The Good, The Fast and The Better Pedestrian Detector

Artur Jordão, William Robson Schwartz
Smart Surveillance Interest Group, Computer Science Department
Universidade Federal de Minas Gerais, Minas Gerais, Brazil

***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). At each node of a decision tree, the method utilizes the PLS to find a decision surface that better splits the samples, 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 combine 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*-Oblique Decision Tree; Partial Least Squares; Filtering Approaches; High-Level Information; Fusion of Detectors.**

I. Introduction

Since the past decade, pedestrian detection has been an active research topic in Computer Vision, mostly because of its direct applications in surveillance and robotics [1]. This task faces many challenges, such as variance in clothing styles and appearance, distinct illumination conditions, frequent occlusion among pedestrians and high computational cost.

According to Benenson et al. [1], 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 [2], [1]. 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 to several advantages that this approach presents including low computational cost to test and its design naturally treats problems with more than two classes [3].

According to the definition of Breiman [4], 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 being considered: 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 [5]. 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) [6], which is a popular technique to dimensionality reduction and regression [7], [8].

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, filtering approaches can be utilized to address the referred problem. Filtering approaches are executed before the feature extraction and classification stage, and they focus on reducing the amount of data that has to be processed, allowing the consideration of fewer samples (detection windows), reducing the computational cost.

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 [9], [10], 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 the 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 [11].

While numerous classification methods and optimization approaches have been investigated, the majority of efforts in pedestrian detection can be attributed to the improvement in features alone and evidences suggest that this trend will continue [2], [1]. In addition, several works show that the combination of features creates a more powerful descriptor which improves the detection [7], [9], [12]. Despite the com-

This work is the result of a M.Sc. dissertation

bination of features provide a better discrimination, pedestrian detection is still dealing with some problems. The existence of false positives, such as 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 [13], [1], [14].

The most recent work regarding high level information, proposed by Jiang and Ma [14], relies on the following hypothesis. If two detectors find the same object, given a specific 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. 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 [15].

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). Regarding the filtering approaches, we demonstrate that the evaluated filters are able to discard a large number of windows without compromising the detection accuracy (Due to the lack of space, the experimental results of the filtering approaches are not in this paper). Finally, regarding the our spatial consensus algorithm, experiments showed that it outperforms the state-of-the-art, achieving the best results in all evaluated datasets.

Contributions. The main contributions of this work are the following. Our first contribution is a novel alternative to generate the oRF, providing a smaller forest when compared with the traditional oRF-SVM [6]. Our second contribution is a detailed study of a series of filtering approaches that provide a lower computational cost to the detection [11]. Finally, our last contribution is a novel late fusion approach that enables to combine multiple detectors improving the detection [15].

The publications achieved with this dissertation 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. (2016). Spatial consensus: A late fusion approach to combine pedestrian detectors. In International Conference on Pattern Recognition (ICPR). Accepted.
3) 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). Accepted.

II. Methodology

In Section II-A, we introduce the steps to build the the oblique decision tree associated with the PLS and SVM. Then, in Section II-B, we present our proposed late fusion algorithm to combine multiple detectors.

Regarding the filtering approaches (The Fast), the following filters were used in our evaluation: entropy filter, magnitude filter, random filtering [8], and saliency map based on spectral residual [16]. Due to the lack of space, the description of the filtering approaches is not in this paper. A detailed discussion regarding the filtering approaches can be found in [11].

A. Oblique Random Forest with PLS (The Good)

This section starts by describing the framework to build an oblique decision tree. Afterwards, we describe how to employ the PLS and SVM with the oblique random forest, respectively.

The steps performed to construct the oblique decision trees composing the oRF are the following. First, we employ feature selection on the data received by the tree. As noticed by Breiman [4], this technique ensures diversity between the trees, presenting an important contribution to improve the accuracy. Second, a starting node (root), Rj, 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 τ 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

$$\Delta L(R_j, s) = L(R_j) - \frac{\mid R_{jls} \mid}{\mid R_j \mid} L(R_{jls}) - \frac{\mid R_{jrs} \mid}{\mid R_j \mid} L(R_{jrs}), \tag{1}$$

where $L(R_j) = \sum_{i=1}^{K} p_i^j (1 - p_i^j)$, $s \in S$ (S is a set of thresholds), K represents the class number and p_i^j is the ratio of class i at the node j. We choose *gini index* because it produces an extremely randomized forest [3].

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 prediction of the random forest for a given sample v is performed by aggregating the predictions of the trees by arithmetic average.

Specifically, to build each node in an oblique decision tree associated with PLS, the samples received by a node have their dimension reduced to a latent space *p-dimensional* using the PLS. Then, the best threshold to split the data samples is obtained using the *gini index* on the regression values given by the PLS.

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

B. Spatial Consensus (The Better)

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, giving more confidence to candidate windows that are more likely to 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 number of false positives.

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 ith detector attributes a score of $[-\infty, +\infty]$ to a given candidate window and the classifier of the jth detector attributes a score between $[0, 1]$, the scores cannot be combined directly. To address this problem, we employ the same procedure used by [14] to normalize the scores. The procedure is described as follows.

First, we fix a set of recall points, e.g, $\{1, 0.9, 0.8, ..., 0\}$. Then, for each detector, we collect the set of scores, τ, that achieve these recall points. Finally, we estimate a function that projects τ_j onto τ_i (details in Section V). 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

$$J = \frac{\text{area}(w_r \cap w_j)}{\text{area}(w_r \cup w_j)}, \tag{2}$$

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

$$score(w_r) = score(w_r) + score(w_j) \times J. \tag{3}$$

The process described above is repeated n times, where n is the number of detectors besides the root detector. Algorithm 1 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 jth detector and p is the amount of all candidate windows in $\{det_j\}_{j=1}^{n}$. Similarly, the approach proposed by [14] (weighted-NMS method) presents complexity of $O(cw \log cw + cw^2)$. Although both methods present a quadratic complexity, p is extremely small because the non-maximum suppression is employed for each detector before presenting the candidate windows to Algorithm 1 (see Section V), which renders the computational cost of both our Spatial Consensus method and the baseline approach [14] to be negligible when compared with the execution time of the individual pedestrian detectors.

Algorithm 1: Spatial Consensus

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

```
for j ← 1 to n do
    project det_j score to det_root score;
    foreach w_r in det_root do
        foreach w_j in det_j do
            compute overlap using Equation 2;
            if overlap >= σ then
                update w_r score using Equation 3;
            end
        end
        if w_r does not presents any matching then
            discard w_r;
        end
    end
end
```

Removing the dependency of root detector. According to the algorithm described above, 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 specify a particular pedestrian detector as the input to the SC algorithm.

To generate windows for the virtual root detector, 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 $(w_i^l, l = 1, 2, ..., k)$ to create a set of windows that will be used to generate a single window belonging to the det_{vr} using $w_i^{vr} = \frac{1}{k}\sum_{l=1}^{k} w_i^l$, 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 described earlier. Once the windows of the virtual root detector had been generated, we can execute the same SC algorithm.

III. Experimental Results

To quantify the detection performance, 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 [2], [1]), where is measured the area under the curve on the interval from 10^{-2} to 10^{0} false

positive per image, in which lower 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 and transit safety), this way, we evaluate the methods under a more rigorous detection.

IV. 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 baselines. At the calibration stage of the oRFs parameters, we utilized the TUD pedestrian dataset as validation set [17].

Feature Extraction. We extract the HOG descriptor for each detection window following the configuration proposed by [18], with blocks of 16×16 pixels and cells 8×8 pixels. This configuration results in a descriptor of 3780 dimensions. We are using these 3780 features during the feature selection process (see Section II-A), for both the oblique random forest to provide a comparison not influenced by the features.

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 feature 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 II-A). 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 set, the best parameters to oRF-SVM were using 200 trees and nF = 400, where it achieved a miss rate of 41.67%. The oRF-PLS obtained the best results with 40 trees and nF = 550, presenting a miss rate of 38.18%.

Influence of the Number of Trees. Table I shows the 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 high. In addition, by computing the standard deviation of the 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 percentage points (p.p) while our proposed method presented a standard deviation of 2.42 p.p. Thus, the use of PLS to build oRF is more adequate than use the SVM once it produces smaller and more accurate forests.

Time Issues. In this experiment, we show that the proposed oRF-PLS is faster than the oRF-SVM. For this purpose, we performed a statistical test between the time (in seconds) to run the complete pipeline detection on an image of 640×480 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.

TABLE I
Miss-rate (low is better) on the TUD Pedestrian dataset in function of the number of trees.

	Number of trees					
	8	16	24	32	40	200
oRF-PLS	45.5	44.3	42.1	43.3	**38.2**	44.7
oRF-SVM	77.2	54.4	58.3	62.6	55.6	**41.7**

Comparison with Baseline Approaches. Our last experiment regarding the oblique random forest compares the proposed oRF-PLS with traditional baselines pedestrian detectors [2], [1]. Our proposed method outperforms common classifiers used in pedestrian detection, e.g., linear SVM (HOG+SVM [19] and QDA (PLS detector [7]) in 8.72 and 2.83 p.p. respectively, on the interval 10^{-2} to 10^{0}. When evaluated on the interval 10^{-2} to 10^{-1}, our method outperformed the HOG+SVM and the PLS detector in 13.11 and 4.61 p.p. respectively. Moreover, the oRF-PLS outperforms a robust partial occlusion method, HOG+LBP [19], in 1.84 and 6.28 p.p to the area in 10^{-2} to 10^{0} and 10^{-2} to 10^{-1}, respectively. According to the results, the proposed oRF-PLS is able to obtain equivalent (or better) results when compared with traditional classifiers.

V. Spatial Consensus Evaluation

This section first evaluates the steps required to execute the spatial consensus algorithm. Finally, compares our method with the baseline and state-of-the-art pedestrian detectors.

Preparing the input detectors. Initially, we need to define det_{root} and a set of detectors $\{det_j\}_{j=1}^{n}$. Due to the large number of pedestrian detectors currently available, there are many options to determine both det_{root} and $\{det_j\}_{j=1}^{n}$ [1], [2]. In this work, we define these detectors as the eleven best ranked pedestrian detectors on the INRIA person dataset. The best ranked detector, the SpatialPolling [20], was set to be the det_{root} and the remaining detectors were set to $\{det_j\}_{j=1}^{n}$. Once specified $\{det_j\}_{j=1}^{n}$, the order of the set members does not affect the result, since all detectors of $\{det_j\}_{j=1}^{n}$ must be evaluated to discard a window of det_{root}.

At the score calibration step, we use the INRIA person dataset to acquire the set of scores τ. Then, 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. However, once the scores are calibrated, we use the estimated regression on the other datasets.

Spatial Consensus vs. weighted-NMS. In this experiment, aiming a fair comparison, we report the results provided by better combination of detectors in each dataset, to both the algorithms. In addition, using the INRIA Person dataset, we estimated the best thresholding σ for the *Jaccard coefficient* as 0.6 for both algorithms.

The weighted-NMS achieved the best results on the INRIA dataset when two detectors are added, Sketch Tokens [21] and Roerei [22], 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 [23] detectors. This combination was not enough to outperform the TA-CNN [24] (current state-of-the-art on this dataset with 34.98%). On the other hand, our approach reached best results adding, beyond these two detectors, the LDCF [25] detector, where we overcome the state-of-the-art in 1.37 p.p.

The best result of the weighted-NMS on the Caltech dataset was achieved combining the Roerei and Franken detectors. 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 use of our approach reduces the difference to most recent state-of-the-art detector (CompACT-Deep [26] - 12.43%) from 16.81% to 10.73%.

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 [18] (miss rate of 46%) on the INRIA person dataset. When it was inserted into $\{det_j\}_{j=1}^{n}$, the miss rate achieved by our method went from 7.95% to 8.90% and to the weighted-NMS the result, was from 14.11% to 16.78%, demonstrating that our algorithm is more robust to less accurate detectors.

Comparison with the state-of-the-art. In this experiment, we compare the results of the proposed SC algorithm with state-of-the-art methods. To perform a fair comparison, we considered the results reported by the authors in their works.

Figures 1(a) and (b) show that our algorithm outperforms the state-of-the-art on the INRIA and ETH datasets achieving miss-rate of 7.95% and 33.61%, respectively. Furthermore, Figure 1(c) shows that our method achieves significant results on the Caltech dataset with miss rate of 19.84%.

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 the Section II-B), 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 *Domain Knowledge* (SC+DK).

Given the definition of the SC+DK, we 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 [24] detector as the det_{root} and the $\{det_j\}_{j=1}^{n}$ was composed of the SpatialPooling and Franken [23] detectors. To the latter, the det_{root} was the CompACF-Deep detector [26] and the $\{det_j\}_{j=1}^{n}$ was composed of the DeepParts [27] and CheckerBoards+ [28].

According to the results shown in Figure 1(b) and 1(c), the use of the aforementioned 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 miss-rate from 10^{-2} to 10^{-1}, where we outperformed the state-of-the-art in 11.15 and 2.84 p.p. on the ETH and Caltech datasets, respectively.

Virtual Root Detector. This 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.

Regarding the results presented in Figure 1, we can notice that the SC+VR outperforms the SC approach (where an initial root detector must be defined) in 0.5 and 3.32 p.p. to INRIA and Caltech, respectively. To the ETH dataset, the miss rate increased 0.13 p.p., in relation to SC approach.

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

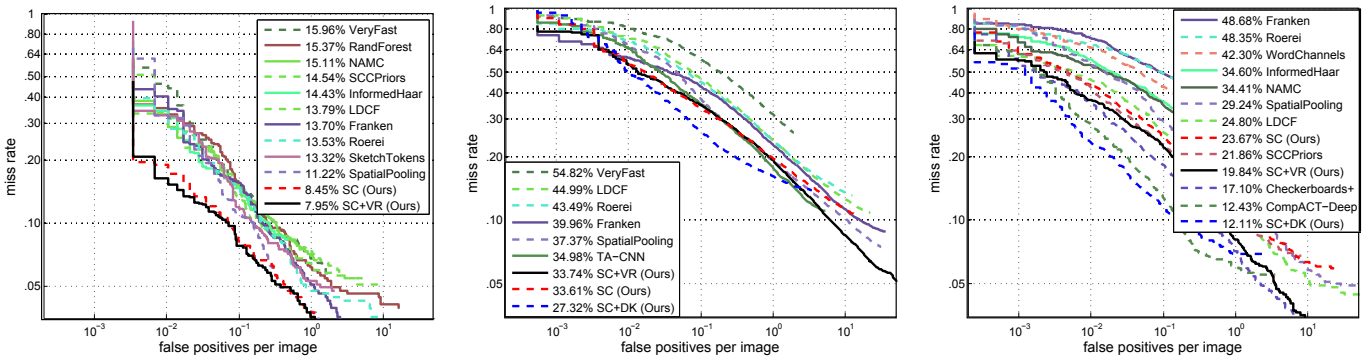
Time issues. As described in Section II-B, the complexity of our method is equal to the weighted-NMS. Although quadratic, both methods run in real time since the traditional NMS is performed for each individual detector before starting the algorithms. Besides, the values of p_{root} and p are corresponding to the number of pedestrians at the scene, which is usually very low. 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 value of p is small, our approach is able to run in real time. To show that, we computed the time average to execute of the SC on an image 640 pixels, using 10 detectors to compose $\{det_j\}_{j=1}^{n}$ and without any parallelization technique. The SC runs in 67 milliseconds, on average (this experiment was executed 10 times). Additionally, the most recent survey of computation cost at the detection pedestrian [29], showed that the faster detector presenting high accuracy is able to process 15 frames per second on a GPU [29]. Therefore, we conclude that our method is able to improve the detection results and could be fast to execute, even though our algorithm requires results of individual detectors.

VI. 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. To this end, we propose a novel oblique random for-



(a) INRIA Person Dataset (b) ETH Pedestrian Dataset (c) Caltech Pedestrian Dataset

Fig. 1. Comparison of our proposed approach with the state-of-the-art. Results using the miss-rate of 10^{-2} to 10^{0} (standard protocol).

est. 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 it presents high computational cost. Besides, our method achieved comparable 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 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 proposed method outperforms the state-of-the-art in two pedestrian detection benchmarks and achieves comparable results on the Caltech dataset. Moreover, we showed that with previous knowledge of the domain, our method outperforms the most powerful detectors in each dataset.

VII. Acknowledgments

The authors 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).

References

[1] R. Benenson, M. Omran, J. Hosang, , and B. Schiele, "Ten years of pedestrian detection, what have we learned?" in *ECCV, CVRSUAD workshop*, 2014.

[2] P. Dollár, C. Wojek, B. Schiele, and P. Perona, "Pedestrian detection: An evaluation of the state of the art," *PAMI*, 2012.

[3] A. Criminisi and J. Shotton, *Decision Forests for Computer Vision and Medical Image Analysis*. Springer Publishing Company, Incorporated, 2013.

[4] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, Oct. 2001.

[5] B. H. Menze, B. M. Kelm, D. N. Splitthoff, U. Köthe, and F. A. Hamprecht, "On oblique random forests." in *ECML/PKDD (2)*, 2011.

[6] A. Jordao and W. R. Schwartz, "Oblique random forest based on partial least squares applied to pedestrian detection," in *ICIP*, (accepted)2016.

[7] W. R. Schwartz, A. Kembhavi, D. Harwood, and L. S. Davis, "Human detection using partial least squares analysis," in *ICCV 2009*, 2009.

[8] V. H. C. de Melo, S. Leao, D. Menotti, and W. R. Schwartz, "An optimized sliding window approach to pedestrian detection." in *ICPR*, 2014.

[9] P. Dollár, Z. Tu, P. Perona, and S. Belongie, "Integral channel features." in *BMVC*, 2009.

[10] P. Dollár, R. Appel, S. Belongie, and P. Perona, "Fast feature pyramids for object detection." in *PAMI*, 2014.

[11] A. Jordao, V. H. C. de Melo, and W. R. Schwartz, "A study of filtering approaches for sliding window pedestrian detection," in *Workshop em Visao Computacional (WVC)*, 2015, pp. 1–8.

[12] J. Marín, D. Vázquez, A. M. López, J. Amores, and B. Leibe, "Random forests of local experts for pedestrian detection," in *ICCV*, 2013.

[13] L. Li, H. Su, E. P. Xing, and F. Li, "Object bank: A high-level image representation for scene classification & semantic feature sparsification," in *NIPS*, 2010.

[14] Y. Jiang and J. Ma, "Combination features and models for human detection," in *CVPR*, 2015.

[15] A. Jordao, J. S. de Souza, and W. R. Schwartz, "Spatial consensus: A late fusion approach to combine pedestrian detectors," in *ICPR*, (accepted)2016.

[16] X. Hou and L. Zhang, "Saliency detection: A spectral residual approach," in *CVPR*, 2007.

[17] M. Andriluka, S. Roth, and B. Schiele, "People-tracking-by-detection and people-detection-by-tracking." in *CVPR*, 2008.

[18] N. Dalal and B. Triggs, "Histograms of Oriented Gradients for Human Detection," in *CVPR*, vol. 1, 2005.

[19] X. Wang, T. X. Han, and S. Yan, "An HOG-LBP human detector with partial occlusion handling," in *ICCV*, 2009.

[20] S. Paisitkriangkrai, C. Shen, and A. van den Hengel, "Strengthening the effectiveness of pedestrian detection with spatially pooled features," in *ECCV*, 2014.

[21] J. J. Lim, C. L. Zitnick, and P. Dollár, "Sketch tokens: A learned mid-level representation for contour and object detection," in *CVPR*, 2013.

[22] R. Benenson, M. Mathias, T. Tuytelaars, and L. J. V. Gool, "Seeking the strongest rigid detector," in *CVPR*, 2013.

[23] M. Mathias, R. Benenson, R. Timofte, and L. J. V. Gool, "Handling occlusions with franken-classifiers," in *ICCV*, 2013.

[24] Y. Tian, P. Luo, X. Wang, and X. Tang, "Pedestrian detection aided by deep learning semantic tasks," in *CVPR*, 2015.

[25] W. Nam, P. Dollár, and J. H. Han, "Local decorrelation for improved pedestrian detection," in *NIPS*, 2014.

[26] Z. Cai, M. Saberian, and N. Vasconcelos, "Learning complexity-aware cascades for deep pedestrian detection," in *ICCV*, 2015.

[27] Y. Tian, P. Luo, X. Wang, and X. Tang, "Deep learning strong parts for pedestrian detection," in *ICCV*, 2015.

[28] S. Zhang, R. Benenson, and B. Schiele, "Filtered channel features for pedestrian detection," in *CVPR*, 2015.

[29] A. Angelova, A. Krizhevsky, V. Vanhoucke, A. Ogale, and D. Ferguson, "Real-time pedestrian detection with deep network cascades," in *BMVC*, 2015.