

**THE GOOD, THE FAST AND THE BETTER  
PEDESTRIAN DETECTOR**



ARTUR JORDÃO LIMA CORREIA.

**THE GOOD, THE FAST AND THE BETTER  
PEDESTRIAN DETECTOR**

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação do Instituto de Ciências Exatas da Universidade Federal de Minas Gerais - Departamento de Ciência da Computação. como requisito parcial para a obtenção do grau de Mestre em Ciência da Computação.

ORIENTADOR: WILLIAM ROBSON SCHWARTZ

Belo Horizonte

Junho de 2016



ARTUR JORDÃO LIMA CORREIA.

**THE GOOD, THE FAST AND THE BETTER  
PEDESTRIAN DETECTOR**

Dissertation presented to the Graduate Program in Ciência da Computação of the Universidade Federal de Minas Gerais - Departamento de Ciência da Computação. in partial fulfillment of the requirements for the degree of Master in Ciência da Computação.

ADVISOR: WILLIAM ROBSON SCHWARTZ

Belo Horizonte

June 2016

© 2016, Artur Jordão Lima Correia..  
Todos os direitos reservados.

Artur Jordão Lima Correia.  
C824g The Good, the Fast and the Better pedestrian  
detector / Artur Jordão Lima Correia.. — Belo  
Horizonte, 2016  
xx, 51 f. : il. ; 29cm

Dissertação (mestrado) — Universidade Federal de  
Minas Gerais - Departamento de Ciência da  
Computação.  
Orientador: William Robson Schwartz

1. Computação - Teses. 2. Visão por computador  
- Teses. 3. Teoria da estimativa - Teses. 4. Detecção de  
pedestres. I.Orientador. II Título.

CDU 519.6\*82.10(043)




UNIVERSIDADE FEDERAL DE MINAS GERAIS  
INSTITUTO DE CIÊNCIAS EXATAS  
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO


## FOLHA DE APROVAÇÃO

The good, the fast and the better pedestrian detector

**ARTUR JORDAO LIMA CORREIA**

Dissertação defendida e aprovada pela banca examinadora constituída pelos Senhores:

  
PROF. WILLIAM ROBSON SCHWARTZ - Orientador  
Departamento de Ciência da Computação - UFMG

  
PROF. DAVID MENOTTI GOMES  
Departamento de Informática - UFPR

  
PROF. JEFFERSSON ALEX DOS SANTOS  
Departamento de Ciência da Computação - UFMG

Belo Horizonte, 24 de junho de 2016.





# Acknowledgments

I am eternally thankful for all support that my parents gave me, allowing me to focus on research and studies.

I would like to thank deeply professor William Robson Schwartz for the outstanding orientation on my graduate study.

A special thanks to my friends: Bruno Salomão, Felipe Casanova, Fernando Plantier, Caio Russi, Arthur Santos, Renan Ferreira (Xisto), Pedro Machado, Anderson Gohara, Guilherme Potje, Thais Lima, Renata Boin and Ana Flávia, for the sincere friendship.

Also, I thank my colleagues in Federal University of Minas Gerais: Luis Pedraza, André Costa, Leandro Augusto, Thiago Rodrigues, Gabriel Gonçalves, Clebson Cardoso, Ricardo Kloss, Jessica Senna, Victor Melo, Antonio Nazaré, Marco Tulio, Rensso Mora, Cássio Elias, Rafael Vareto, Carlos Caetano, Ramon Pessoa, Raphael Prates, César Augusto.

I would like to thank the Brazilian National Research Council – CNPq (Grant #477457/2013-4), the Minas Gerais Research Foundation – FAPEMIG (Grants APQ-00567-14 and PPM-00025-15) and the Coordination for the Improvement of Higher Education Personnel – CAPES (DeepEyes Project).



# Resumo

Detecção de pedestres é um bem conhecido problema em Visão Computacional, principalmente por causa de sua direta aplicação em vigilância, segurança de trânsito e robótica. Na última década, vários esforços têm sido realizados para melhorar a detecção em termos de acurácia, velocidade e aprimoramento de *features*. Neste trabalho, nós propomos e analisamos técnicas focando sobre estes pontos. Primeiro, nós desenvolvemos uma acurada *random forest* oblíqua (oRF) associada com *Partial Least Squares* (PLS). O método utiliza o PLS para encontrar uma superfície de decisão, em cada nó de uma árvore de decisão. Para mensurar as vantagens providas pelo PLS sobre o oRF, nós comparamos o método proposto com a *random forest* oblíqua baseada em SVM. Segundo, nós avaliamos e comparamos abordagens de filtragem para reduzir o espaço de busca e manter somente regiões de potencial interesse para serem apresentadas para os detectores, acelerando o processo de detecção. Resultados experimentais demonstram que os filtros avaliados são capazes de descartar um grande número de janelas de detecção sem comprometer a acurácia. Finalmente, nós propomos uma nova abordagem para extrair poderosas *features* em relação à cena. O método combina resultados de distintos detectores de pedestres reforçando as hipóteses humanas, enquanto que suprime um significativo número de falsos positivos devido à ausência de consenso espacial quando múltiplos detectores são considerados. A abordagem proposta, referida como *Spatial Consensus* (SC), supera os resultados de todos os métodos de detecção de pedestres previamente publicados.



# Abstract

Pedestrian detection is a well-known problem in Computer Vision, mostly because of its direct applications in surveillance, transit safety and robotics. In the past decade, several efforts have been performed to improve the detection in terms of accuracy, speed and feature enhancement. In this work, we propose and analyze techniques focusing on these points. First, we develop an accurate oblique random forest (oRF) associated with Partial Least Squares (PLS). The method utilizes the PLS to find a decision surface, at each node of a decision tree, that better splits the samples presented to it, based on some purity criterion. To measure the advantages provided by PLS on the oRF, we compare the proposed method with the oRF based on SVM. Second, we evaluate and compare filtering approaches to reduce the search space and keep only potential regions of interest to be presented to detectors, speeding up the detection process. Experimental results demonstrate that the evaluated filters are able to discard a large number of detection windows without compromising the accuracy. Finally, we propose a novel approach to extract powerful features regarding the scene. The method combines results of distinct pedestrian detectors by reinforcing the human hypothesis, whereas suppressing a significant number of false positives due to the lack of spatial consensus when multiple detectors are considered. Our proposed approach, referred to as Spatial Consensus (SC), outperforms all previously published state-of-the-art pedestrian detection methods.

**Keywords:** Random Forest, Oblique Decision Tree, Partial Least Squares, Filtering Approaches, High-Level Information, Fusion of Detectors, Pedestrian Detection.



# List of Figures

1.1	Detection pipeline used to find people in images. . . . .	2
3.1	Pipeline detection and its respective section. . . . .	13
3.2	Decision tree split types (the bars represent the information gain). . . . .	15
3.3	Translucent areas demonstrate regions eliminated by filtering stage for different filtering approaches. . . . .	18
3.4	Different regions of the image (detection windows) captured by sliding windows approach and their respective magnitude images where $M$ is the average gradient magnitude computed from each region using Equation 3.10. . . . .	19
3.5	Sliding window approach on saliency map. . . . .	21
3.6	Detection results and their respective heat map. . . . .	22
3.7	Different aspects between our proposed Spatial Consensus algorithm and the weighted-NMS [Jiang and Ma, 2015]. . . . .	25
4.1	Image examples from the datasets used in this work. . . . .	28
4.2	Log-average miss rate achieved in each bootstrapping iteration using oRF-PLS and oRF-SVM, respectively, on validation set. . . . .	30
4.3	Log-average miss-rate (in percentage points) on the validation set as a function of the number of trees. . . . .	31
4.4	Comparison of our oRF-PLS approach with the state-of-the-art. . . . .	32
4.5	Tradeoff between scale factor and number of windows generated for a $640 \times 480$ image. . . . .	33
4.6	Threshold approaches analyzed to be used as rejection criteria in the saliency map. . . . .	34
4.7	Relationship between rejection percentage and recall achieved by filters (assuming that an ideal detector was employed after the filtering stage). . . . .	35
4.8	Number of false positives as a function of the number of detectors added and the threshold $\sigma$ (results on INRIA Person dataset). . . . .	39

4.9	Comparison between our proposed method with the baseline in terms of improvement and depreciation (according to $det_{root}$ ) of the log-average miss rate. . . . .	40
4.10	Comparison of our proposed approach with the state-of-the-art. . . . .	44



# List of Tables

2.1	Overview of state-of-the-art detectors on INRIA person dataset, sorted by log-average miss-rate. . . . .	11
4.1	Miss rate obtained at $10^0$ FPPI with different scale factors. . . . .	34
4.2	Miss rate at $10^0$ FPPI applying the filtering stage on the detectors. . . . .	36
4.3	INRIA Person Detectors Accumulation. . . . .	37
4.4	ETH Detectors Accumulation. . . . .	37
4.5	Caltech Detectors Accumulation. . . . .	37



# Contents

<b>Acknowledgments</b>	<b>ix</b>
<b>Resumo</b>	<b>xi</b>
<b>Abstract</b>	<b>xiii</b>
<b>List of Figures</b>	<b>xv</b>
<b>List of Tables</b>	<b>xvii</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Motivation . . . . .	4
1.2 Objectives . . . . .	5
1.3 Contributions . . . . .	5
1.4 Work Organization . . . . .	6
<b>2 Related Work</b>	<b>7</b>
<b>3 Methodology</b>	<b>13</b>
3.1 Oblique Random Forest with Partial Least Squares . . . . .	14
3.1.1 Partial Least Squares . . . . .	14
3.1.2 Oblique Random Forest . . . . .	15
3.1.3 Bootstrapping . . . . .	17
3.2 Filtering Approaches . . . . .	18
3.2.1 Entropy Filter . . . . .	18
3.2.2 Magnitude Filter . . . . .	19
3.2.3 Random Filtering . . . . .	20
3.2.4 Saliency Map based on Spectral residual . . . . .	21
3.3 Spatial Consensus . . . . .	22
3.3.1 Removing the Dependency of the Root Detector . . . . .	24

<b>4</b>	<b>Experimental Results</b>	<b>27</b>
4.1	Datasets . . . . .	27
4.2	Oblique Random Forest Evaluation . . . . .	29
4.2.1	Feature Extraction . . . . .	29
4.2.2	Tree Parameters . . . . .	29
4.2.3	Bootstrapping Contribution . . . . .	30
4.2.4	Influence of the Number of Trees . . . . .	30
4.2.5	Time Issues . . . . .	31
4.2.6	Comparison with Baselines . . . . .	31
4.3	Filtering Approaches . . . . .	33
4.3.1	Scaling Factor Evaluation . . . . .	33
4.3.2	Saliency Map Threshold . . . . .	34
4.3.3	Number of Discarded Windows . . . . .	35
4.3.4	Filtering Approaches Coupled with Detectors. . . . .	35
4.4	Spatial Consensus . . . . .	36
4.4.1	Preparing the Input Detectors . . . . .	37
4.4.2	Jaccard Coefficient Influence . . . . .	38
4.4.3	Weighted-NMS Baseline . . . . .	39
4.4.4	Spatial Consensus vs. weighted-NMS . . . . .	39
4.4.5	Influence of a Less Accurate Detector . . . . .	41
4.4.6	Comparison with the State-of-the-Art . . . . .	41
4.4.7	Domain Knowledge . . . . .	41
4.4.8	Virtual Root Detector . . . . .	42
4.4.9	Limitations of the Method . . . . .	42
4.4.10	Time Issues . . . . .	43
<b>5</b>	<b>Conclusions</b>	<b>45</b>
5.1	Future Works . . . . .	46
	<b>Bibliography</b>	<b>47</b>

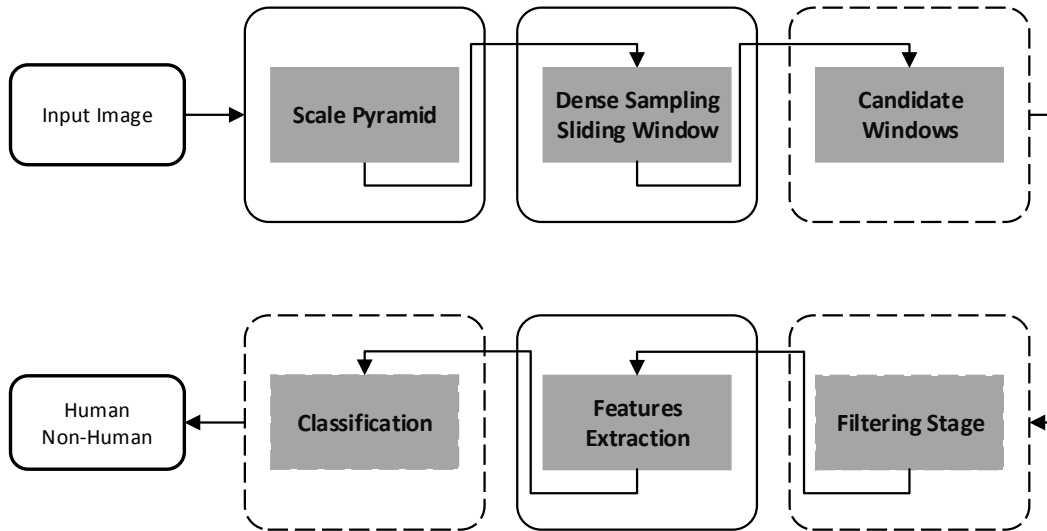
# Chapter 1

## Introduction

Since the past decade, pedestrian detection has been an active research topic in Computer Vision, mostly because of its direct applications in surveillance, transit safety and robotics [Benenson et al., 2014]. This task faces many challenges, such as variance in clothing styles and appearance, distinct illumination conditions, frequent occlusion among pedestrians and high computational cost.

Figure 1.1 introduces the steps employed by traditional approaches to detect pedestrians in an image. First, the image is downsampled by a scale factor generating a set of new images, this procedure is named scale pyramid. Then, a window slides on each image of the pyramid yielding several candidate windows. Once the candidates have been generated, they might be presented to an optional filtering stage, employed to remove a large number of windows quickly. Finally, for each candidate window, features are extracted and presented to a classifier that assigns a score, which will be considered as the likelihood of having a pedestrian located at the particular location in the image. Different challenges are found throughout this pipeline and this work addresses some of them. More specifically, we tackle these challenges by acting on three main points: classification, candidate rejection, and fusion of detectors.

According to Benenson et al. [2014], the most promising pedestrian detection methods are based on deep learning and random forest. Despite accurate, deep learning approaches (commonly convolutional neural networks) require a powerful hardware architecture and considerable amount of samples to learn a model. Moreover, the best results associated to such approaches are comparable with simpler methods [Dollár et al., 2012; Benenson et al., 2014]. On the other hand, random forest approaches are able to run on simple CPU architecture and can be learned with fewer samples. The increasing number of studies based on this classifier is due several advantages that this approach presents including low computational cost to test, probabilistic output and



**Figure 1.1.** Detection pipeline used to find people in images.

it naturally treats problems with more than two classes [Criminisi and Shotton, 2013].

Following the definition of Breiman [2001], a random forest is a set of decision trees, in which the response is a combination of all tree responses at the forest. We can classify a random forest according to the type of the decision tree that it is composed: orthogonal or oblique. In the former type, each tree node creates a boundary decision axis-aligned, i.e, it divides the data selecting an individual feature at a time. The latter type separates the data by oriented hyperplanes, providing better data separation and shallower trees [Menze et al., 2011]. Inspired by these features, in the first part of this work, we propose a novel oblique random forest (oRF) associated with Partial Least Squares (PLS) [Jordao and Schwartz, 2016], which is a popular technique to dimensionality reduction and regression [Schwartz et al., 2009, 2011; de Melo et al., 2014].

Even providing an accurate detection, the proposed method based on oblique random forest leads to a high computational cost, since each detection window must be projected in each node at the tree (path from the root to the leaf) to obtain its confidence. This is a drawback of this class of oblique random forest. However, several pedestrian detection optimization approaches can be utilized to address the referred problem. The majority of the optimization approaches focuses on four main aspects, namely: (i) computing fast features [Nam et al., 2014; Dollár et al., 2014]; (ii) cascades of rejection [Ko et al., 2013, 2014]; (iii) parallelization and use of GPUs [Masaki et al.,

2010; Benenson et al., 2012a]; and (iv) filtering regions of interest [Silva et al., 2012; de Melo et al., 2014]. Among the aforementioned approaches, filtering regions of interest is a simple and effective way of speeding up the detection. Filtering approaches are performed before of the feature extraction and classification stage, and focus on reducing the amount of data that has to be processed, allowing the consideration of fewer samples (detection windows), reducing the computational cost. Such approaches are based on two prior knowledges: (1) only a subset of all detection windows contains the target object (the distribution between pedestrian and non-pedestrian is largely unbalanced) and (2) several detection windows cover the same object at the scene [de Melo et al., 2013; Silva et al., 2012; de Melo et al., 2014].

Although filtering approaches are effective, it is unclear which filters are more appropriate according to the detector employed since there is not a study evaluating this relationship. Even though similar studies have been performed in previous works [Dollár et al., 2009, 2014], where several techniques to improve the detection rate were evaluated, to the best of our knowledge, there is not a comparison among filters in terms of efficiency and robustness, i.e., the ability of rejecting candidate windows while preserving a high detection rate. This motivated the second part of our work, where we evaluate and compare filtering approaches to both reduce the search space and keep only potential regions of interest to be presented to detectors [Jordao et al., 2015].

While numerous classification methods and optimization approaches have been investigated, the majority of efforts in pedestrian detection in the last years can be attributed to the improvement in features alone and evidences suggest that this trend will continue [Dollár et al., 2012; Benenson et al., 2014]. In addition, several works show that the combination of features creates a more powerful descriptor which improves the detection [Schwartz et al., 2009; Dollár et al., 2009; Marín et al., 2013]. Despite the combination of features provide a better discrimination, pedestrian detection is still dealing with some problems. The existence of false positives, such as lampposts, tree and plates, which are very similar to the human body, is a difficult problem to solve. To address this problem, previous works employed high level information regarding the scene to refine the detections [Schwartz et al., 2011; Li et al., 2010; Benenson et al., 2014; Jiang and Ma, 2015].

The most recent work regarding high level information, proposed by Jiang and Ma [2015], relies on the following hypothesis. If two detectors find the same object, given a determined overlapping area, the window with lower response is discarded and its confidence multiplied by a weight is added to the kept window. This is powerful because in the event of a true positive, the discarded window helps to increase the confidence of the kept one, while in the case of a false positive, it contributes to decrease

the confidence. However, when the windows do not overlap, their method keeps both, which might increase the number of false positives (details in Section 3.3). Aiming at tackling such limitation, in the third part of this work, we propose a novel late fusion method called *Spatial Consensus (SC)* to combine multiple detectors [Jordao et al., 2016].

According to the experimental results, the proposed oblique random forest based on PLS (oRF-PLS) achieves comparable results when compared with traditional methods based on HOG features. Besides, we demonstrate that a smaller forest is produced when compare to the oblique random forest based on SVM (oRF-SVM). In the experiments considering the filtering approaches, we demonstrate that the evaluated filters are able to discard a large number of windows without compromising the detection accuracy. Finally, regarding the spatial consensus algorithm, experiments showed that it outperforms the state-of-the-art, achieving the best results in all evaluated datasets.

## 1.1 Motivation

An important application involving the pedestrian detection is to improve the efficient of the work of a human operator. For instance, large surveillance centers demand which a single operator observes several cameras at the same time to find suspicious activities. However, studies show that in a short time the concentration is lost since this activity is routine and monotonous [Smith, 2004]. To avoid that, pedestrian detection algorithms might be employed to attract the operator’s attention to a determined camera (or another surveillance device) and relevant regions of the scene, improving the efficient of the work. Another target in detect people in images is directed to automatic systems applications, e.g, driving assistance and robotics. In these applications, the pedestrian detection assists on the decision-making, focusing on avoiding damage to the humans and the environment. The issues listed above require a robust and accurate pedestrian detection, these requirements motivated us to propose and study a series of techniques focused on improvement of pedestrian detection.

Our first center of attention regards the classification stage associated with oblique random forest. Such class of random forest is commonly generated using the SVM as oriented hyperplane (details in Section 3.1.2). This inspired the first part of our work, where we demonstrate experimentally that the PLS provide a more accurate oblique random forest than SVM [Jordao and Schwartz, 2016].

Due to numerous projections (one for each node at the tree that composes the forest), the oblique random forest presents high computational cost. This fact encouraged



the second part of our work, in which we consider several optimization approaches to keep only regions of the scene where there is the object of interest [Jordao et al., 2015]. Therewith, a smaller number of candidate windows are propagated to the classification stage to allow a faster pedestrian detection without compromising the detection accuracy.

The promising results using high level information regarding the scene to refine detections [Schwartz et al., 2011; Li et al., 2010; Benenson et al., 2014; Jiang and Ma, 2015] motivated the third part of our work, where we propose a novel late fusion method to combine the responses coming from multi-detectors [Jordao et al., 2016].

## 1.2 Objectives

This work targets the problem of finding people in images through use distinct ways in different stages of the detection (see Figure 1.1). We can divide the objectives into three main parts, as follows. First, we intend to demonstrate the advantage of the PLS as alternative to build the oblique random forest. To this end, we employed another accurate classifier to produce the oblique random forest, the SVM. Second, we intend to evaluate the behavior of the filters approaches when employed on different detectors. To this analysis, we collect the main filters used in the pedestrian detection context. Third, we demonstrate that information coming from multiple detectors can improve the detection, increasing the confidence of true positives. To evidence that, we propose a novel late fusion method that enable such combination and we showed experimentally that our method is a more suitable choice to fuse detectors when compared with the weighted-NMS (a recent approach to combine detectors) [Jiang and Ma, 2015].

## 1.3 Contributions

Our first contribution is a novel alternative to generate the oRF to providing a smaller forest when compared with the traditional oRF-SVM [Jordao and Schwartz, 2016]. Our second contribution is a detailed study of a series of filtering approaches that provide a lower computational cost to the detection [Jordao et al., 2015]. Finally, our last contribution is a novel late fusion approach that enable to combine multi-detectors improving the detection [Jordao et al., 2016].

The publications achieved with this work are listed as follows.

1. Jordao, A., de Melo, V. H. C., and Schwartz, W. R. (2015). A study of filtering approaches for sliding window pedestrian detection. In Workshop em Visao

Computacional (WVC), pages 1-8.

2. Jordao, A., de Souza, J. S., and Schwartz, W. R. A Late Fusion Approach to Combine Multiple Pedestrian Detectors. In IEEE Transactions on Image Processing (ICPR).
3. Jordao, A. and Schwartz, W. R. Oblique random forest based on partial least squares applied to pedestrian detection. In IEEE International Conference on Image Processing (ICIP).

## 1.4 Work Organization

In Chapter 2, we review the main pedestrian detection techniques, features and approaches to decrease the computational cost as well as methods with focus on improve the detection results. Chapter 3 starts by describing the pipeline detection employed by pedestrian detectors. Afterwards, we introduce some concepts regarding the PLS. Then, we describe the oblique random forest and as use the PLS into oblique random forest. Next, we explain each filtering approach studied in this work. Finally, we describe the steps of our proposed late fusion method. In Chapter 4, we present the experiments executed to validate the oblique random forest based on PLS, the filtering approaches and the late fusion algorithm and discuss the results obtained. Finally, Chapter 5 provides the conclusions and directions to future works.

# Chapter 2

## Related Work

In this chapter, we present an overview regarding the main approaches employed in the pedestrian detection context. Initially, we discuss the main feature descriptors employed to describe human samples and background samples. Then, we review approaches used to reduce the computational cost to enable faster detection. Finally, we demonstrate techniques applied after the detection stage to improve the detection.

The detector based on the Histogram of Oriented Gradients (HOG) features proposed by Dalal and Triggs [2005] enabled impressive advances in several object recognition tasks, mainly on the pedestrian detection problem. On their initial work, Dalal and Triggs proposed to divide the detection windows in blocks of  $16 \times 16$  pixels with shift of  $8 \times 8$  pixels between blocks to compute the HOG features. Zhu et al. [2006] then showed that extracting HOG with different block sizes and strides, could lead to a more discriminative descriptor. Following the work of Zhu et al. [2006], Schwartz et al. [2009] employed similar block configurations to extract HOG features and with the addition of extra information provided by co-occurrence and color frequency features, the detector proposed by Schwartz et al. [2009] was able to reducing considerably the false positives. However, these features when combined yield a high dimensional feature space, rendering many traditional machine learning techniques intractable. To address that, the authors employed the partial least squares (PLS) to reduce the high dimensional feature space onto a low dimensional latent space before projecting itself to Quadratic Discriminant Analysis (QDA) performs the classification.

Similarly to Schwartz et al. [2009], several works showed that the combination of features creates a more powerful descriptor that improves the detection [Dollár et al., 2009; Marín et al., 2013]. A classical example of feature combination widely-used is the HOG with local binary pattern (LBP), HOG+LBP [Wang et al., 2009]. This merge has been shown efficient since HOG describes the shape information while the LBP capture

the texture of the object, both important clues to find people in images. Marín et al. [2013] employed this combination to describe human regions with high discriminative power, achieving a detector robust to partial occlusions. In contrast to Marín et al. [2013], Costea and Nedeveschi [2014] combined HOG+LBP and LUV color channels in a high level visual words named word channels allowing detection of pedestrians of different sizes on single scale image, which considerably reduces the computational cost.

Another feature combination that present good results to object detection are the Integral Channel Features (ICF) [Dollár et al., 2009]. Proposed by Dollár et al. [2009], the ICF features consists on ten channels of features: HOG (6 channels), LUV color channels (3 channels) and normalized gradient magnitude (1 channel). All these feature channels are extracted using the Integral Image trick, which render the feature extraction process extremely fast [Gerónimo and López, 2014]. Due to its simplicity and low computational cost, ICF features are the most predominant features explored in pedestrian detection, as illustrates Table 2.1. That table lists the main state-of-the-art pedestrian detectors on INRIA person dataset and synthesizes the essential features of each detector instead of discussing each one individually. An important aspect to be pointed out is that the Adaboost classifier is usually a preferential choice since its classification is very fast, mainly when combined with ICF features.

Adaboost classifier consists on an ensemble of classifiers that are combined to make prediction once test samples are presented. Generally, weak classifiers as decision stumps and orthogonal decision forest are chosen to compose the ensemble. However, some works [Criminisi and Shotton, 2013; Marín et al., 2013] have shown promising results when using strong classifiers (for instance SVM) on the ensemble. Inspired by these works [Criminisi and Shotton, 2013; Marín et al., 2013], we analyze, in the first part of our work (Section 3.1), the performance of the PLS as alternative to the SVM to creating ensemble members, focusing on oblique decision trees.

An alternative to enable a faster pedestrian detection independently of features and the classifier utilized are two main class: parallelization and use of GPUs [Masaki et al., 2010; Benenson et al., 2012b], and filtering regions of interest [Hou and Zhang, 2007; Silva et al., 2012; de Melo et al., 2014]. The latter is a simple but effective manner of speeding up the detection. In the next paragraph, we review the main filtering approaches applied to object recognition tasks.

Based on the observation that different images present similar log spectrum, Hou and Zhang [2007] proposed a filtering approach to remove the redundant information and preserve the non trivial parts of the scene. Their saliency detector aims at reducing the computational cost without knowing any prior information regarding the image.

To find objects in the image  $I$ , the authors applied a threshold,  $\tau$ , on the saliency map  $S(I)$ . This threshold was empirically estimated as  $\tau = 3E(S(I))$ , where  $E$  represents the average of intensity in the saliency map. Silva et al. [2012] proposed an extension of Hou and Zhang [2007] to pedestrian detection, where a saliency map was build for multiple scales. Different from of Hou and Zhang [2007], Silva et al. [2012] computed  $\tau$  based on a trade-off between false negative and selected regions. Following a different direction, de Melo et al. [2014] proposed a random filtering based on a uniform distribution. Their work demonstrated that selecting 14% of all candidate windows<sup>1</sup> is enough to cover around 83% of the people on the INRIA Person dataset. Moreover, the authors proposed a method named location regression where each window is displaced by  $\delta x$  and  $\delta y$  adjusting itself better on the pedestrian improving the detection. Also aiming to discard candidate windows, Singh et al. [2012] employed a filtering technique to remove regions of the images unlikely to contain objects. In their work, the energy gradient is utilized to discard regions of the image (named patches) with low energy (e.g sky patches). Even though effective, it is unclear which filters are more appropriate for a given detector since there are not studies evaluating this relationship. This motivated the second part of our work (Section 3.2), where we evaluate, compare and improve the filtering approaches described above.

Another line of research that has been explored in pedestrian detection is the use of high level information regarding the objects in the scene to improve detection. Since these approaches are used after the detection, we can call themselves of *post-processing* approaches. The high level information in *post-processing* approaches can be obtained by using the raw response of a single detector [Schwartz et al., 2011] or by combining distinct detectors [Li et al., 2010]. While Schwartz et al. [2011] proposed an approach to learn a classifier using the raw responses of a general pedestrian detector, Li et al. [2010] combined several pre-trained general object detectors, aiming at producing a powerful image representation. The authors noted that distinct detectors yielded complementary information achieving a better scene classification.

The combination of results obtained by multiple detectors has also been explored for pedestrian detection. Ouyang and Wang [2013] proposed a method to combine multiple detectors into a single detector to address the problem of groups of people. Their method learns the unique visual pattern of occluded regions using the responses of other detectors. In addition, Jiang and Ma [2015] combined multiple detectors via a weighted-NMS algorithm. In contrast to the traditional non-maximum suppression algorithms, the weighted-NMS does not simply discard the window with lower score

---

<sup>1</sup>It is usual to used the terms detection windows and candidate windows to denotes the regions of the image where will performed features extraction and classification.

(given the Jaccard coefficient), but it uses the score to weight the kept window.

The successful results of approaches such as in [Li et al., 2010; Schwartz et al., 2011; Ouyang and Wang, 2013; Jiang and Ma, 2015] rely on the hypothesis that regions containing a pedestrian have a strong concentration of high responses, different from false positive regions, where the responses have a large variance (low and high responses). Inspired by these observations, the last part of this work proposes a novel late fusion method, the *Spatial Consensus*, to capture additional information provided by a set of detectors of simpler and low computational cost manner, since it does not require the employment of machine learning techniques.

In the work proposed by Jiang and Ma [2015], the candidate windows without overlap are preserved, which might increase the miss rate. This occurs because it is expected that the false positives of distinct detectors reside in dissimilar regions at the scene. Therewith, it will not be overlapped and consequently will not be suppressed by the weighted-NMS, keeping the false positive of both the detectors, increasing the miss rate. On the other hand, in our Spatial Consensus approach, we remove windows without overlapping (windows that do not present spatial consensus when multiple detectors are considered), improving the detection since the likelihood of false positives provided by distinct detectors be isolated is high.

**Table 2.1.** Overview of state-of-the-art detectors on INRIA person dataset, sorted by log-average miss-rate. Training column: INRIA/Caltech model trained using INRIA and Caltech datasets; INRIA+ model trained using INRIA dataset with additional data.

Detector	Feat. Type	Classifier	Occlusion Handled	training
SpatialPolling [Paisitkriangkrai et al., 2014]	Multiple	pAUCBoost	-	INRIA/Caltech
S.Tokens [Lim et al., 2013]	ICF	AdaBoost	-	INRIA+
Roerei [Benenson et al., 2013]	ICF	AdaBoost	-	INRIA
Franken [Mathias et al., 2013]	ICF	AdaBoost	X	INRIA
LDCF [Nam et al., 2014]	ICF	AdaBoost	-	Caltech
I.Haar [Zhang et al., 2014]	ICF	AdaBoost	-	INRIA/Caltech
SCCPriors [Yang et al., 2015]	ICF	AdaBoost	X	INRIA/Caltech
NAMC [C. Toca and Patrascu, 2015]	ICF	AdaBoost	-	INRIA/Caltech
R.Forest [Marín et al., 2013]	HOG+LBP	D.Forest	X	INRIA/Caltech
W.Channels [Costea and Nedeveschi, 2014]	WordChannels	AdaBoost	-	INRIA/Caltech
V.Fast [Benenson et al., 2012a]	ICF	AdaBoost	-	INRIA

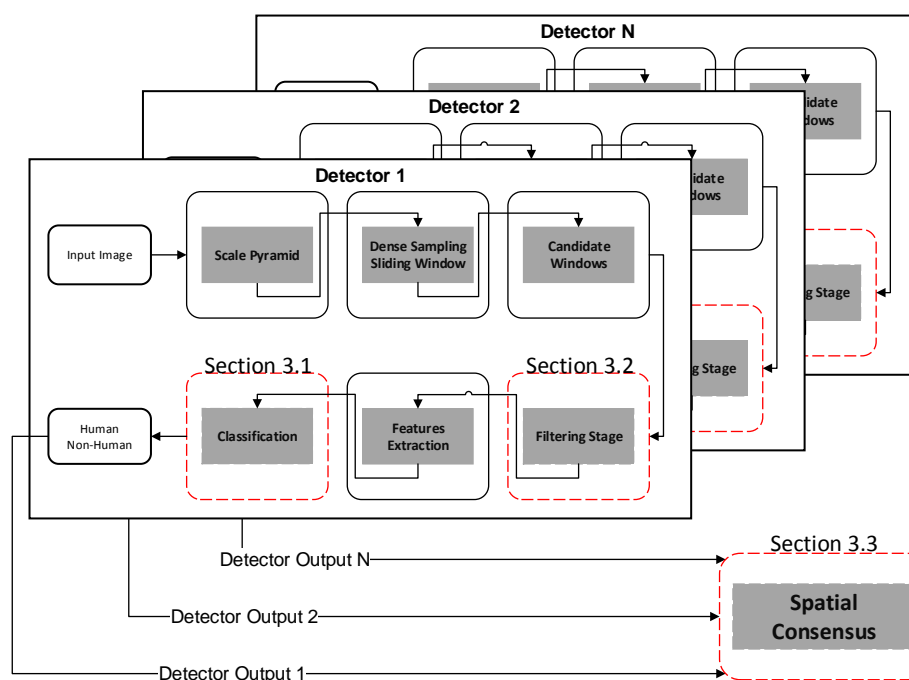




# Chapter 3

## Methodology

An overview of our methodology is summarized in Figure 3.1, where we show at which stage of pipeline the described method is operating. In Section 3.1, we introduce a



**Figure 3.1.** Pipeline detection and its respective section. Red dashed lines denotes where our work operates.

brief mathematical definition of the PLS, the main features of the oblique decision tree and how the oRF-PLS and oRF-SVM are built, respectively. Section 3.2 describes the steps performed by each filtering approach evaluated as well as its properties. Finally,

in Section 3.3, we present our proposed late fusion algorithm to combine multiple detectors.

## 3.1 Oblique Random Forest with Partial Least Squares

This section starts by giving a brief mathematical definition of the Partial Least Squares (PLS). Afterwards, we describe the features of the oblique random forest as well as its build process. Last, we describe how to employ the PLS and SVM with the oblique random forest and an adaptive bootstrapping procedure to improve the performance of the oblique random forest.

### 3.1.1 Partial Least Squares

The PLS is a technique widely employed to model the relationship between variables (features) utilized in several application areas [Rosipal and Krämer, 2006]. A brief definition of the PLS is shown below, detailed mathematical definitions can be found in Wold [1985] and Rosipal and Krämer [2006].

Let  $X \subset R^m$  be the matrix representing  $n$  data in  $m - dimensional$  space of features,  $y \subset R$  be the label class, in this work a  $1 - dimensional$  vector. The method decomposes  $X$  and  $y$  as

$$X = TP^T + E, \quad y = Uq^T + f, \quad (3.1)$$

where  $T$  and  $U$  are  $n \times p$  matrices of variables in latent space,  $p$  is a parameter of algorithm.  $P$  and  $q$  corresponds to matrix  $m \times p$  and vector  $1 \times p$  of loadings, in this order. The residuals are represented by  $E$  and  $f$  matrices of size  $n \times m$  and  $n \times 1$ , respectively. The PLS, constructs a matrix of weight  $W = \{w_1, w_2, \dots, w_p\}$ , where the  $i$ th column represents the maximum covariance ( $cov$ ) between the  $i$ th element of the matrix  $T$  and  $U$  as denotes the Equation 3.2. This procedure is made using the nonlinear iterative partial least squares (NIPALS) algorithm [Wold, 1985].

$$[cov(t_i, u_i)]^2 = \max_{w_i} [cov(Xw_i, y)]^2 \quad (3.2)$$

Besides dimensionality reduction, the PLS can be used for regression [Schwartz et al., 2011; de Melo et al., 2014], applying the matrix of weight  $W$  on the feature

vector,  $v_i$ . To this end, first we compute the regression coefficients,  $\beta_{m \times 1}$ ,

$$\beta = W(P^T W)^{-1} T^T y, \quad (3.3)$$

then the regression response to a features vector  $v_i$  is

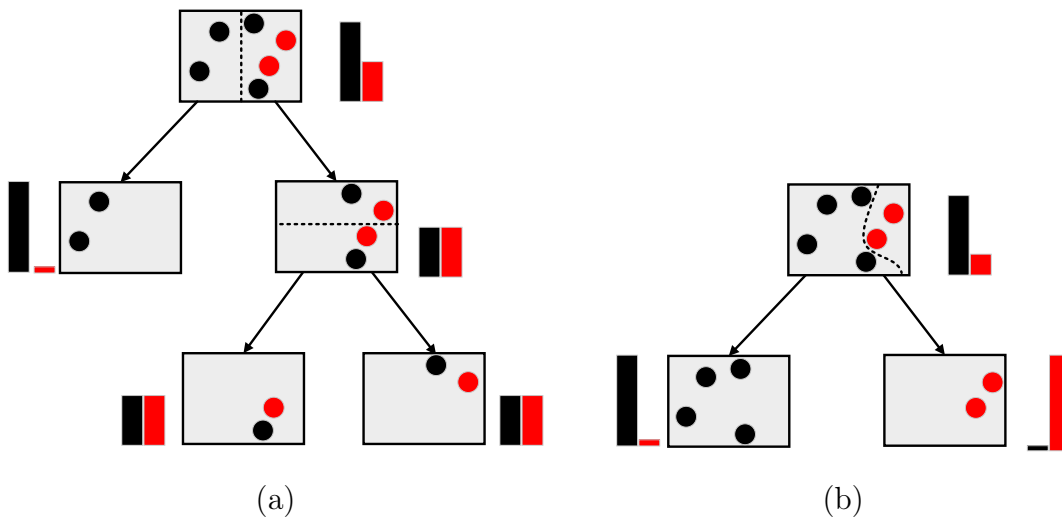
$$y_{v_i} = \bar{y} + \beta^T v_i, \quad (3.4)$$

where  $\bar{y}$  represents the average of  $y$ .

An important aspect of the PLS regarding the traditional dimensionality reduction techniques, e.g, principal component analysis (PCA) [Shlens, 2005], is that it considers the class label in the construction process of the matrix of weights  $W$ . Schwartz et al. [2009] showed that the PLS is able to separate data better than PCA, in the pedestrian detection context. In view of their results [Schwartz et al., 2009], we opt to utilize the PLS as dimensionality reduction technique as well as regression model.

### 3.1.2 Oblique Random Forest

Figure 3.2 illustrates the main advantage provided by oblique random forest (oRF). As can be observed, the samples are separated by oriented hyperplanes (Figure 3.2 (b)), achieving a better partition of the space that induces to shallower trees.



**Figure 3.2.** Decision tree split types (the bars represent the information gain).  
 (a) Orthogonal split, tree with depth 2. (b) Oblique split, tree with depth 1.

The steps performed to construct the decision trees composing the oRF are the following. First, we employ feature selection on the data received by the tree. As

noticed by Breiman [2001] and Criminisi and Shotton [2013], this technique ensures decorrelation (or diversity) between the trees, presenting an important contribution to improve the accuracy. In particular, the bagging mechanism also provides diversity on the random forest [Breiman, 2001]. However, as reported by Criminisi and Shotton [2013], several works are abandoning the use of such method. Therefore, in this work we discard the use of bagging since a considerable number of samples is required to build each oblique decision tree. Second, a starting node (root),  $R_j$ , is created with all data. The creation of a node estimates a decision boundary (hyperplane) to separate the presented samples according to their classes. Third, the data samples are projected onto the estimated hyperplane and a threshold  $\tau$  is applied on its projected values splitting the samples between in two children ( $R_{jr}$ ,  $R_{jl}$ ). The samples below this threshold are sent to the left child,  $R_{jl}$ , and samples equal or above to the threshold are sent to its right child,  $R_{jr}$ . This procedure is recursively repeated until the tree reaches a specified depth or another stopping criterion.

To estimate the threshold that better separates the data samples, we employ the *gini index* as quality measure. The *gini index* is computed in terms of

$$\Delta L(R_j, s) = L(R_j) - \frac{|R_{jls}|}{|R_j|} L(R_{jls}) - \frac{|R_{jrs}|}{|R_j|} L(R_{jrs}), \quad (3.5)$$

where

$$L(R_j) = \sum_{i=1}^K l_i^j (1 - l_i^j), \quad (3.6)$$

in which  $s \in S$  ( $S$  is a set of thresholds),  $K$  represents the class number and  $l_i^j$  is the label of class  $i$  at the node  $j$ . We choose *gini index* because it produces an extremely randomized forest [Criminisi and Shotton, 2013].

Once the trees have been learned, given a testing sample  $v$ , each node sends it either to the right or to the left child according to the threshold applied to the projected sample. For a tree, the probability of a sample to belong to class  $c$  is estimated combining the responses of the nodes in the path from the root to the leaf that it reaches at the end. The final response for a sample  $v$  presented to the forest is given by

$$l(c|v) = \frac{1}{F} \sum_{i=1}^F l_i(c|v), \quad (3.7)$$

where  $F$  is the number of trees composing the forest.

To build each node in an oblique decision tree associated with PLS, the samples  $P$  received by a node have its dimension reduced to a latent space *p-dimensional* using

PLS. The value to  $p$  is set by validation (see Section 4.2.2). Subsequently, the regression coefficients  $\beta$  are estimated using the Equation 3.4. Finally, the best threshold to split the training data samples, is obtained using the *gini index* on the regression values given by Equation 3.4.

The difference to build the oRF-SVM is that the received data samples do not have their dimensionality reduced and instead computing the regression coefficients, a linear SVM<sup>1</sup> is learned at each tree node. The remaining of the process is equal. This way, the approaches can be compared only in terms of better data separation and generalization.

### 3.1.3 Bootstrapping

The idea of bootstrapping consists in retraining an initial model  $F$ , using negative sample considered hard to classify (hard negatives samples). These hard negative samples are found according to a threshold applied on the prediction performed by  $F$  in a pool of negative examples  $S$ . The samples above of this threshold are introduced into a set  $N$ . It is important to mention that this set  $S$  are negative samples distinct of the negative examples used to generate the initial model  $F$ . Once model  $F$  classified

---

#### Algorithm 1: Bootstrapping

---

**input** : Samples to hard negative mining  $S$ , Iterations  $K$   
**output**: Forest  $F$

- 1 **for**  $iteration = 1$  *until*  $K$  **do**
- 2     Find hard negatives samples ( $N$ ) in  $S$ , using the current forest  $F$ .
- 3      $X = P \cup N$ , where  $P$  is the set of positive samples.
- 4     Train  $n$  new trees using  $X$ .
- 5     Add these new trees  $n$  into current forest  $F$ .
- 6      $iteration = iteration + 1$ .
- 7 **end**

---

all the negative examples in  $S$ , it is updated using the initial positives samples and the samples in set  $N$ . This procedure is repeated  $K$  times. Algorithm 1 is a variation of the bootstrapping proposed by Marín et al. [2013] focused on random forest and summarizes the steps above mentioned. Our experiments (Section 4.2.3) showed that, for each bootstrapping iteration, the log-average miss-rate decreases (lower is better). However, once four iterations are reached, the accuracy saturates.

---

<sup>1</sup>We are using linear SVM because it has been shown appropriate to pedestrian detection [Dalal and Triggs, 2005; Dollár et al., 2012; Benenson et al., 2014].

In particular, our bootstrapping procedure ensures diversity among the trees at the forest, since in each iteration different negative samples are utilized to produce  $n$  new trees, as we explained before.

## 3.2 Filtering Approaches

This section describes each filtering approach and its properties. The following filtering approaches are used in our study: the entropy filter, the magnitude filter, the random filtering and the saliency map based on spectral residual. A common feature among them is illustrated in Figure 3.3, in which all removed regions do not contain the object of interest (in our context, the pedestrian).



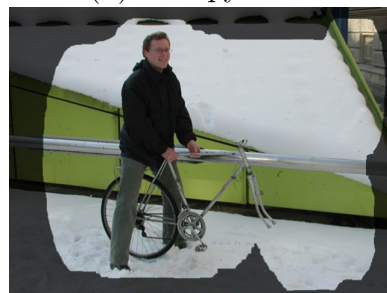
(a) Input image



(b) Entropy filter



(c) Magnitude filter



(d) Saliency map based on spectral residual

**Figure 3.3.** Translucent areas demonstrate regions eliminated by filtering stage for different filtering approaches. Some filters removed more regions than others, yet, all preserved the pedestrian (the random filtering, also considered in this work, was not showed since it is difficult to be visualized).

### 3.2.1 Entropy Filter

The main idea of this filter is to extract a histogram of gradient orientation for each detection window to reject those windows related with histograms presenting low entropy. For instance, homogeneous (flat) regions in the image present lower entropy due

to its more uniform distribution when compared with windows containing a human (rich on edges in a given orientation).

The computation process is the following. Initially, we compute the image derivatives  $I_x'$  and  $I_y'$ , regarding the  $x$  and  $y$ , using a  $3 \times 3$  Sobel mask [Gonzalez and Woods, 1992]. Then, we estimate the orientation ( $0^\circ$  to  $180^\circ$ ) for each pixel  $i$  using

$$\theta_i = \arctan \left( \frac{I_{y'}(x,y)}{I_{x'}(x,y)} \right). \quad (3.8)$$

Afterwards, we generate a histogram  $h$  incrementing its respective bin  $\theta_i$  by the magnitude of the pixel<sup>2</sup> (the number of bins was set experimentally to be nine). Finally, we normalize  $h$  using the *L1-norm* to become a probability distribution and estimate its entropy as

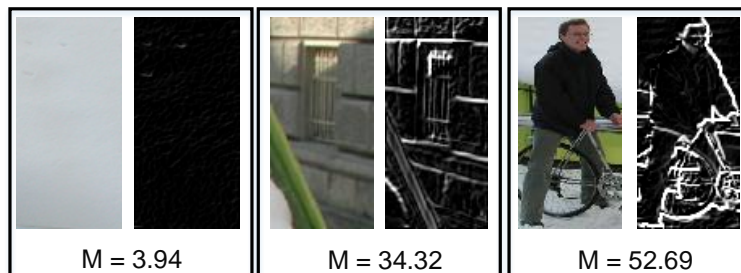
$$E(w) = - \sum_{i=0}^{\text{bins}} (a(h_i) \log(a(h_i))), \quad (3.9)$$

where  $a(h_i)$  denotes the value of the normalized bin  $i$  and  $E(w)$  is the entropy to detection window  $w$ .

### 3.2.2 Magnitude Filter

The average of the gradient magnitude within a detection window can be used as a cue for discriminating humans from the background. As illustrated in Figure 3.4, there is a gap between the magnitude values of regions containing background and regions containing humans. Therefore, we can utilize this interval to reject detection windows with background. This is a similar procedure that Singh et al. [2012] employed to discard image patches without relevant information.

<sup>2</sup>We accumulate the magnitude to soften the contribution of noisy pixels to  $h$ .



**Figure 3.4.** Different regions of the image (detection windows) captured by sliding windows approach and their respective magnitude images where  $M$  is the average gradient magnitude computed from each region using Equation 3.10.

To this filter, we initially compute the image derivatives as in the entropy filter. Then, we sum all values inside the detection window as

$$M(w) = \frac{1}{D} \sum_i \sum_j \left( \sqrt{I_x^2(i, j) + I_y^2(i, j)} \right), \quad (3.10)$$

where  $D$  is the window area.

This filter is relatively simple and our experiments demonstrate that a large number of windows can be discarded. Besides, it presents two important aspects: (1) when using integral images, the average magnitude can be computed using only four arithmetic operations yielding a faster filtering stage; (2) the magnitude is a feature widely used to create the image descriptors, such as the HOG, therewith detectors based on such descriptors do not have extra computational cost after this filtering stage.

### 3.2.3 Random Filtering

The random filtering technique consists in randomly selecting a sufficiently large amount  $\tilde{m}$  of windows from the set of detection windows  $W$ , which has cardinality  $m = |W|$  [de Melo et al., 2014]. The approach relies on the Maximum Search Problem theorem [Schölkopf and Smola, 2002] to ensure that every person will be covered. The theorem provides a set of tools that allows to estimate the required number of windows  $\tilde{m}$  to be selected.

The problem is formulated as follows. Let  $W = \{w_1, \dots, w_m\}$  be the set of  $m$  detection windows generated by the sliding window approach. In this problem, one needs to find a window  $\hat{w}_i$  that maximizes the criterion  $\mathcal{F}[w_i]$ , which evaluates whether a detection window covers a pedestrian or not. The problem is usually solved by evaluating each window  $w_i$  regarding such criteria, thus requiring  $m$  evaluations. However, such evaluations are expensive since the number of windows is large. The Maximum Search Problem states that one does not need to evaluate every window. By selecting a random subset  $\tilde{W} \subset W$  sufficiently large, it is very likely, that the maximum over  $\tilde{W}$  will approximate the maximum over  $W$  (with a confidence  $\eta$ ).

The size  $\tilde{m} = |\tilde{W}|$  of this random selection can be estimated by

$$\tilde{m} = \frac{\log(1 - \eta)}{\ln(n/m)}, \quad (3.11)$$

where  $\eta$  is the desired confidence,  $n$  denotes the number of elements in  $W$  which has  $\mathcal{F}[w_i]$  smaller than the maximum of  $\mathcal{F}[w_i]$  among the elements in  $\tilde{W}$ .

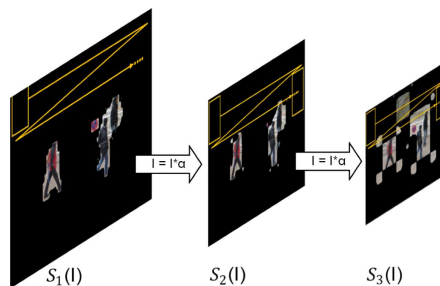


In their initial work, de Melo et al. [2014] proposed an extra stage, the location regression, where each selected windows is displaced by  $\delta X$  and  $\delta Y$  adjusting itself better on the pedestrian. Despite this procedure improve the detection performance, the  $\delta X$  and  $\delta Y$  values must be previously learned in a training step. Hence, since we are evaluating the random filtering only on the selection stage and the focus of our study is to evaluate simple filtering approaches, we disregard the location regression since it depends on previous learning.

### 3.2.4 Saliency Map based on Spectral residual

In their work, Hou and Zhang [2007] observed that images share the same behavior when viewed from the log spectral domain. Using this feature, the authors proposed a method to capture the saliency regions of the image removing redundant information and preserving the non-trivial regions in the scene. Following Silva et al. [2012], we apply the saliency map on multi-scales as this procedure outperforms the original method proposed in Hou and Zhang [2007]. Moreover, we demonstrate that the choice of the threshold used to discard regions of the image is essential to reject a large number of detection windows without compromising accuracy.

As mentioned in Chapter 2, the proposed threshold used to consider a detection window valid is based on the global mean of the saliency map. In this work, we propose two alternative thresholds: (1) the amount of the saliency pixels within a detection window is greater or equal to one, and (2) the sum of the saliency pixels within a detection window is greater than 10% of its dimension. In our experimental results, we show that the latter proposed thresholding allows to discard a larger number of candidate windows, without affecting the detection rate.



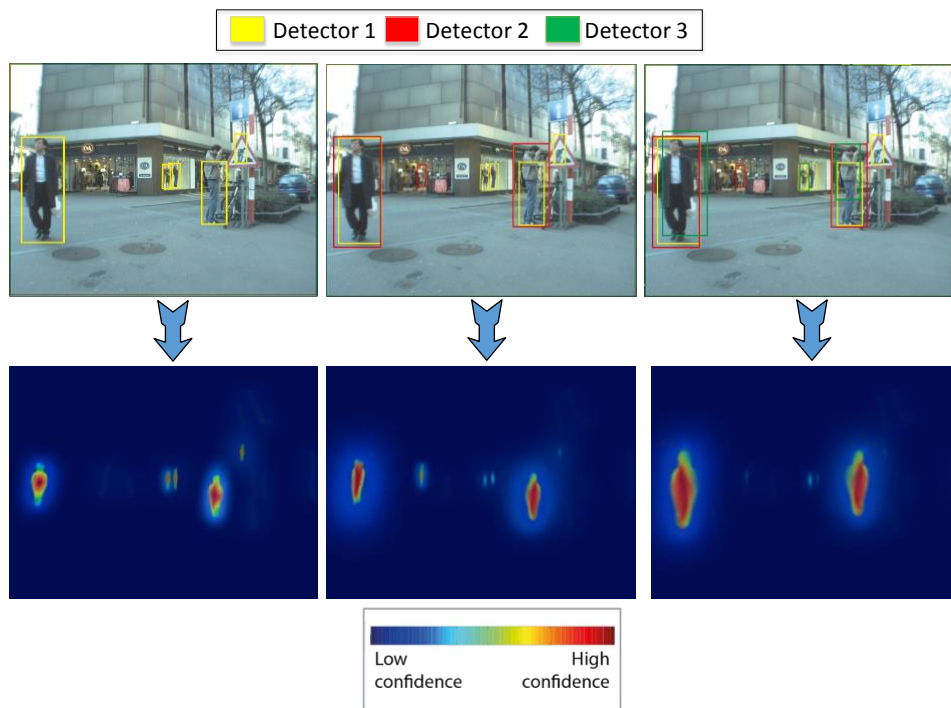
**Figure 3.5.** Sliding window approach on saliency map.

To this filtering stage, we apply the sliding window approach as following. First, we generate a saliency map  $S$  for the image  $I$  following the process proposed in Hou and Zhang [2007]. Afterwards, we scan  $S(I)$  via the sliding window technique. Then, the

image is downsampled by a scale factor and the process above is repeated, as illustrated in Figure 3.5. In other words, we can consider which each output image of the scale pyramid (see Figure 1.1) is a  $S(I)$ .

### 3.3 Spatial Consensus

This section describes the steps of our proposed algorithm to combine multiple detectors iteratively. Using the responses coming from these detectors, we weight their scores and give, giving more confidence to candidate windows that really belong to a pedestrian (our hypothesis is that regions containing pedestrians have a dense concentration of detection windows from multiple detectors converging to a spatial consensus) and eliminating a large amount of false positives, as illustrates Figure 3.6.



**Figure 3.6.** Detection results and their respective heat map. From the left to the right. First image only one detector is being used to generate the heat map, but in the second and third images two and three detectors, respectively, are used to generate the heat map. Each bounding box color represents the results of a distinct detector. As can be noticed, the addition of more detectors reduces the confidence of false positives with similar human body structure and reinforces the pedestrian hypothesis (best visualized in color).

The first issue to be solved when performing detector response combination (late

fusion) is to normalize the output scores to the same range because different classifiers usually produce responses in a different ranges. For instance, if the classifier used by the  $i$ th detector attributes a score of  $[-\infty, +\infty]$  to a given candidate window and the classifier of the  $j$ th detector attributes a score between  $[0, 1]$ , the scores cannot be combined directly. To address this problem, in this work we employ the same procedure used by Jiang and Ma [2015] to normalize the scores. The procedure steps are described as follows. First, we fix a set of recall points, e.g.,  $\{1, 0.9, 0.8, 0.7, \dots, 0\}$ . Then, for each detector, we collect the set of scores,  $\tau$ , that achieve these recall points. Finally, we estimate a function that projects  $\tau_j$  onto  $\tau_i$  (details in Section 4.4.1).

After normalizing the scores to the same range, we combine the candidate windows of different detectors as follows. Let  $det_{root}$  be the root detector from which the window scores will be weighted based on the detection windows of the remaining detectors in  $\{det_j\}_{j=1}^n$ . For each window  $w_r \in det_{root}$ , we search for windows  $w_j \in det_j$  that satisfies a specific overlap according to the *Jaccard coefficient* given by

$$J = \frac{\text{area}(w_r \cap w_j)}{\text{area}(w_r \cup w_j)}, \quad (3.12)$$

where  $w_r$  and  $w_j$  represent windows of  $det_{root}$  and  $det_j$ , respectively. Finally, we weight  $w_r$  in terms of

$$\text{score}(w_r) = \text{score}(w_r) + \text{score}(w_j) \times J. \quad (3.13)$$

The process described above is repeated  $n$  times, where  $n$  is the number of detectors besides the root detector. Algorithm 2 represents the aforementioned process.

Regarding the computational cost, the asymptotic complexity of our method is denoted by

$$O(cw_{root} \times \sum_{j=1}^n cw_j) = O(cw_{root} \times p) = O(cw^2),$$

where  $cw_{root}$  is the number of candidate windows of  $det_{root}$ ,  $cw_j$  denotes the number of detection windows of the  $j$ th detector and  $p$  is the amount of all candidate windows in  $\{det_j\}_{j=1}^n$ . Similarly, the approach proposed by Jiang and Ma [2015] (weighted-NMS method) presents complexity of  $O(cw \log cw + cw^2)$ . Although both methods present a quadratic complexity,  $p$  is extremely low because the non-maximum suppression is employed for each detector before presenting the candidate windows to Algorithm 2 (see Section 4.4.10), which renders the computational cost of both our Spatial Consensus method and the baseline approach [Jiang and Ma, 2015] to be negligible when compared with the execution time of the individual pedestrian detectors.

Our approach differs from the weighted-NMS method [Jiang and Ma, 2015] in two

---

**Algorithm 2:** Spatial Consensus

---

**input** : Candidate windows of  $det_{root}$  and  $\{det_j\}_{j=1}^n$   
**output:** Updated windows of  $det_{root}$

```

1 for  $j \leftarrow 1$  to  $n$  do
2   project  $det_j$  score to  $det_{root}$  score;
3   foreach  $w_r$  in  $det_{root}$  do
4     foreach  $w_j$  in  $det_j$  do
5       compute overlap using Equation 3.12;
6       if  $overlap \geq \sigma$  then
7         update  $w_r$  score using Equation 3.13;
8       end
9     end
10    if  $w_r$  does not presents any matching then
11      discard  $w_r$ ;
12    end
13  end
14 end

```

---

main aspects: (1) instead of inserting the candidate windows of  $det_{root}$  and  $det_j$  into a single set and performing weighted-NMS (see Section 4.4.3), we fix  $det_{root}$  and weight its windows using the  $det_j$  windows responses. In this way, we reduce the possibility of errors added by choosing a window that covers poorly the pedestrian according to the ground-truth, as illustrated in Figure 3.7 (a) - the suppression made by weighted-NMS algorithm, the chosen window will be the orange and thus we lose the pedestrian, generating one false positive and one false negative; (2) in the weighted-NMS [Jiang and Ma, 2015], windows without overlap will be kept, as illustrated the Figure 3.7 (b). On the other hand, our approach (step 11 in Algorithm 2) remove such a window even if it presents high confidence score. This is the key point that enables our approach to be powerful in eliminating hard false positives.

### 3.3.1 Removing the Dependency of the Root Detector

According to the algorithm described in the previous section, the execution of the SC algorithm requires the selection of a root detector. To address this restriction, we propose a generation of a “virtual” root detector, referred to as *virtual root detector*. The idea behind building this virtual root detector is to increase the flexibility of the algorithm – this way, we do not need to specify a particular pedestrian detector as input to the SC algorithm (see Algorithm 2).

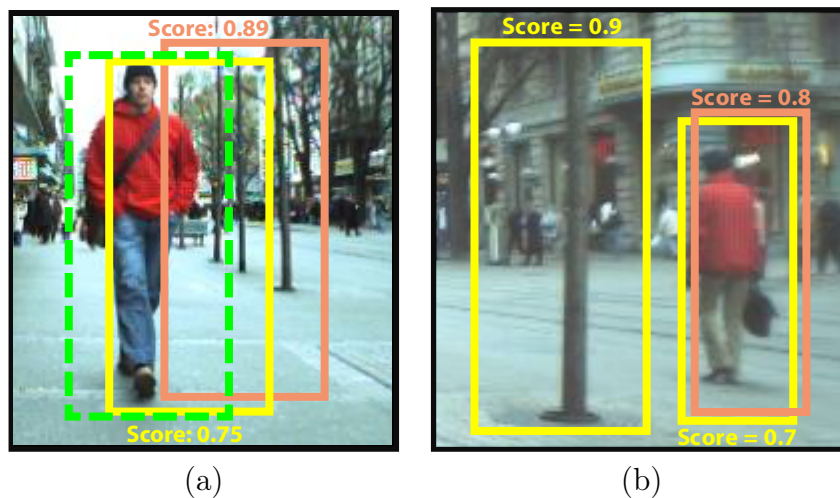
To generate windows for the virtual root detector ( $det_{vr}$ ), let us consider the

set of detectors  $\{det_j\}_{j=1}^n$ . For a detection window  $w_i^j \in det_j$  with dimensions  $(x, y, width, height)$ , we search for overlapping windows in the remaining detectors  $(w_i^l, l = 1, 2, \dots, k)$  to create a set of windows that will be used to generate a single window belonging to the  $det_{vr}$  using

$$w_i^{vr} = \frac{1}{k} \sum_{l=1}^k w_i^l, \quad (3.14)$$

where  $k$  is the number of overlapping windows to the window  $w_i^j$ . Finally, we assign a constant  $C$  (for instance,  $C = 1$ ) to this novel window. This constant contains the score of this window and its value will be updated after executing the SC algorithm.

Once the windows of the virtual root detector had been generated, we can execute the SC algorithm. However, the steps 10 to 11 of the algorithm are inoperative, since we will always have windows presenting overlapping.



**Figure 3.7.** Different aspects between our proposed Spatial Consensus algorithm and the weighted-NMS [Jiang and Ma, 2015]. Yellow and orange boxes indicate the detection coming from  $det_{root}$  and  $det_j$ , respectively, and the dashed green box shows the ground-truth annotation. (a) Our Spatial Consensus algorithm will maintain the yellow box (true positive), since this window belongs to  $det_{root}$ , while the weighted-NMS will maintain the orange box (false positive) because it is the window with higher score, leading to higher miss rate and reduced recall; (b) The SC algorithm will remove the false positive in yellow since it has no spatial support of other detectors, while the weighted-NMS will keep this false positive window due to its high score (best visualized in color).



# Chapter 4

## Experimental Results

We start this chapter describing the benchmarks employed through of our work. Then, we present the experiments, results and discussion regarding the oRFs, filtering approaches and spatial consensus, respectively.

The majority of the methods were implemented using the Smart Surveillance Framework (SSF) [Nazare et al., 2014], except to generate the saliency map (see Section 3.2.4), where we use its version that is available online<sup>1</sup>.

To measure the detection accuracy, we employed the standard protocol evaluation used by state-of-the-art called *reasonable set* (a detailed discussion regarding this protocol of evaluation can be found in [Dollár et al., 2009; Dollár et al., 2012]), in which only pedestrians with at least 50 pixels high and under partial or no occlusion are considered. The *reasonable set* measures the log-average miss rate of the area under the curve on the interval from  $10^{-2}$  to  $10^0$  (low values are better). However, in some experiments, we report the results using the interval from  $10^{-2}$  to  $10^{-1}$ . The area under curve in this interval represents a very low false positive rate (that is a requirement to real applications, e.g., surveillance, robotics and transit safety), this way, we evaluate the methods under a more rigorous detection. We used the code available in the toolbox<sup>2</sup> of the Caltech pedestrian benchmark to perform the evaluations.

### 4.1 Datasets

We compare our work with the state-of-the-art methods on three challenging widely-used pedestrian detection benchmarks: INRIA Person, ETH and Caltech. An extra dataset was used as validation set (TUD pedestrian) to calibrate the oRF parameters.

---

<sup>1</sup><http://www.saliencytoolbox.net/>

<sup>2</sup>[www.vision.caltech.edu/Image\\_Datasets/CaltechPedestrians/](http://www.vision.caltech.edu/Image_Datasets/CaltechPedestrians/)

However, we prefer not report the results of the other methods in this dataset since it is more utilized to pedestrian detectors that are part based [Andriluka et al., 2008].

Figure 4.1 illustrates the different scenarios of the datasets. As can be noticed, the datasets present high variability in the terms of illumination, human pose and background, rendering the pedestrian detection a challenging task.



**Figure 4.1.** Image examples from the datasets used in this work.

**INRIA Person dataset.** Providing rich annotations and high quality images, INRIA Person dataset still remains as the most employed dataset in pedestrian detection [Dalal and Triggs, 2005]. This dataset provides both positive and negative sets of images for training and testing, where there is a wide variation in illumination and weather conditions.

**ETH Pedestrian dataset.** Composed of images with size  $640 \times 480$  pixels, the ETH dataset provides stereo information. In this dataset are available four video sequences, one for training and three for test [Ess et al., 2007]. The large pose and people height variation make this dataset a challenging pedestrian detection dataset.

**Caltech Pedestrian detection.** Nowadays, this is the most predominant and challenging benchmark in pedestrian detection. Caltech dataset consists of urban environment images acquired from a moving vehicle [Dollár et al., 2012]. This dataset provides about 50,000 labeled pedestrians. Moreover, it has been largely utilized by methods designed to handle occlusions since such labels are available.

**TUD Pedestrians.** This dataset provides 250 images for test, all with dimension of  $640 \times 480$  pixels. Its training samples provide labeled human parts, hence, it is commonly used in part based approaches [Andriluka et al., 2008]. Since its images are



composed from people of side view, we are using this dataset as validation set (only to the experiments of the oRFs), aiming at measure the power of generalization of the models by considering that they were learned on side view samples.

## 4.2 Oblique Random Forest Evaluation

This section details the experimental setup utilized to validate our proposed oblique random forest as well as the comparison between our method with the baseline and the state-of-the-art.

### 4.2.1 Feature Extraction

We extract the HOG descriptor for each detection window following the configuration proposed by Dalal and Triggs [2005], with blocks of  $16 \times 16$  pixels and cells  $8 \times 8$  pixels. This configuration results in a descriptor of 3780 dimensions. We are using these 3780 features during the feature selection process (see Section 3.1.2), for both the oblique random forest to provide a comparison not influenced by the features.

### 4.2.2 Tree Parameters

To tune the parameters for both oRFs, we adopted the grid search technique where each parameter is placed as a dimension in a grid. Each cell in this grid represents a combination of the parameters.

In this experimental validation, we focus on the impact of two aspects in our forests: numbers of trees and number of features used in the feature selection stage. We are using the term  $nF$  to denote the number of features randomly selected to create a tree node (as explained in Section 3.1.2). To both oRFs, the maximum depth allowed at the growing stage of the tree is 5. In some preliminary experiments, we noticed that increasing the depth, the gain does not improve considerably. Therefore, we fixed this depth, which reduces considerably the search space in the grid search technique. On the validation dataset, the best parameters to oRF-SVM were using 200 trees and  $nF = 400$ , where it achieved a log-average miss rate of 41.67%. The oRF-PLS obtained the best results with 40 trees and  $nF = 550$ , presenting a log-average miss rate of 38.18%.

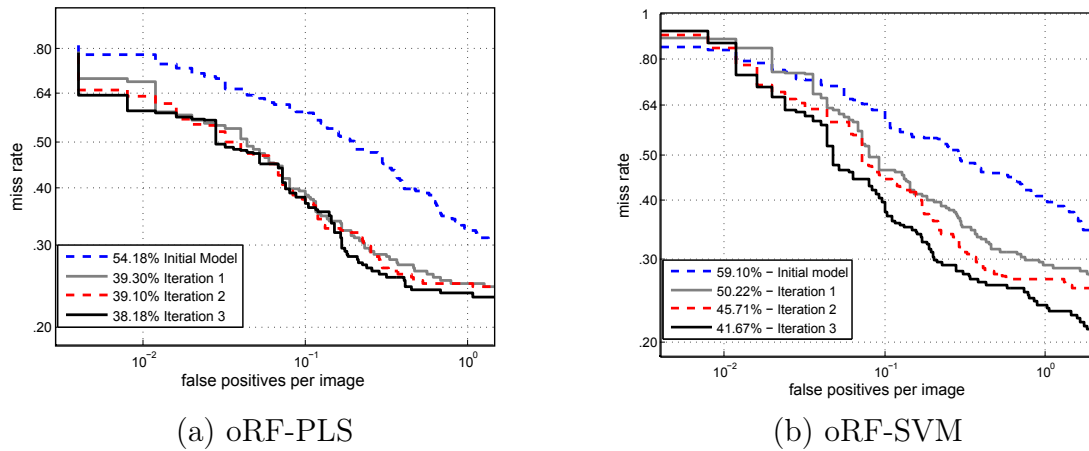
Differently from oRF-SVM, the oRF-PLS has an extra parameter to be tuned, the number of dimensions,  $p$ , required by PLS technique (see Section 3.1.1). By evaluating the accuracy in the validation set, the best value found to  $p$  was of 6. In our experiments, we noticed that varying  $p$  slightly the log-average miss rate increases sub-

stantially. For instance, modifying  $p$  from 6 to 8 the log-average miss rate goes from 38.18% to 42.98%. Therefore, this is a crucial parameter to oRF-PLS.

It is important to mention that the number of trees composing the forest is considering bootstrapping iterations (see Section 3.1.3).

### 4.2.3 Bootstrapping Contribution

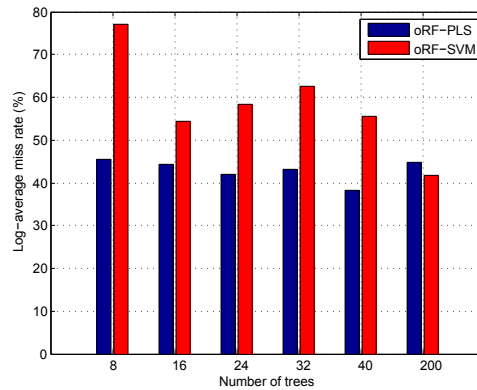
As can be noticed in Figure 4.2, the log-average miss rate presents a significant reduction to each bootstrapping iteration. From the initial model to the third iteration, the log-average miss rate decreases 16 percentage points (p.p) to oRF-PLS against 17.43 p.p. to oRF-SVM. This improvement is achieved since in each bootstrapping iteration, the forest finds more hard negative samples and these examples, when used to produce more trees, allow the current forest be more robust to false positives. In addition, for each bootstrapping iteration, the computational cost increases since the forest becomes larger, hence, more projections are performed.



**Figure 4.2.** Log-average miss rate achieved in each bootstrapping iteration using oRF-PLS and oRF-SVM, respectively, on validation set.

### 4.2.4 Influence of the Number of Trees

Figure 4.3 shows the log-average miss rate obtained by each approach on the validation set, as a function of the number of trees composing the forest. According to the results, with the same number the trees (except 200), the detection accuracy of oRF-PLS outperforms the oRF-SVM. Furthermore, to achieve competitive results, the oRF-SVM demands a larger number of trees, which renders the computational cost extremely



**Figure 4.3.** Log-average miss-rate (in percentage points) on the validation set as a function of the number of trees.

high (see Section 4.2.5). In addition, by computing the standard deviation of the log-average miss rate, we can notice that the oRF-SVM is more sensitive to variation of the number of trees to presenting a standard deviation of 10.58% while our proposed method presented a standard deviation of 2.42%. Thus, the use of PLS to build oRF is more adequate than use the SVM since it produces smaller and more accurate forests.

### 4.2.5 Time Issues

In this experiment, we show that the proposed oRF-PLS is faster than oRF-SVM. For this purpose, we performed a statistical test (visual test [Jai, 1991]) among the time (in seconds) to run the complete pipeline detection on an image of  $640 \times 480$  pixels. To each approach, we execute the pipeline 10 times and compute its confidence interval using 95% of confidence. The oRF-PLS obtained a confidence interval of [270.2, 272.44] against [382.72, 392.72] achieved by the oRF-SVM. As can be observed, the confidence intervals does not overlap, showing that the methods present statistical differences regarding the execution time, being the proposed method faster.

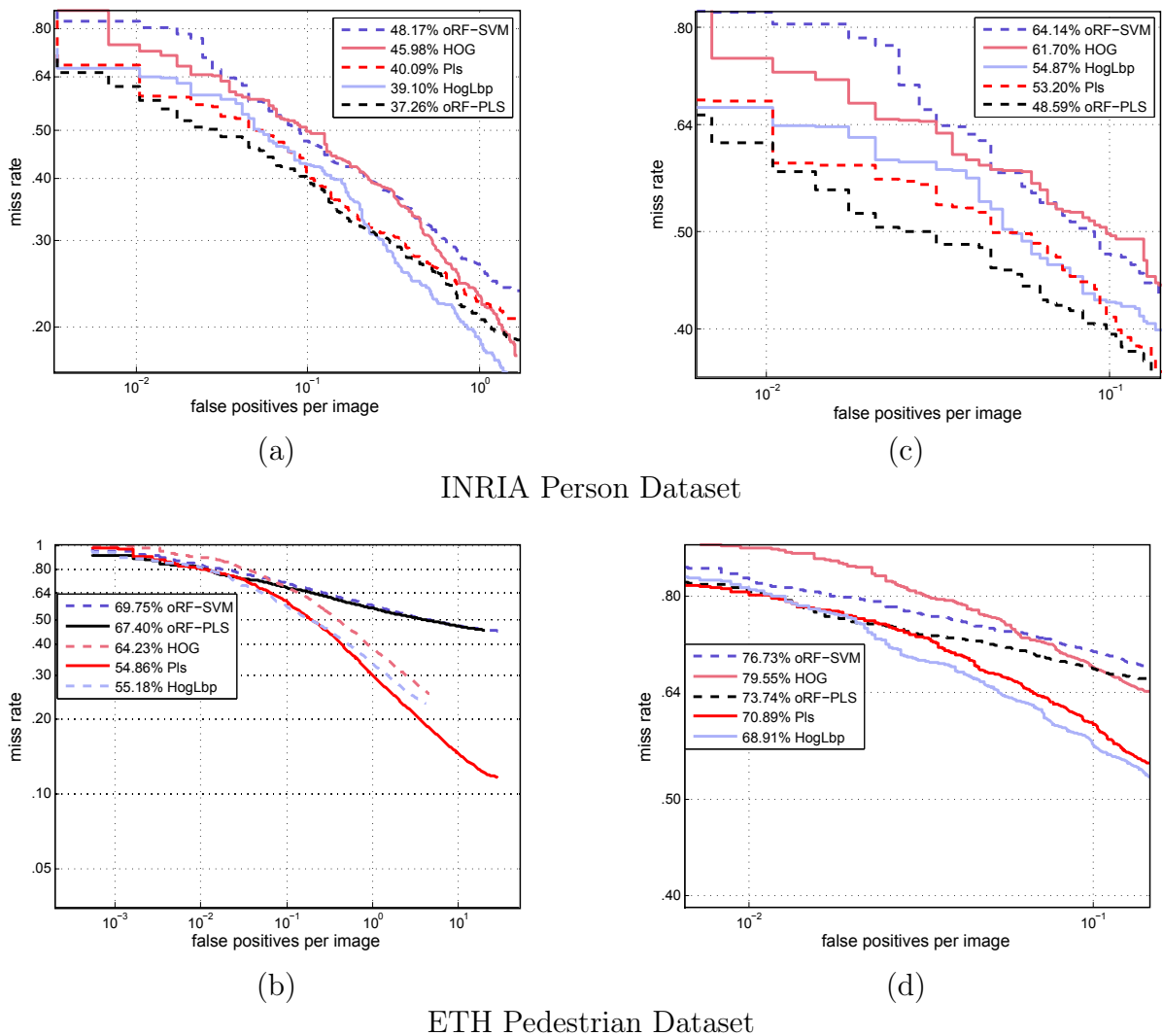
### 4.2.6 Comparison with Baselines

Our last experiment regarding the oblique random forest compares the proposed oRF-PLS with traditional state-of-the-art pedestrian detectors [Dollár et al., 2012; Benenson et al., 2014]. The first row in Figure 4.4 shows that our proposed method outperforms traditional classifiers used in pedestrian detection, e.g., linear SVM (HOG+SVM [Wang et al., 2009] and QDA (Pls [Schwartz et al., 2009])). Moreover, the oRF-PLS outper-

forms a robust partial occlusion method, HOG+LBP [Wang et al., 2009], in 1.84 and 6.28 percentage points to the area in  $10^{-2}$  to  $10^0$  and  $10^{-2}$  to  $10^{-1}$ , respectively.

When evaluated on the ETH pedestrian dataset, showed in the second row in Figure 4.4, the accuracy of our method decreases. However, its result still overcomes the oRF-SVM in 2.35 and 2.99 percentage points on the area in  $10^{-2}$  to  $10^0$  and  $10^{-2}$  to  $10^{-1}$ , respectively.

According to results showed in this section, the proposed oRF-PLS is able to obtain equivalent (or better) results when compared with traditional classifiers.



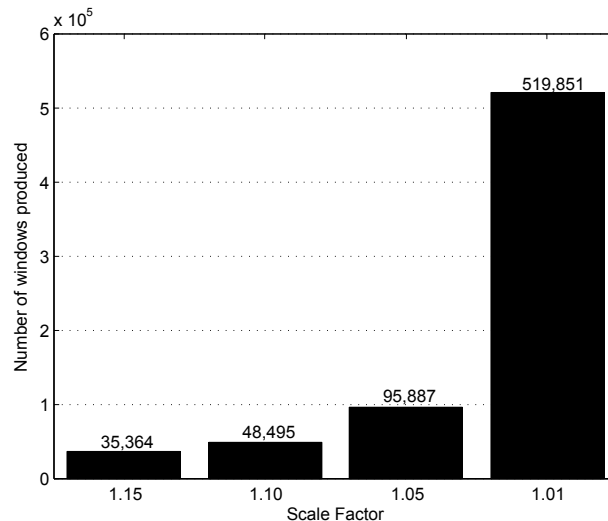
**Figure 4.4.** Comparison of our oRF-PLS approach with the state-of-the-art. The first column reports the results using the log-average miss-rate of  $10^{-2}$  to  $10^0$  (standard protocol). The second column report the results using the area of  $10^{-2}$  to  $10^{-1}$ .

## 4.3 Filtering Approaches

In this section, we evaluate several aspects of the filtering approaches and describe the experimental setup employed throughout of our analyze.

### 4.3.1 Scaling Factor Evaluation

Pedestrians can have different heights in an image due to their distance to the camera [Dollár et al., 2012]. Therefore, to ensure that all people have been covered by detection windows, a common technique is to employ a pyramid scale, keeping fixed the sliding window size. To generate this pyramid, we employ an iterative procedure that scales the image by a scale factor  $\alpha$ , in which the new image is generated by applying this scale factor to the previously generated image. In the first experiment, we



**Figure 4.5.** Tradeoff between scale factor and number of windows generated for a  $640 \times 480$  image.

evaluate the impact of the scaling factor on the number of detection windows generated, as well as the miss rate obtained by the detectors.

Figure 4.5 shows that the number of windows increases quickly depending on  $\alpha$ . For a  $640 \times 480$  image, while the sliding window algorithm generates 10 scales with  $\alpha = 1.15$ , this number increases to 171 scales when  $\alpha = 1.01$ . Table 4.1 presents the results achieved by the detectors at  $10^0$  false positive per image (FPPI), value commonly used to report the results in pedestrian detection [Dollár et al., 2012; Benenson et al., 2014].

The results indicate that denser samplings yield a lower miss rate and emphasizes the use of filtering approaches, which enables the usage of small values for  $\alpha$ , since large

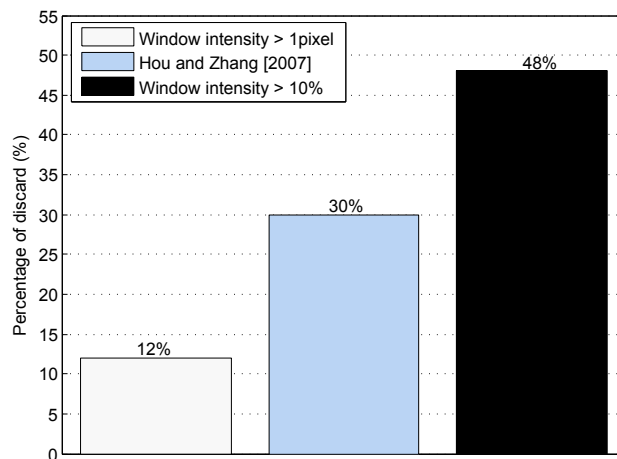
**Table 4.1.** Miss rate obtained at  $10^0$  FPPI with different scale factors.

Scale factor $\alpha$	HOG+SVM	PLS+QDA	oRF-PLS	oRF-SVM
1.15	0.34	0.33	0.28	0.26
1.10	0.33	0.29	0.28	0.24
1.05	0.31	0.29	0.27	0.23
1.01	0.29	0.27	0.25	0.22

part of the generated windows will be removed in the filtering stage and will not be presented to the classifier. However, throughout of the next experiments, we are using  $\alpha = 1.15$ , since it is a typical value used in pedestrian detection [Benenson et al., 2014] and, this way, our results can be compared directly with the original detectors.

### 4.3.2 Saliency Map Threshold

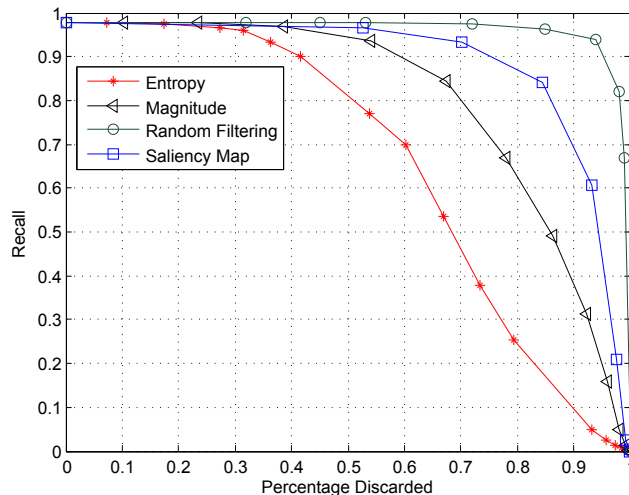
Our next experiment evaluates the power of the saliency map to discard candidate windows using different threshold approaches (as discussed in Section 3.2.4). According to the results showed in Figure 4.6, the proposed threshold approach is able to discard a larger number of detection windows, demonstrating to be more suitable than the threshold approach proposed in [Hou and Zhang, 2007]. It is important to mention that all the thresholds evaluated have been set to achieve the same recall to provide a fair comparison.

**Figure 4.6.** Threshold approaches analyzed to be used as rejection criteria in the saliency map.

### 4.3.3 Number of Discarded Windows

Figure 4.7 presents the percentage of rejected windows achieved by the filters assuming that an ideal detector<sup>3</sup> were to be used afterwards. In this experiment, we fixed  $\alpha$  as 1.15, which generated a total of 15,956,718 detection windows for all testing images of the INRIA person dataset. One may notice that some filters were able to reject more than 30%, while preserving the same recall rate as obtained without window rejection.

According to the results in Figure 4.7, the entropy filter was able to reject a small number of windows when compared to the other filters. Besides, this filter presented the largest increase of miss rate when a larger percentage of detection windows were discarded. The magnitude filter demonstrated to be effective to discriminate background windows from humans. It was able to reject up to 50% of the candidate windows conserving the recall rate above 90%. The random filtering and saliency map presented a powerful ability to reject candidate windows, discarding around 70% while keeping the recall rate above of 90%.



**Figure 4.7.** Relationship between rejection percentage and recall achieved by filters (assuming that an ideal detector was employed after the filtering stage).

### 4.3.4 Filtering Approaches Coupled with Detectors.

Our last experiment regarding the filtering approaches evaluates the distinct behavior of the filters when employed before different detectors. First, we defined ranges of rejection percentages (30 – 40, 41 – 50 and 51 – 60). We use these ranges to determine

<sup>3</sup>We consider an ideal detector, the one that if there were a detection window containing a person, it would classify that window as such (presenting a person). Therefore, it would achieve the maximum recall for any false positive per image rate.

**Table 4.2.** Miss rate at  $10^0$  FPPI applying the filtering stage on the detectors. Values between parentheses indicate the percentage of discarded detection windows.

	Entropy	Magnitude	R. Filtering	S. Map
HOG+SVM	0.39(31%)	0.34(38%)	0.36(32%)	0.33(38%)
	0.45(41%)	0.35(44%)	0.36(45%)	0.34(48%)
	0.58(53%)	0.38(54%)	0.37(53%)	0.34(52%)
PLS+QDA	0.36(31%)	0.33(38%)	0.33(32%)	0.30(38%)
	0.42(41%)	0.34(44%)	0.35(45%)	0.30(48%)
	0.56(53%)	0.36(54%)	0.35(53%)	0.31(52%)
oRF-PLS	0.31(31%)	0.28(38%)	0.30(32%)	0.26(38%)
	0.38(41%)	0.30(44%)	0.30(45%)	0.27(48%)
	0.51(53%)	0.31(54%)	0.30(53%)	0.28(52%)
oRF-SVM	0.30(31%)	0.27(38%)	0.28(32%)	0.25(38%)
	0.37(41%)	0.29(44%)	0.29(45%)	0.25(48%)
	0.51(53%)	0.31(54%)	0.28(53%)	0.25(52%)

the same rejection ratio among the filters, since we are only interested in analyzing the relationship between filter and detector. We reported the miss rate fixed at  $10^0$  FPPI. The results are reported in Table 4.2. In the evaluation of the number of discarded windows, the random filtering outperformed all approaches. However, in this experiment, the detectors performed poorly when evaluating the windows selected by this approach. This happens due to its random essence, since windows that fit the pedestrian might be slightly misplaced from the pedestrian’s center. Hence, as holistic detectors are trained considering a centered window, the classifier assigns a low score to that sample, even though it satisfies the evaluation protocol.

The results obtained indicate that for a fixed recall (Figure 4.7), each filter is able to reject a percentage distinct of candidate windows, being the saliency map the most efficient since it is able to discard a large number of candidate windows and reduce the miss rate. Moreover, when more windows are discarded, the detectors are effected differently according to filter being applied.

## 4.4 Spatial Consensus

This section starts by describing the steps required to execute the spatial consensus algorithm, the parameters that affect its performance and the baseline utilized to compare our proposed method, respectively. Finally, we compare our method with the state-of-the-art and present its limitations.



### 4.4.1 Preparing the Input Detectors

First, we need to define  $det_{root}$  and a set of detectors  $\{det_j\}_{j=1}^n$  (as explained in Section 3.3). Due to the large number of pedestrian detectors currently available, there are many options to determine both  $det_{root}$  and  $\{det_j\}_{j=1}^n$  [Benenson et al., 2014; Dollár et al., 2012]. In this work, we define these detectors as the top eleven best ranked pedestrian detectors on the INRIA person dataset (Table 2.1). The best ranked detector, the SpatialPolling [Paisitkriangkrai et al., 2014], was set to be the  $det_{root}$  and the remaining detectors were attributed to  $\{det_j\}_{j=1}^n$ . The columns of Tables 4.3, 4.4 and 4.5 show the detectors used in  $\{det_j\}_{j=1}^n$  in each dataset<sup>4</sup>. In Algorithm 2, each  $det_j$  is considered one after another, iteratively, to weight the  $det_{root}$ .

At the score calibration step, we use the INRIA person dataset to acquire the set of scores  $\tau$ . Next, to map the  $\{det_j\}_{j=1}^n$  score to  $det_{root}$  score, we consider a linear regression. From the scatter plot between  $\tau_{root} \times \tau_j$ , we observed that a linear regression is a suitable choice to perform this mapping.

<sup>4</sup>Some top ranked detectors of INRIA are not available to the ETH and Caltech Datasets.

**Table 4.3.** INRIA Person Detectors Accumulation. The initials SC refers to our proposed method and the initials W-NMS refers to our baseline the weighted-NMS [Jiang and Ma, 2015]. The results are measured in log-average miss-rate (lower is better).

$\sigma$		S.Tokens	Roerei	Franken	LDCF	I.Haar	SCCPriors	NAMC	R.Forest	W. Channels	V.Fast
0.5	SC (Ours)	10.78%	9.57%	9.44%	9.77%	9.54%	9.66%	9.92%	9.58%	9.10%	<b>9.08%</b>
	W-NMS	<b>10.60%</b>	11.22%	12.91%	12.42%	12.48%	12.37%	12.38%	14.70%	14.81%	14.11%
0.6	SC (Ours)	10.79%	8.85%	8.63%	9.21%	8.96%	9.30%	9.54%	9.17%	<b>8.45%</b>	9.75%
	W-NMS	10.12%	<b>9.74%</b>	11.75%	11.84%	12.72%	13.48%	13.65%	16.11%	14.81%	14.11%
0.7	SC (Ours)	14.98%	11.11%	10.56%	10.73%	10.61%	10.77%	10.94%	10.20%	<b>9.60%</b>	9.97%
	W-NMS	11.37%	<b>10.01%</b>	13.58%	14.81%	15.59%	16.14%	17.79%	24.88%	14.81%	14.11%

**Table 4.4.** ETH Detectors Accumulation. Columns with "–" express detection results not available on the ETH dataset. The initials SC refers to our proposed method and the initials W-NMS refers to our baseline the weighted-NMS [Jiang and Ma, 2015]. The results are measured in log-average miss-rate (lower is better).

	S.Tokens	Roerei	Franken	LDCF	I.Haar	SCCPriors	NAMC	R.Forest	W. Channels	V.Fast
SC (Ours)	–	35.63%	34.60%	<b>33.61%</b>	–	–	–	34.15%	–	33.98%
W-NMS	–	<b>35.19%</b>	39.69%	40.93%	–	–	–	48.75%	–	49.31%

**Table 4.5.** Caltech Detectors Accumulation. Columns with "–" express detection results not available on the Caltech dataset. The initials SC refers to our proposed method and the initials W-NMS refers to our baseline the weighted-NMS [Jiang and Ma, 2015]. The results are measured in log-average miss-rate (lower is better).

	S.Tokens	Roerei	Franken	LDCF	I.Haar	SCCPriors	NAMC	R.Forest	W. Channels	V.Fast
SC (Ours)	–	36.90%	37.90%	27.86%	27.10%	24.73%	24.60%	23.78%	<b>23.67%</b>	–
W-NMS	–	40.58%	<b>40.54%</b>	48.75%	49.64%	45.94%	44.33%	44.13%	44.68%	–

Due to difficulty to obtain the exact results due to parameter setup, we preferred not to implement some detectors<sup>5</sup>. Therefore, throughout of the experiments we are using the results provided by authors and this fact forced us to use INRIA test to calibrate the scores only for one of ours experiments (to produce Table 4.3). However, once the scores are calibrated, we use the estimated regression on the other datasets. To the domain knowledge experiment (see Section 4.4.7), we utilize a random video subsequence available in the ETH and Caltech to calibrate the scores.

It is important to mention that before combining the detectors by Algorithm 2 or by the weighted-nms algorithm Jiang and Ma [2015], we assume that all detectors performed non-maximum suppression (NMS) individually. This initial NMS is performed to suppress overlapping detections from the same detector and that is essential to reduce the number of candidate windows since it will influence the running time of both algorithms.

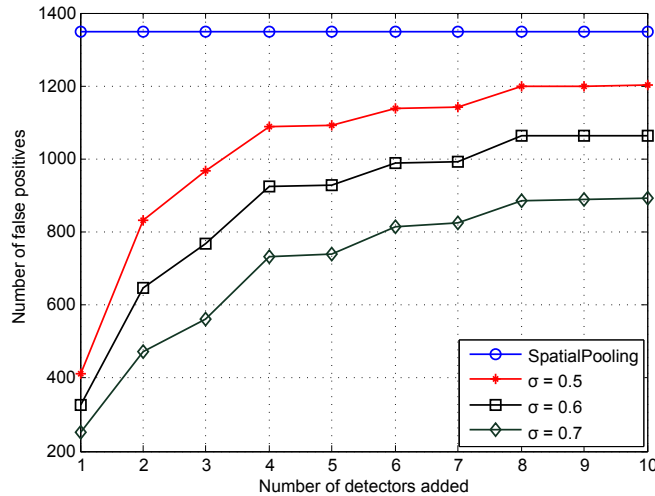
#### 4.4.2 Jaccard Coefficient Influence

The idea behind this experiment is to evaluate the influence of the threshold,  $\sigma$ , applied to the *Jaccard coefficient* (step 6 of Algorithm 2). Increasing this threshold not only forces that only windows better aligned with  $det_{root}$  contribute to the weight of its windows, but also discards a larger number of windows since the likelihood of having a match with windows in other detectors of  $\{det_j\}_{j=1}^n$  reduces. In this experiment, we consider  $\sigma$  equals to 0.5, 0.6 and 0.7, as reported in Table 4.3. Furthermore, we evaluate the influence of  $\sigma$  on the number of false positives, as shown in Figure 4.8.

According to the results, a more restrict (larger)  $\sigma$  reduces the number of false positives since more candidate windows are removed. However, more true positives are also discarded and the detection accuracy decreases slightly (see Table 4.3). According to Figure 4.8, the  $det_{root}$  by itself obtained 1350 false positives and when ten detectors were added with our algorithm, this value decreased to 1065 and 894 using  $\sigma = 0.6$  and  $\sigma = 0.7$ , respectively. Nonetheless, according to Table 4.3, the log-average miss rate increased from 9.75% to 9.97%. As shown in Table 4.3, the best results (aiming at preserving the lowest miss rate) were obtained using  $\sigma = 0.6$  for both methods. Thus we consider this value for the experiments on ETH and Caltech datasets.

---

<sup>5</sup>This consideration also implicates because some results are not available for the ETH and Caltech dataset (see Table 4.4 and 4.5 for details).



**Figure 4.8.** Number of false positives as a function of the number of detectors added and the threshold  $\sigma$  (results on INRIA Person dataset). The number of false positives obtained by the  $det_{root}$  alone (without our method) is shown as line because it is constant.

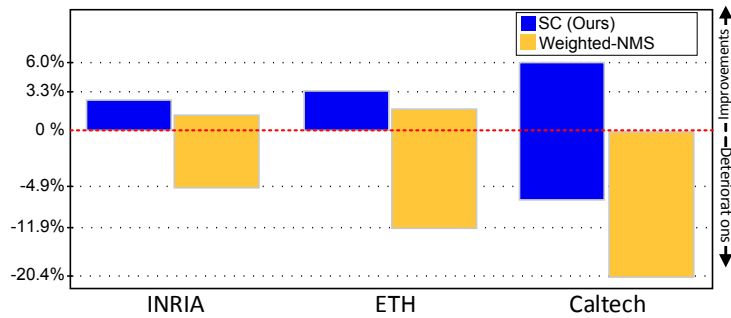
### 4.4.3 Weighted-NMS Baseline

The method proposed by Jiang and Ma [Jiang and Ma, 2015] can be described in four main steps. First, the detection window responses of the combined detectors must be normalized to the same score range. Then, the windows of both detectors are inserted in a set  $U$ . Afterwards,  $U$  is sorted in descending order of the scores. Finally, when a window at position  $i$  of  $U$  presents overlap higher than the threshold ( $\sigma$ ) with a window at position  $j$  of  $U$  ( $j > i + 1$ ), the NMS process is applied and the window with the lower score is discarded and its score contributes to the kept window (in the same way that is performed in the step 7 of Algorithm 2).

The method proposed in Jiang and Ma [2015] accepts only two detectors as input. Therefore, to enable multiple detectors, we insert all windows of the detectors that we want to combine into  $U$ . To enable a fair comparison, we consider that the  $det_{root}$  windows always is in  $U$ .

### 4.4.4 Spatial Consensus vs. weighted-NMS

In our experiments, we evaluate the performance of adding multiples detectors to extract the spatial consensus to improve the pedestrian detection. According to results shown in Tables 4.3, 4.4 and 4.5, the weighted-NMS was better than our method when adding only one detector to improve  $det_{root}$ . This occurs because the probability of detection windows without overlapping is higher when few detectors are considered,



**Figure 4.9.** Comparison between our proposed method with the baseline in terms of improvement and depreciation (according to  $det_{root}$ ) of the log-average miss rate. Values above of the red dashed line denote improvement whereas values below show deterioration.

thereby we may discard true positives windows (Step 11 of Algorithm 2), decreasing the accuracy. However, when more detectors are added, our approach outperformed considerably the weighted-NMS, achieving lower miss rates, as shown in the tables.

The weighted-NMS achieved the best results on the INRIA dataset when two detectors were added, Sketch Tokens [Lim et al., 2013] and Roerei [Benenson et al., 2013], outperforming the state-of-the-art by 1.48 p.p. On the other hand, the best result of our approach is achieved adding nine detectors, improving the state-of-the-art in 2.77 p.p. (8.45%). On the ETH dataset, the weighted-NMS method achieved its best result, 35.19%, by combining Roerei and Franken detectors. However, this combination was not enough to outperform the TA-CNN [Tian et al., 2015b] (current state-of-the-art on this dataset with 34.98%). On the contrary, our approach reached best results adding, beyond these two detectors, the LDCF [Nam et al., 2014] detector, where we overcome the state-of-the-art in 1.37 p.p. (33.98%).

The best result of the weighted-NMS on the Caltech dataset was achieved combining the Roerei and Franken detectors. However, this combination increased the  $det_{root}$  miss rate from 29.24% to 40.54%. On the other hand, we achieved best results adding eight detectors and decreasing the  $det_{root}$  miss rate from 29.24% to 23.16%. In addition, the employment of our approach reduces the difference to most recent state-of-the-art detector (CompACT-Deep [Cai et al., 2015] - 12.43%) from 16.81 to 10.73 p.p.

A summary of the comparison between the proposed method and the weighted-NMS is shown in Figure 4.9. This figure synthesizes the best improvements and the smallest deterioration of the miss rate for both methods in each dataset. We conclude that our method is more suitable to perform fusion between multiple detectors than the weighted-NMS [Jiang and Ma, 2015].

### 4.4.5 Influence of a Less Accurate Detector

To evaluate the robustness of our method to the addition of a detector with high false positive rate, we introduced the HOG detector right after the V. Fast [Benenson et al., 2012a] on the INRIA person dataset. When it was inserted into  $\{det_j\}_{j=1}^n$ , the miss rate achieved by our method was of 8.90% against 16.78% of the weighted-NMS algorithm, demonstrating the robustness our method to the addition of less accurate detectors.

### 4.4.6 Comparison with the State-of-the-Art

In this experiment, we compare the results of the proposed *Spatial Consensus* algorithm with state-of-the-art methods, where we utilized the results provided by the authors in their works, aiming at a fair comparison.

Figures 4.10(a) and 4.10(b) show that our algorithm outperforms the state-of-the-art on the INRIA and ETH datasets achieving log-average miss-rate (low values are better) of 8.45% and 33.61%, respectively in these datasets. In addition, Figure 4.10(c) shows that our method achieves significant results on the Caltech dataset, improving in 6.08 p.p. the  $det_{root}$  used (SpatialPooling).

An important goal of pedestrian detection is to significantly minimize false alarms for applications such as video surveillance in which they may cause damage to environment as well to humans [Angelova et al., 2015]. To indicate that our method is suitable for the requirement of very low false positive rates, we report our results using the area under curve from  $10^{-2}$  to  $10^{-1}$  (values where the false positive rates are extremely small). Figures 4.10(d), (e), and (f) show these results. As can be noticed, our method further enhances the detection accuracy, demonstrating to be appropriate to applications that need to operate at very low false positive rates.

### 4.4.7 Domain Knowledge

This experiment evaluates the impact of using domain knowledge regarding the dataset to assign the detectors to  $det_{root}$  and to  $\{det_j\}_{j=1}^n$ , i.e., instead of following the ordering based on the INRIA dataset (as discussed in Section 4.4.1), we attribute the top ranked detector to  $det_{root}$  and the remaining ten best ranked detectors to  $\{det_j\}_{j=1}^n$ , according to results achieved on that particular dataset. We call this procedure *Spatial Consensus + Domain Knowledge* (SC+DK).

Given the definition of the SC+DK, we will now describe the detailed configuration where we achieved the best results on the ETH and Caltech dataset, respectively. To the former dataset, we specified the TA-CNN [Tian et al., 2015b] detector as the

$det_{root}$  and the  $\{det_j\}_{j=1}^n$  was composed of the SpatialPooling and Franken [Mathias et al., 2013] detectors. To the latter, the  $det_{root}$  was the CompACF-Deep detector [Cai et al., 2015] and the  $\{det_j\}_{j=1}^n$  was composed of the DeepParts [Tian et al., 2015a] and CheckerBoards+ [Zhang et al., 2015] detectors. It is worth mentioning that due to lack of some results in the detection on the INRIA person dataset, in this experiment, we utilized the own dataset to perform the score calibration stage (see Section 4.4.1).

According to the results shown in Figure 4.10(b) and 4.10(c), the use of this extra knowledge, allowed our method to outperform all previously published state-of-the-art methods in 7.66 and 0.32 p.p. on the ETH and Caltech datasets, respectively. Such improvements are even more emphasized when considering the log-average miss-rate from  $10^{-2}$  to  $10^{-1}$ , as shown in Figure 4.10(e) and 4.10(f), where we outperformed the state-of-the-art in 11.15 and 2.84 p.p. on the ETH and Caltech datasets, respectively.

#### 4.4.8 Virtual Root Detector

Our last experiment evaluates the proposed approach to remove the requirement of specify a root detector. Different from techniques that we presented so far, which use the best pedestrian detector as root detector, in the virtual root detector approach, referred to as SC+VR, we utilize it only to calibrate the scores (see Section 3.3).

Regarding the results presented in Figure 4.10, we can notice that the virtual root detector outperforms the conventional approach, where an initial root detector must be defined, in 0.5 p.p and 3.32 p.p. to INRIA and Caltech, respectively. To the ETH dataset the log-average miss rate increased 0.13, in relation to conventional approach.

According to these results, we conclude that the virtual root detector enables the SC algorithm has more flexibility, without compromising the accuracy.

#### 4.4.9 Limitations of the Method

A limitation of our proposed method is that it was not necessarily able to improve the  $det_{root}$  according to every single  $det_j$  introduced. For instance, in the Caltech dataset, our method does not outperform the SCCPriors [Yang et al., 2015] detector which was one of the detectors added to weight the  $det_{root}$  windows. This occurs because  $det_j$  can cover pedestrians that  $det_{root}$  does not cover and as only  $det_{root}$  windows are considered (Step 3 of Algorithm 2), even  $det_j$  covering more pedestrians, it cannot help  $det_{root}$  in this issue. It is important to mention that this limitation is only to the conventional SC approach.

Another question that affects both weighted-NMS and our method, is the high variability of results for a particular detector in different datasets. For instance, according to Figure 4.10(a) and Figure 4.10(b), the Roerei detector [Benenson et al., 2013] is one of the best detectors on the INRIA and ETH datasets but its accuracy drops considerably in the Caltech dataset, as can be seen in Figure 4.10(c). This behavior might interfere in our algorithm. For instance, according to Tables 4.3 and 4.4, when introducing the Roerei to weight the  $det_{root}$ , the miss rate decreases, but for the Caltech dataset, the miss rate increases (Table 4.5). This issue led us to use the ordering criterion (as discuss previously), since which we do not know whether determined detector will have the same behavior on other datasets [Dollár et al., 2012; Benenson et al., 2014]. Besides, using this ordination renders the SC more general whereas this ordering can be fixed only once and utilized over other datasets.

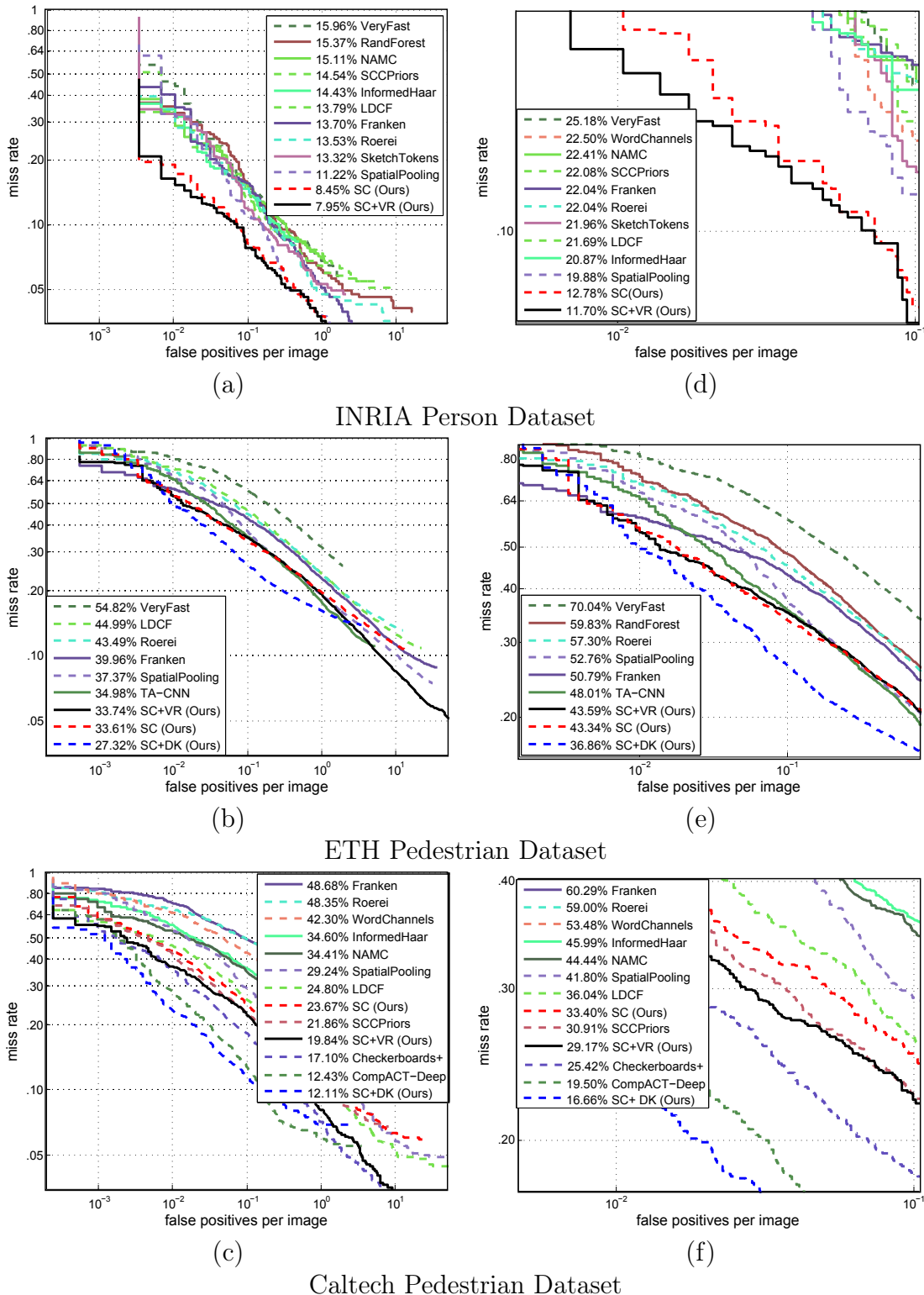
#### 4.4.10 Time Issues

As described in Section 3.3, the complexity of our method is equal to the weighted-NMS. Although presenting a quadratic complexity, both methods run in real time since the traditional NMS is performed for each individual detector before starting the algorithms (see Section 4.4.1). Besides, the values of  $p_{root}$  and  $p$  are corresponding to the number of pedestrians at the scene, which is low, in general. To verify that these values are extremely small, we collected the average of people per image in the INRIA person and the ETH (*seq#2*) datasets. The values are 3.3 and 43.6, respectively (not large enough to impact the computational time of our algorithm).

Since the values of  $p$  are small, our approach is able to run in real time. To show that, we computed the time average to execute of the SC on a  $640 \times 480$  image, using 10 detectors to compose  $\{det_j\}_{j=1}^n$  and without any parallelization technique. The SC runs in 2.17 milliseconds on average, by executing this experiment 10 times. Additionally, the most recent survey of computation cost at the detection pedestrian [Angelova et al., 2015] showed that the faster detector presenting high accuracy is able to process 15 frames per second<sup>6</sup>, running on an NVIDIA K20 Tesla GPU [Angelova et al., 2015]. Based on these results, we conclude that our method is capable of improving the detection results and could be fast to execute, even though our algorithm requires results of individual detectors.

---

<sup>6</sup>Results provided by author.



**Figure 4.10.** Comparison of our proposed approach with the state-of-the-art. The first column reports the results using the log-average miss-rate of  $10^{-2}$  to  $10^0$  (standard protocol). The second column reports the results using the area of  $10^{-2}$  to  $10^{-1}$ .



# Chapter 5

## Conclusions

This work faces the problem of finding pedestrian in images. Throughout this work, different methods are proposed and analyzed to address three main challenges listed below.

The first one, it is to distinguish humans from background features. In this step, an accurate classifier is required to separate correctly the examples. Given this requirement, we propose a novel oblique random forest associated with PLS. The method consists on utilize the PLS to find a decision surface at each node in a decision tree. We compare the proposed method with the oblique random forest based on SVM. Our experimental results demonstrated that a smaller forest is generated when using the PLS instead SVM, which is ideal to such type of random forest since each decision tree (that composes the forest) presents high computational cost. Besides, our method achieved comparable state-of-the-art results, when compared with traditional classifiers employed in the pedestrian detection.

The second one, it is associated with the computational cost required to provide a faster detection. Our experiments showed that a denser sampling induces to a better detection. However, the computational cost increase proportionally. Aiming to around this problem, we analyze several filtering approaches to quickly discard parts of the image without losing relevant information to the pedestrian detection task. Our experiments allowed us to perform a quantitative analysis on the number of detection windows rejected by the filtering stage. Furthermore, we demonstrated that each detector has different behavior (miss rate) according to filter applied.

The last one, focuses on improving the detection using the high-level information regarding the scene. To this end, we propose a novel approach to combine results of distinct detectors. The method bases itself in using the responses coming from multiple detectors to reinforce more consistent human hypothesis whereas reducing and

discarding false positives. The proposed method outperforms the state-of-the-art in two pedestrian detection benchmarks and achieves comparable results on the challenging Caltech dataset. Furthermore, we demonstrated that with previous knowledge of the domain, our method outperforms the most powerful detectors in each dataset.

## 5.1 Future Works

On the results showed in this work, we conclude that the most promising improvement in the detection can be attributed to combination of detectors. This combination outperforms the most powerful feature utilized to describe human samples, in terms of differentiate it of hard false positives. Under this circumstance, as future work we intend to explore others way to combine the detection coming from different detectors.

# Bibliography

- (1991). Book review: The art of computer systems performance analysis: Techniques for experimental design, measurement, simulation, and modeling by raj jain (john wiley & sons 1991). *SIGMETRICS Perform. Eval. Rev.*, 19(2):5--11. Reviewer-Al-Jaar, Robert Y.
- Andriluka, M., Roth, S., and Schiele, B. (2008). People-tracking-by-detection and people-detection-by-tracking. In *CVPR*.
- Angelova, A., Krizhevsky, A., Vanhoucke, V., Ogale, A., and Ferguson, D. (2015). Real-time pedestrian detection with deep network cascades. In *BMVC*.
- Benenson, R., Mathias, M., Timofte, R., and Gool, L. J. V. (2012a). Pedestrian detection at 100 frames per second. In *CVPR*.
- Benenson, R., Mathias, M., Timofte, R., and Van Gool, L. (2012b). Pedestrian Detection at 100 Frames per Second.
- Benenson, R., Mathias, M., Tuytelaars, T., and Gool, L. J. V. (2013). Seeking the strongest rigid detector. In *CVPR*.
- Benenson, R., Omran, M., Hosang, J., , and Schiele, B. (2014). Ten years of pedestrian detection, what have we learned? In *ECCV, CVRSUAD workshop*.
- Breiman, L. (2001). Random forests. *Mach. Learn.*, 45(1):5--32.
- C. Toca, M. C. and Patrascu, C. (2015). Normalized autobinomial markov channels for pedestrian detection. In *BMVC*.
- Cai, Z., Saberian, M., and Vasconcelos, N. (2015). Learning complexity-aware cascades for deep pedestrian detection. In *ICCV*.
- Costea, A. D. and Nedeveschi, S. (2014). Word channel based multiscale pedestrian detection without image resizing and using only one classifier. In *CVPR*.

- Criminisi, A. and Shotton, J. (2013). *Decision Forests for Computer Vision and Medical Image Analysis*. Springer Publishing Company, Incorporated.
- Dalal, N. and Triggs, B. (2005). Histograms of Oriented Gradients for Human Detection. In *CVPR*.
- de Melo, V. H. C., ao, S. L., and Schwartz, W. R. (2013). Pedestrian detection optimization based on random filtering. In *Workshop of Works in Progress (WIP) in SIBGRAPI 2013*.
- de Melo, V. H. C., Leao, S., Menotti, D., and Schwartz, W. R. (2014). An optimized sliding window approach to pedestrian detection. In *ICPR*.
- Dollár, P., Appel, R., Belongie, S., and Perona, P. (2014). Fast feature pyramids for object detection. In *PAMI*, pages 1532–1545.
- Dollár, P., Tu, Z., Perona, P., and Belongie, S. (2009). Integral channel features. In *BMVC*.
- Dollár, P., Wojek, C., Schiele, B., and Perona, P. (2009). Pedestrian detection: A benchmark. In *CVPR*.
- Dollár, P., Wojek, C., Schiele, B., and Perona, P. (2012). Pedestrian detection: An evaluation of the state of the art. *PAMI*, 34.
- Ess, A., Leibe, B., and Gool, L. J. V. (2007). Depth and appearance for mobile scene analysis. In *ICCV*.
- Gerónimo, D. and López, A. M. (2014). Vision-based pedestrian protection systems for intelligent vehicles. pages i–x, 1–114.
- Gonzalez, R. C. and Woods, R. E. (1992). Digital image processing. pages I–XVI, 1–716.
- Hou, X. and Zhang, L. (2007). Saliency detection: A spectral residual approach. In *CVPR*.
- Jiang, Y. and Ma, J. (2015). Combination features and models for human detection. In *CVPR*.
- Jordao, A., de Melo, V. H. C., and Schwartz, W. R. (2015). A study of filtering approaches for sliding window pedestrian detection. In *Workshop em Visao Computacional (WVC)*, pages 1–8.

- Jordao, A., de Souza, J. S., and Schwartz, W. R. (2016). A late fusion approach to combine multiple pedestrian detectors. In *ICPR*.
- Jordao, A. and Schwartz, W. R. (2016). Oblique random forest based on partial least squares applied to pedestrian detection. In *IEEE International Conference on Image Processing (ICIP)*.
- Ko, B. C., Jeong, M., and Nam, J. (2014). Fast human detection for intelligent monitoring using surveillance visible sensors. *Sensors*, 14(11):21247--21257.
- Ko, B. C., Kim, D.-Y., Jung, J.-H., and Nam, J.-Y. (2013). Three-level cascade of random forests for rapid human detection. *Optical Engineering*, 52(2):027204–027204.
- Li, L., Su, H., Xing, E. P., and Li, F. (2010). Object bank: A high-level image representation for scene classification & semantic feature sparsification. In *NIPS*.
- Lim, J. J., Zitnick, C. L., and Dollár, P. (2013). Sketch tokens: A learned mid-level representation for contour and object detection. In *CVPR*.
- Marín, J., Vázquez, D., López, A. M., Amores, J., and Leibe, B. (2013). Random forests of local experts for pedestrian detection. In *ICCV*.
- Masaki, I., Horn, B. K., Bilgiç, B., et al. (2010). *Fast human detection with cascaded ensembles*. PhD thesis, Massachusetts Institute of Technology.
- Mathias, M., Benenson, R., Timofte, R., and Gool, L. J. V. (2013). Handling occlusions with franken-classifiers. In *ICCV*.
- Menze, B. H., Kelm, B. M., Splitthoff, D. N., Köthe, U., and Hamprecht, F. A. (2011). On oblique random forests. In *ECML/PKDD (2)*.
- Nam, W., Dollár, P., and Han, J. H. (2014). Local decorrelation for improved pedestrian detection. In *NIPS*.
- Nazare, A. C., dos Santos, C. E., Ferreira, R., and Schwartz, W. R. (2014). Smart Surveillance Framework: A Versatile Tool for Video Analysis. In *WACV*.
- Ouyang, W. and Wang, X. (2013). Single-pedestrian detection aided by multi-pedestrian detection. In *CVPR*.
- Paisitkriangkrai, S., Shen, C., and van den Hengel, A. (2014). Strengthening the effectiveness of pedestrian detection with spatially pooled features. In *ECCV*.

- Rosipal, R. and Krämer, N. (2006). Overview and recent advances in partial least squares. In *in Subspace, Latent Structure and Feature Selection Techniques, Lecture Notes in Computer Science*, pages 34--51. Springer.
- Schölkopf, B. and Smola, A. J. (2002). *Learning with kernels: support vector machines, regularization, optimization and beyond*. MIT press.
- Schwartz, W. R., Davis, L. S., and Pedrini, H. (2011). Local Response Context Applied to Pedestrian Detection. In *CIARP*, pages 181–188.
- Schwartz, W. R., Kembhavi, A., Harwood, D., and Davis, L. S. (2009). Human detection using partial least squares analysis. In *ICCV 2009*.
- Shlens, J. (2005). A tutorial on principal component analysis. In *Systems Neurobiology Laboratory, Salk Institute for Biological Studies*.
- Silva, G., Schnitman, L., and Oliveira, L. (2012). Multi-scale spectral residual analysis to speed up image object detection. In *SIBGRAPI*.
- Singh, S., Gupta, A., and Efros, A. A. (2012). Unsupervised discovery of mid-level discriminative patches. In *ECCV (2)*.
- Smith, G. J. D. (2004). Behind the screens: Examining constructions of deviance and informal practices among CCTV control room operators in the UK. *Surveillance and Society*, 2(2/3):376--395.
- Tian, Y., Luo, P., Wang, X., and Tang, X. (2015a). Deep learning strong parts for pedestrian detection. In *ICCV*.
- Tian, Y., Luo, P., Wang, X., and Tang, X. (2015b). Pedestrian detection aided by deep learning semantic tasks. In *CVPR*.
- Wang, X., Han, T. X., and Yan, S. (2009). An HOG-LBP human detector with partial occlusion handling. In *ICCV*.
- Wold, H. (1985). Partial Least Squares. In *Encyclopedia of Statistical Sciences*, volume 6, pages 581–591. Wiley, New York, NY, USA.
- Yang, Y., Wang, Z., and Wu, F. (2015). Exploring prior knowledge for pedestrian detection. In *BMVC*.
- Zhang, S., Bauckhage, C., and Cremers, A. B. (2014). Informed haar-like features improve pedestrian detection. In *CVPR*.

Zhang, S., Benenson, R., and Schiele, B. (2015). Filtered channel features for pedestrian detection. In *CVPR*.

Zhu, Q., Yeh, M., Cheng, K., and Avidan, S. (2006). Fast human detection using a cascade of histograms of oriented gradients. In *CVPR*.