published dataset [54] contains more partially occluded pedestrians annotated than ours, which means that the PobleSec dataset remains the second largest dataset in the literature in terms of partially occluded pedestrians.

A random forest of local experts

As previously introduced, detecting partially occluded pedestrians may increase the chances of reducing the number of casualties. In the literature [18, 30], part-based approaches are claimed to be robust against partial occlusions as well as pose variability. This highlight and the success of our previous approach motivate us to develop a new method based on local experts.

In Chapter 5 we address the objective O3, *i.e.*we propose a new method for pedestrian detection. The main aim in this part of the thesis was to present an accurate detector when compared with the state-of-the-art but also robust to partial occlusions. In our proposal, during the training process, every single weak classifier of each tree was selecting the most discriminant local patch via an optimization process on the holistic descriptor. This local patch was chosen from an initial random subset of all possible patches. The main differences at this point with respect to the original random forest were the optimization process in which the weak classifier was obtained and the use of richer descriptors. The main difference with respect to the standard framework is that in each node the optimization process is not only based on a maximization of a purity measure, but also on a maximum-margin optimizer which minimizes the classification error over the samples of the node. This was achieved by making use of a local expert sustained by a linear vector machine (SVM) which assured the optimal hyperplane that splits the training samples at each node. Besides, it is important to note that previous works were not using the random forest approach to classify a candidate window but to locate the center position, with a certain confidence, of the target with respect to each patch.

Additionally, to speed-up our method we integrated the random forest into a soft cascade. Basically, this approach permitted to increase the efficiency while keeping the original accuracy provided by the RF. It is also worth mentioning that, while efficiency was not our main concern, the resulting detector achieved a comparable speed to the fastest approaches in the literature as reported in [18]. Besides, our RF can be easily extended to other class objects as well as incorporating new features such motion, multi-resolution or colour for further improvement. To validate our method, we conducted several evaluations on four different pedestrian datasets, INRIA [10], Caltech [18], Daimler [19] and ETH [21]. In these evaluations, we also compared the current state-of-the-art with our method. To evaluate the performance against partial occlusions we made use of the partial occlusion evaluation subset of Caltech and PobleSec.

Conclusions

Finally, Chaper 6 draws the main conclusions of this thesis.

Chapter 2

State of the art

In this chapter we describe the complete[1] architecture of a pedestrian detection system based on the framework proposed by Gerónimo *et al.* [29] (see Fig. 2.1). For this purpose, we explain the different stages that form part of a complete system: from the data acquisition process to the final output detections. In each stage, we discuss the issues that have to be undertaken and the different solutions existent in the state-of-the art. Thus, the objective of this chapter is two-fold: first to provide a general view of a complete pedestrian detection system, and second to provide an exhaustive review of the state-of-the-art and how the existent methods solve the issues found when building a complete system. In order to write several parts of this chapter we made use of [18, 29] as a especially rich source of documentation.

2.1 Pedestrian detection system architecture

In this section we describe the components of a complete pedestrian detection system (see Fig. 2.1), leaving out the tracking and application modules. We start in section 2.1.1 by describing the data acquisition component, which addresses the issue of which sensors are used by the system. Although this component is not shown in Fig. 2.1, it is a fundamental part of the system, as different sensors provide different input images to the system. Next, we see in section 2.1.2 the preprocessing stage, which addresses issues coming from the exposure, gain and calibration. Section 2.1.3 describes the foreground segmentation module, which provides the regions of the image where the presence of a pedestrian is plausible. Typically, this module outputs a set of candidate windows that are later evaluated by the classifier. Section 2.1.4 describes the classification component. This part of the system comprises indeed two submodules, which are the visual representation and the classifier applied to this representation. There have been a large variety of methods proposed in the literature for making the classification stage efficient and accurate, and we provide a review of the most important works in section 2.1.4 Finally, in section 2.1.5 we see the last component of the system, which is the

[1] In our case we do not include the tracking stage.

detection refinement. This typically consists of grouping the multiple overlapped detections obtained from the classification stage, in order to obtain a spatially precise set of detections.

2.1.1 Data Acquisition

Sensors can be divided into two different types, active and passive ones. While the first ones, for example, radars, lidars and laser scanners, transmit signals in a specific direction or working space to later obtain their reflection, the second ones, such as cameras (CCD and CMOS) composed by millions of photosensitive components, capture light and convert it into an electronic signal. As pointed by Gerónimo *et al.* [30], systems based on active sensors usually encounter problems when distinguishing pedestrians from other objects in urban scenarios. This is mainly due to the reflectance properties. Besides, the cost of these sensors is much higher than the cost of the passive ones. Hence, most of the pedestrian detection systems in the literature rely on passive sensors [29].

Cameras can work in a wide electromagnetic spectrum range. Based on this range, cameras are divided into three different subsets: Visible spectrum (VS), near infrared (NIR) and thermal infrared (TIR). Figure 2.2 shows the same scene using VS and TIR sensors. Since pedestrian detection is typically addressed in daytime, VS cameras are by far the most used [29]. Nevertheless, NIR and TIR cameras are also used for detecting pedestrians under other circumstances[2], such as night time, fog or heavy rain. During daytime NIR and TIR sensors can also be used to support VS cameras. When using a specific set of headlights, *e.g.*xeon ones, VS cameras are capable of capturing a near infrared range, which gives an additional advantage to the VS cameras with respect to the rest. The methods we introduce in next sections work in the visible spectrum.

2.1.2 Preprocessing

In most recent works, issues coming from exposure, gain or calibration are rarely mentioned by authors. However, the posterior stages involved during the detection procedure may be affected if these problems are not handled properly. Only few works have addressed this problem. For instance, Nayar *et al.* [50] proposed some locally adaptive dynamic range approaches in order to adjust the exposure. Recent cameras are starting to capture images using a high dynamic range (HDR), which provides a highly contrasted images and is determinant in order to reduce the impact of the issues coming from the exposure. Note that many detection systems fail on poorly contrasted images. Hence, incorporating HDR images is key for improving the system performance. Besides, to handle exposure, recent cameras use more sophisticated automatic aperture systems, which benefits the final performance.

[2] Depending on the situation, a specific active sensor can be more appropriate than the others.

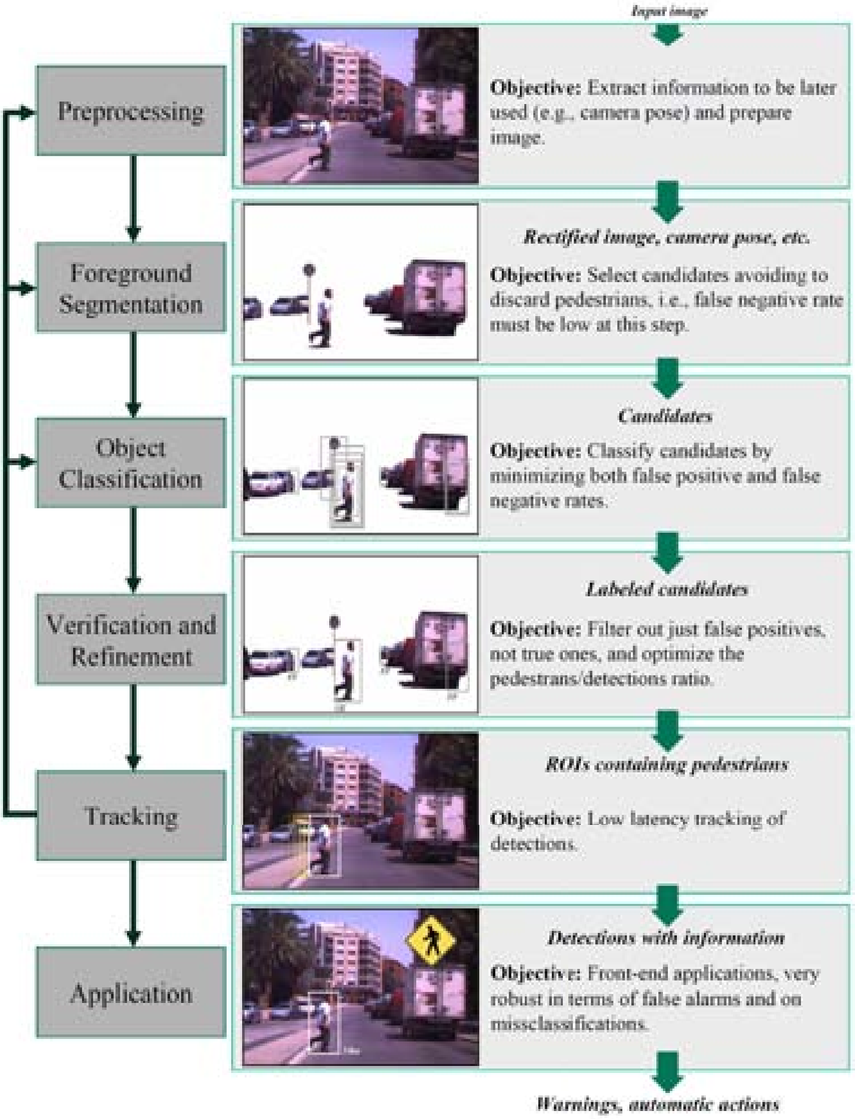
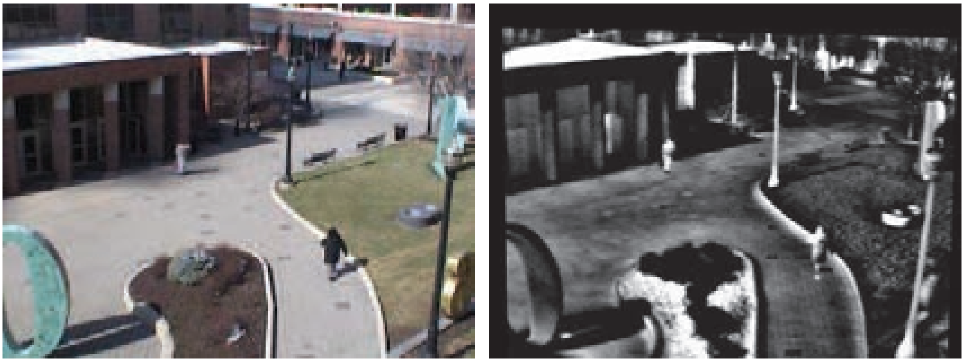

Input image
Preprocessing
Objective: Extract information to be later used (e.g., camera pose) and prepare image.
Foreground Segmentation
Rectified image, camera pose, etc.
Objective: Select candidates avoiding to discard pedestrians, i.e., false negative rate must be low at this step.
Object Classification
Candidates
Objective: Classify candidates by minimizing both false positive and false negative rates.
Verification and Refinement
Labeled candidates
Objective: Filter out just false positives, not true ones, and optimize the pedestrans/detections ratio.
Tracking
ROIs containing pedestrians
Objective: Low latency tracking of detections.
Application
Detections with information
Objective: Front-end applications, very robust in terms of false alarms and on missclassifications.
Warnings, automatic actions

Figure 2.1: On-board pedestrian detection system architecture proposed by Gerónimo *et al.* [29] (figure from the original thesis).


(a) VS

(b) TIR

Figure 2.2: Out-door scene during daytime. (a) using a VS camera sensor, and (b) using a TIR camera sensor (images from [12]).

2.1.3 Candidate generation

In this section we focus on the most widely used candidate generation technique, which is a simple sliding window. In fact, the most successful methods in the literature rely on a sliding window strategy. Non sliding window approaches such as segmentation [33] or key point [40, 65] based methods, as pointed by Dollár *et al.* [18], tend to fail for low to medium resolution settings. Almost all the sliding window based methods, except one of them [2], use a pyramid in order to handle different detection scales [10]. In the case of Benenson *et al.* [2], the authors propose instead of resizing the image to apply multi-scale classifiers over the image for efficiency purposes.

The common sliding window approach yields a set of candidates to be sent to the classification stage. Due to the exhaustive scanning, this technique brings two main disadvantages: 1) the large number of possible candidates (usually thousands of them), which makes it infeasible to achieve a real-time performance, and 2) irrelevant regions are also scanned, which may increase the number of false positives and the computational time that implies evaluating these candidates. Therefore, reducing the number of candidates and avoiding irrelevant regions of the image is crucial at this point. For this purpose, several authors have proposed different segmentation techniques which can be categorized into three different types: 2D, stereo and motion. In the next section we describe several state-of-the-art detection systems in which some of the previous techniques are integrated.

2.1.4 Classification

After the candidate generation step, each window needs to be classified. The common approach in this phase consists in using a discriminative model based on a learning algorithm previously trained with examples and counterexamples. These methods define a feature space in which each positive and negative training example is represented by a descriptor. Then, the algorithm, using the previous representation space, builds a model to discern between the positive and negative examples. Finally, using the learnt model the system classifies each candidate window.

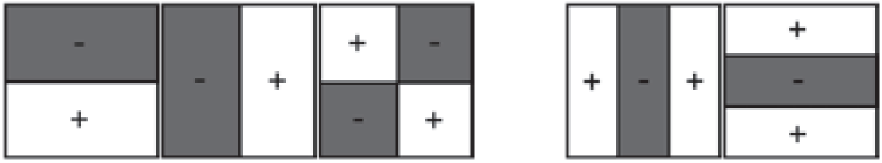
As introduced in Chapter 1, most of the works in the literature focus on two main objectives: 1) designing new features, and 2) developing machine learning algorithms. As a result, various representations and learning algorithms have been proposed for pedestrian detection [18, 19, 30]. Concerning features the most relevant works include the use of Haar wavelets, and other Haar-like features; edge orientation histograms (EOH); histograms of oriented gradients (HOG) inspired on the scale-invariant feature transform (SIFT); local receptive fields (LRF); integral histograms; colour histograms such as color-self similarity (CSS); covariance features; local binary patterns (LBP); histograms of flow (HOF) coming from motion; stixels coming from depth; and other features coming from orientations, depth, motion and segmentation. Some authors have proposed combining features coming from different sources showing in some cases an increase of the performance [44, 81]. Regarding the learning algorithms, boosting classifiers such as AdaBoost; Neural Networks (NN); linear support vector machines (SVM); histogram intersection kernel SVM (IKSVM); latent SVM; Multiple kernel SVM; and other SVM variants have been recurrently used in the literature. Next, we review the state-of-the-art based on the previous machinery.

One of the first sliding window approaches for object detection was proposed by Papageorgiou *et al.* [57] at the beginning of last decade. This method relies on Haar wavelet features together with a non-linear SVM using a quadratic function to map the features into a higher dimensional representation space. The Haar wavelet features consist in an overcomplete dictionary of local, oriented, multi-scale intensity differences between adjacent rectangular regions. This type of representation can be interpreted as a derivative at a large scale. After this work, the same authors together with Mohan [48] presented a part-based method, using again Haar features and a classification procedure that consists of two layers: in the first layer, a quadratic SVM classifier is used in order to classify the parts of the pedestrian, and in the second layer a linear SVM classifier is used in order to obtain a single score for the whole pedestrian based on the classification scores of the parts.

Zhao *et al.* [92], introduced a *feed forward neural network* with gradient magnitude features. The main core of the classifier is a NN fed with gradient information coming from stereo-based segmentation.

Later, Viola and Jones [78] proposed AdaBoost cascades, which consists of several rejection levels of AdaBoost classifiers. Additionaly, the authors incorporate integral images for fast computation of Haar-like features. In [80], the same authors use Haar-like features to model motion information. These Haar-like features are an extension of the original dictionary (see Fig. 2.3). Mikolajczyk *et al.* [47] introduced a coarse-to-fine cascade approach using AdaBoost based on orientation features in order to learn different body parts. Then, the different part detectors are assembled into a probabilistic framework.

In [66] Sashua *et al.* , using SIFT-inspired features [43], introduced a part-based approach. In this case, an AdaBoost classifier is used in order to classify each candidate window. As weak classifiers the authors proposed a total of 117 classifiers coming from thirteen



(a) (b)

Figure 2.3: (a) Original Haar features, and (b) Haar-like features.

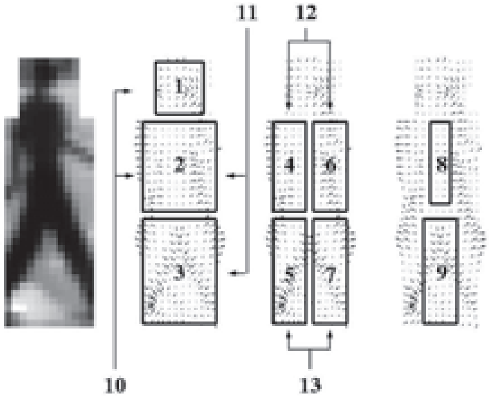


Figure 2.4: The configuration of the nine sub-regions defined in the part-based approach proposed by Shashua *et al.* [66] over the gradient image (figure extracted from [66]).

original part classifiers (see Fig. 2.4). Each one of the thirteen part classifiers was trained with nine different training subsets.

Dalal and Triggs [10] presented a new descriptor named HOG and a linear SVM as the learning method. The HOG features are obtained by dividing the window into a block-based structure. Then, each block is divided again into 2×2 cells from which a histogram of gradient orientations is extracted. The final descriptor is the result of concatenating all these histograms. Zhu *et al.* [93] proposed to use the AdaBoost as a feature selector over a large set of HOG blocks of different sizes to later classify each window via a rejection cascade. This work achieved similar performance when compared with [10], but with less computation time.

Wu and Nevatia [87] presented a new part-based method based on a new set of features, termed edgelets, which encode the intensity and the shape information of an edge (see Fig. 2.5). Each part is then trained using these features and AdaBoost. Finally, the part responses are combined by a probabilistic approach in which cases of multiple and possibly inter-occluded humans are considered. This method focused on detecting multiple and partially occluded people in cluttered scenes. Later, the same authors also introduced a method

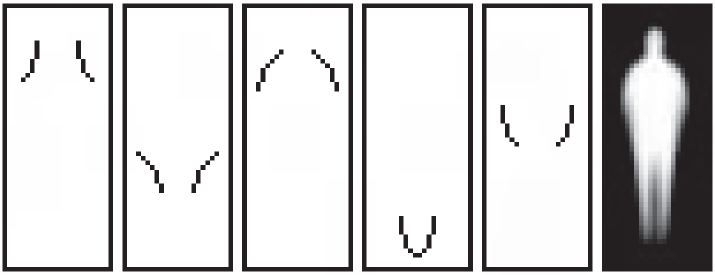


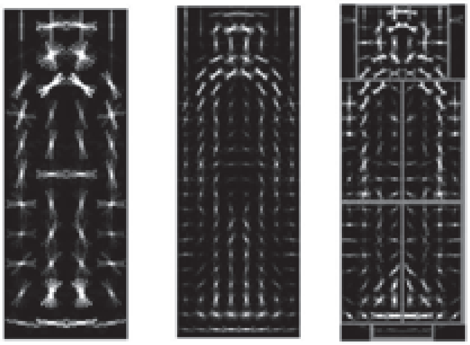
Figure 2.5: The first five features (edgelet) selected by the Adaboost approach in [86], and the prior distribution (images extracted from the original paper).

that combines object detection and segmentation based on edgelets and AdaBoost [86], and combined edgelet and HOG features with AdaBoost and SVM learning algorithms for both VS and TIR imagery [91].

Tuzel *et al.* [72] presented a new approach based on the covariance matrices as object descriptors using a LogiBoost algorithm on Riemannian manifolds. In their work the authors propose the use of Riemannian manifolds in order to deal with the fact that the used descriptors (covariance matrices) do not live in a vector space. Later works such as [71] have used a similar idea to build a part-based approach.

In [55], Pang *et al.* used multiple instance learning (MIL) through a boosting learning approach named Logistic Multiple Instance Boost. In this work, in order to efficiently use the histogram feature, the method utilizes a decision stump based on a graph embedding. MIL has also been used in other works for automatically determining the position of parts without any kind of supervision [25, 42]. Felzenszwalb *et al.* [25] proposed a deformable part-based approach that models the unknown part positions as latent variables in an SVM framework, termed LatSVM. Recently, several authors have been extending this work. For instance, Park *et al.* [58] extended this work to a multi-scale approach (see Fig. 2.6), Ouyang *et al.* [54] to a deep model approach with occlusion handling, and Pedersoli *et al.* [60] proposed a coarse-to-fine approach to accelerate the detector.

Maji *et al.* [45] proposed an approximation of the histogram intersection kernel for use with SVMs (HIKSVM) together with multi-level oriented edge energy features (similar to HOG, but simpler). The main advantage of the HIKSVM is based on the fact that it can be computed in logarithmic time, or approximately constant time, while consistently outperforming the linear kernel. Following this insight, Walk *et al.* [81] made use of the HIKSVM accompanied with multiple features. In particular, the authors proposed the CSS descriptor combined with HOG and HOF features, this last features proposed by Dalal in [9]. Wang *et al.* [82], combine HOG and LBP, and make use of linear SVM as classifier. In this case, the authors also introduce an occlusion handling approach based on the linearity of the algorithm and the block-based structure. Similar to HOG, the LBP is computed in the entire window divided by cells.



(a) (b) (c)

Figure 2.6: Multi-resolution templates based on the HOG descriptor proposed originally by Dalal and Triggs [10]. (a) low resolution template, (b) high resolution template, and (c) high resolution template with parts (original figure from [58]).

In [77], Vedaldi *et al.* proposed a three layer cascade in which the first stage is evaluated by a fast linear SVM, the second one by a quasi-linear SVM and the final one by a non-linear SVM. This framework permits to apply a MKL classifier in a reasonable time while outperforming previous approaches.

Dollar *et al.* [15] proposed an extension of [79] in which Haar-like features are computed over LUV colour channels, grayscale, gradient magnitude and gradient magnitude quantized by orientation (see Fig. 2.7). This work at the same time has been recently extended. In the first case, the same authors [14] sped up the previous framework. The authors named it the Fast Pedestrian detector in the West (FPDW). Basically, the feature responses computed at a single scale are used to approximate feature responses at nearby scales, which considerably reduces the computational time. In the second case, the same authors, based on the idea that detector responses at nearby locations and scales are correlated, introduced a crosstalk cascade [13] in which nearby detectors are sharing information between each other to achieve a higher efficiency. In the third and last case, Benenson *et al.* [2] extended [14]. In particular, Benenson *et al.* proposed a pedestrian detection system based on a soft cascade approach without image resizing and the use of stixels (efficient features coming from detph). In this case, the authors use a multi-scale classifier with a fixed number of scales. Then, to approximate nearby scales in between the defined ones, authors follow a similar procedure compared to [14].

2.1.5 Detection refinement

Once all the possible candidates have been classified, the next step consists of grouping multiple overlapped detections to provide one single detection per target. For such task, a simple algorithm that provides one detection per pedestrian is clustering. The two non-maximum

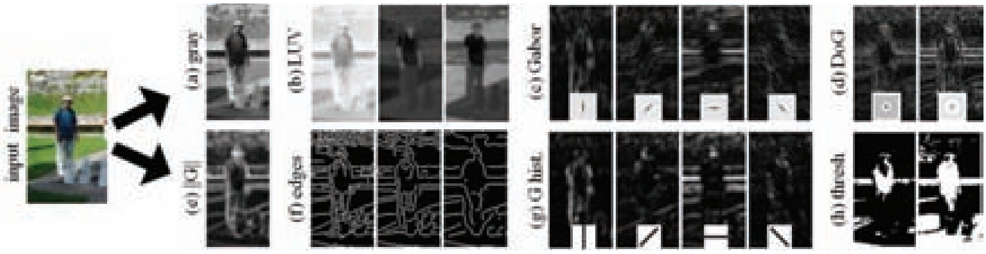


Figure 2.7: From left to right, the original image; the multiple image channels of the input image; and the features extracted from the different channels (figure from [15]).

suppression (NMS) clustering techniques most extended in the literature rely on: the Mean Shift (MS) algorithm [9] and the pairwise max (PM) suppression [25]. In the former, the algorithm finds the minimum set of detection windows which best adjust to the targets in the image. The second one makes a pairwise comparison of all the detections, and if the area of overlap between a pair exceeds a certain threshold, the one with lower confidence is suppressed.

2.2 Evaluation Methodologies

The standard evaluation methodology to assess the performance of the different pedestrian detectors is usually known as *per-image evaluation*. Some authors also evaluate the performance of the classifier, termed *per-window evaluation*. Both evaluation methodologies have been recurrently used in the literature [18, 19, 30]. However, in object detection, the per-image evaluation tends to be the standard evaluation methodology [63] because the main concern in real applications is the performance at the detection level. The former evaluation type provides a curve that depicts the tradeoff between detection rate (*i.e.*, the percentage of mandatory pedestrians that are actually detected) and the number of false positives per image (FPPI). The second evaluation type, given a classifier, yields the trade-off between miss classifications and the number of false positive windows (FPPW). Depending on the author, instead of detection or classification rate the plot show the missrate vs the FPPI/FPPW. The same happens with the axes, some authors set them into the logarithm scale and other do not. Additionally, the average performance of each curve can be shown. It is also worth to mention that some authors differ on the evaluation range according to the FPPI and FPPW. In this thesis, we initially focus on the range 10^{-1} to 10^{0} as is more interesting from the real application point-of-view. Later, in Chap. 5, we show results in the range 10^{-2} to 10^{0} for benchmarking purposes following the recent tendency in the literature.

In order to identify a window as a true positive or a false positive, the most common criterion used in object recognition is the PASCAL VOC criterion [23]. Given a detection window W_d, and a window labelled as mandatory pedestrian W_l,the following ratio is computed:

$$r(W_d, W_l) = \frac{a(W_d \cap W_l)}{a(W_d \cup W_l)} \quad . \tag{2.1}$$

Based on this ratio, W_d is considered a true positive if there is a W_l for which $r(W_d, W_l) > 0.5$, and otherwise, W_d is considered a false positive. Undetected mandatory pedestrians count as false negative, *i.e.*, those W_l for which there is no W_d with $r(W_d, W_l) > 0.5$. If, given a W_l, more than one W_d passes the true positive criterion (*i.e.*, multiple detections), only one of them is considered, and the rest are considered as false positives. Note that such criterion usually affects training because of the bootstrapping.

Chapter 3

Exploring virtual worlds

Over the last years video games have achieved a high realism thanks to the recent advances in computer graphics. In this chapter we explore the potential of using virtual worlds for pedestrian detection. We first assess the viability of using virtual data for training a pedestrian detector by comparing it with a real detector in real images. This is done by first developing an urban scenario and then recording several video sequences to later create the dataset to train the virtual detector. Moreover, we investigate other interesting issues. In particular, we explore the impact in the detection performance with respect to the different number of virtual models, the total number of examples and the pose distribution. To validate our experiments we make use of one of the largest datasets in the literature [19] and for the detector a linear SVM with HOG features, the machinery most used in the literature. The results obtained in this chapter demonstrate that a virtual detector can be successfully applied in a real scenario. Besides, the experiments conducted in this work provide several useful insights related to the number of models, examples and the pose distribution. These last contributions bring new possible lines of research.

3.1 Introduction

State-of-the-art detectors rely on machine learning algorithms trained with labelled samples, *i.e.*, *examples* (pedestrians), and *counterexamples* (background). Therefore, in order to build robust pedestrian detectors the quality of the training data is fundamental. Last years various authors have publicly released their pedestrian datasets [10, 17, 19, 30, 84] which have gradually become more challenging (bigger number of samples, new scenarios, occlusions, etc.). In the last decade, the traditional research process has been to present a new database containing images from the world, and then researchers developed new and improved detectors [18, 19, 41, 72]. In this chapter, we explore the possibilities that a virtual *computer generated* database, free of real-world images, can offer to this process.

The use of Computer Graphics in Computer Vision is not novel. Grauman *et al.* [32] exploit computer generated images to train a probabilistic model to infer multi-view pose estimation. Such generated images are obtained by using a software tool. More specifically, the shape model information is captured through different multiple cameras obtaining the contour of the silhouettes simultaneously, and the structure information is formed by a fixed number of 3D body part locations. Later, shape and structure features are used together to construct a prior density using a mixture of probabilistic PCAs. Finally, given a set of silhouettes a new shape's reconstruction is obtained to infer the structure. Broggi *et al.* [5] use synthesised examples for pedestrian detection in infrared images. More specifically, a 3D pedestrian model is captured from different poses and viewpoints in which the background is later modelled.

Finally, instead of following a learning-by-examples approach to obtain a single classifier, a set of templates is used by a posterior pedestrian detection process based on template matching. Enzweiler *et al.* [20] enlarge a set of examples by transforming the shape of pedestrians (labelled in real images) as well as the texture of pedestrians and background. The pedestrian classifier is learnt by using a discriminative approach (NNs with LRFs and Haar features with SVM are tested). Since these transformations encode a generative model, the overall approach is seen as a generative-discriminative learning paradigm. The generative-discriminative cycle is iterated several times in a way that new synthesised examples are added in each iteration by following a probabilistic selective sampling to avoid redundancy in the training set. These examples are later used to train a model to be used in a detector. Marín *et al.* [46] use a commercial game engine to create a virtual pedestrian dataset to train a synthetic model to test in real images. Then, they show the comparison between real and virtual models revealing the similarity of both detectors in terms of HOG features and SVM classifier.

More recently, Pishchulin *et al.* [61] employ a rendering-based reshaping method to generate a synthetic training set using real subjects (similar to [20]) from only few persons and views. In this case, they collected a dataset of eleven subjects each represented in six different poses corresponding to a walking cycle. Eight different viewpoints are then used to capture each pose. Later, they gradually change the height of each pose (15 higher/15 smaller) to obtain 20400 positive examples in total. Finally, they explore how the number of subjects used during the training process affects the performance as well as the combination between real and synthesised models. The same authors [62] also explore the use of synthetic data obtained from a 3D human shape model in order to complement the image-based data. Such 3D human shape and pose model is obtained through a database of 3D laser scans of humans which describes shape and pose variations. Similar to [61] best performance is obtained when training models in different datasets and then combining them.

The reviewed proposals can be divided into the using synthesised examples coming from real data and the ones using only virtual examples. While promising detectors based on synthesised examples are still needed of real images in which models are obtained through sophisticated systems [61, 62], the virtual worlds generated using just synthetic data offer a large number of available models and possibilities without using real data. In this chapter we focus on learning pedestrian models in such virtual worlds to be used in real world detection and how different settings perform when testing in real data. In particular, following the approach in [46] we learn such appearance using virtual samples in order to detect pedestrians in real images (Fig. 3.1). Besides, we extend the approach published with specific analysis on the required number of virtual models and training examples to get a satisfactory performance, and present new results on how the pose influences the performance.

The process is as follows. We record training sequences in realistic virtual cities and train appearance-based pedestrian classifiers using HOG and linear SVM, a baseline method for building such classifiers that remains competitive for pedestrian detection in the ADAS context [18, 19]. As Marín *et al.* we test such classifiers in the same publicly available dataset, Daimler AG [19]. The obtained results are evaluated in a per-image basis and compared with the classifier obtained when using real samples for training. In this work, we specially focus on exploring the impact in the performance w.r.t the different number of virtual models, the total number of examples and the pose distribution.

The structure of the chapter is as follows. Section 3.2 introduces the datasets used for training (real world and virtual world ones) and testing (only real world images). Section 3.3 details the conducted experiments, which use the real- and virtual-pedestrian models in a complete detection system. Section 3.4 presents the results and corresponding discussions. Finally, section 3.5 summarizes the conclusions and future work.

3.2 Datasets

The lack of publicly available large datasets for pedestrian detection in the ADAS context has been a recurrent problem for years [18, 19, 30]. For instance, INRIA dataset [10] has been the most widely used for pedestrian detection. However, it contains photographic pictures in which people are mainly close to the camera and in focus. Moreover, there are backgrounds that do not correspond to urban scenarios, which are the most interesting and difficult ones for detecting pedestrians from a vehicle.

Fortunately, three more adapted datasets for the ADAS context have recently been made publicly available. They have been presented by Caltech [17], Daimler [19], and the Computer Vision Center [30]. In the current work, we perform the experiments in Daimler dataset since it comes from one of the most relevant automotive companies worldwide, thus, we can expect the images to be quite representative for ADAS. Our proposed virtual dataset is also focused on ADAS images, but acquired from a virtual car in a computer graphics generated world. In the next sections we summarise the details of both datasets.

3.2.1 Real images

We summarize the main characteristics of Daimler's dataset. In fact, it consists of a training set and different testing sets.

Training set

The images of this set are grayscale and were acquired at different times of day and locations (Fig. 3.3).

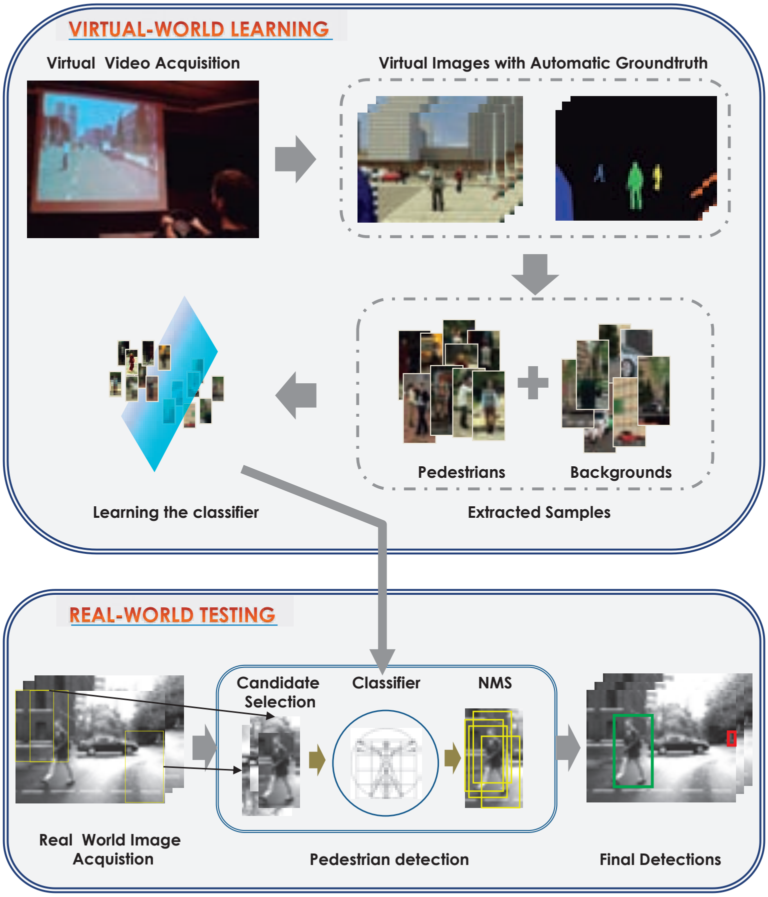


Figure 3.1: Training a pedestrian classifier in virtual-world scenarios for a pedestrian detector operating in real world.

Examples. The original training frames with pedestrians are not publicly available, but cropped pedestrians are. From 3915 manually labelled pedestrians, 15660 were obtained by applying small vertical and horizontal random shifts (*i.e.*, jittering) and mirroring, and then put publicly available. The size of each cropped example is 48×96 pixels, which comes from the 24×72 pixels of the contained pedestrian plus an additional margin of 12 pixels per side. All the original labelled pedestrians are at least 72 pixels high, thus, some of the samples come from downscaling but none from upscaling. All the samples contain pedestrians that are upright and not occluded.

Counterexamples. 6744 pedestrian-free frames were delivered. Their resolution is 640×480 pixels. Thus, to gather cropped counterexamples these frames must be sampled. Conceptually, the sampling process we use can be described as follows. We need counterexamples of the same dimensions than the cropped pedestrian examples, *i.e.*, 48×96 pixels. Therefore, we can select windows of size $48k^i \times 96k^i$ pixels, where k is the scale step (1.2 in our case) and $i \in \{0, 1, 2, ...\}$, provided that they are fully contained in the image we are sampling. Then, we can downscale the counterexamples by a factor k^i using, for instance, bi-cubic interpolation. In practice, we implement this sampling idea by using a pyramid of the frame to be sampled and then by cropping windows of size 48×96 pixels at each layer [9], which is closely related to the scanning strategy used by the final pedestrian detector (Sect. 3.3).

Testing set

The testing set consists in a sequence of 21790 grayscale frames of 640×480 pixels. The sequence was acquired on-board while driving during 27 minutes through urban scenarios. Moreover, this testing set does not overlap the training set. The testing set includes 56492 manually labelled pedestrians. The labels contain also an additional information indicating whether they are of *mandatory* detection or not. Basically, the pedestrians labelled as non-mandatory are those either occluded, not upright, or smaller than 72 pixels high. There are 2459 mandatory pedestrians in total. Frames of the training set can be seen with overlayed results in Sect. 3.4 (Fig. 3.11).

3.2.2 Virtual images

In order to obtain virtual images, the first step consists in building virtual scenarios. We have carried it out by using the video game Half-Life 2 [70]. This game allows to run maps created with an editor named *Hammer* (included in the Valve's software package), as well as to add modifications (*a.k.a. mods*). We use Hammer to create realistic virtual cities with roads, streets, buildings, traffic signs, vehicles, pedestrians, different illumination conditions, etc. Once we start to *play*, the pedestrians and vehicles move through the virtual city by respecting physical laws (*e.g.*, pedestrians do not float and cannot be at the same place than other solid objects at the same moment) as well as by following their artificial intelligence (*e.g.*, vehicles move on the road).

In order to acquire images in virtual scenarios we use the *mod* created by the company ObjectVideo. Taylor *et al.* [69] show the usefulness of such a *mod* for designing and validating people tracking algorithms for video surveillance (static camera). A relevant functionality consists in providing pixel-wise groundtruth for human targets (Fig. 3.2). However, since the aim in [69] is to test algorithms under controlled conditions, all the work is done with virtual scenarios without considering real world images. Thus, the work we present in this chapter is not actually related to [69] apart from the use of the same Half-Life 2 *mod*.

In fact, we created an application to augment the functionalities of such a *mod* with the possibility of moving a virtual camera as if we were driving. In particular, in order to *drive* through a virtual city we introduced a camera with a given height as well as pitch, roll and yaw angles, and then we move it keeping these parameters constant. The only constraint that must be ensured is that these parameters are compatible with a camera forward facing the road from inside a vehicle, for instance, as if it was placed at the rear view mirror behind the

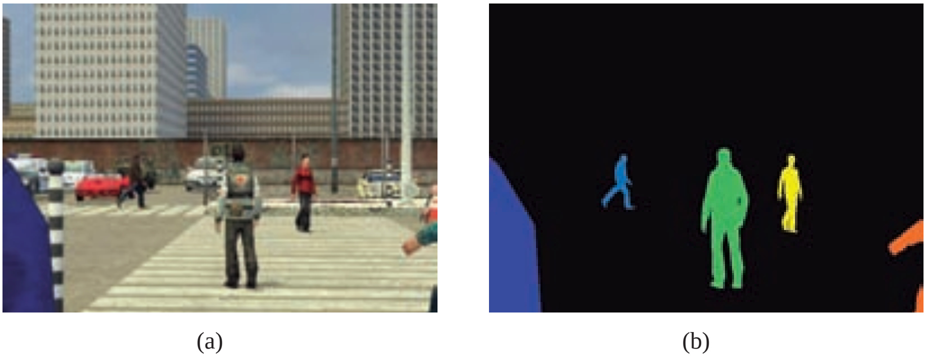


Figure 3.2: Virtual image with corresponding automatically generated pixel-wise groundtruth for pedestrians. (a) Original image, and (b) the pixel-wise groundtruth image, where each pedestrian has a different color label.

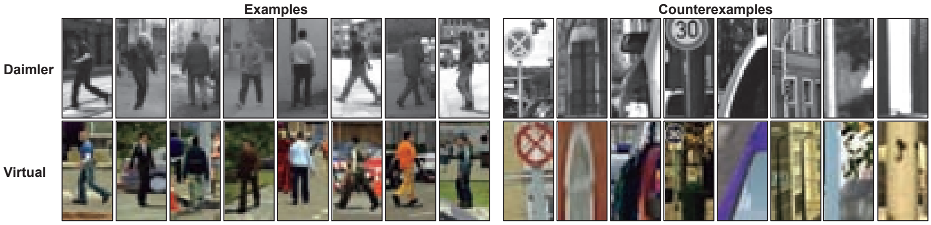


Figure 3.3: Examples and counterexamples taken from real images (Daimler's dataset) and from virtual ones.

windshield. Finally, in order to emulate the dataset of Daimler, we set the resolution of our virtual camera to 640×480 pixels.

As in [46] we created four virtual cities which, in fact, correspond to a single one in terms of graphical primitives, *i.e.*, we only changed some building, ground and object textures so that they look different as well as the overall illumination to emulate different daytimes. In order to introduce more variability in terms of pedestrians (aspect ratio and clothes) in these cities with respect to the ones in [46], we added new human models into the game, finally achieving a set of 60 different pedestrians (see Figure 3.4). Note that since they are articulated moving models seen from a moving camera, each virtual pedestrian can be imaged with different poses and backgrounds. Figure 3.3 plots samples of the virtual training set that we describe in the rest of this section.

Examples. We recorded forty video sequences by driving through the virtual cities. In total we obtained 100075 frames, at 5fps, which corresponds to 5 hours, 33 minutes and 35

	Training Set Cropped pedestrians (jitter and mirroring included) & Background frames	**Training process** 1st round: cropped pedestrians / cropped background & Bootstrapping: additional cropped background	**Testing sets**
Daimler	15660 & 6744	15660 / 15560 & All False Positives	Full set: 21790 frames Mandatory set: 973 frames
Virtual	1440, 4320, 12960 & 1219	1440, 4320, 12960 / 2438 & All False Positives	

Table 3.1: Training and testing settings for our experiments.

seconds. The virtual car was driven without any preferred *plan of route*. Along the way we captured images containing pedestrians in different poses and with different backgrounds. Since we can obtain the groundtruth of the virtual pedestrians automatically, we consider only those upright, non-occluded, and with a height equal or larger than 72 pixels in the captured images (pedestrians *taller* than 72 pixels require further down scaling as we will see) like in the training set of Daimler database. This gives us 7973 pedestrians to consider in order to construct the set of examples for training. It is worth mentioning that in order to have automatically labelled examples analogous to the manually labelled ones of Daimler's training set (*i.e.*, with the torso centered with respect to the horizontal axis), we cannot just take the bounding boxes corresponding to the pixel-wise groundtruth. Instead, we apply the following process to each virtual pedestrian:

1. For some pedestrian poses, the bounding box obtained from the pixel-wise groundtruth is such that the torso is not well centered in the horizontal axis, so we automatically correct this. More specifically, we project the pedestrian groundtruth into its horizontal axis. Then we take the location of the maximum of the projection as the horizontal center of the torso. Finally, we shift the initial pixel-wise bounding box so that its horizontal center matches the one of the torso.
2. Then, we modify the location of the sides of the pedestrian bounding box preserving the previous re-centering, but enforcing the same aspect ratio and proportional background margins than the pedestrians in the training set of Daimler (*i.e.*, 24/72 and 12/72, respectively). This is automatically achieved by simply applying standard rule of proportionality.
3. The bounding box at this point can still be larger than the canonical bounding box of the pedestrian examples of Daimler's training set, *et al.* , larger than 48×96 pixels. Thus, the final step consists in performing a down scaling using bi-cubic interpolation.

Counterexamples. In order to collect the counterexamples for training, we used the same four virtual cities than to obtain the examples, but now without pedestrians inside, *i.e.*, we drove through these *uninhabited* cities to collect pedestrian-free video sequences. We collect frames from these sequences in a random manner but assuring a minimum distance of five frames between any two selected frames, which is a simple way to increase variability. In fact, the images where taken with an initial resolution of 720×1280 pixels, and laler scaled to 480×640. In total we have 1219 frames without virtual pedestrians, so they can be sampled to gather virtual counterexamples. The sampling process is, of course, the same than the one previously described for Daimler (Sect. 3.2.1).

The video sequences were taken with the highest graphics quality allowed by the game engine. In particular, the changed settings were: the model detail, which controls the number of polygons and extra detailing; the texture detail; the color correction, which increases the realism; the antialiasing mode, which smooths the jagged lines when rendering; the high definition rate (HDR), which gives a higher vivid and contrasting lighting; and the filtering mode, which determines how clear is the detail of the textures as they fade into the distance to the camera.

3.3 Experiment design

3.3.1 Pedestrian detector components

In order to detect pedestrians we need a pedestrian classifier learnt from the training set by using specific *features* and a *learning machine*. With this classifier we *scan a given image* looking for pedestrians. Since multiple detections can be produced by a single pedestrian, we also need a mechanism to *select the best detection*. The procedures we use for features extraction, machine learning, scanning the images, as well as selecting the best detection from a cluster of them, are the same no matter if the classifier was learnt using virtual images or real ones (*et al.* , from Daimler). Let us briefly review which are these components in our case.

Features and learning machine.

The combination of the histograms of oriented gradients (HOG) features and linear SVM learning machine, proposed by Dalal *et al.* in [10], has been proven as a competitive method to detect pedestrians in the ADAS context [19]. Similar conclusions are also obtained when using a large ADAS-inspired dataset for testing in [17]. In fact, recent proposals that outperform HOG/linear-SVM when using the INRIA dataset include both HOG and linear SVM as core ingredients [82]. Thus, we think that HOG/linear-SVM stands as a relevant baseline method for learning pedestrian classifiers, so we use it in our experiments. In particular, we follow the settings suggested in [10] for both HOG and linear SVM, as it is also done in [19]. A minor difference comes from the fact that in Daimler's datasets the images are grayscale while the virtual images are RGB. This issue is easily handled by just taking at each pixel the gradient orientation corresponding to the maximum gradient magnitude among the RGB channels (as in [10] for INRIA dataset).

Scanning strategy.

We use the first type of sliding window described in Chap. 2, which consists in a sliding window approach implemented through a pyramid to handle different detection scales. [9]. We could consider the sliding window parameters found in [19] as the best in terms of pedestrian detection performance for the so-called *generic pedestrian detection* case with Daimler's testing set. However, at this stage we followed the settings proposed in [9].

Selecting the best detection.

To group multiple overlapped detections and (ideally) provide one single detection per target we follow an iterative confidence- and overlapping- based approach, *i.e.* a kind of *non-maximum-suppression*. This technique, used by I. Laptev in [38], consists of four basic steps: 1) create a new cluster with the detection of highest confidence; 2) recompute the cluster with the mean of the detections overlapping the new cluster; 3) iterate to step 2 until the cluster position does not change; 4) delete the detections contained in the cluster and iterate to 1 while there are detections. This NMS approach has been also used in the bootstrapping stage.

For us, as described in Chap. 2, a pedestrian detector consists of a pedestrian classifier, plus the above seen techniques of sliding window and non-maximum-suppression. Therefore, we are not considering tracking of pedestrians, but we think this does not affect the aim of this chapter.

3.3.2 Training

Training with Daimler dataset

We train the HOG/linear-SVM classifier with the 15660 provided examples and collect also 15560 counterexamples by sampling the 6744 provided pedestrian-free images as explained in Sect. 3.2.1, which is the approach followed in [19]. In addition, we also apply one *bootstrapping* step, *i.e.*, with the first learnt classifier we run the corresponding pedestrian detector on the 6744 pedestrian-free frames and collect false positives to enlarge the number of counterexamples and retrain. This bootstrapping technique is known to provide better classifiers [10, 19, 49] than simply train once. In our case, instead of collecting 15660 false positives, like in [19], all the false positives considered by the initial classifier are collected during bootstrapping. Thus, the final classifier is trained using 15660 examples and over 90000 counterexamples. By following such technique we found to increase the performance of the final detector.

Training with virtual dataset

Learning a classifier by using the virtual training set is analogous to the Daimler case, *i.e.*, HOG/linear-SVM and one bootstrapping stage are used. In our experiments we do not only explore the feasibility of using virtual data for learning a reliable pedestrian detector, but also the total number and variability between pedestrian models that is required and how the pose influences the detection. First, we test how many different models are at least required to obtain similar performance as the real dataset. Later, given a fixed number of models, we assess the performance of different number of examples, in particular, 1440, 4320, and 12960. Next, different adapted pose distributions are generated and compared one another. And finally, a final detector is trained using real and virtual data.

In order to assess how robust the training process is to changes, when using different training examples coming from the same virtual world, we perform a random selection of the

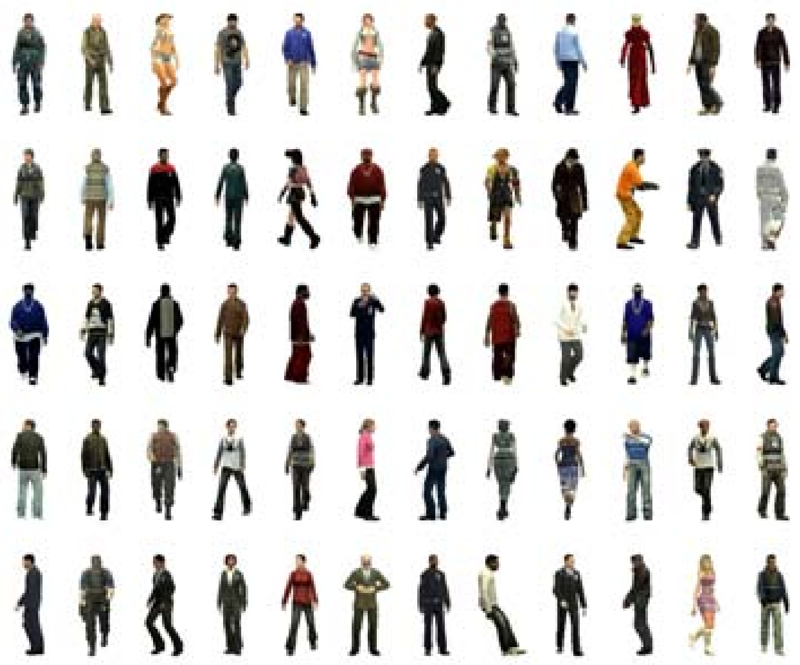


Figure 3.4: Sample of each pedestrian model used in our virtual world. In total there are sixty different models (models from garrysmod.org).

total number of labelled pedestrians into five subsets with the aim of conducting five trainings regarding each of the experiments described above in which every model is proportionality represented in each random subset. Then, from the 80000 total pedestrians annotated (≥ 72 pixels tall) to conduct our experiments we extract a thousand of them -in this case 1080, the closest multiple of total number of pedestrians, 60 (see Figure 3.4)- per training subset. In order to emulate Daimler training set, we apply two jitters and a mirroring per jitter, which means that in total every subset contains 4320 examples. Note that the randomness when all the models are used comes in terms of pose and illumination, while in the case of reduced number of models, it comes also from the pedestrian models themselves (the ones that have been selected).

Moreover, to ensure a certain variability on the subsets of examples, every subset is generated so that every pedestrian model selected has a gap of five frames between examples, thus making the differences come from pose and background.

In order to generate the different pose distribution sets, we first define the aspect-ratio of the legs in each pedestrian sample. This is the horizontal projection length of the bottom of the shape-mask image -a quarter of the image- divided by the total length of the bottom window (see bottom right image in Figure 3.6 (a)). Then, we split the aspect-ratio into nine

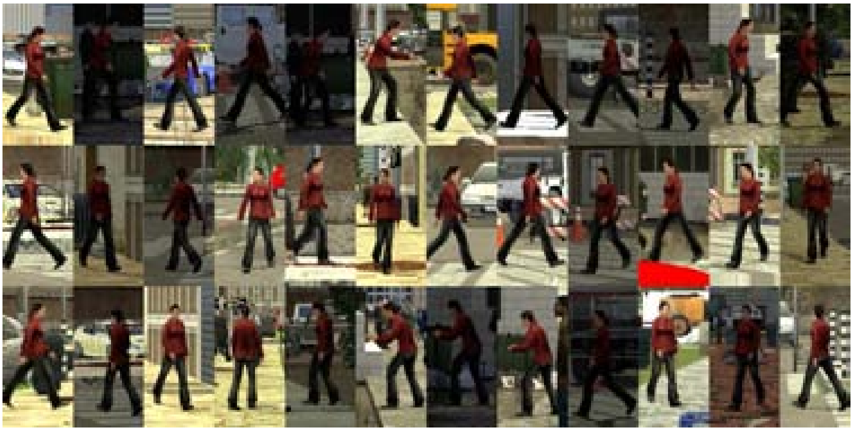


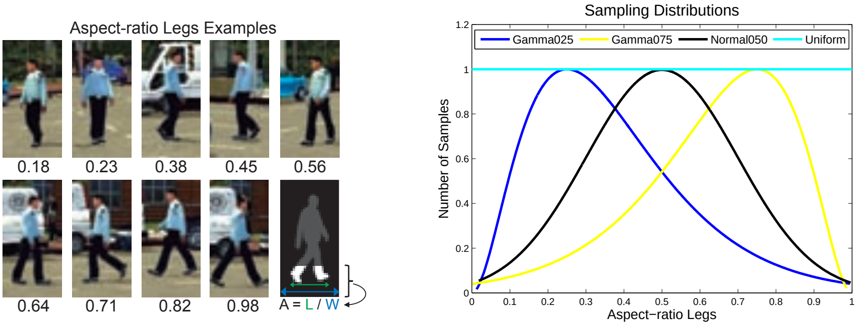
Figure 3.5: A virtual pedestrian through different backgrounds and illuminations.

different ranges. Later, we define four different distributions in terms of aspect-ratio (Uniform, Normal050, Gamma025 and Gamma075 -the numbers 025, 050 and 075 related to the distribution names, represent the peak of each distribution-), which define the way we sample the training samples with respect to their pose. Finally, we randomly sample subsets based on the adapted distributions through the defined ranges. In Figure 3.6 (b) we show the sampling distributions for Gamma025, Gamma075, Normal050, and Uniform. For instance, in Gamma025 more virtual pedestrians with closed legs than opened ones are selected, while in Gamma075 occurs the opposite.

3.3.3 Testing

In order to reduce the computational time of the experiments, rather than using the 21790 testing frames from Daimler, we rely on a representative but reduced testing set as performed in [46]. Specifically, those frames in which there is at least a mandatory pedestrian to detect (3.2.1) are first selected, then one every two frames is taken out. This final set of frames is considered as *mandatory testing set*. There are 973 of such frames and they contain 1193 mandatory pedestrians.

We use the mandatory testing set to evaluate all the pedestrian detectors associated to the classifier learnt with Daimler's training set and the same for the other detectors related to the virtual training set.



(a)

(b)

Figure 3.6: Aspect-ratio legs distributions. (a) Examples with different aspect-ratio legs values of the same model - the bottom right image in (a) shows how the aspect-ratio is computed. (b) Sampling distributions: Gamma025, Gamma075, Normal050, and Uniform.

3.4 Results

In this section we describe the obtained results following the settings summarised in the previous section.

To assess the performance of the different pedestrian detectors we use *per-image evaluation* detailed in Chap. 2. We choose to plot missrate detection rate versus FPPI both in logarithm scale. We restrict the range for plotting such curves to $[10^{-1}, 10^{0}]$, which means that we focus our interest in those regions that allow between one false positive per ten images and one FPPI. Similarly to [18], instead of using a single point on the curve to compare the performances, we compute the log-average miss rate at nine points on the curve equally distributed over the logarithmic x-axis. For the evaluation, we follow the PASCAL VOC criterion described in Chap. 2. The sliding window can be defined as a triple $(\Delta_x, \Delta_y, \Delta_s)$, in which the first two parameters denote the spatial stride, and the third parameter is the scale step. In our case, the triple used in our experiments is $(8, 8, 1.2)$.

Figure 3.7 shows the obtained performance curves of the virtual and real approaches following the PASCAL criterion (see Chap. 2). In Figure 3.7 (a) the mean and the standard deviation of the 5-experiment-based of the virtual based training are shown. The virtual approach is conducted by using a fixed number of 4320 positive samples with all the pedestrian models proportionally represented. The standard deviation illustrates how robust the approach is when generating different random subsets, being its value 0.66 points. In Figure 3.7 (b) the best curve and the worst curve are plotted versus the Daimler baseline approach. As it can be seen, the results reveal the similarity of both datasets. The difference between the best virtual-world-based curve and the real-world-based one less than one point, and comparing the worst virtual-world-based curve and the real-world-based one such difference is still less than five points when comparing the AUC average. Altogether, the results, as presented in [46], allow us to contemplate that the differences of the learnt virtual-world-based detec-

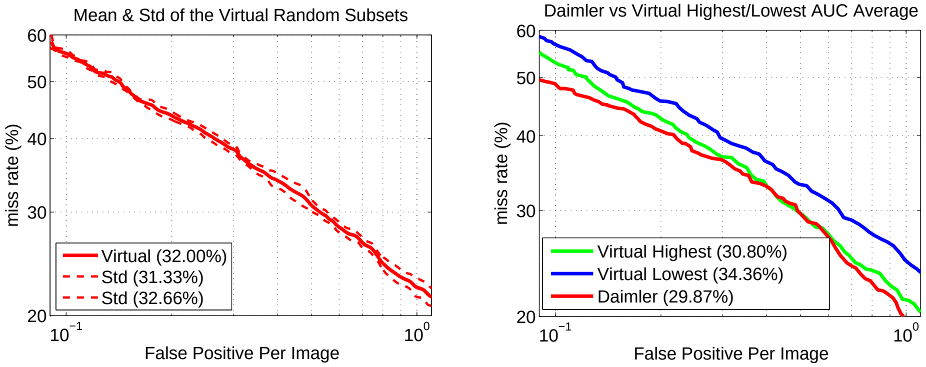
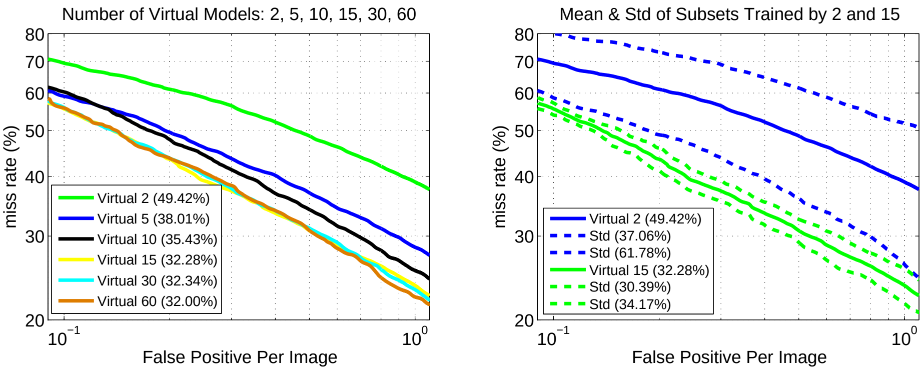


Figure 3.7: Per-image evaluation of the pedestrian detectors following PASCAL criterion [22]. Left: mean performance and standard deviation obtained in the 5 experiments with virtual samples and the sixty models over the mandatory testing set. Right: performances of virtual highest and lowest AUC average versus Daimler performance. The percentage between parenthesis next to each curve description represents the AUC average between range $[10^{-1}, 10^{0}]$.

tor and the real-world-based one are minimal in terms of performance. Figure 3.11 shows some qualitative results when using the real-world-based and virtual-world-based detectors. As expected, both detectors are quite similar in particular detections, however, they seem to slightly differ in the false positives.

Next we show the performances when using different number of models in the training procedure. Figure 3.8 (a) reproduces the different curves when using 2, 5, 10, 15, 30 and 60 pedestrians models. In this case, the mean of each experiment related to the number of pedestrian models is shown. As the AUC average indicates, the performance tends to saturate when 15 models are used. Note that these models have been captured through different illumination, background and pose (see Figure 3.5). In figure 3.8 (b), we can see that when the number of models is reduced the standard deviation increases: when using a number of 15 the standard deviation value is 1.89 points, while when using 2 the value is 12.36 points. While the average performance when using a reduced number of pedestrian models is worse than the ones with 30, and 60, in some cases detectors learnt with 10 models achieve almost the same performance than the ones trained with more models. This reflects that depending on the models selected to train we can achieve a higher or lower performance, meaning that some virtual pedestrians suit more than the others real ones, always in terms of HOG features.

Figure 3.9 (a) shows the performance of the approaches when training with different total number of positive examples. In this case the curves show the mean of five random experiments when training with 1440, 4320, and 12960. The results manifest that training with 1440 is not enough to achieve the desired performance, while training with 4320 or 12960 do. Accordingly, the classifier converges with already 4320. In Figure 3.9 (b) we show the different performance between detectors trained on real, virtual and their combination. The performance when using both data altogether outperform the ones using separate data. In this case, four points better than just using independent sets. Indeed, it seems that real and virtual



(a) (b)

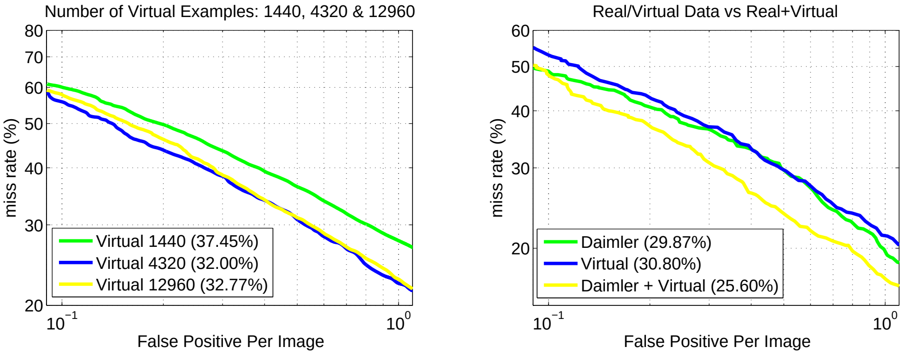
Figure 3.8: Per-image evaluation of the pedestrian detectors using different number of different models. (a) mean performances obtained in the 5 experiments with virtual samples for each fixed number of pedestrian models: 2, 5, 10, 15, 30 and 60. (b) mean and standard deviation of the subsets generated by 2 and 15 models. Both plots have been obtained over the mandatory test.

data can be complementary, outperforming in this case both detectors. This complementarity could be explained by the fact that both detectors detect almost the same pedestrians and fail in different background as already mentioned.

Next experiments evaluate the effect of the pose of training models in the performance. In such experiments we model four different distributions to assess their behaviour when testing on real images. Figure 3.10 (a) shows the different mean performances versus the original one (with no selection). As it can be seen, the performance that suits more the dataset and has closer performance to the original one is the called Gamma025, which samples are more likely to be in a close legs pose than a lateral walking one. Figure 3.10 (b) shows the comparison between the original and the Gamma025 distributions, in which it can be seen the similarity.

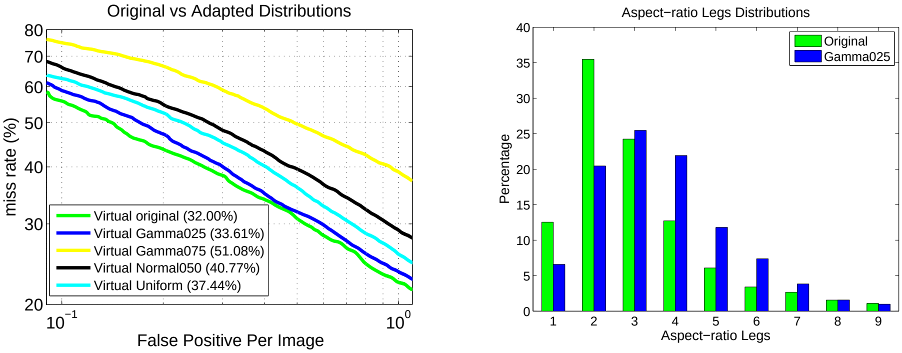
3.5 Conclusions

In this chapter we have explored the potential of virtual worlds in order to train appearance-based models for PPSs. The machine learning algorithm used to classify in our experiments is the linear SVM based on HOG features, a *de facto* standard in pedestrian detection. The whole detection pipeline consists of a sliding-window, the classification and finally, a non-maximum-suppression procedure. We first compare the virtual detector versus the real one in real images, and conclude that both performances are fairly the same. Several experiments with different virtual datasets to explore the possibilities in pedestrian detection, and concretely in appearance, that are carried to assess what virtual data can offer. In particular, we demonstrate that just few virtual pedestrian models can achieve almost the same accuracy of



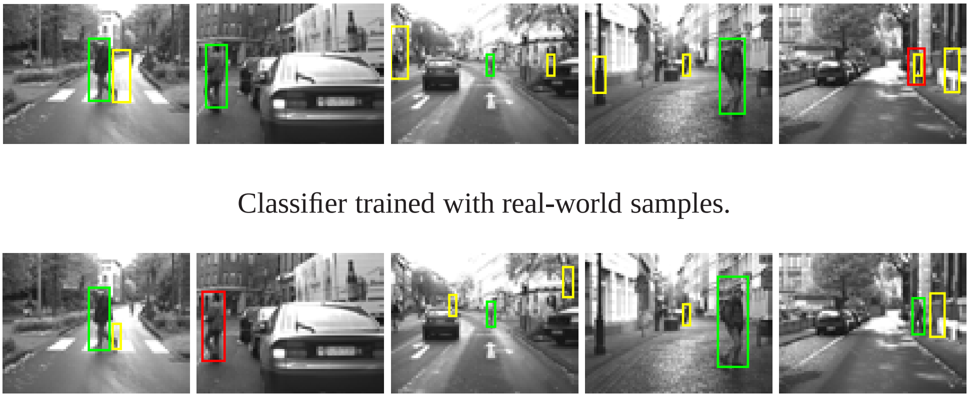
(a) (b)

Figure 3.9: Per-image evaluation of the different experiments. (a) Comparison of virtual detectors using different number of training examples: 1440, 4320, and 12960 pedestrians. (b) Comparison of different detectors trained on real and virtual data separately, and altogether.



(a) (b)

Figure 3.10: Experiments on the pose variation. (a) shows the different performances between the adapted distributions and the original one. (b) shows the comparison between pedestrian poses found in the original distribution and the gamma025 distribution. Each bar k, with $k \in \{1, \ldots, 9\}$, shows the percentage of pedestrians that belong to the aspect-ratio range $[0.1k, 0.1(k+1)[$.



Classifier trained with real-world samples.

Classifier trained with virtual-world samples.

Figure 3.11: Qualitative results at 10^0 FPPI taken when following PASCAL VOC criterion. Top row: using the pedestrian detector based on Daimler's training set. Bottom row: using the pedestrian detector corresponding to the training with virtual samples, in particular with the classifier of highest detection rate at 10^0 FPPI. Green bounding boxes are right detections, yellow ones are false positives and red ones misdetections.

the detectors trained with more models, which means that adding new models does not bring any improvement. This same conclusion can be seen in [61]. Besides, we investigate whether increasing the total number of examples used can benefit the detection or not. In this case the obtained results show that the performance converges. However, this fact can come from the machine learning and features used, so in future experiments we will test other widely used features such as Haar-like [78] features or LBP [51]. Finally, we study how the pose in pedestrians can influence in detection. The different pose distributions used proves that, in this specific case, the original distribution, in which the number of pedestrians walking across the camera is small compared to the others (*i.e.*standing/front-rear), achieves the best performance. These last results expose once again the similarity between virtual and real data when performing urban scenarios. Besides, combination results show the complementarity of real and virtual, outperforming the baseline. Therefore, the work done so far indicates that the virtual scenario generation stage is an important key to achieve state-of-the-art performance.

Once we finished this work, we assessed the performance of the virtual detector in other datasets (see Fig.3.12). The results showed that the detector was no longer reaching the same accuracy as the detector trained with the new dataset. At this point, we found we were facing a domain adaptation context problem. This new research line was out of the scope of this thesis, thus we decided to continue with our two other objectives (see Chap. 1). This new line of research has been continued by one of the PhD students in our group as the main topic of his thesis [75]. On the other hand, as future work we plan to use other targets in our occlusion handling framework such as vehicles.

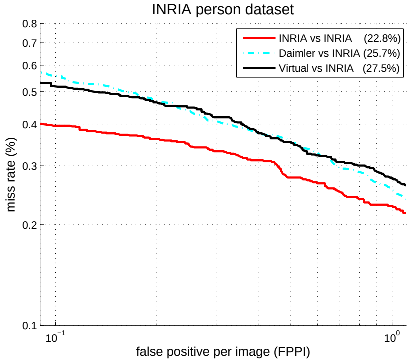


Figure 3.12: Per-image evaluation in the INRIA person dataset. The three different curves correspond to the detectors trained using INRIA, Daimler and Virtual datasets, and tested in the INRIA person dataset. As can be seen, Daimler and Virtual detectors obtain similar accuracy, but worst accuracy than INRIA detector. These results highlight the domain adaptation context problem (figure from [75]).

Chapter 4

Occlusion Handling via Random Subspace Classifiers

In this chapter we propose a general method to address partial occlusions for human detection in still images. The Random Subspace Method (RSM) is chosen for building a classifier ensemble robust against partial occlusions. The component classifiers are chosen on the basis of their individual and combined performance on a hold-out validation set containing examples of partially occluded pedestrians. For this purpose, we make use of the framework developed in the last chapter in order to generate a large variety of examples of partial occlusions that can happen in a real situation.

The main contribution of the work presented in this chapter lies in our approach's capability to improve the detection rate when partial occlusions are present without compromising the detection performance on non occluded data. In contrast to many recent approaches, we propose a method which does not require manual labelling of body parts, defining any semantic spatial components, or using additional data coming from motion or stereo. Moreover, the features used in the holistic classification are reused by the RSM, which means that there is no additional computational cost. The method can also be easily extended to other object classes. The experiments are performed on three large datasets: the INRIA person dataset, the Daimler Multicue dataset, and a new challenging dataset, called *PobleSec*, in which a considerable number of targets are partially occluded. The different approaches are evaluated at the classification and detection levels for both partially occluded and non-occluded data. The experimental results show that our detector outperforms state-of-the-art approaches in the presence of partial occlusions, while offering performance and reliability similar to those of the holistic approach on non-occluded data. The datasets used in our experiments have been made publicly available for benchmarking purposes.

4.1 Introduction

As introduced in the first chapter, most promising pedestrian detection methods can be divided into two different groups: Holistic, which consist in detecting the pedestrian as a whole target, and Part models, which rely on the combined detection of the different parts of the body. Holistic methods offer robustness with respect to illumination, background and texture changes, whereas part-based methods have an advantage for different poses [18]. In all cases, the presence of partial occlusions causes a significant degradation of performance, even for part-based methods which are supposed to be robust in that respect [18].

Expectedly, detection in the presence of partial occlusions has sparked significant interest [8, 28, 31, 44, 82, 85]. For instance, an accident in which a vehicle hits a pedestrian is likely to occur when the pedestrian is not in full view to the driver, *e.g.*, when it appears from behind a parked car. Captured in a sequence of images, several frames prior to the accident will contain a partially occluded human figure. Therefore, accurate detection in the presence of partial occlusion is of paramount importance when building driver assistance systems.

Current methods for handling occlusion lack generalisation, either because additional information is required (coming from manual annotations of the parts or from other sensors), or they are tied to a specific object class [31, 44, 82, 85]. Therefore, our aim is to introduce a general method for automatic, accurate and robust detection of human figures in the presence of partial occlusion.

Image windows framing partially occluded persons tend to be misclassified due to the fact that, given the descriptor of the whole window, the features corresponding to the occluded areas can be interpreted by the classifier as noise or background. Accordingly we argue that an appropriate solution for these situations is to apply classifiers trained on regions less likely to be occluded. More specifically, we propose to learn the different regions of the window by using random subspace classifiers [34], and subsequently find the optimal ensemble through a bespoke selection strategy.

The proposed approach brings several benefits: 1) the approach is generic, therefore applicable to any class of objects; 2) as the random subspace classifiers are trained in the original space, no further feature extraction is required; 3) the detection is done on monocular intensity images, unlike other methods for which stereo and motion information are mandatory [44]; and 4) during training, we only require a subset of images with and without partial occlusion; other detection methods require delineation of the occluded area.

Following our previous work [46], here we use a virtual-world based dataset with the occlusion labelling available by design. We also introduce a new real world dataset with occluded pedestrians for testing.

The remainder of this chapter is organised as follows. Section II introduces the related work. Section III presents the method from a generic point of view. Section IV, presents a particular implementation for human detection. Section V, relates the design followed in our experiments. In Section VI we validate and discuss our method. Finally, Section VII draws the main conclusions and future work.

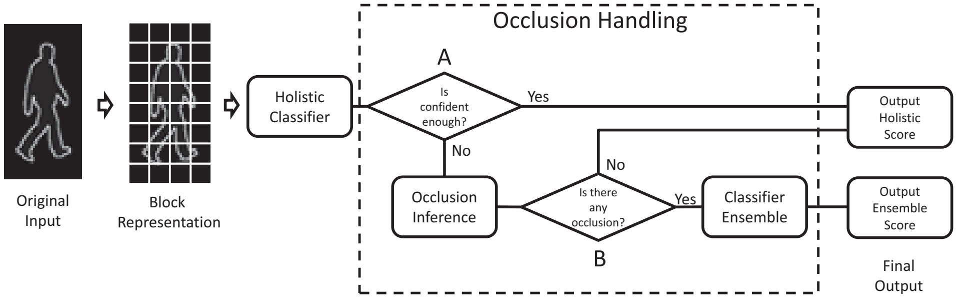


Figure 4.1: Occlusion handling scheme. From left to right, the steps for classifying a window.

4.2 Related Work

Dollar *et al.* [18] evaluated state-of-the-art detectors under occlusions, and demonstrated that both holistic and part-based methods have similar unsatisfactory performance. This is attributed to the fact that these methods are not specifically designed for handling occlusions.

Very few methods from the literature handle occlusions explicitly. In [8], Dai *et al.* propose a part-based method for face and car detection. The method consists of a set of substructure-detectors, each of which is composed of detectors related to the different parts of the object. The disadvantage of this method is that the different parts of the object need to be manually labelled in the training dataset, in particular, eight parts for face detection and seven parts for cars.

A general approach based on the response of different part detectors and a whole-object segmentation process is introduced in [85] by Wu *et al.* The method requires a hierarchical object-parts design with eleven components making up the head, the torso and the legs. The edge pixels of the object that positively contribute to the part detectors are extracted and used together with the part detector responses to obtain a joint likelihood of multiple objects. In this joint likelihood an occlusion reasoning is applied. In case of finding any inter-object occlusions, the occluded parts are ignored. The main drawback of this method is that it requires a manual spatial alignment of the objects, which has to be adapted to each object class. In addition, it requires a special camera set-up in which the camera has to look down on the ground-plane.

Wang *et al.* [82] propose a new scheme to handle occlusions. More concretely, the response at a local level of the Histograms of Oriented Gradients (HOG) [6] descriptor is used to determine whether or not such local region contains a human figure. Then, by segmenting the binary responses over the whole window, the algorithm infers the possible occlusion. If the segmentation process does not lead to a consistent positive or negative response for the entire window, an upper/lower-body classifier is applied. The drawback of this method is that it makes use of a pre-defined spatial layout that characterises a pedestrian but not any other object class.

A mixture of experts for handling partial occlusion is presented in [44] by Enzweiler *et al.* The component layout the authors use is composed by three overlapped regions: head, torso and legs. Then, during the classification process, expert weights are computed to focus on the unoccluded region through a segmentation process applied to the depth and motion images. While the authors demonstrate the robustness of their method against partial occlusions, the drawback of this approach is that it requires both stereo vision and motion information, which limits its applicability if we do not have this additional information. Furthermore, the method is based on a pre-defined spatial layout that is characteristic of the pedestrian, which limits its applicability for other classes of objects.

Gao *et al.* [28] tackle occlusions by identifying and using in the training process cells of pixels which belong to the object in the bounding box. The method outputs not just the detection but also the inferred segmentation. However, the method requires the tedious task of manual labelling all the cells that belong to the object in the training set.

In [31], Girshick *et al.* propose an extension of the deformable part-based detector [25] with occlusion handling. Specifically, the method tries to place the different body parts over the window. Then, if some of the parts are not matched, the method tries to fit in their designated place occluding objects learned from the data. The obvious inconvenience of such an approach is the need of learning the objects that occlude the target. Besides, to extend the method to other classes a different occlusion reasoning has to be defined.

Here we propose a method for detecting human figures in still images, which can handle occlusion automatically. Manual annotation or defining specific parts/regions of the window are not needed. Our method is based on an ensemble of random subspace classifiers obtained through a selection process. It is worth mentioning that, as the random subspace classifiers use the original feature space, there is no additional feature extraction cost. Similar to [82] and [44], the proposed approach uses a segmentation process to find the unoccluded part of a candidate-window. An ensemble is applied only in uncertain cases. In particular, the proposed method generalises the inference process presented in [82] by extending it to multiple descriptors.

4.3 Occlusion Handling Method

4.3.1 Proposal Outline

We present a general method for handling partial occlusions (see Fig. 4.1). In such a design, the window is described by a block-based feature vector. The resulting feature vector is evaluated by the holistic classifier. If the confidence given by the holistic classifier falls into an ambiguous range (Fig. 4.1-A), then an occlusion inference process is applied by using the block responses. Finally, if the inference process determines that there is a partial occlusion (Fig. 4.1-B), an ensemble classifies the window. Otherwise, the final output is given by the holistic classifier. Notice that, in order to obtain a more accurate decision, we apply the ensemble only when partial occlusion is suspected. In the following, we explain in detail the components shown in Fig. 4.1.

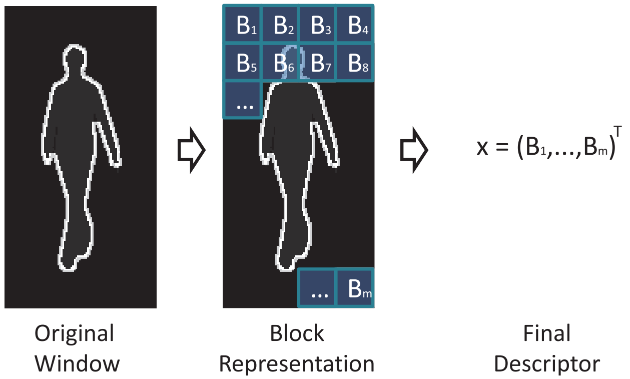


Figure 4.2: Block-based representation. From left to right, the original input, then the division into blocks (note that Blocks can overlap), and finally, the feature descriptor.

4.3.2 Block Representation

Our detection system relies on using a block-based representation, one of the most successful descriptor types in use today [18]. A well-known example of such descriptor is the HOG of Dalal *et al.* [82], although there exist many other examples [64, 72]. In section 4 we explain our specific choice for this work. Fig. 4.2 illustrates the idea of this type of representation, where the window descriptor $\mathbf{x} \in \mathbf{R}^n$ is defined as the concatenation of the features extracted from every predefined block $\mathbf{B}_i$, $i \in \{1, \dots, m\}$. A block is a fixed subregion of the window as shown in Fig. 4.2. Our method also allows the blocks to overlap. The descriptor is denoted as $\mathbf{x} = (\mathbf{B}_1, \dots, \mathbf{B}_m)^T$.

The feature vector $\mathbf{x}$ is passed to a holistic classifier H:

$$\begin{array}{rccc} H: & \mathbf{R}^n & \longrightarrow & (-\infty, +\infty) \\ & \mathbf{x} & \longmapsto & H(\mathbf{x}) \end{array} \tag{4.1}$$

where the feature space dimension, n, is $n = m \cdot q$, being q the number of features per block.

The higher the value returned by the function H the higher the confidence that there is a pedestrian in the given window. Note that the function H can be any classifier that returns a continuous-valued output, for example, a hyperplane learnt with an SVM.

4.3.3 Occlusion Inference and Posterior Reasoning

In order to detect if there is a partially occluded human figure in the image, we make use of a procedure similar to the one of Wang *et al.* [82]. First, we determine whether the score of the holistic classifier is ambiguous. For example, the response from an SVM classifier can be

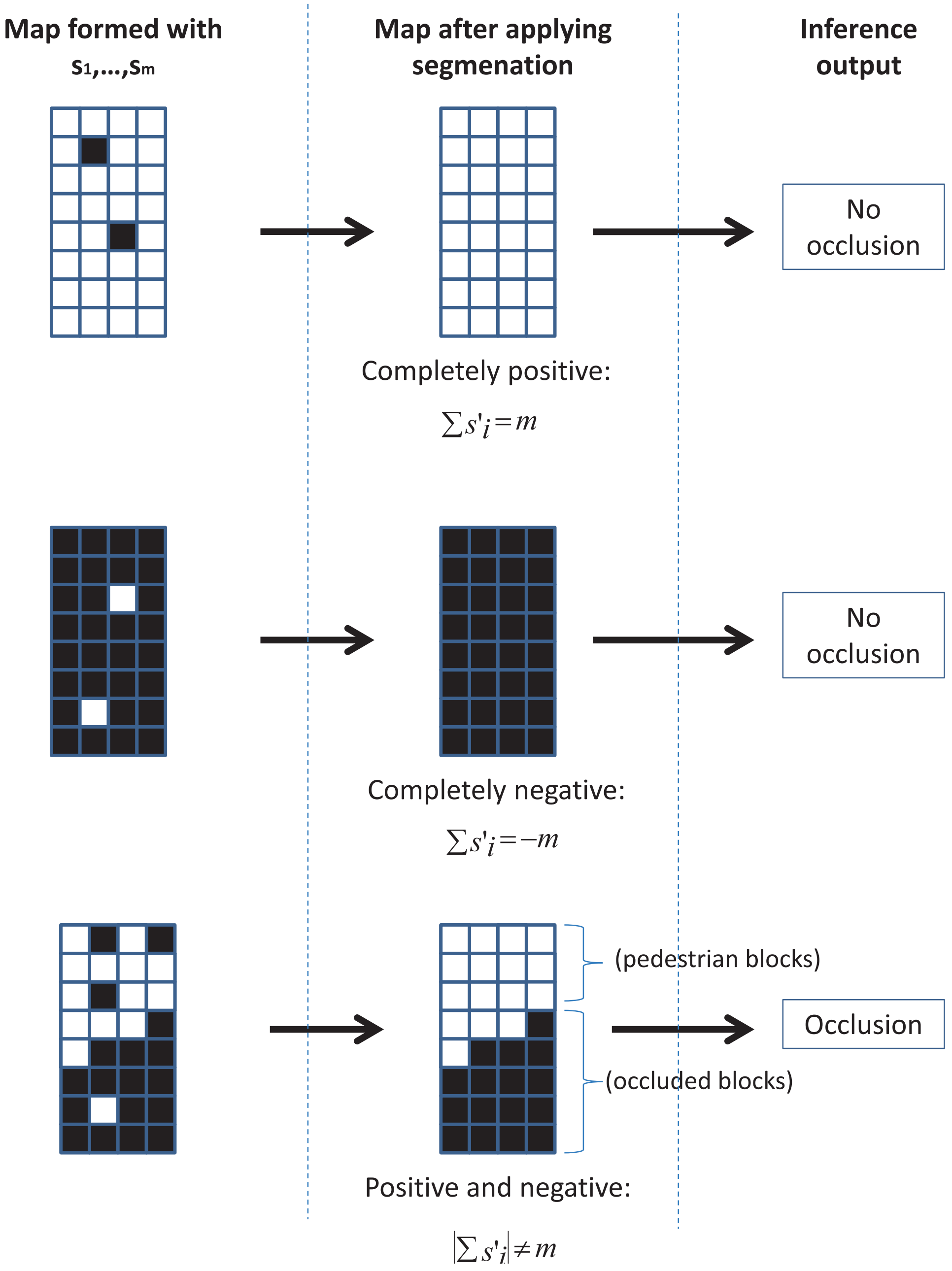


Figure 4.3: Occlusion inference and posterior reasoning. From left to right, the initial map formed by the local responses s_i; in the middle, the output after segmentation, s'_i; at the right, the three inference outputs.

perceived as ambiguous if it is close to 0. When the output is ambiguous, an occlusion inference process is applied. This is based on the responses obtained from the features computed in each block. In particular, for every block B_i, $i \in \{1, \ldots, m\}$ we define a local classifier h_i:

$$\begin{array}{rccc} h_i : & \mathbf{R}^q & \longrightarrow & (-\infty, +\infty) \\ & \mathbf{B}_i & \longmapsto & h(\mathbf{B}_i) \end{array} \tag{4.2}$$

where the classifier h_i takes as input the i-th block $\mathbf{B}_i$ of the window, and provides as output the likelihood that the block $\mathbf{B}_i$ is part of the pedestrian or, otherwise, is part of an occluding object or background.

The algorithm for the occlusion inference and the posterior reasoning is described in Alg. 1. For each block $\mathbf{B}_i$ we obtain a discrete label s_i by thresholding the local response $h_i(\mathbf{B}_i)$ (see Alg. 1). The discrete label s_i indicates whether the block $\mathbf{B}_i$ is part of the pedestrian ($s_i = 1$) or is part of an occluding object or background ($s_i = -1$). Once we have determined this for all the blocks, we can define a binary map as illustrated in Fig. 4.3, and then apply a segmentation algorithm on this binary map. The objective of applying segmentation is to remove spurious responses and to obtain spatially coherent regions. As a result of this segmentation, we obtain spatially coherent block labels s'_i (see Fig. 4.3), and we can determine if there is actually an occlusion or not.

Algorithm 1: The occlusion inference and posterior reasoning (Fig. 4.1-B) pseudo-code.

```
Input: $\mathbf{B}_1, \ldots, \mathbf{B}_m$
Output: Found partial occlusion
Procedure:
foreach $i \in 1, \ldots, m$ do
    Calculate $h_i(\mathbf{B}_i)$;
    $s_i := sign(h_i(\mathbf{B}_i))$;
end
$(s'_1, \ldots, s'_m)$ := seg($s_1, \ldots, s_m$);
if $|\sum s'_i| \neq m$ then
    return true;// There are occluded blocks
else
    return false;// Pedestrian or Background
end
```

In Algorithm 1, $(s_1, \ldots, s_m)$ represents the binary image given by the sign of the local responses $(h_1(\mathbf{B}_1), \ldots, h_m(\mathbf{B}_m))$, being $s_i \in \{-1, 1\}$, $\forall i \in \{1, \ldots, m\}$. After obtaining the local responses s_i, the algorithm returns $(s'_1, \ldots, s'_m)$ as the result of applying a segmentation process over the binary image, where again $s'_i \in \{-1, 1\}$ $\forall i$. Finally, the algorithm returns a boolean confirming whether there is a partial occlusion depending on the responses. More concretely, if all the responses s'_i are negative, we interpret that such window only contains background. If the responses are all positive, then we consider that there is a pedestrian with no occlusions. Finally, if there are both, positive and negative values, we consider that there is a partial occlusion (see Fig. 4.3).

4.3.4 Ensemble of Local Classifiers

In general, partial occlusions can vary considerably in terms of shape and size; hence a flexible model is needed. We propose an adapted Random Subspace Method (RSM) [34, 35] for this task. In particular, we propose to use classifiers trained on random locally distributed blocks; the collection of such classifiers is subsequently browsed to find an optimal combination. Our adapted RSM is introduced below (see Fig. 4.4).

Block-based Random Subspace Classifiers

Given $I = \{1, \ldots, m\}$ the set of block indices, in the k-th iteration we generate a random subset J_k of indices, where $J_k \subset I$. This selection process is carried on until we obtain T different subsets of indices $J_1, \ldots, J_T$. The k-th subset J_k contains m_k indices, where this number can vary across different iterations.

Given the k-th subset $J_k = \{j_1^k, ..., j_{m_k}^k\}$, we define a subspace formed with the blocks indexed by $J_k : \{B_{j_1^k}, ..., B_{j_{m_k}^k}\}$. For each subspace, we train an individual classifier g_k. Thus, the decision function of each base classifier of the ensemble can be expressed as a composition of functions:

$$\begin{array}{ccccc} \mathbf{R}^{m\cdot q} & \xrightarrow{P_k} & \mathbf{R}^{m_k\cdot q} & \xrightarrow{g_k} & (-\infty, +\infty) \\ \mathbf{x} = \begin{pmatrix} \mathbf{B}_1 \\ \vdots \\ \mathbf{B}_m \end{pmatrix} & \longmapsto & \begin{pmatrix} \mathbf{B}_{j_1^k} \\ \vdots \\ \mathbf{B}_{j_{m_k}^k} \end{pmatrix} & \longmapsto & (g_k \circ P_k)(\mathbf{x}) \end{array} \tag{4.3}$$

where P_k denotes the projection from the original space to the subspace defined by J_k, and g_k the corresponding classifier trained in such subspace. For simplicity of notation, from now on, we will use g_k instead of $(g_k \circ P_k)$.

The algorithm for the random subspace classifiers generation is described in Alg. 2, where D is the training set, $\mathbf{x}_j$ denotes the j-th sample and l_j its respective label. Given the J_k indices we apply a segmentation algorithm to the binary image $(r_1, \ldots, r_m)$, where $r_i = 1$ if the i-th block forms part of J_k, and $r_i = -1$ otherwise (see Fig. 4.5 left image). The segmentation is intended, again, as a means of obtaining spatial coherence in the selected blocks (see Fig. 4.5 right image). As a result of this segmentation process we obtain a new binary image from which we construct a new set J'_k. In particular, let r'_i be the binary

Algorithm 2: Our random subspace classifiers pseudo-code.

```
Input: Training dataset $D = \{(\mathbf{x}_j, l_j) | 1 \leq j \leq n\}, T$
Output: $g_1, \ldots, g_T$
Procedure:
$I := \{1, \ldots, m\}$;
$\mathcal{J} := \{\emptyset\}$;
$k := 1$;
while $k \leq T$ do
    Randomly select a subset $J_k \subset I$ with $J_k \neq \emptyset$;
    Given $J_k$ generate the according $(r_1, \ldots, r_m)$;
    $(r'_1, \ldots, r'_m)$:=seg$(r_1, \ldots, r_m)$;
    Obtain $J'_k$ from $(r'_1, \ldots, r'_m)$;
    if $|\sum r'_i| \neq m \wedge J'_k \notin \mathcal{J}$ then
        Train $g_k$ in $D_k = \{(P'_k(\mathbf{x}_j), l_j) | 1 \leq j \leq n\}$;
        $\mathcal{J} := \mathcal{J} \cup \{J'_k\}$;
        $k := k + 1$;
    end
end
```

value of the i-th block after segmentation, then we define $J'_k = \{i : r'_i = 1\}$, *i.e.*, the set of blocks that are positive in the segmented binary map (see Fig. 4.5 right image). Then, if the binary image $(r'_1, \ldots, r'_m)$ obtained after applying segmentation has all its values set to 1 (the resulting classifier would be the holistic classifier), to -1 (no subspace can be defined) or $J'_k \in \mathcal{J}$ (which means that we have already trained a classifier in the subspace defined by J'_k) we discard this set. Otherwise, we train a classifier in the set D_k defined by the projection P'_k, which is characterised by the indices in J'_k.

Note that, in the original RSM a fixed number of features are randomly selected from the original space, *i.e.*, all the subspaces have the same dimension. In our case, the dimension m_k may differ from one random subspace to the next as $m_k = |J'_k|$. This way, the classifiers are trained in areas with different sizes.

Algorithm 2 is used for generating $g_1, \ldots, g_T$ trained on random blocks. Based on that, we obtain our final ensemble through the selection strategy described below.

Classifier Selection (N-Best Strategy)

The accuracy of g_k, $k \in \{1, \ldots, T\}$ in our ensemble depends on the discriminative strength of the local region where this classifier is applied. In order to filter out the less accurate classifiers, our system uses the N-best algorithm [59]. A validation set is used (see Section 4.5.1) to select a subset of classifiers which work best when combined. For this purpose, the

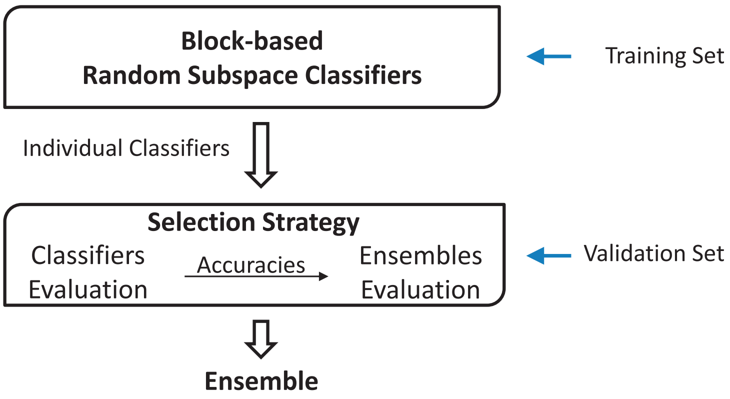


Figure 4.4: Training of the adapted random subspace method for handling partial occlusion.

algorithm first sorts the classifiers by their individual performance on the validation set and evaluates how many best classifiers form the optimal ensemble. The single best classifier is considered first. Then an ensemble is formed by the first and the second classifiers and evaluated on the validation set. The third classifier is added, and the ensemble evaluated again, and so on. We apply a weighted average for calculating the final decision, in which weights are related to the individual performances (see Eq. 4.4). The ensemble with the highest accuracy is selected among the nested ensembles. One of the most important advantages of this strategy is its linear order of complexity regarding the number of evaluations. For an ensemble of T classifiers, we need T individual evaluations plus $T-1$ combined evaluations, giving complexity $\mathcal{O}(T)$. Besides, during the evaluations it is not necessary to re-compute the features.

Final Ensemble

Given $\mathbf{x}$ and the classifiers g_k selected after the N-best strategy, the combined decision can be finally expressed as:

$$E(\mathbf{x}) = \sum_{k \in S} \omega_k g_k(\mathbf{x}) \quad , \tag{4.4}$$

where S is the set of the classifier indices that form the optimal ensemble, with $|S| \leq T$, and ω_k their corresponding weights. We derive ω_k using the validation set described in Section 4.5.1.

Combining holistic and part classifier responses is a common technique used in part-based approaches [25, 82]. In our case, if the score given by the ensemble is not confident enough (*i.e.*, the score is smaller than a fixed threshold th), we combine both scores. More precisely, we apply a linear combination between them:

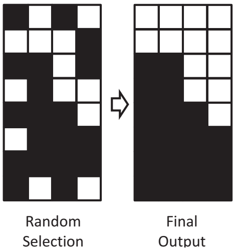


Figure 4.5: Adapted random block selection. On the left, the initial randomly selected blocks (in white), and on the right the blocks selected after applying segmentation to obtain spatially coherent regions.

$$C(\mathbf{x}) = \alpha H(\mathbf{x}) + (1-\alpha)E(\mathbf{x}) \ \ , \tag{4.5}$$

where α weights the scores of both classifiers. In Section 4.5.4 we describe how to obtain the best parameters for our method.

4.4 Human Detection with Occlusion Handling

In the previous section, we presented a general method to handle partial occlusions for object detection. In order to illustrate and validate our approach, in this section we describe in detail a particular instantiation of our method for the class of humans.

In order to apply our method to pedestrians, we make use of both linear SVMs and HOG descriptors, which have been proven to provide excellent results for this object class. In addition to HOG descriptor, we also test our system using the combination of the HOG and the Local Binary Pattern (LBP) descriptor [52], which has recently been proposed by Wang et al. [82] for human detection. In the following we explain very briefly each of these components.

Given a training dataset D, the linear SVM finds the optimal hyperplane that divides the space between positive and negative samples. Thus, given a new input $\mathbf{x} \in \mathbf{R}^n$, the decision function of the holistic classifier can be defined as:

$$H(\mathbf{x}) = \beta + \mathbf{w}^T \cdot \mathbf{x} \ \ ,$$

where $\mathbf{w}$ is the weighting vector, and β is the constant bias of the learnt hyperplane. Motivated by its success, we also propose to use the linear SVM as the learning algorithm for the base classifiers described in Sec. 4.3.4.

The HOG descriptor was proposed by Dalal *et al.* [10] for human detection. Since then, the descriptor has grown in popularity due to its success. These features are widely used now in object recognition and detection. They describe the body shape through a dense extraction of local gradients in the window. Usually, each region of the window is divided into overlapping blocks where each block is composed of cells. A histogram of oriented gradients is computed for each cell. The final descriptor is the concatenation of all the blocks' features in the window.

The LBP descriptor proposed first by Ojala *et al.* [52] has been successfully used in face recognition and human detection [1, 82, 90]. These features encode texture information. In order to compute the cell-structured LBP descriptor, the window is divided into overlapping cells. Then, each pixel contained in a cell is labelled with the binary number obtained by thresholding its value to its neighbour pixel values. Later, for each cell a histogram is built using all the binary values obtained in the previous step. Finally, the cell-structured LBP is the result of concatenating all the histograms of binary patterns in such window.

The HOG-LBP is the concatenation of both descriptors, HOG and LBP. These two descriptors complement each other, as they combine shape and texture information. Besides, this combination has been proven to outperform the original HOG descriptor [18]. Note that in our case we interpret every cell LBP as a block, thus a block HOG-LBP represents the concatenated block HOG and the cell LBP computed in the same region.

Following the formulation proposed by Wang *et al.* [82], the constant bias β can be distributed to each block $\mathbf{B}_i$ by using the training data (see Eq. 10 in [82]). This technique allows the possibility to rewrite the decision function of the whole linear SVM as a summation of classification results. Then, using this formulation we can define the local classifiers described in the previous Sect. 4.3.3 as:

$$h_i(\mathbf{B}_i) = \beta_i + \mathbf{w}_i^T \cdot \mathbf{B}_i \quad ,$$

where $\mathbf{w}_i$ and β_i are the corresponding weights and distributed bias for each block $\mathbf{B}_i$, respectively. By defining the local classifiers this way, no additional training per block is required. Moreover, when computing the holistic classifier, the local classifiers are implicitly computed, which means that there is no extra cost.

In this work, instead of just using HOG features to infer whether there is a partial occlusion [82], we extend the process to rely on both, HOG and LBP features. Thus, the response of each h_i is given by all the features computed in the same block i. As in [82], the segmentation method used in our implementation is based on the mean shift algorithm [6], whose libraries are publicly available[1]. The mean shift weights are set to $w_i = |h_i(\mathbf{B}_i)|$.

4.5 Experimental Design

In this section, we outline the set-up followed in our experiments. We describe in detail the different datasets used, as well as the procedure conducted during the training and the testing phases. As explained in Section 4.3.4, as part of our training procedure we make use of a hold out validation set. In order to obtain this validation set we propose the use of virtual pedestrians, a sample of which is shown in Fig. 4.7. The Daimler multi-cue dataset, published

[1] http://coewww.rutgers.edu/riul/research/code/EDISON/index.html

recently [44], is proposed for evaluating the different approaches at the classification level. The INRIA person dataset [10], in which almost none of the pedestrians are occluded, is used to assess the detectors under no occlusions. To evaluate the detector under partially occluded data, we compiled a new dataset, called *PobleSec*, in which a significant number of partially occluded pedestrians are annotated.

4.5.1 Validation dataset

For the validation stage, we need partially occluded data where only the bounding box of the entire object needs to be specified. Recently, the use of synthetic data in Computer Vision has grown in popularity [37, 46, 61, 76] due to their multiple advantages (no manual annotation is required, easy generation of more samples, the possibility of reproducing difficult scenarios, etc.). In this work, we generate a validation set of partially occluded pedestrians needed in the training process (see Fig. 4.4). In particular, using the same game engine as in our previous work [20], we built a scenario with 50 different human models (see Fig. 4.6), and created four different variations by introducing illumination, texture and object changes. Afterwards, we recorded 40 video sequences with a freely moving virtual camera, and extracted only positive examples in which humans were partially occluded (see Fig. 4.7). For validating the classifiers learnt in the INRIA dataset we extracted humans whose bounding boxes were at least 96 pixels tall (around 8000 positive samples in total), and for the classifiers learnt in the Daimler dataset, bounding boxes of height 72 pixels or more (over 12000 examples). Negative images (without humans) were extracted from the same scenario with its different variations. Note that real data with the corresponding label (partially/non-occluded) could also be used in the classifier selection. For the classifiers learnt in the INRIA and the Daimler datasets, we rescaled the extracted humans to the same sizes, *i.e.*, 64×128 and 48×96, respectively.

4.5.2 Datasets

INRIA person dataset

This dataset was proposed by Dalal *et al.* [10], and it is still one of the most widely used datasets in human detection. The data is already divided into training and testing subsets. The annotations are provided for the original positive images (those containing pedestrians). The images come from a personal digital image collection, and pedestrians are shown in different poses against a variety of backgrounds (indoors, urban, rural) in which people are normally standing or walking. Examples and counterexamples in the training set are normalised to 64×128 pixels, in which pedestrians are downscaled to a height of 96 pixels (a margin of 16 pixels is added around them). We use the INRIA training set for training the classifiers and the testing set to evaluate the detectors under no occlusions (see Table 4.1 for more detail).

Daimler multi-cue dataset

In 2010, Enzweiler *et al.* [44] published a new dataset, also divided into training and testing parts (see Table 4.1). We used the same partition of the data in our experiment. Two different evaluations at the classification level are done, one assessing the classifiers against partially occluded pedestrians, and the other one only using non-occluded pedestrians. For each labelled pedestrian, Enzweiler *et al.* [44] generated additional samples by geometric jittering. The provided images were captured from a vehicle-mounted calibrated stereo camera rig (grayscale) in an urban environment. The authors also supply the stereo and flow images corresponding to each sample. Only cropped examples and counterexamples are provided, which have a resolution of 48×96 pixels and a margin of 12 pixels around each side. Non-pedestrian samples contain a bias towards more difficult patterns in terms of shape, which means that hard negative examples are also provided.

PobleSec dataset

In order to evaluate the different approaches under partial occlusions at per-image level, we have created a new challenging dataset, called *PobleSec*. We captured 327 positive images with a digital camera with a resolution of 640×480. The images have been taken in urban scenarios in Barcelona and both non-occluded and partially occluded pedestrians are annotated. *PobleSec* dataset has a similar number of labelled pedestrians to the Daimler Partially Occluded dataset. The details of the datasets used in the training and testing stages are shown in Table 4.1.

	Training				Testing				
	# pedestrians	# non pedestrians	# pos. images	# neg. images	# non-occluded pedestrians	# partially occluded pedestrians	# non pedestrians	# pos. images	# neg. images
INRIA	1208	-	614	1218	566	-	-	288	453
Daimler	6514	32465	-	-	3201	620	16235	-	-
PobleSec	-	-	-	-	-	1117	-	593	-

Table 4.1: Comparison of the different pedestrian datasets. The number of humans shown are the total number of labelled ones.

4.5.3 Implementation details

Following the same procedure as Dalal *et al.* [10], we train the holistic classifier by simply feeding the linear SVM with the positive samples and 10 random negative samples per negative image. Once the classifier is trained, we run the detector over the training negative images keeping all the false positive samples (also named hard negatives). Later, we retrain

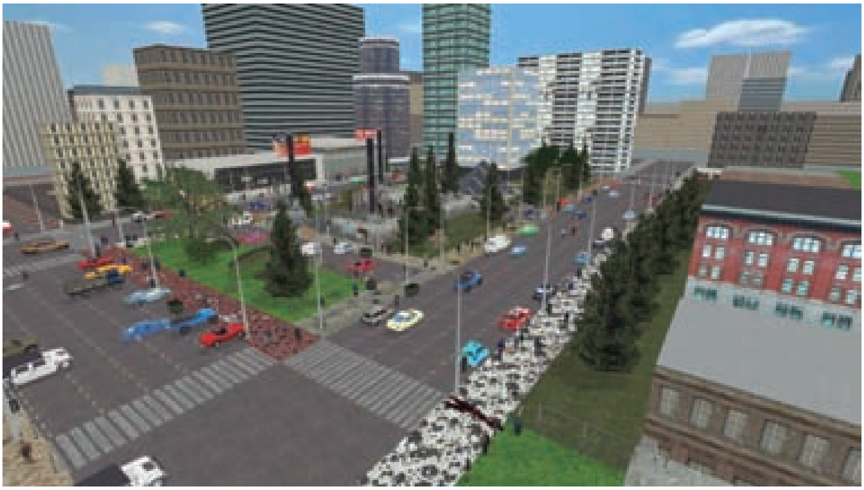


Figure 4.6: Virtual scenario.

the classifier by using the initial and new hard negatives. For the upper/lower-body classifiers used in Wang's method and for the random subspace classifiers, the initial training is done by using the samples obtained at the first bootstrapping step in the holistic training. Next, we conduct an additional bootstrapping for each one of them (using only the corresponding dimensions). The holistic classifier is also retrained. This means that all the classifiers undergo a second bootstrapping phase.

The training with both INRIA and Daimler data is performed using only intensity images. For the different classifiers trained in the Daimler dataset, no additional bootstrapping is done, as positive and negative cropped samples are already provided. In our experiments we use the original size of the windows (in contrast to [44], where the windows were scaled to 36×84 pixels with 6 pixels of margin for their specific component layout). Observe that in this work we only focus on handling occlusion based on features extracted from intensity, so there is no need to follow their specific layout. We implemented Wang's method using both HOG and HOG-LBP descriptors following the same procedure as originally proposed [82].

In our implementation, the HOG descriptor of each window consists of 7×15 blocks with a spatial shift of 6 pixels for the Daimler data, and 8 pixels for the INRIA data. This leads to overlapping blocks for both data sets. Each block is divided into 2×2 cells of a fixed number of pixels. We applied 6×6 cells for the Daimler data and 8×8 cells for the INRIA data. The histogram of oriented gradients with 12 and 9 orientation bins were computed, respectively. The HOG feature vector is normalised using a L2_HYS norm. For the LBP descriptor, we compute cell structures using the same block HOG size with the same spatial shift. This means that both descriptors are computed in the same region. The L1-sqrt norm is applied for the normalization. In order to remove the aliasing effect when scaling the images (in the training procedure and the detection evaluation), we incorporate a bilinear interpolation.

	α	th	Ambiguous range
Wang *et al.* [82]	0.7	1.5	$[-2, 1]$
Our method	0.3	2	$[-2, 1]$

Table 4.2: Best parameters for Wang's method and our method.

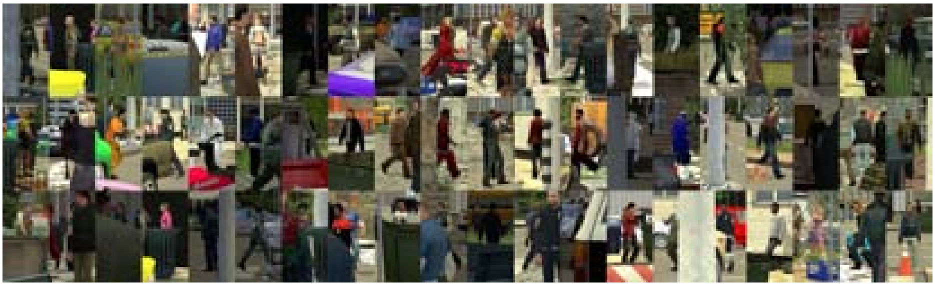


Figure 4.7: Partially occluded examples included in the validation set.

4.5.4 Training methodology

Different methodologies have been proposed in the literature to conduct the validation stage. Following [36], we use the hold-out protocol (H-method). It has low-computational cost and high reliability for large data sets, and is reproducible when training and testing data are specified. We divided the validation set into halves, one for estimating the individual performance of each base classifier, and the other for evaluating the N-best ensemble (see Sect. 4.3.4). The human images were randomly split between the two halves.

In Table 4.5.4 we show the best parameters found by using our virtual dataset for both occlusion handling methods (Wang's approach and our approach). In particular, we found the best values for: the ambiguous range defined in Section 4.3.1 (see Fig. 4.1-A); the weights w_k, the classifier score threshold th, and the weight α defined in Section 4.3.4; the minimum and maximum random subspace dimensions used in our adapted RSM (15 and 90 blocks, respectively); and the MeanShift parameters.

4.5.5 Performance Evaluation

In this chapter, in addition to the per-image evaluation, described in Chap. 2, we also evaluate the classification rate (per-window) for benchmarking purposes. On one hand, the classification system assigns a continuous-valued output to each input window related to the likelihood that the window contains a human. The detection system, on the other hand, employs a sliding window for different scales through a HOG/HOG-LBP features pyramid. Thus, for each image a group of detections is returned with their respective confidences. Later, a verification refinement is conducted to prune several detections of the same pedestrian through a confidence based non-maximum suppression process. Similarly to [18] and the previous chapter, we compute the log-average miss rate.

Following the triplet defined in Chap. 3, $(\Delta_x, \Delta_y, \Delta_s)$, in which the first two parameters denote the spatial stride, and the third parameter denotes the scale step, we use the same parameter values, $(8, 8, 1.2)$. For the evaluation criterion we use the PASCAL VOC criterion (see Chap 2).

For the experiments performed in the *PobleSec* dataset, we consider those labels mandatory in which the pedestrian are completely inside the frame, partially occluded and at least 96 pixels tall. Analogous to [18], we normalise all bounding boxes to have a width of 0.41 times the height during the per-image evaluation. For each classifier g_k, $k \in \{1, \ldots, T\}$ described in Sec. 4.3.4, its respective weight w_k is set to be proportional to the log-average classification rate between 10^{-4} and 10^{-1} FPPW. The weights w_k are normalised to sum to one.

4.6 Results

In this section we describe and discuss the experimental results. Two state-of-the-art methods are compared with our approach, the holistic method and Wang's one with partial occlusion handling. To prove its viability, our approach should be tested for partially occluded as well as non-occluded data.

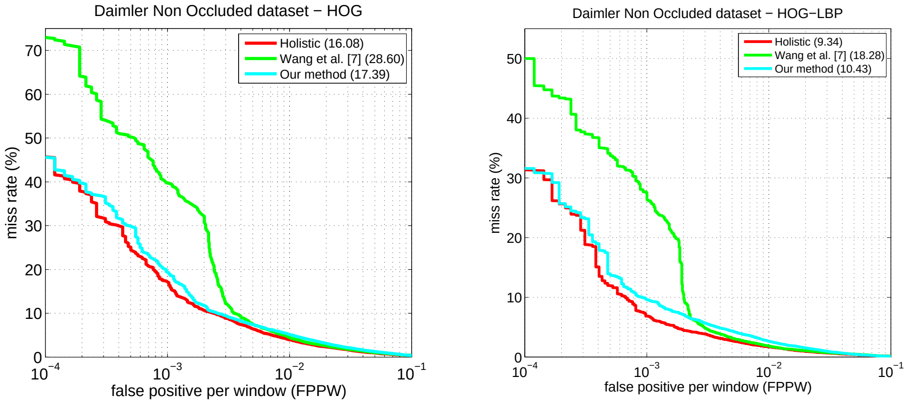
4.6.1 Per Window

Figure 4.8 shows the results on the Daimler Non Occluded dataset at per-window level. As can be seen in Fig. 4.8 (a), the performances using HOG features between our approach and the holistic approach are similar (around 1 percentage point in log-average between performances). Wang's method, instead, shows a higher miss rate at low false positive per window. In Fig. 4.8 (b) we show the performances of the extended HOG-LBP methods. Again, the performances of our approach and the holistic approach are almost equivalent, which corroborates the HOG results. However, Wang's method, like when using HOG features alone, has a higher miss rate at low false positive per window.

In Fig. 4.9, we show the curves for the three different methods using HOG and HOG-LBP features on the Daimler Partially Occluded dataset. Fig. 4.9 (a) shows that, for HOG, Wang's approach is 2 percentage points better than the holistic approach, whereas our approach was 5 percentage points better. Fig. 4.9 (b) shows that both methods with explicit handling of occlusion outperform the baseline approach in the HOG-LBP feature space.

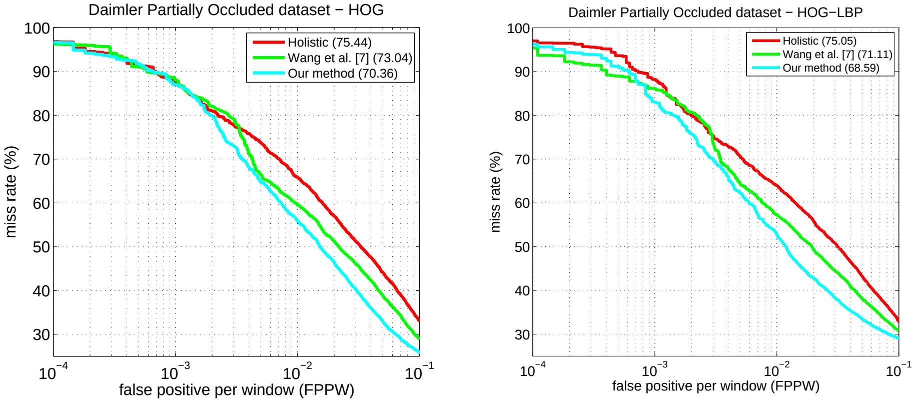
4.6.2 Per Image

In Fig. 4.10 we show the per-image evaluation using HOG and HOG-LBP on the INRIA testing dataset. Both sub-figures indicate that the occlusion handling does not degrade the performance of the classifier for either Wang's or our method compared to the holistic approach.



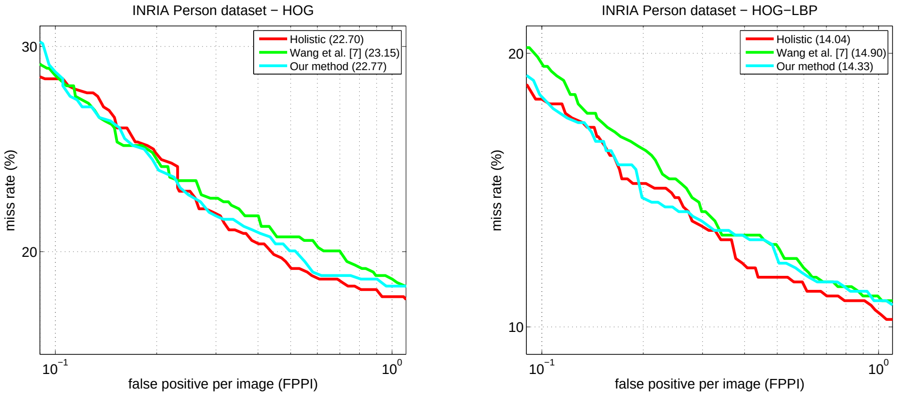
(a) (b)

Figure 4.8: Per-window evaluation on Daimler Non Occluded dataset of the three different methods. (a) Evaluation using HOG features. (b) Evaluation using HOG-LBP features. In parenthesis the log-average miss rate between 10^{-4} and 10^{-1}.



(a) (b)

Figure 4.9: Classification comparison on Daimler Partially Occluded dataset. (a) Evaluation of the different methods using HOG features. (b) Performance curves of the methods using HOG-LBP. In parenthesis the log-average miss rate between 10^{-4} and 10^{-1}.



(a) (b)

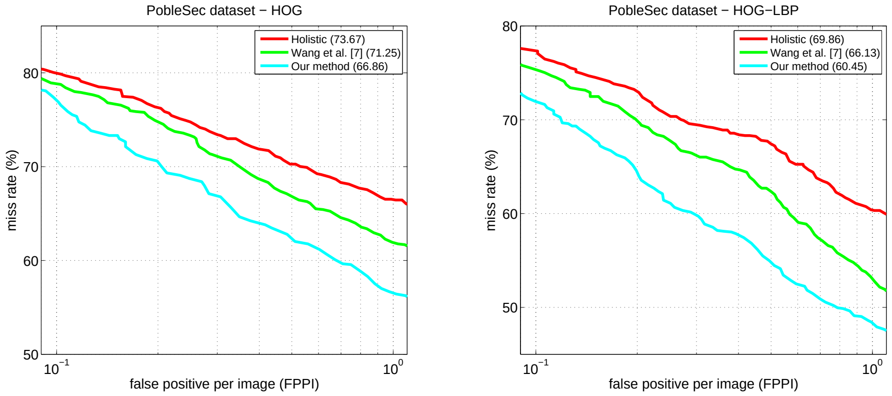
Figure 4.10: Detection curves on the INRIA testing dataset. (a) Evaluation of the different methods on the test set using HOG features. (b) Performance curves of the approaches using HOG-LBP features. In parenthesis the log-average miss rate between 10^{-1} and 10^{0}.

Figure 4.11 shows the detection curves on the *PobleSec* dataset using both HOG and HOG-LBP features. Only partially occluded humans were used in this evaluation as described earlier. The holistic method fails for both HOG and HOG-LBP features. The best performance is demonstrated by our method for both feature spaces. When using the HOG descriptor, our approach outperforms the holistic approach by 7 percentage points on average, and Wang's method by 4 percentage points. When using the HOG-LBP descriptor our approach outperforms the holistic method by 9 percentage points and Wang's method by 6 percentage points. In contrast to the other methods, our extended HOG-LBP based approach outperforms the HOG based one.

In Figures 4.13 and 4.14 we show a qualitative comparison between the different approaches at one FPPI using HOG and HOG-LBP descriptors. As can be seen, in both cases, the holistic approach is able to detect certain pedestrians which are partially occluded. However, it does not detect those with a higher level of occlusion. Both occlusion handling methods exhibit better performance by detecting cases missed by the holistic approach. Our approach manages to detect true positives where both other methods fail. This can be seen, for example, in the third and fifth columns of frames in both figures. When both methods have the same true positive detections, Wang's method tends to introduces more false detections, as seen in the second column of frames in Fig. 4.13.

4.6.3 Discussion

After having presented and analyzed the results, we discuss here the points where the proposed framework shows a performance superior to both the holistic method [10] and Wang's method [82].



(a) (b)

Figure 4.11: Per-image curves generated on the *PobleSec* dataset. (a) Evaluation of the different methods on the test set using HOG features. (b) the three different curves using HOG-LBP features. In parenthesis the log-average miss rate between 10^{-1} and 10^{0}.

As we have seen, both Wang's method and ours provide a significantly better performance than the holistic method when there are partial occlusions. This is due to the fact that the holistic method makes use of all the features in the window, including those ones that correspond to occluded parts. The latter features add noise to the classifier's decision, and significantly reduce the performance of the holistic method (see Fig. 4.11). In contrast, both Wang's method and our method focus only on the non occluded regions of the window. This fact makes these methods more robust when we have partial occlusions, as shown in Fig. 4.11.

Now let us discuss the difference in performance between our method and Wang's method in the presence of partial occlusions, and explain the technical reasons why our method performs better in this case. Wang's method divides the window into two disjoint regions (upper/lower), therefore, destroying the relationship between features from the two parts. However, this relationship might be important for handling different types of partial occlusions. In contrast, our classifier model consists in an ensemble obtained through a selection process under which a large number of classifiers responsible for differently shaped parts of the window is used (see Fig. 4.12). Therefore, in our method the relationship between features from different parts of the window is maintained, in contrast with Wang's method. The model obtained with our method is more complete leading to a higher accuracy.

Based on the score of the classifier for each individual block, Wang's method selects the part of the window (upper or lower) that contains a lower number of occluded blocks. The drawback of this method is that, many times, the individual blocks are not very informative, and therefore the score obtained for these blocks is noisy. This leads to a poor part selection if we use Wang's method. In contrast, in our method the selection is based on performance statistics over a validation data set which contains only partially occluded samples. This

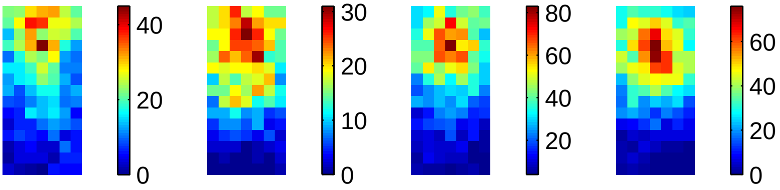


Figure 4.12: Heat-maps of which features (blocks) are used in each of our final ensembles. For each block in the window, the figure shows a score (color) equal to the number of classifiers that use the block. From left to right, the heat-maps corresponding to the 48×96 classifiers using HOG and HOGLBP, and the 64×128 ones using HOG and HOGLBP, respectively.

drives our method to finding and using, collectively, regions in the window that are frequently non-occluded.

Finally, let us discuss the performance of the three methods (our method, Wang's method and the holistic one) in the situation where there are no occlusions. In this case, the three methods perform similarly (see Fig. 4.10). The conceptual reason why this happens is that both Wang's method and our method only handle the cases inferred as partial occluded targets. The rest of the windows are evaluated by the holistic method. This common design brings comparable performance to the holistic method for non-occluded targets and a significant improvement against partial occluded ones.

In Figure 4.12 we show four different heat-maps. Each one of them indicates which features (blocks) are actually used in each of our final ensembles (read figure's caption for more details). On one hand, the uneven shading in all the heat-maps shows that features from all parts of the window are present in the ensemble, be it only in a small number of classifiers. This fact demonstrates one of the advantages of our method described above, which consists of preserving and drawing upon relationships between features in the whole window. On the other hand, the large blue area in the bottom half of the window shows that the lower part is rarely useful (also supported by the study performed in [18]). These circumstances together with the results shown in this section highlight the benefit of relying on a supervised statistical learning of the type of occlusions that a given class typically undergoes, *i.e.*, in opposition to making a specific hard assumption about such occlusions (*e.g.*, upper/lower selection).

4.7 Conclusions and Future Work

In this chapter, we have presented a general approach for human detection in still images with the presence of partial occlusion. The method is based on a modified random subspace classifier ensemble. The method can be easily extended to other objects, and allows to incorporate other block-based descriptors. Two of the most widely used descriptors in the literature of pedestrian detection have been implemented, HOG and HOG-LBP. The linear SVM was used as the base classifier. We evaluated our approach on two large datasets, INRIA and Daimler. The INRIA data is considered a standard benchmark for human detection. We designed and

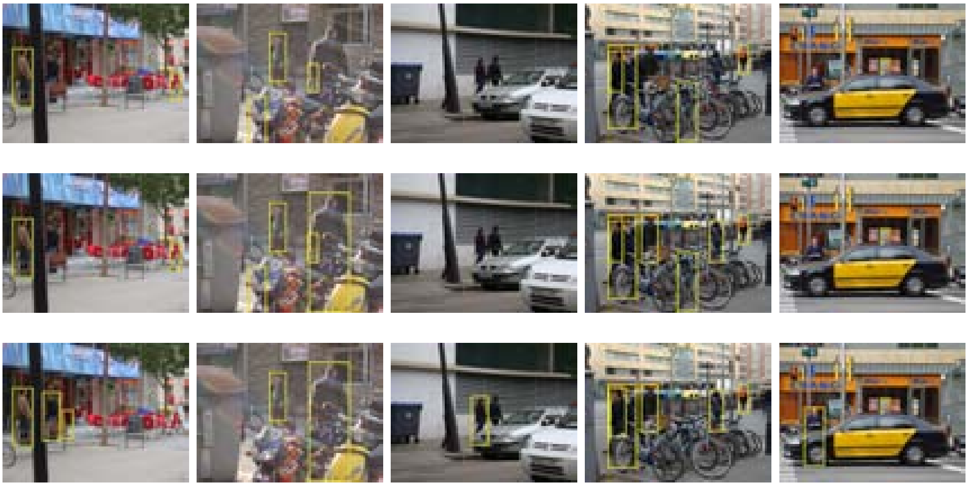


Figure 4.13: Per-image results at one FPPI using HOG features. Top row, the detections using the holistic detector without occlusion handling. Middle row, the detections using Wang's detector. Bottom row, the detections using our method.

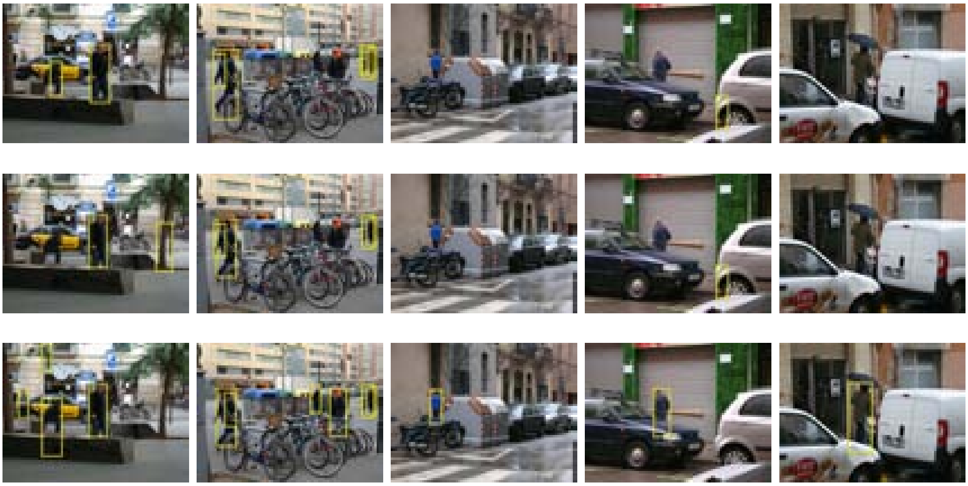


Figure 4.14: Per-image results at one FPPI using HOGLBP features. Top row, the detections using the holistic detector without occlusion handling. Middle row, the detections using Wang's method. Bottom row, the detections using our method.

released for public use a new challenging dataset called *PobleSec*. The virtual-reality dataset for per-image detection is also released for public use. Both per-window and per-image evaluations have shown that the proposed approach works on a par with the holistic approach when no occlusions are present and outperforms both holistic and Wang's approaches for detection of partially occluded pedestrian images.

As future work, we plan on adding new descriptors, using new kernels (through embedding techniques), and applying our method to other objects.

Chapter 5

Random Forest of Local Experts for Pedestrian Detection

In the previous chapter we presented a novel ensemble of local classifiers based on the Random Subspace Method. In this chapter, we push this idea further and present a new method that effectively combines multiple ensembles of rich local experts. We achieve this by adapting the classical Random Forest framework in order to work with discriminant local models. While the objective of the last chapter was to present a method robust against partial occlusions, in this chapter we focus on learning a rich model that is able to cope well with the large intra-class variability typical of pedestrians, especially due to the multiple articulated poses that they can adopt. For this purpose, we present a new Random Forest which combines multiple discriminant local experts. This combination provides flexibility in the learned spatial arrangements and a certain robustness against partial occlusions, even though the method is not specifically designed for occlusion handling, in contrast with our previous method.

The proposed method works with rich block-based representations such as HOG and LBP, in such a way that the same features are reused by the multiple local experts, so that no extra computational cost is needed with respect to a holistic method. Furthermore, we demonstrate how to integrate the proposed approach with a cascaded architecture in order to achieve not only high accuracy but also an acceptable efficiency. In particular, the resulting detector operates at five frames per second using a laptop machine. We tested the proposed method with well-known challenging datasets such as Caltech, ETH, Daimler, and INRIA. The method proposed in this work consistently ranks among the top performers in all the datasets, being either the best method or having a small difference with the best one.

5.1 Introduction

Pedestrian detection is an extremely challenging task due to the large intra-class variability caused by different articulated poses and clothing, cluttered backgrounds, abundant partial occlusions and frequent changes in illumination. The seminal work of Dalal and Triggs [10] showed the importance of using rich block-based descriptors such as the Histograms of Oriented Gradients (HOG) representation, which provides both robustness and distinctiveness. Building upon this work, other authors have proposed additional features that enrich the visual representation, including the use of color through self-similarity features (CSS) [81], texture through block-based Local Binary Patterns (LBP) [82], and the design of efficient gradient-based features via integral channels [13–15].

All of these approaches are holistic, in the sense that the whole pedestrian is described by a single feature vector and is classified at once. Recently, some authors have proposed successful methods for combining local detectors [8, 24, 85] and integrating the evidence from multiple local patches [26, 39, 68]. This type of approaches provides more flexibility in the spatial configuration of the different parts of the object, which leads to higher adaptability to the different poses of the pedestrian. Furthermore, it provides higher robustness against partial occlusions and atypical part appearances [26]. The most promising local part-based approach, proposed by Felzenszwalb et al. [24] has shown state-of-the-art results in several challenging datasets, being consistently ranked among the top performers.

Regarding the classification method, most approaches have made use of linear SVM classifiers [11, 24, 81, 82], which combine both the strength of the SVM machinery and the efficiency of a linear computation. AdaBoost is also a popular classifier for pedestrian detection, typically used in the presence of large numbers of features [15, 16, 83], or for speeding up the detection through cascaded layers of Boosting [14, 79, 93]. In particular, the use of cascades has made it possible to obtain close to real-time performance in the detection stage, especially when combined with integral features [14].

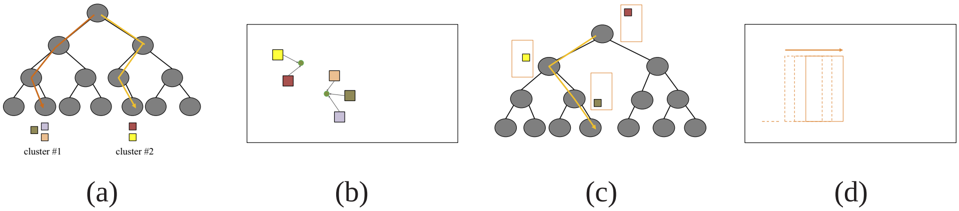


Figure 5.1: (a) Classification of individual patches in the Hough Forest [26, 39, 68], (b) detection by Hough voting, (c) classification of image windows in the proposed method used in a (d) sliding window framework.

Recently, Random Forest ensembles [26,39,68] have been proposed as an alternative type of ensemble classifier for pedestrian detection. However, traditionally based on simple pixel comparisons, their detection accuracy has remained moderate. In this chapter, we propose a novel pedestrian detection approach that combines the flexibility of a part-based model with the fast execution time of a Random Forest classifier. In this proposed combination, the role of the part evaluations is taken over by local expert evaluations at the nodes of the decision tree. As an image window proceeds down the tree, a variable configuration of local experts is evaluated on its content, depending on the outcome of previous evaluations. Thus, our proposed approach can flexibly adapt to different pedestrian viewpoints and body poses. At the same time, the decision tree structure ensures that only a small number of local experts are evaluated on each detection window, resulting in fast execution. The proposed detection system was evaluated with a variety of well-known pedestrian datasets such as Caltech [18], Daimler [19], ETH [21] and INRIA [10], where it consistently ranks among the top performers. This is on a par with the most successful part-based detection system [24], while our method presents far less design complexity and higher computational efficiency.

The rest of this chapter is organized as follows. Section 5.2 describes the state-of-the-art approaches related to ours, section 5.3 describes key concepts of the standard Random Forest classifier, section 5.4 introduces the proposed method, section 5.5 provides results and section 5.6 summarizes the work and discusses its contributions.

5.2 Related work

Closely related to our work, there are recent patch-based methods that make use of a specific type of Random Forests (RF) called Hough Forests (HF) [26, 39, 68]. Before explaining the HF, let us summarize briefly the RF framework. In RF, each node corresponds to a simple binary test that is applied to the input data. Depending on the outcome of this test, processing continues with the left or right child node until a leaf node is reached. Each leaf node stores a probability distribution over class labels, which is taken as the corresponding tree's classification confidence when the leaf node is reached.

The HF approach takes up this idea, but applies it on a patch level. Here, the leaf nodes take up the role of visual words, and each of them stores a vote distribution for the relative position of the object center. The votes from activated leaf nodes are combined in a Hough Voting space, and object locations are determined as local maxima of the vote distribution.

The HF classifier is significantly different from the RF classifier proposed in the present work. While HF is used for classifying the individual local patches, the proposed RF is used for classifying the entire object at once. For this purpose, at training time each node of the tree receives a subset of window samples containing the entire object and decides which local patch is most discriminant based on the given data (see Figs. 5.1(c)-(d)). As a result, each tree of the forest provides an ensemble of local experts, where each expert is specialized in a different local patch of the object.

An important difference with respect to [26, 39, 68] is that in these approaches the collection of local patches is sampled beforehand from the window and introduced into the tree. Therefore, each node of the tree is forced to learn each patch of the collection, regardless of whether or not this patch is discriminant for classifying the whole object. In contrast, in the proposed method each node of the tree automatically selects the local patch that is found to be the most discriminant one, based on the subset of samples received. Furthermore, by using the RF machinery the local patch selected by each node complements, in a discriminative sense, the local patches selected by its ancestors in the tree, obtaining a strong ensemble of local experts. At the end of the process each tree of the forest has selected a different collection of discriminant local patches, increasing the robustness and generalization capability of the final classifier.

5.3 Standard weak learner model

Before discussing the classifier proposed in our work, let us first introduce the basic concepts and notation of the Random Forest ensemble [7]. For lack of space we restrict the explanation to only the standard weak learner model, and we refer to [7] for an in-depth description of the RF classifier.

Given a tree of the forest, we follow the notation in [7] and denote as S_j the set of samples received by the j-th internal or split node of this tree. We denote as $h(\vec{v};\theta_j) \in \{0,1\}$ the split function associated with this node, where $\vec{v}$ is a feature vector and θ_j is the set of parameters defining the split function. The split function acts as a weak classifier that is part of the ensemble defined by the whole tree. At training time, the j-th node receives a subset of samples S_j, and based on this data the classifier $h(\vec{v};\theta_j)$ is trained. This is done by finding the optimal parameters θ_j for this classifier. At test time, the j-th node receives the feature vector $\vec{v}$, and this vector is passed to either the left or the right child depending on the output of $h(\vec{v};\theta_j) \in \{0,1\}$.

Criminisi et al. [7] define a general framework for defining the split function $h(\vec{v};\theta_j)$. In this framework, the set of parameters θ_j is defined as $\theta_j = (\phi,\psi,\tau)$, where the parameter ϕ is defined as a feature selection function that allows to disregard the noisy features in $\vec{v}$, the parameter ψ defines a geometric transformation that maps the data to a space where it is separable, and the parameter τ is a threshold that permits to classify the points.

In order to clarify the ideas, let us consider a common instantiation of this general framework. The feature selector ϕ is defined as the function $\phi(\vec{v}) = \vec{u}$ where $\vec{u} \in \mathbb{R}^s$ contains a subset of components of $\vec{v} \in \mathbb{R}^d$, $s < d$. The geometric transformation is parameterized by a vector $\vec{\psi} \in \mathbb{R}^s$ defining a linear projection $\phi(\vec{v}) \cdot \vec{\psi}$ over the selected features [1]. Finally, the split function $h(\vec{v};\theta)$ is defined as $[\phi(\vec{v}) \cdot \vec{\psi} < \tau]$, where [] is the indicator function. As a result, the classification is performed by first selecting some of the components of $\vec{v}$, then projecting the resulting vector, and then applying the threshold τ.

Let $\mathcal{T}$ be the search space where the parameters θ_j live. The optimal parameters θ_j are estimated as follows:

1. Randomly sample a small subset $\mathcal{T}_j \subset \mathcal{T}$.
2. For each $\theta \in \mathcal{T}_j$ do:

   (a) Split the set S_j into two subsets:

$$\mathcal{S}_j^L = \{\vec{v} \in \mathcal{S}_j : h(\vec{v};\theta) = 0\}$$
$$\mathcal{S}_j^R = \{\vec{v} \in \mathcal{S}_j : h(\vec{v};\theta) = 1\}$$

   (b) Evaluate the goodness of the previous partition using some measure of purity

[1] Using some abuse of notation, we write $\vec{\psi}$ as a vector in order to express the linear projection $\phi(\vec{v}) \cdot \vec{\psi}$. However, in the general framework of Criminisi et al. [7] the parameter ψ defines a generic geometric transformation, and thus should be expressed as a function in the general case.

such as the information gain:

$$I(\theta) = H(\mathcal{S}_j) - \sum_{child \in \{L,R\}} \frac{|\mathcal{S}_j^{child}|}{|\mathcal{S}_j|} H(\mathcal{S}_j^{child}) \tag{5.1}$$

where $H(\mathcal{S})$ is the entropy:

$$H(\mathcal{S}) = -\sum_{c \in \mathcal{C}} p(c) \log(p(c)).$$

3. Define the parameters for node j as:

$$\theta_j = \arg \max_{\theta \in \mathcal{T}_j} I(\theta)$$

5.4 Proposed method

In this work we define a novel ensemble of local experts based on an averaged combination of random decision trees. We first describe the main differences with respect to the standard RF framework by using generic pattern recognition concepts. Afterwards we will introduce the concepts specific to pedestrian detection, and introduce our ensemble of local experts.

5.4.1 Weak learner model

The main difference with respect to the standard framework is that in each node the optimization of the parameters θ is not only based on a maximization of a purity measure (Eq. 5.1), but also on a maximum-margin optimizer which minimizes the classification error over the samples of the node $\mathcal{S}_j$. In particular, this is done by optimizing the linear transformation $\vec{\psi}$ based on the linear SVM learning algorithm. Later on we will see that the joint use of this learner together with and an appropriate feature selector $\phi(\vec{v})$ provides the desired ensemble of local experts.

Keeping the discussion still under generic pattern recognition terms, the optimization process for each node j is composed of the following steps:

1. Randomly generate a subset $\{\phi_1, \ldots, \phi_K\}$ of K feature selectors $\phi_k(\vec{v})$. The generation of these K feature selectors is explained in section 5.4.2.

2. For $k = 1, \ldots, K$ do:

   (a) Let $\mathcal{S}_j^{\phi_k}$ be the transformed set of samples: $\mathcal{S}_j^{\phi_k} = \{\phi_k(\vec{v}) : \vec{v} \in \mathcal{S}_j\}$.

   (b) Obtain a discriminant linear transformation $\vec{\psi}_k$ by learning a linear SVM classifier over the transformed samples $\mathcal{S}_j^{\phi_k}$.

(c) Find the threshold τ_k that maximizes the purity (Eq. 5.1) of the following partition:

$$\begin{aligned} \mathcal{S}_j^L = & \quad \{\vec{v} \in \mathcal{S}_j : \vec{\psi}_k^T \cdot \phi_k(\vec{v}) \leq \tau_k\} \\ \mathcal{S}_j^R = & \quad \{\vec{v} \in \mathcal{S}_j : \vec{\psi}_k^T \cdot \phi_k(\vec{v}) > \tau_k\} \end{aligned}$$

Note that the projected values $\vec{\psi}_k^T \cdot \phi_k(\vec{v})$ are classification scores provided by the previously learned linear SVM classifier [2].

(d) Let $P_k = I(\phi_k, \vec{\psi}_k, \tau_k)$ be the maximum purity value obtained in the previous step.

3. Let $k^* = \arg\max_{k=1,\ldots,K} P_k$. Define the split function for node j as:

$$h(\vec{v}; \theta_j) = [\vec{\psi}_{k^*} \cdot \phi_{k^*}(\vec{v}) \leq \tau_{k^*}] \ . \tag{5.2}$$

The most important difference between the proposed weak learner model and the standard one (Section 5.3) lies in the optimization of the linear transformation in step 2.b. In our case, this is carried out by a discriminant optimizer such as the linear SVM learner. This learner obtains a hyperplane that optimally separates the set of training samples at each node. In contrast, in the standard weak learner model (Section 5.3) the optimization consists of randomly generating a few transformations and then evaluating each transformation together with the rest of parameters (the feature selector ϕ and threshold τ) in order to obtain the combination that maximizes the purity of the resulting partition. While the latter approach also provides a discriminant classification of the samples, there is no guarantee that the resulting hyperplane provides an optimal maximum-margin discrimination. In addition to this conceptual difference, the use of a discriminant classifier such as the linear SVM, together with an appropriate definition of the feature selector ϕ (see section 5.4.2) allows us to train our ensemble of local experts inside the RF framework.

5.4.2 Feature selector

We define our ensemble of local experts through the definition of an appropriate feature selector $\phi(\vec{v})$. Fig. 5.2 shows an illustration of the idea: given an image window, a block based descriptor $\vec{v}$ such as HOG is extracted by partitioning the window into $N \times M$ blocks [3]. Given this block-based descriptor $\vec{v}$, each feature selector ϕ_k defines a rectangular region formed by contiguous blocks (see Fig. 5.2).

In particular, the k-th feature selector ϕ_k is generated by randomly selecting the coordinates (i, j) of the top-left block, and randomly generating the width W and height H of the rectangular area, where $1 \leq W \leq L$ and $1 \leq H \leq L$, with L the predefined maximum size.

[2] Technically, the vector $\vec{\psi}$ is obtained as $\vec{\psi} = (\vec{w}^T, b)^T$, where $\vec{w}$ and b are the weights and offset provided by the linear SVM classifier. The feature selector $\phi(\vec{v})$ is then obtained as $\phi(\vec{v}) = (\vec{z}^T, 1)^T$ where $\vec{z}$ contains a subset of the components in $\vec{v}$. This way we have: $\vec{\psi}^T \cdot \phi(\vec{v}) = \vec{w}^T \cdot \vec{z} + b$.

[3] Note that neighbour blocks usually overlap each other, although this is not illustrated in the Fig. 5.2.

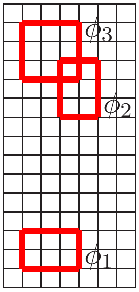


Figure 5.2: Illustration of our feature selector. In here, we show a conceptual grid of blocks.

Given the previous definition of the feature selector ϕ_k, the k-th local expert is defined as $E_k(\vec{v}) = \vec{\psi}_k^T \cdot \phi_k(\vec{v})$. As explained in section 5.4.1, the transformation $\vec{\psi}_k$ is learned by using a discriminant learner such as linear SVM, using the transformed samples $\mathcal{S}_j^{\phi_k}$ as training set. This is equivalent to extracting a local block-based feature vector from the same rectangular area across the different image windows introduced into the node, and feeding them to a learner that obtains a model of this part of the window. In our case, however, an explicit extraction of local descriptors is not necessary, making the approach computationally efficient.

Note that there is a very large number of possible feature selectors ϕ_k that can be defined over a typical block-based descriptor such as HOG or HOG-LBP, and not all of them provide the same discriminatory power. Using the weak learner defined in Section 5.4.1, the j-th node randomly generates a fixed number K of feature selectors ϕ_k, learns the corresponding local experts E_k and selects the most discriminant one according to the given data. The selected local expert E_k also complements, in a classification sense, the ones selected by the other nodes in the same branch of the tree. This is due to the fact that the data samples received by the node j depend on the classification provided by its ancestors. As a result, each tree of the forest provides an ensemble of local experts which are both discriminant and complementary.

5.4.3 Definition of other RF components

The rest of the RF components are defined in a standard way [7]. This comprises the type of randomness, and the aggregation rule used for obtaining the final output of the forest. Regarding the type of randomness, we do not use bagging in this work, i.e., each tree of the forest receives the whole training set. This choice is recommended in the analysis of Criminisi et al. [7] and gave us slightly better results in preliminary tests.

Regarding the output of the forest, let $p_t(c|\vec{v})$ be the probability that the window $\vec{v}$ belongs to class c, computed by the t-th tree of the forest. This probability is obtained during the training stage. Every leaf stores the class distribution of the training samples that reach it, and then each leaf probability is set according to this distribution. Given this, we use the average as aggregation rule in order to compute the probability for the whole forest: $p_{\mathcal{F}}(c|\vec{v}) = \frac{1}{T}\sum_{t=1}^{T} p_t(c|\vec{v})$, where T represents the number of trees in the RF $\mathcal{F}$.

5.4.4 Bootstrapping procedure

We use bootstrapping at training time in order to select a subset of negative windows from the large pool of possible negatives. For this purpose, we propose to use an efficient procedure that consists of the following steps:

1. Set the initial training set as $\mathcal{S} = \mathcal{P} \cup \mathcal{N}$, where $\mathcal{P}$ is the set of cropped pedestrians, and $\mathcal{N}$ is an initial set of negative windows that are randomly sampled.
2. Set the initial forest as $\mathcal{F} = \emptyset$
3. For $i = 1, \dots, N_{boot}$ do:
    (a) Train M new trees using the training set $\mathcal{S}$. Add the trees to the current forest $\mathcal{F}$.
    (b) Use the current forest $\mathcal{F}$ for detecting false positives in the training images. Consider these false positives as negative samples and add them to the training set $\mathcal{S}$.
    (c) Use the new training set $\mathcal{S}$ for updating the leaf probabilities $p(c|\vec{v})$ for all the trees in the current forest $\mathcal{F}$.

Our strategy, when compared with [68], allows to reduce the number of hard negatives obtained at each iteration. This is mainly due to the fact that at each iteration more trees are responsible for classifying, and that all the probabilities $p(c|\vec{v})$ stored at the leaf nodes are updated using the entire training set (which slightly increments their discriminative ability). Moreover, it is worth mentioning that the training time is reduced thanks to the smaller number of negative samples introduced at each iteration.

5.4.5 Soft Cascade

In order to speed up the detection of objects, we propose to use a Soft Cascade (SC) architecture [4]. Let T be the total number of trees in the forest, M be the number of trees used in an initial layer, η be a predefined rejection threshold (see Section 5.5.1), and $\vec{v}$ be the block-based representation of the current window. We propose the following SC algorithm:

```
1. $score \leftarrow \frac{1}{M} \sum_{t=1}^{M} p_t(c = 1|\vec{v})$
2. $t \leftarrow M$
3. While $score > \eta$ and $t < T$ do:
    (a) $score \leftarrow \frac{1}{t+1} (score \cdot t + p_{t+1}(c = 1|\vec{v}))$
    (b) $t \leftarrow t + 1$
4. If $score < \eta$ reject window $\vec{v}$, otherwise output $score$.
```

The cascade works by first gathering enough evidence for the window $\vec{v}$, through the use of M trees in the initial layer (step 1). After this initialization, a new tree is added at each layer of the cascade (step 3.a) and the score is updated. The process continues until all the trees have been added or the score is lower than η. In this case the window is rejected and the evaluation stops.

As we will see in the results section, the SC provides a significantly faster detection. This is due to the fact that a large majority of windows are rejected at early stages of the cascade, and thus there is no need to compute the probability for all the trees of the forest on these windows.

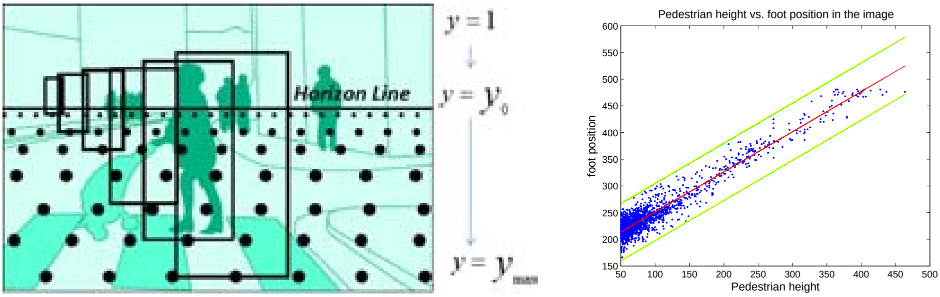
5.4.6 Candidates generation pruning

The components introduced so far can be employed for detecting generic object classes (not just pedestrians). We introduce now an additional component that is specific for pedestrian detection. In particular, we make use of a-priori geometry information, based on three different assumptions: i) the frames are compensated for car pitch motion, ii) the pedestrians are standing on the floor, and iii) the floor is flat.

Using projective geometry, we have $h \approx \frac{Hf}{d}$, where h is the height of the pedestrian in the image, H is the height of the pedestrian in the world, d is the distance of the pedestrian to the camera and f is the focal length. Furthermore, if (x, y) is the position of the pedestrian in the image, the vertical coordinate y is inversely proportional to the distance d: $y \propto \frac{1}{d} + y_0$ (see Fig. 5.3(a)). This way, the pedestrian appears at the bottom of the image (i.e., $y = y_{max}$) when it approaches the camera, and it appears close to the horizon line (i.e., $y = y_0$) when the distance d in the real world tends to infinite (see Fig. 5.3(a)).

Combining the both equations, we have: $h \propto Hf(y - y_0)$, which indicates a linear relationship between the height of the pedestrian h and its vertical position y in the image. In order to verify this relationship, the actual values of h and y were measured for all the pedestrians from the Caltech training set. Fig. 5.3(b) shows these values plotted as blue points, where the horizontal axis represents h and the vertical axis represents y. The red line corresponds to the linear regression for $\hat{y}$, the green lines show the standard deviation, and the yellow lines delimit the possible range of values where all the samples are contained. The data clearly shows a strong correlation between the pedestrian height h and its vertical position y, where the variability is due to the different height H of the pedestrians, which follows a normal distribution, and the fact that the ground is not completely flat in the real world.

Given the previous observations, in this work we propose a simple method for discarding unrealistic window candidates before introducing them to the classifier. At training time, we compute the linear regression and standard deviation (as in Fig. 5.3(b)) for the pedestrians in the training set. Assuming that the test images have been captured under similar conditions, at test time we discard all the window candidates whose height and vertical position (h, y) fall outside the range defined by the standard deviation interval. In practice, this is done as follows: for each level of the scale-space pyramid we only need to extract the visual representation $\vec{v}$ and the classification score for those windows whose vertical position y fall in a certain range.



(a) (b)

Figure 5.3: Projective geometry (a), linear regression of pedestrian height h versus vertical position y in the image (b)

The proposed candidate pruning provides both a speed up in the detection step and a reduction of false positives appearing in regions of the image where it is not physically plausible to have a pedestrian. An evaluation of the impact of both factors is provided in section 5.5.

5.5 Results

The INRIA dataset [10] is currently being used in the literature as a training-validating dataset. Then, once the best parameters are found during the validation, authors usually report additional results on other challenging datasets. We followed a similar procedure.

Due to the large number of parameters, most of them were estimated by testing just a few reasonable values. The selected values are described in Section 5.5.1. There were two parameters, however, that were exhaustively optimized using a validation set (*i.e.* the INRIA testing dataset). In Section 5.5.2, we describe the corresponding experiments. In Section 5.5.3, we provide a comparison against the best state-of-the-art methods on these other datasets. Finally, Section 5.5.4 provides an evaluation of the computational cost obtained with different alternatives.

5.5.1 Experimental setup

In this work, we evaluated the use of both HOG [10] and HOG-LBP [82]. Some modifications were introduced into the HOG-LBP descriptor: i) we used the same spatial partition as in HOG for LPB, resulting in 105 spatial blocks; ii) we did not interpolate the pixels around the compared central one, in order to prevent the texture information from being distorted; iii) in order to add robustness against noise, we used an offset when comparing the central pixel with its neighbours; and iv) we only used the luminance channel. Altogether, these changes permitted us to reduce the computational cost while maintaining an accuracy similar to the one of the original definition [82].

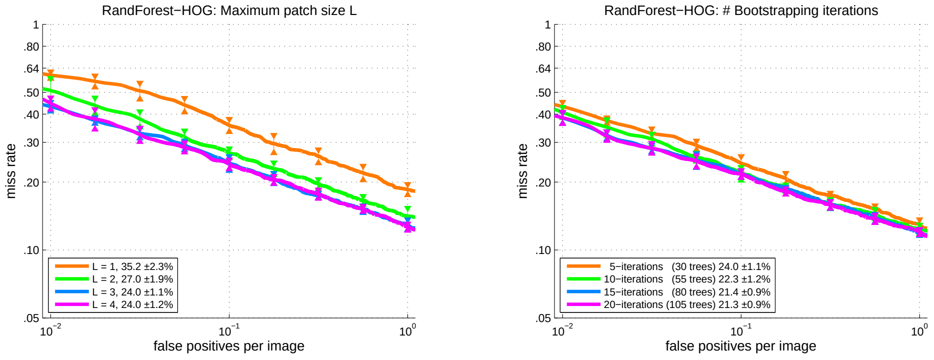
Regarding the computation of the sliding window, we used a step size of eight pixels, which allows to reuse overlapping blocks. For the multi-resolution pyramid we set the scale stride to 1.05.

During the Random Forest construction we used the following stopping criterion. A node is no longer split if either of the following conditions occurs: a) its depth is larger than 6 levels; b) the subset of samples contains less than 10 samples; or c) the percentage of samples from the same class is above 99%. This type of stopping criterion is standard [7], and the specific values were observed to provide good results on the INRIA dataset.

We used a fixed threshold $\eta = 0.1$ in the SC (section 5.4.5). In the Bootstrapping procedure (section 5.4.4) the hard negatives are defined as those negative samples whose classification confidence is larger than 0.25.

Similar to the previous chapters, we performed the standard per-image evaluation used in pedestrian detection [18,19,30]. However, in order to quantify the performance and compare our approach to other state-of-the-art methods we used the well-known Caltech pedestrian toolbox [18]. The unique difference between our previous framework and the Caltech one is the range in which the log-average miss-rate is computed. In their case the compute it in the range 10^-2 to 10^0.

The CGP step described in Section 5.4.6 was only used in the Caltech dataset, in order to obtain a fair comparison with the best performer in this dataset [58], that also uses a similar CGP component. Sections 5.5.3 and 5.5.4 show the performance of our system both including a CGP step and not including it.



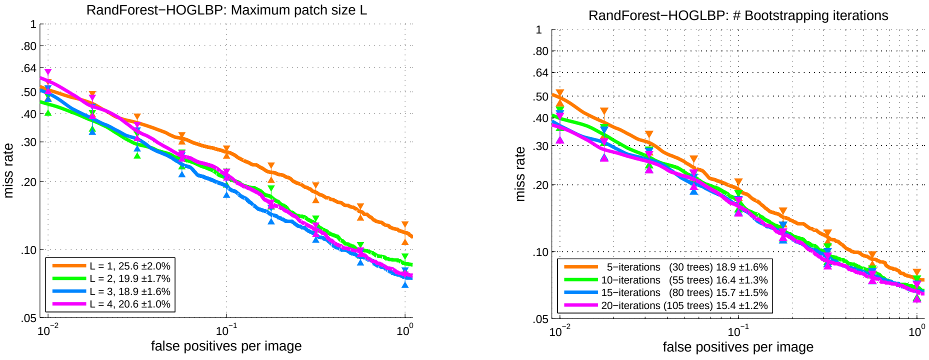
(a) (b)

Figure 5.4: Validation results using HOG. (a) Performance as a function of maximum patch size L, (b) performance as a function of the number of bootstrapping rounds.

5.5.2 Estimation of parameters

Two parameters were exhaustively optimized using the test set of the INRIA dataset. The first one is the maximum patch size L selected by each local expert (see section 5.4.2). This parameter represents the compromise between having an expert that is based on *local* (*i.e.* small) regions and the use of distinctive (*i.e.* large) regions. In Fig. 5.5(a) we can see that the accuracy increases as we increase the maximum patch size, until it reaches 3×3 blocks. Permitting larger patches leads to lower accuracy.

The second parameter is the number of bootstrapping iterations N_{boot}. This parameter is important due to the high computational cost of each bootstrapping iteration, which makes it necessary to estimate the minimum number of iterations that provide an accuracy converging to the maximum. In Fig. 5.5(b) we can see that the accuracy saturates with 20 iterations (we also tried 25 and 30 iterations, but the accuracy was no longer increasing). The results using HOG features were analogous to the HOG-LBP ones.



(a) (b)

Figure 5.5: Validation results using HOGLBP. (a) Performance as a function of maximum patch size L, (b) performance as a function of the number of bootstrapping rounds.

5.5.3 Comparison with the state-of-the-art

Our final detector was evaluated using three well-known, challenging datasets: Caltech [18], Daimler [19] and ETH [21]. Results are shown in Fig. 5.6, where we also include results on INRIA for completeness, and where we compare the accuracy of our approach against the best methods of the state-of-the-art. Regarding the Caltech dataset, most of the works in the literature only use the so-called “Reasonable” subset, so that we use this subset as reference. However, we also show results on the “Overall” and the “Partial occlusion” subsets [18] for completeness. We only use our CGP approach in the Caltech testing dataset (where we used the Caltech training data for estimating the CGP parameters).

Our method matches or outperforms the state-of-the-art methods in all three datasets. Only in the Caltech “Reasonable” subset three methods outperform our approach (if we do not include the CGP component), although the third best performer has a similar accuracy to the one of our method. If we use CGP, the accuracy increases. In this case, our method matches the second best performer (MultiFtr-Motion [81]) and it is outperformed by only one method

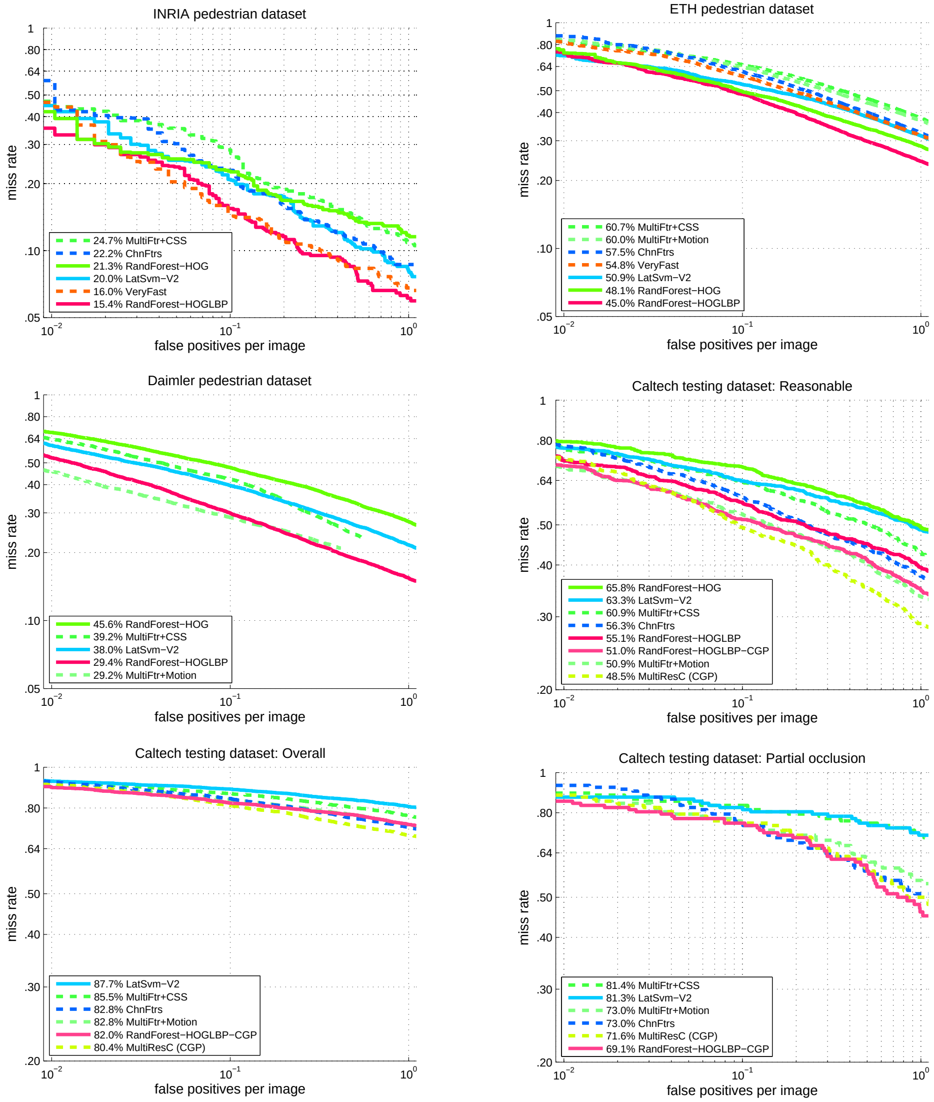


Figure 5.6: Miss rate versus false positive per image curves in the INRIA, Daimler, ETH and Caltech testing. For the Caltech testing dataset we show results under three different conditions: reasonable, overall and partial occlusion (please refer to [18] for further details).

(MultiResC (CGP) [58]) in the reasonable evaluation subset. It is worth mentioning that these methods make use of additional sources of information (multi-resolution in MultiResC (CGP) [58] and motion in MultiFtr-Motion [81]). These sources can also be incorporated in our approach. In fact, both [81] and [58] make use of block-based representations in order to include these sources of information, so that they can be integrated in our framework with moderate changes. This would further increase the accuracy of our method.

Regarding the rest of the other Caltech evaluation subsets ("Overall" and "Partial occlusion"), we can see that the conclusions are maintained. In all the cases, our method is in the top positions. In particular, the RandForest-HOGLBP-CGP outperforms the MultiResC (CGP) in the "Partial occlusion" subset. Additionally, in Fig. 5.7 we show the results of our random forest compared to the linSVM, our previous occlusion handling method, and Wang's approach [82] in the PobleSec dataset. These results are consistent with the fact that local-patch based methods are usually more robust against partial occlusions than the holistic ones.

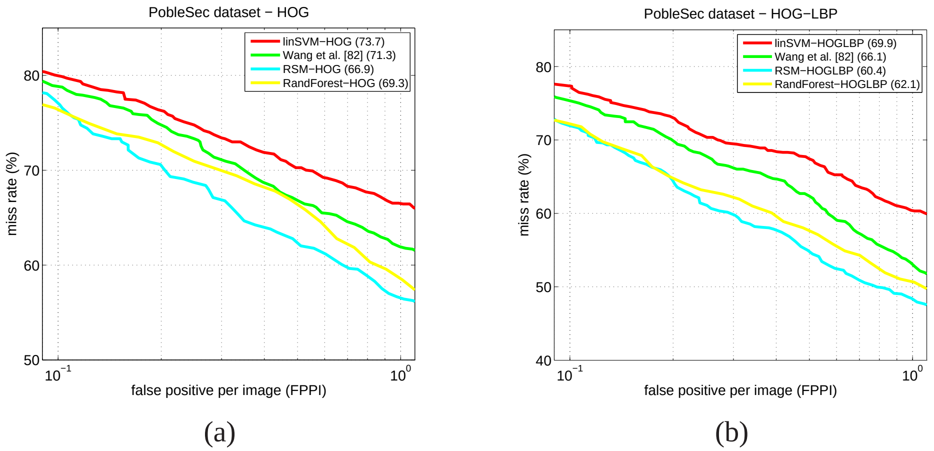


Figure 5.7: Per-image evaluation in PobleSec. (a) Performance using HOG features, and (b) performance using HOGLBP. We refer to our occlusion handling approach as RSM.

5.5.4 Testing speed

In order to evaluate the computational cost, we used a laptop machine with a i7-2860QM CPU at 2,50GHz. Furthermore, we parallelized our code in order to compute several scales of the pyramid at the same time, and in order to compute several trees of the forest at the same time (this last parallelization was only performed for our baseline and it was not performed when using the SC component).

Table 5.1 shows the runtime of the proposed approach, including the baseline without any speed-up, the use of SC and the use of both SC and CGP. If we consider pedestrians with a minimum height of 96 pixels, the system operates at 4 fps with HOG, and 3 fps with

HOG-LBP. This can be further sped up if we use some hardware optimization. For this purpose, we used AVX instructions in order to implement the dot product involved in the SVM classification. In this case, we reached 5.9 fps with HOG and 4.6 fps with HOG-LBP. These times make the resulting system fairly fast in comparison with the state-of-the-art, as evaluated in [18]. As an example, the two fastest detectors evaluated in that survey operate at 1.2 fps and 6.5 fps, while the proposed approach operates at 1.9 fps without any optimization (using only the SoftCascade and excluding the CGP step) and at 4.6 fps if we use both AVX instructions and the CGP step. At the same time, the proposed approach ranks in the top positions in terms of accuracy, as shown in this section.

	$\geq$ 50 pixels		$\geq$ 96 pixels	
	HOG	HOG-LBP	HOG	HOG-LBP
RandForest	0.15	0.09	0.75	0.53
SoftCascade	0.60	0.45	2.51	1.88
SoftCascade + CGP	1.23	0.93	4.01	3.17

Table 5.1: Frames per second (fps) obtained when detecting pedestrians in the Caltech test dataset. Second column shows the fps when detecting pedestrians with a minimum height of 50 pixels using HOG and HOG-LBP. And the third column, the fps for pedestrians with a minimum height of 96 pixels with HOG and HOG-LBP.

5.6 Conclusions

In this chapter, we have presented a novel approach for estimating ensembles of local experts through the RF framework. The proposed approach works with rich block-based descriptors which are reused by the different experts of the ensemble in such a way that each expert selects the most discriminant local patch based on this descriptor. Making use of the RF framework, the patches selected by each tree are both discriminant and complementary, and at the end of the process the forest estimates a diverse collection of ensembles providing both robustness and generalization capabilities. As part of the work, we show how to integrate the proposed RF classifier into a SC architecture.

Altogether the proposed work provides an interesting framework that permit to match the best approaches in terms of accuracy, as measured across several challenging datasets, without including additional sources of information such as motion, multi-resolution or colour. These sources can be easily integrated in the future in order to further increase accuracy. At the same time, we showed how the proposed architecture permits to obtain a quasi real-time performance at test time, on pair with some of the fastest detection approaches. This is due to the integration of the SC component.

Chapter 6

Conclusions

Pedestrian detection is one of the most challenging tasks in Computer Vision. This is mainly due to the range of variability in terms of the appearance caused by the different clothing, poses, sizes and partial occlusions. Moreover, weather conditions may also influence the detector performance, as well as the illumination, the quality of the camera sensor, and highly textured elements present in the background. The research in this field has been mainly concentrating its efforts on two different lines of investigation: 1) design of features, and 2) learning machine algorithms.

In this dissertation we can divide our work into two different stages. In the first stage of the thesis, we have explored how computer graphics could benefit pedestrian detection. In particular, we have addressed the question of whether or not a pedestrian detector trained using virtual data could be successfully applied in a real scenario. A positive answer to this question brings the possibility of reducing or even removing the tiring time of annotating required to collect the training data, which, with no doubt, can be considered a significant contribution.

In the near future, such a virtual data will be naturally collected from the simulators that the automotive industry is developing. In the meantime, as a proof of concept, we have developed an entire virtual city using a mapping tool. To achieve a high realism we have made use of all the possibilities the game engine provided. First, we have introduced elements such as buildings, roads, streets, traffic signs, different ground textures, trash on the floors, trees, dust and cars. Second, we have used over 50 different human models with type of male and female bodies and clothing. Third, we have used the game's artificial intelligence to move the human and car models around the city. Fourth, we have used the best video options available in the game (HDR, anti-aliasing, high quality textures, etc.). Fifth, different types of illuminations. Once we have finalized the development, we have driven through the virtual city with an on-board virtual camera, recording several video-sequences using four different illuminations (we tried to reproduce different daytime moments). Next, we have randomly generated a virtual dataset. Then, we have trained two different detectors, one using the virtual data and another one using a real dataset. For training the detector, we have used the HOG descriptor accompanied with a linear SVM. Both, the HOG descriptor and the SVM are two of the most used features and learning machine algorithms, respectively, in the literature. The number of training samples and the algorithm parameters are the same in both cases. Finally, the two detectors are validated in one large real dataset. The final results demonstrate the viability of using virtual data for pedestrian detection. In fact, both detectors achieve a similar accuracy. We detected domain adaptation problems when applying our virtual-world based detector to new real-world images. This domain adaptation issue gave rise to a full new PhD developed in parallel to this one.

As mentioned before, another challenge is to detect partially occluded pedestrians. Accordingly, In the second part of this dissertation, we have focused on improving the detection against partial occlusions. To this end, two different methods have been proposed: a novel framework for occlusion handling, and a robust detector based on a random forest of local experts ensemble.

We have presented a novel method to handle partial occlusions. After investigating the problem, we have found that in most cases occluded parts tend to confuse the classifier as they can be interpreted as noise or background, and therefore lead to a misclassification. Our belief at this point is that, if the classifier is able to avoid such regions and rely only on those in which no occlusion is present, the chances of handling these cases may increase. Following this hypothesis, we present a general framework in which only those candidates inferred as partially occluded are again evaluated by a Random Subspace Method (RSM). To infer a candidate window as a partially occluded pedestrian we take advantage of the block-based structure and the linearity of the SVM used as the holistic classifier. More concretely, we divide the global response into local responses and then analyse them to figure out if there is a possible partially occluded pedestrian. The key of RSM is how the ensemble has been trained. In particular, the final ensemble we have obtained is the result of a training stage in which the best random classifiers against partial occlusions are selected and then combined. Here, each random classifiers is trained on a rigid compact region of the window formed by blocks. To validate our proposal, we have carried out two different evaluations, per window and per image. For both evaluations, the final detector has been confronted to non-occluded data and partially occluded one. When evaluating the performance against partial occlusions, our algorithm outperforms the current state-of-the-art. Some of the advantages of our method are: it is not class dependent; it can be extended to other block-based descriptors; and other sources of information such motion, stereo or multiresolution can be used for further improvement. Besides, the RSM reuses the same features used by the holistic SVM, which means that no additional computational cost is needed. The experiments have been carried out using two different sets of features, related to shape and texture, in the first case using only HOG and in the second case combining HOG and LBP features. We make publicly available the two datasets created in this work for benchmarking purposes. The virtual dataset used in the validation stage, which includes only partially occluded pedestrians. The real pedestrian dataset to assess the performance at the per image level against partial occlusions, which includes images taken in urban scenarios in Barcelona.

Finally, we have explored how to capture the appearance variability of pedestrians (clothe, pose, view, occlusion) using a ensemble of local experts, where such an ensemble consists in a random forest. Concretely, at every node of each random decision tree the most discriminant local expert is selected during the training process. As basic (node/expert) learning machine we have used HOG+LBP features with the linear SVM to assure an optimum split for each tree node, through a maximum-margin optimization. The robust ensemble is then the result of the complementary joined trees. Later, we integrate the final random forest into a Soft Cascade (SC). The SC is introduced to increase the efficiency while keeping the original accuracy. Finally, we have performed several evaluations in different pedestrians datasets. Moreover, we have assessed the speed of the system. The results show that while matching the best approaches in the state-of-the-art, our method achieves a quasi real-time performance when testing, on a pair with some of the fastest detectors in the literature. Besides, the RF presents some advantages similar to those of our previous occlusion handling method. The initial features are reused by each node and tree of the random forest, the method can be easily extended to other object classes, and it can also benefit from other sources of information such as motion or depth.

As a future work, we would like to assess the detection of other targets such as vehicles using our occlusion handling framework and our random forest of local experts. We would also like to implement two other random forest variants. More concretely, a Extremely Randomized Trees (ERT) ensemble, and a Random Ferns one. ERT have the advantage of reducing the training time with respect the RF. Random Ferns are also faster to train than the RF and they can also be parallelized at the decision tree level. It would be also interesting to include additional sources of information such as stereo, multi-resolution or colour into our system. Finally, we plan to integrate the random forest of local experts proposed in this thesis into our previous occlusion handling framework.

List of Publications

This dissertation has led to the following communications:

Journal Papers

- Javier Marín, David Vázquez, Antonio M. López, Jaume Amores, and Ludmila I. Kuncheva. Occlusion handling via random subspace classifiers for human detection. ***IEEE Transactions on Systems, Man, and Cybernetics - Part B*** 2013.

Conference Contributions

- Jiaolong Xu, David Vázquez, Antonio M. López, Javier Marín, and Daniel Ponsa. Learning a Multiview Part-based Model in Virtual World for Pedestrian Detection. ***In IEEE Intelligent Vehicles Symposium***, 2013.

- David Vázquez, Antonio M. López, Daniel Ponsa, and Javier Marín. Cool world: domain adaptation of virtual and real worlds for human detection using active learning. ***In NIPS Domain Adaptation Workshop: Theory and Application***, 2011.

- David Vázquez, Antonio M. López, Daniel Ponsa, and Javier Marín. Virtual Worlds and Active Learning for Human Detection. ***In 13th International Conference on Multimodal Interaction***, 2011.

- Javier Marín, David Vázquez, David Gerónimo, and Antonio López. Learning Appearance in Virtual Scenarios for Pedestrian Detection. ***In 23rd IEEE Conference on Computer Vision and Pattern Recognition***, 2010.

Chapter Books

- Javier Marín, David Gerónimo, David Vázquez, and Antonio M. López. Pedestrian Detection: Exploring Virtual Worlds. ***In Handbook of Pattern Recognition: Methods***

and Application. iConcept Press, 2012.

Submitted Journals

- David Vázquez, Antonio M. López, Javier Marín, Daniel Ponsa and David Gerónimo. Virtual and Real World Adaptation for Pedestrian Detection. ***IEEE Transactions on Pattern Analysis and Machine Intelligence***, 2013.

Bibliography

[1] Timo Ahonen, Abdenour Hadid, and Matti Pietikäinen. Face description with local binary patterns: Application to face recognition. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 28(12):2037–2041, Dec. 2006.

[2] R. Benenson, M. Mathias, R. Timofte, and L. Van Gool. Pedestrian detection at 100 frames per second. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages 2903–2910, 2012.

[3] R. Bishop. *Intelligent Vehicle Technologies and Trends*. Artech House, Inc., 2005.

[4] Lubomir Bourdev and Jonathan Brandt. Robust object detection via soft cascade. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, 2005.

[5] A. Broggi, A. Fascioli, P. Grisleri, T. Graf, and M. Meinecke. Model–based validation approaches and matching techniques for automotive vision based pedestrian detection. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages 1–, San Diego, CA, USA, 2005.

[6] D. Comaniciu and P. Meer. Mean shift: a robust approach toward feature space analysis. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 24(5):603–619, May 2002.

[7] A. Criminisi, J. Shotton, and E. Konukoglu. Decision forests for classification, regression, density estimation, manifold learning and semi-supervised learning. Technical report, Microsoft Research, 2011.

[8] S. Dai, M. Yang, Y. Wu, and A. Katsaggelos. Detector ensemble. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages 1–8, Minneapolis, Minnesota, USA, 2007.

[9] N. Dalal. *Finding People in Images and Videos*. PhD Thesis, Institut National Polytechnique de Grenoble / INRIA Rhône-Alpes, 2006.

[10] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages 886–893, San Diego, CA, USA, 2005.

[11] N. Dalal, B. Triggs, and C. Schmid. Human detection using oriented histograms of flow and appearance. In *Proc. the European Conf. on Computer Vision*, 2006.

[12] James W. Davis and Vinay Sharma. Background-subtraction using contour-based fusion of thermal and visible imagery. *Journal of Computer Vision and Image Understanding*, 106(2-3):162–182, May 2007.

[13] P. Dollár, R. Appel, and W. Kienzle. Crosstalk cascades for frame-rate pedestrian detection. In *Proc. the European Conf. on Computer Vision*, 2012.

[14] P. Dollár, S. Belongie, and P. Perona. The fastest pedestrian detector in the west. In *Proc. British Machine Vision Conference*, 2010.

[15] P. Dollar, Z. Tu, P. Perona, and S. Belongie. Integral channel features. In *Proc. British Machine Vision Conference*, 2009.

[16] P. Dollár, Z. Tu, H. Tao, and S. Belongie. Feature mining for image classification. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, 2007.

[17] P. Dollár, C. Wojek, B. Schiele, and P. Perona. Pedestrian detection: A benchmark. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages 304–311, 2009.

[18] P. Dollár, C. Wojek, B. Schiele, and P. Perona. Pedestrian detection: An evaluation of the state of the art. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 34(4):743–761, April 2012.

[19] M. Enzweiler and D. Gavrila. Monocular pedestrian detection: Survey and experiments. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 31(12):2179–2195, Dec. 2009.

[20] M. Enzweiler and D.M. Gavrila. A mixed generative-discriminative framework for pedestrian classification. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages 1–8, Anchorage, AK, USA, 2008.

[21] A. Ess, B. Leibe, and L. Van Gool. Depth and appearance for mobile scene analysis. In *Proc. IEEE Int. Conf. on Computer Vision*, 2007.

[22] M. Everingham and al. The 2005 PASCAL visual object classes challenge. In *Proceedings of the First PASCAL Challenges Workshop, LNAI, Springer-Verlag*, 2006.

[23] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (VOC) challenge. *Int. Journal on Computer Vision*, 88(2):303–338, Jun. 2010.

[24] P. Felzenszwalb, R. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 2010.

[25] P. Felzenszwalb, D. McAllester, and D. Ramanan. A discriminatively trained, multiscale, deformable part model. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages 1–8, Anchorage, AK, USA, 2008.

[26] J. Gall and V. Lempitsky. Class-specific hough forests for object detection. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, 2009.

[27] T. Gandhi and M. M. Trivedi. Pedestrian protection systems: issues, survey, and challenges. *IEEE Trans. on Intelligent Transportation Systems*, 8(3):413–430, 2007.

[28] Tianshi Gao, Benjamin Packer, and Daphne Koller. A segmentation-aware object detection model with occlusion handling. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages 1361–1368, Colorado Springs, CO, USA, 2011.

[29] D. Gerónimo. *A global approach to vision-based pedestrian detection for advanced driver assistance systems*. PhD thesis, Computer Vision Center, Universitat Autònoma de Barcelona, 2010.

[30] D. Gerónimo, Antonio M. López, Ángel D. Sappa, and T. Graf. Survey of pedestrian detection for advanced driver assistance systems. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 32(7):1239–1258, July 2010.

[31] Ross B. Girshick, Pedro F. Felzenszwalb, and David McAllester. Object detection with grammar models. In *Neural Information Processing Systems*, pages 442–450, Granada, Spain, 2011.

[32] Kristen Grauman, Gregory Shakhnarovich, and Trevor Darrell. Inferring 3d structure with a statistical image-based shape model. In *Proc. IEEE Int. Conf. on Computer Vision*, pages 641–648, Nice, France, 2003.

[33] Chunhui Gu, Joseph J. Lim, Pablo Arbelaez, and Jitendra Malik. Recognition using regions. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages 1030–1037, 2009.

[34] Tin Kam Ho. The random subspace method for constructing decision forests. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 20(8):832–844, Aug. 1998.

[35] L. I. Kuncheva, J. J. Rodriguez, C. O. Plumpton, D. E. J. Linden, and S. J. Johnston. Random subspace ensembles for fMRI classification. *IEEE Trans. on Medical Imaging*, 29(2):531–542, Feb. 2010.

[36] Ludmila I. Kuncheva. *Combining Pattern Classifiers: Methods and Algorithms*. Wiley-Interscience, 2004.

[37] B. Kuneva, A. Torralba, and W. T. Freeman. Evaluation of image features using a photorealistic virtual world. In *Proc. IEEE Int. Conf. on Computer Vision*, pages 2282–2289, Barcelona, Spain, 2011.

[38] I. Laptev. Improving object detection with boosted histograms. *Image and Vision Computing*, 27(5):535–544, 2009.

[39] B. Leibe, A. Leonardis, and B. Schiele. Robust object detection with interleaved categorization and segmentation. *Int. Journal on Computer Vision*, 2008.

[40] Bastian Leibe, Edgar Seemann, and Bernt Schiele. Pedestrian detection in crowded scenes. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages 878–885, 2005.

[41] Z. Lin and L.S. Davis. A pose-invariant descriptor for human detection and segmentation. In *Proc. the European Conf. on Computer Vision*, volume 4, pages 423–436, Marseille, France, 2008.

[42] Zhe Lin, Gang Hua, and L.S. Davis. Multiple instance ffeature for robust part-based object detection. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages 405–412, 2009.

[43] David G. Lowe. Distinctive image features from scale-invariant keypoints. *Int. Journal on Computer Vision*, 60(2):91–110, 2004.

[44] B. Schiele M. Enzweiler, A. Eigenstetter and D. M. Gavrila. Multi-cue pedestrian classification with partial occlusion handling. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages 990–997, San Francisco, CA, USA, 2010.

[45] S. Maji, A. Berg, and J. Malik. Classification using intersection kernel support vector machines is efficient. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages 1–8, Anchorage, Alaska, USA, 2008.

[46] Javier Marín, David Vázquez, David Gerónimo, and Antonio M López. Learning appearance in virtual scenarios for pedestrian detection. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages 137–144, San Francisco, CA, USA, 2010.

[47] Krystian Mikolajczyk, Cordelia Schmid, and Andrew Zisserman. Human detection based on a probabilistic assembly of robust part detectors. In *Proc. the European Conf. on Computer Vision*, volume I, pages 69–81, 2004.

[48] Anuj Mohan, Constantine Papageorgiou, and Tomaso Poggio. Example-based object detection in images by components. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 23(4):349–361, 2001.

[49] S. Munder and D.M. Gavrila. An experimental study on pedestrian classification. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 28(11):1863–1868, 2006.

[50] S.K. Nayar and V. Branzoi. Adaptative dynamic range imaging: optical control of pixel exposures over space and time. In *Proc. IEEE Int. Conf. on Computer Vision*, 2003.

[51] T. Ojala, M. Pietik ainen, and D. Harwood. Performance evaluation of texture measures with classification based on Kullback discrimination of distributions. In *Proc. Int. Conf. in Pattern Recognition*, pages 582–685, 1994.

[52] Timo Ojala, Matti Pietikäinen, and David Harwood. A comparative study of texture measures with classification based on featured distributions. *Pattern Recognition*, 29(1):51–59, Jan. 1996.

[53] World Health Organization. *Global status report on road safety 2013: supporting a decade of action*. World Health Organization, 2013.

[54] Wanli Ouyang and Xiaogang Wang. A discriminative deep model for pedestrian detection with occlusion handling. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages 3258–3265, 2012.

[55] Junbiao Pang, Qingming Huang, and Shuqiang Jiang. Multiple instance boost using graph embedding based decision stump for pedestrian detection. In *Proc. the European Conf. on Computer Vision*, pages 541–552, 2008.

[56] Yanwei Pang, He Yan, Yuan Yuan, and Kongqiao Wang. Robust CoHOG feature extraction in human-centered image/video management system. *Proc. IEEE on Systems, Man, and Cybernetics. B, Cybernetics.*, 42(2):458–468, April 2012.

[57] Constantine Papageorgiou and Tomaso Poggio. A trainable system for object detection. *Int. Journal on Computer Vision*, 38(1):15–33, 2000.

[58] Dennis Park, Deva Ramanan, and Charless Fowlkes. Multiresolution models for object detection. In *Proc. the European Conf. on Computer Vision*, 2010.

[59] D. Partridge and W. B. Yates. Engineering multiversion neural-net systems. *Neural Computation*, 8(4):869–893, May 1996.

[60] Marco Pedersoli, Andrea Vedaldi, and Jordi Gonzàlez. A coarse-to-fine approach for fast deformable object detection. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages 1353–1360, 2011.

[61] Leonid Pishchulin, Arjun Jain, Christian Wojek, Mykhaylo Andriluka, Thorsten Thormaehlen, and Bernt Schiele. Learning people detection models from few training samples. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages 1473–1480, 2011.

[62] Leonid Pishchulin, Arjun Jain, Christian Wojek, Thorsten Thormaehlen, and Bernt Schiele. In good shape: Robust people detection based on appearance and shape. In *22nd British Machine Vision Conference (BMVC)*, Dundee, UK, 2011. Oral.

[63] Jean Ponce, T. L. Berg, M. Everingham, D. Forsyth, M. Hebert, Svetlana Lazebnik, Marcin Marszałek, Cordelia Schmid, C. Russell, A. Torralba, C. Williams, Jianguo Zhang, and Andrew Zisserman. Dataset issues in object recognition. In *Towards Category-Level Object Recognition*, pages 29–48. Springer, 2006.

[64] William Robson Schwartz, Aniruddha Kembhavi, David Harwood, and Larry S. Davis. Human detection using partial least squares analysis. In *Proc. IEEE Int. Conf. on Computer Vision*, pages 24–31, 2009.

[65] Edgar Seemann, Bastian Leibe, Krystian Mikolajczyk, and Bernt Schiele. An evaluation of local shape-based features for pedestrian detection. In *Proc. British Machine Vision Conference*, 2005.

[66] A. Shashua, Y. Gdalyahu, and G. Hayun. Pedestrian detection for driving assistance systems: single-frame classification and system level performance. In *IEEE Intelligent Vehicles Symposium*, pages 1–6, 2004.

[67] John Sousanis. World vehicle population tops 1 billion units. *Ward AutoWorld*, 2011.

[68] D. Tang, Y. Liu, and T.-K. Kim. Fast pedestrian detection by cascaded random forest with dominant orientation templates. In *BMVC*, 2012.

[69] G.R. Taylor, A.J. Chosak, and P.C. Brewer. OVVV: Using virtual worlds to design and evaluate surveillance systems. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages 1–8, Minneapolis, MN, USA, 2007.

[70] Valve Software Corporation. http://www.valvesoftware.com, 2004.

[71] D. Tosato, M. Farenzena, M. Cristani, and V. Murino. Part-based human detection on riemannian manifolds. In *IEEE International Conference on Image Processing*, pages 3469–3472, 2010.

[72] O. Tuzel, F. Porikli, and P. Meer. Pedestrian detection via classification on Riemannian Manifold. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 30(10):1713–1727, Oct. 2008.

[73] UN-ECE. *Economic Comision for Europe - Statistics of Road Traffic Accidents in Europe and North America*, volume LI. United Nations, 2007.

[74] Department of Economic United Nations and Social Affairs. The 2010 revision. *World Population Prospects*, 2010.

[75] David Vázquez, Antonio M. López, Daniel Ponsa, and Javier Marín. Virtual worlds and active learning for human detection. In *ACM International Conference on Multimodal Interaction*, pages 393–400, 2011.

[76] David Vázquez, Antonio M. López, Daniel Ponsa, and Javier Marín. Virtual worlds and active learning for human detection. In *ACM International Conference on Multimodal Interaction*, pages 393–400, Alicante, Spain, 2011.

[77] A. Vedaldi, V. Gulshan, M. Varma, and A. Zisserman. Multiple kernels for object detection. In *Proc. IEEE Int. Conf. on Computer Vision*, pages 606–613, 2009.

[78] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages 511–518, Kauai, HI, USA, 2001.

[79] P. Viola and M. Jones. Robust real-time face detection. *Int. Journal on Computer Vision*, 57(2):137–154, July 2004.

[80] Paul A. Viola, Michael J. Jones, and Daniel Snow. Detecting pedestrians using patterns of motion and appearance. In *Proc. IEEE Int. Conf. on Computer Vision*, pages 734–741, 2003.

[81] S. Walk, N. Majer, K. Schindler, and B. Schiele. New features and insights for pedestrian detection. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages 1030–1037, San Francisco, CA, USA, 2010.

[82] X. Wang, T.X. Han, and S. Yan. An HOG–LBP human detector with partial occlusion handling. In *Proc. IEEE Int. Conf. on Computer Vision*, pages 32–39, Kyoto, Japan, 2009.

[83] Christian Wojek and Bernt Schiele. A performance evaluation of single and multi-feature people detection. In *DAGM Symposium Pattern Recognition*, 2008.

[84] Christian Wojek, Stefan Walk, and Bernt Schiele. Multi-cue onboard pedestrian detection. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages 1–8, 2009.

[85] B. Wu and R. Nevatia. Detection and segmentation of multiple, partially occluded objects by grouping, merging, assigning part detector responses. *Int. Journal on Computer Vision*, 82(2):185–204, April 2009.

[86] Bo Wu and R. Nevatia. Simultaneous object detection and segmentation by boosting local shape feature based classifier. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages 1–8, 2007.

[87] Bo Wu and Ramakant Nevatia. Detection of multiple, partially occluded humans in a single image by bayesian combination of edgelet part detectors. In *Proc. IEEE Int. Conf. on Computer Vision*, pages 90–97, 2005.

[88] Yanwu Xu, Xianbin Cao, and Hong Qiao. An efficient tree classifier ensemble-based approach for pedestrian detection. *Proc. IEEE on Systems, Man, and Cybernetics. B, Cybernetics.*, 41(1):107–117, Feb. 2011.

[89] Yanwu Xu, Dong Xu, Stephen Lin, Tony X. Han, Xianbin Cao, and Xuelong Li. Detection of sudden pedestrian crossing for driving assistance systems. *Proc. IEEE on Systems, Man, and Cybernetics. B, Cybernetics.*, 42(3):729–739, Jun. 2012.

[90] Junge Zhang, Kaiqi Huang, Yinan Yu, and Tieniu Tan. Boosted local structured HOG-LBP for object localization. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages 1393–1400, Colorado Springs, CO, USA, 2011.

[91] Li Zhang, Bo Wu, and R. Nevatia. Pedestrian detection in infrared images based on local shape features. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages 1–8, 2007.

[92] Liang Zhao and Charles E. Thorpe. Stereo- and neural network-based pedestrian detection. *IEEE Transactions on Intelligent Transportation Systems*, 1(3):148–154, 2000.

[93] Q. Zhu, S. Avidan, M.-C. Yeh, and K.-T. Cheng. Fast human detection using a cascade of histograms of oriented gradients. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, 2006.