Fast and Robust Optimization Approaches for Pedestrian Detection

Victor Hugo Cunha de Melo*, David Menotti (*Co-advisor*)†, William Robson Schwartz (*Advisor*)*
*Computer Science Department, Universidade Federal de Minas Gerais, Belo Horizonte, Brazil
†Computer Science Department, Universidade Federal de Ouro Preto, Ouro Preto, Brazil
Email: victorhcmelo@dcc.ufmg.br, menottid@gmail.com, william@dcc.ufmg.br

***Abstract*—The large number of surveillance cameras available nowadays in strategic points of large cities aims to provide a safe environment. However, the huge amount of visual data provided by the cameras prevents its manual processing, requiring the application of automated methods. Among such methods, pedestrian detection plays an important role in reducing the amount of data. However, the currently available methods are unable to process such large amount of data in real time. Therefore, there is a need for the development of optimization techniques. Towards accomplishing the goal of reducing costs for pedestrian detection, this Master's thesis proposed two optimization approaches. Our first approach proposes a novel optimization that performs a random filtering in the image to select a small number of detection windows, allowing a reduction in the computational cost. Our results show that accurate results can be achieved even when a large number of detection windows are discarded. The second approach consists of a cascade of rejection based on Partial Least Squares (PLS) combined with the propagation of latent variables through the stages. Our results show that the method reduces the computational cost by increasing the number of rejected background samples in earlier stages of the cascade.**

***Keywords*-Pedestrian detection; random filtering; location regression; cascade of rejection; partial least squares.**

I. Introduction

Video surveillance has been around us for almost a century and recently it suffered a huge growth due to the reduction in prices of the cameras and the increasing network connectivity [1]. Nowadays, we have a growing availability of visual data captured by surveillance cameras, which provides safer environments for people whom attend monitored environments. However, the large number of cameras to be monitored and consequently the large number of images that must be interpreted, precludes an effective manual processing and require a significant number of people dedicated to analyzing visual data. The ubiquity of video surveillance is advantageous for protection, but it is harder to monitor.

Granted that the manual analysis of large amounts of visual data is challenging, the automatic understanding and interpretation of activities performed by humans in videos show great interest because such information can assist the decision making process of security agents. Among the automatic approaches for understanding and interpretation, pedestrian detection plays an important role, since pedestrians are the most important agents in the scene. They can be found in several environments, representing a key information for numerous applications. Given their importance, we are interested in monitoring them to determine how they interact with the environment. Therefore, we want to know their location and what activities they are performing to infer whether they may harm someone or something might harm them.

Although remarkable progress has been achieved in the past years for pedestrian detection [2], the problem still remains open due to its difficult nature [3], which includes changes in appearance due to different types of clothing, illumination changes and pose variations, low quality of the data acquired, and the small size of the pedestrian, which make the detection process harder. In addition, a large number of applications require a high performance and reliable detection results, outlining the need for efficient and accurate pedestrian detection approaches.

There are several optimization approaches to reduce the computational cost of pedestrian detectors that may be grouped into three major categories, namely, *filtering*, *parallelization and GPGPUs*, and *cascades of rejection* [4]. We turn our focus to filtering and cascade of rejection techniques, since our proposed approaches fits into these two categories. Although parallelization/GPGPU algorithms are not addressed by this work, our proposed approaches may benefit from them since they are complementary.

Among filtering approaches, there are several solutions to reduce the amount of data to be processed. These approaches are based on branch-and-bound techniques, saliency detectors, among others [5]. Although most of the techniques allows to reduce the amount of data and may be used as a preliminary step to the classifier, they still might present unnecessary evaluations.

Cascades of rejection are a widely employed approach to reduce the computational cost in object detection. They are composed of multiple stages, each one composed of a classifier or an ensemble of them. The main idea behind this approach is to use simple classifiers to discard detection windows that are easy to classify, while the remaining windows advance through the cascade, where more complex classifiers are used. This process leads to a significant reduction in computational cost [6], [7].

Moving towards the reduction of the computational cost in pedestrian detection, this Master's thesis proposed two novel

This work refers to a M.Sc. thesis defended in February, 2014.

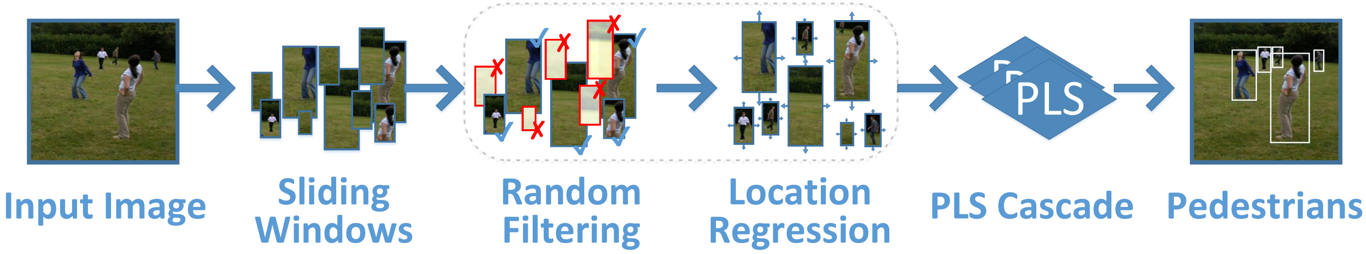

Fig. 1. Fluxogram describing the proposed optimization approach, the random filtering combined with location regression, and the PLS Cascade.

optimization approaches for reducing the computational cost of pedestrian detection, namely, the *random filtering* and the *PLS Cascade*. These optimization approaches focus on the generation of the detection windows and on the classifier. Our random filtering approach, described in Section II-A, aims at rejecting detection windows by evaluating only a few of them; consequently, a large amount of windows are preemptively discarded without cost. Later, we correct the misplaced windows using location regression, which has a low computational cost since it requires the extraction of simple and sparse features. Therefore, our proposed filtering method is able to achieve a considerably speedup.

Different from the previous approaches, the PLS Cascade (described in Section II-B), proposes a cascade of classifiers using a combination of Partial Least Squares (PLS) and Variable Importance on Projection (VIP) aiming at reducing the number of projections required by the PLS detector [8]. In addition, different from approaches such as [6] and [7], the proposed cascade propagates information (without increasing the computational cost) to later stages to increase the discriminability of the classifiers instead of maintaining all feature descriptors as candidates during all stages. Our resulting approach is faster to train than the conventional cascades; the usage of the VIP allows to reject more samples in the earlier stages, and the computational cost of the PLS Detector is considerably reduced.

The main contributions provided by this Master's thesis are: (1) A new filtering approach, which can be applied on any sliding window based detector; (2) The application of location regression to predict a pedestrian's correct location, given a shifted detection window; (3) Reduction of the computational cost of the PLS Detector, a widely employed pedestrian detector; (4) The application of VIP for feature ordering for fast training cascades of rejection; and, (5) The usage of the VIP for rejecting more samples in the earlier stages.

II. Proposed Approaches

The proposed optimization approach consists of the following steps (shown in Figure 1). Given an input image, in the first step we apply the traditional sliding window algorithm, which scans the input image with a window of fixed size in a range of scales, generating a set of detection windows. Such detection windows are presented to the random filtering which selects a random set of detection windows and adjusts them properly using a location regression (described in Section II-A). Later, the filtered and adjusted set of detection windows is presented to the last step of our methodology, the PLS Cascade, to reject detection windows that are easily classified as background, while windows that are harder to predict advances through the stages of the cascade (described in Section II-B).

It is worth noting that the proposed optimization approaches are mutually independent, such that the random filtering is optional for the execution of the PLS Cascade, and the converse is also true. Therefore, we may use random filtering with any other detector based on sliding window, and the PLS Cascade might be employed stand-alone. In this work, we focus on the PLS Detector [8] since it is widely used in the literature and achieves high detection rates on several pedestrian detection data sets.

A. Random Filtering

The sliding window algorithm generates detection windows in a wide range of scales and strides, yielding a set of overlapping windows with high redundancy, which highlights the need for a filtering approach. To reduce the amount of data processed by the pedestrian detector, we propose a method based on a random filtering followed by adjustments on the detection window locations. Here, we randomly select a fraction of windows that will be presented to a classifier. To ensure that every pedestrian is still detected, we rely on the Maximum Search Problem (MSP) theorem [9]. The problem of classifying windows as containing pedestrians or not may be seen as the task of finding a subset of windows containing pedestrians from a finite set of windows. Similarly to the majority of maximum search problems, the exact solution is computationally expensive (every sample has to be evaluated). Instead, the Maximum Search Problem states that it is possible to find almost optimal approximate solutions by randomly selecting a percentage of the samples to be evaluated.

Although the random filtering can provide a small subset of detection windows, such that almost every person in the image is covered, these windows might not provide the exact location of the pedestrian. Hence, this pedestrian might be missed due to the low response achieved by the classifier. Therefore, we employ an extra step before presenting the window to the classifier to adjust the window location to the

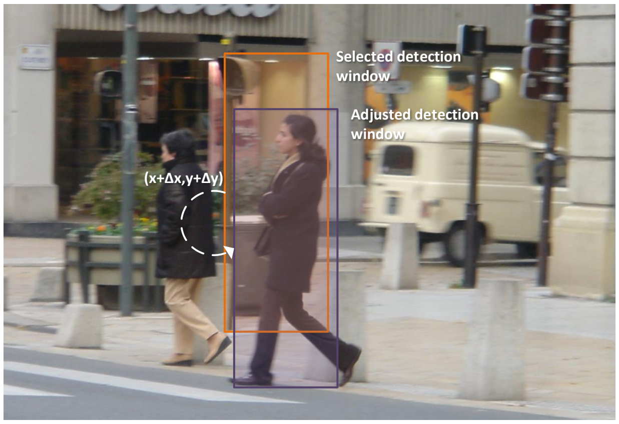

Fig. 2. Example of performing location regression to adjust the detection window location.

pedestrian. Aiming at adjusting the bounding box delimited by a detection window, we learn a regression model (referred to as *location regression*) to correct it to the pedestrian's location. In this problem, we want to find displacements Δx and Δy such that, when added to the centroid (G_x, G_y) of a given window, they move the detection window to the correct position of a pedestrian. Unlike [10], our proposed method learns the regression model during an offline phase.

To create the location regression model, we first need to generate a training set to be presented to the learning algorithm. Given a training sample, we generate a set of displaced windows with the respective differences $(\Delta x, \Delta y)$ to their correct position. This set of displaced windows is generated in all directions, as long as the Jaccard coefficient between the ground-truth bounding box and the displaced window is greater than 50% [3]. This ensures that we have a portion of the pedestrian within the window.

Once the training set is created, features descriptors are extracted from the windows and associated to the displacements. Ideally, such descriptors should be simple enough to preserve a low computational cost. Then, a regression with two dependent variables, Δx and Δy, is learned. Even though we have employed a regression based on Partial Least Squares due to its numerical stability and robustness to multicollinearity, other methods could have been applied.

During the testing phase, the location regression corrects the detection windows' location before presenting them to the classifier, as illustrated in Figure 2.

B. PLS Cascade

Designed to model relations between observed variables, the Partial Least Squares method (PLS) constructs a set of predictor variables (latent variables) as a linear combination of the original predictors, represented in a matrix X (feature matrix), containing one sample per row (the reader is referred to the work of Rosipal et al. [11] for more details). The responses associated with the samples are stored in a vector y, which are the class labels in the pedestrian detection problem [12]. Although PLS allows accurate detection in high-dimensional feature sets, the method presents a high computational cost [3], [10]. To reduce this cost, we propose the application of Partial Least Squares method in the context of a cascade framework, referred to as *PLS Cascade* (depicted in Figure 3).

In the proposed cascade, the feature descriptors are ranked by Variable Importance on Projection (VIP) so that more discriminative descriptors are used first in the cascade aiming at the rejection of a large number of samples in early stages. Derived from PLS, the Variable Importance on Projection (VIP) provides a score for each variable on the original feature space (matrix X), so that it is possible to rank the variables according to their predictive power in the PLS model. A higher score indicates that the variable presents more importance [12].

As a side effect of ranking the feature descriptors, the later stages will use less discriminative feature descriptors. Hence, to improve the detection rate in later stages of the cascade, we also propose to propagate the latent variables T_i from the ith stage to the next $(i+1)$th stage such that discriminative information is also available in later stages without the need for reconsideration of feature descriptors that were already used in previous stages.

III. Experiments

In this section, we present our evaluation of the proposed approaches. Section III-A addresses the random filtering approach and evaluates the effectiveness of location regression. Section III-B explores the PLS Cascade. The evaluation was conducted in the INRIA Person Dataset [13], a widely employed dataset for pedestrian detection.

A. Random Filtering

We evaluate the performance of random filtering and location regresion considering the following setup. We use the PLS Detector [8] as our baseline (any other sliding window based detector could be used instead). The detector was trained using the same Histograms of Oriented Gradients (HOG) setup used by Dalal and Triggs [13], i.e, a feature vector with 3,780 dimensions. To execute the location regression, we consider two feature descriptors, namely, pixel intensity and HOG [13].

The first experiment examines the random filtering to determine whether it misses a pedestrian or not when used by itself, which is shown by applying random filtering and evaluating the results obtained with respect to the ground-truth (i.e. a perfect classifier). Then, we evaluate whether the location regression is able to improve the detection results when applied after the random filtering. Afterwards, we evaluate the detector's behavior when presented to the windows selected by random filtering and the ones adjusted by location regression. Finally, we present the computational cost of the proposed approach.

Ground-truth Comparison. To verify the applicability of the Maximum Search Problem theorem, presented in Section II-A, this experiment determines the ratio of pedestrians that are covered[1] by at least one detection window as a function of

[1] A window is considered covered when the Jaccard coefficient is greater than 0.5.

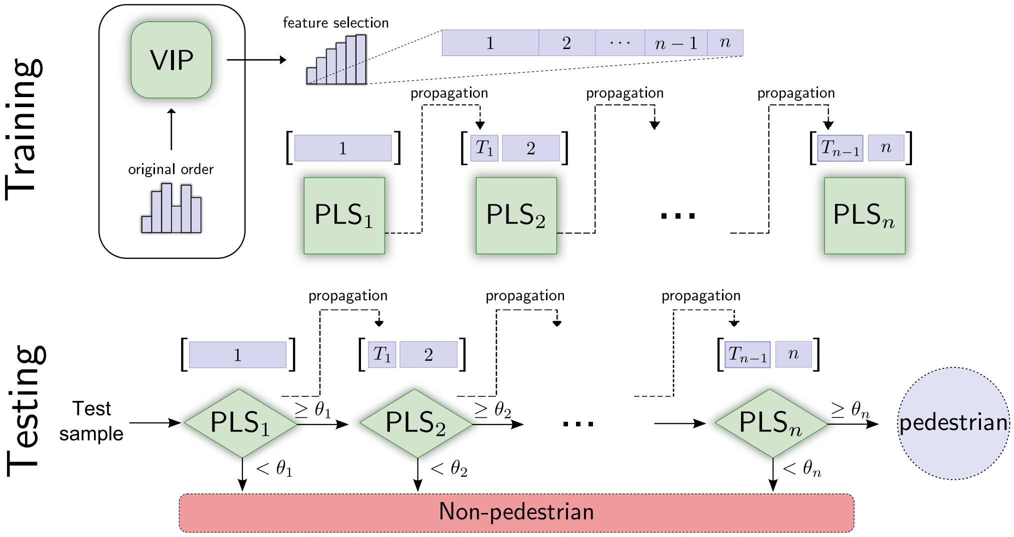

Fig. 3. Overall layout of the proposed *PLS cascade* using Partial Least Squares with latent variable propagation and Variable Importance on Projection (VIP) for feature ranking. Initially, the descriptors are extracted from the image and sorted using VIP, which ranks variables by their discriminative power. According to their rankings, the variables are set to stages, which allows to increase the number of discarded samples in the early stages. Each stage adds features until it reaches a desired false positive and miss rates. Hence, a PLS model is created using these features to classify the samples presented to this stage. Since features that have already been considered are not used in the later stages, the low-dimensional feature set T_i (latent variables) are propagated to avoid using only features with less discriminative power.

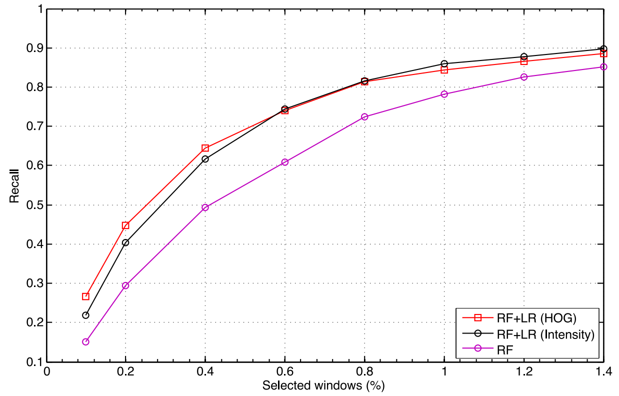

Fig. 4. Results of the random filtering approach. Achievable recall as a function of the number of selected windows, evaluated on the INRIA data set (RF: random filtering, LR: location regression).

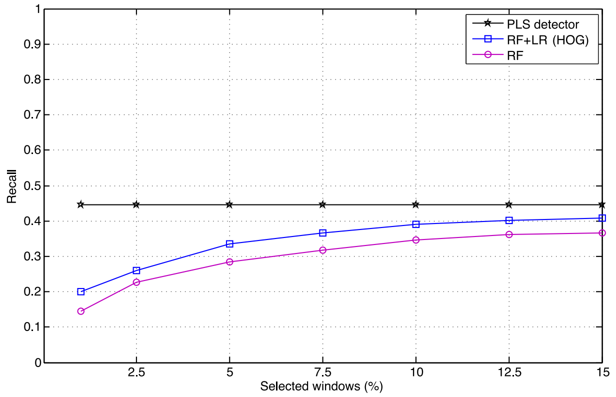

Fig. 5. Recall achieved at 1 FPPI when the selected detection windows are presented to the PLS detector. The PLS detector is shown as a line because it is executed with 100% of the detection windows (without filtering).

the percentage of selected windows. This can be verified according to their correct position given by the ground-truth. The random filtering, depicted in Figure 4 as RF (purple line), shows that a random selection of 1.4% of detection windows is enough to detect 83% of the pedestrians on the INRIA data set (if we consider a classifier that provides perfect results). Note that according to the MSP theorem, approximately 0.2% would be enough to approximate the maximum (find at least one pedestrian). However, since, on average, two people are present in each image, this value increases. In addition, we cannot achieve maximum recall score in this experiment because we are not padding the images, which means that people near to the edge of the images cannot be fit within a detection window.

Location regression. After applying the random filtering, we adjust the detection windows using location regression. Figure 4 reports the results achieved when applying the technique, using either pixel intensity or HOG as feature descriptor. As we can see, the regression is able to correct the position of the detection windows and, consequently, increase the recall achieved by the random filtering to a recall of 0.9 when 1.4% of the detection windows are selected. In addition, we observe that both feature descriptors, pixel intensity and HOG, obtained comparable results. Note that these results show the maximum achievable recall if the detector provided perfect results.

Pedestrian detector. Although random filtering misses only few pedestrians, these windows still need to be presented to a classifier, which may not obtain high accuracy due to some displacement of windows regarding to the person's location.

TABLE I

RELATIVE SPEEDUP ACHIEVED WITH THE PROPOSED METHOD WHEN COMPARED TO ORIGINAL DETECTOR ALONE (RF: RANDOM FILTERING, LR: LOCATION REGRESSION USING HOG).

SETUP	PERCENTAGE OF SELECTED WINDOWS							
	1%	2.5%	5%	7.5%	10%	12.5%	15%	100%
PLS Detector	–	–	–	–	–	–	–	1.00×
RF	67.91×	37.64×	21.81×	15.58×	12.10×	9.62×	8.45×	–
RF+LR	64.83×	35.13×	20.40×	14.59×	11.39×	9.09×	7.97×	–

This experiment evaluates how that may affect the accuracy of the detector/classifier. In the following experiments, we discuss only results achieved with HOG, because it has lower dimensionality and consequently is less subject to issues regarding the curse of dimensionality.

The results in Figure 5 show the recall obtained at one false positive per image (FPPI). Even after executing the random filtering, the accuracy is still comparable to the original detector (black line), which considers 100% of the detection windows, i.e., no windows are discarded. However, to achieve similar results, the number of selected detection windows had to be larger than the result achieved by the ground truth experiment previously described. This indicates that, although the correct detection windows have been selected, the PLS detector does not provide high responses for all the correct windows. By using the location regression, we could improve the random filtering results, increasing the recall to 40.9% (close to the 45% achieved by the original detector).

Computational cost. The results in Table I show the speedup for the experiments reported on Figure 5. The random filtering was able to achieve significant reduction in the computational cost, which also justifies its usage. In addition, by comparing the last two rows in Table I one may note that the employment of the location regression presents a low overhead.

B. PLS Cascade

We compared the PLS cascade with the cascade proposed by Zhu et al. [7] (using PLS for classification, instead of SVM) and with the PLS detector proposed by Schwartz et al. [8]. To establish a fair comparison, we have used the same 3,780 feature descriptors employed to learn the PLS cascade to learn Zhu's cascade and the PLS detector. The results are reported in False Positives per Window (FPPW).

According to Figure 6, the miss rate achieved by the PLS cascade (28.35% at 10^{-4} FPPW) is smaller than the one achieved by Zhu's cascade (40.16%). In addition, the number of samples discarded in the early stages is greater when the proposed cascade is considered (e.g., 67.45% of the detection windows are rejected by PLS cascade at the first stage and 44.46% by the Zhu's cascade), which makes the PLS cascade a faster and more accurate method.

When compared to the PLS detector, the proposed cascade achieved a higher miss rate at 10^{-4} (17.38% for the PLS detector and 28.35% for the PLS cascade), according to Figure 6. Even though the miss rate is higher, the proposed cascade performs only 7.49% of the projections required by the PLS detector (Figure 7), making the PLS cascade a promising approach, which should focus mainly on the use of a larger number of feature descriptors, an aspect that is usually necessary for cascade approaches (e.g., as much as $98,928$ descriptors were used by Zhu et al. [7] to achieve similar results obtained by Dalal and Triggs [13] with only 3,780 descriptors with their SVM-based detector).

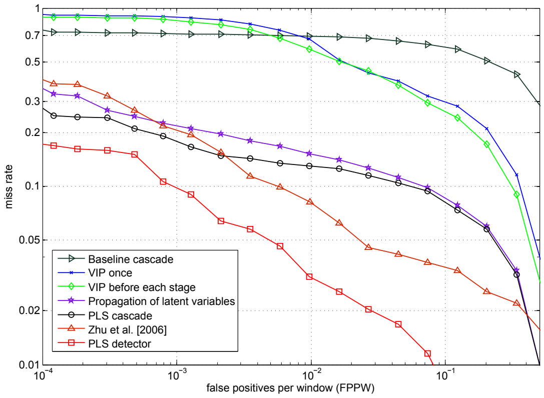

Fig. 6. Results of the proposed PLS Cascade. We compare different setups of the proposed cascaded and different pedestrian detectors. The plot is reported in a detection error tradeoff plot (lower and left-most plots are better).

C. Discussion and Remarks

Random filtering allowed a great reduction in the number of detection windows processed, without significantly increasing the computational cost. The percentage of selected windows estimated by the Maximum Search Problem might be seen as a lower limit of the real estimation of the number of selected windows. Applying location regression to correct the windows selected by random filtering allows to increase the detection rate, which could be explored to select an even smaller number of windows without greatly affecting the computational cost. Random filtering was not able to achieve the same recall obtained by the PLS Detector stand-alone, mainly due to the generalization of the classifier for non-centralized pedestrians.

PLS Cascade allowed to reduce the number of projections performed by the PLS Detector. The experiments have shown that VIP allows faster training and rapid window rejection in earlier stages of the cascade. The cumulative strategy of propagation has obtained better results than the noncumulative, since it incorporates features of every previous stage. Finally, there is no extra cost on computing the feature space onto a low dimensional one, since it is already done when performing the PLS regression.

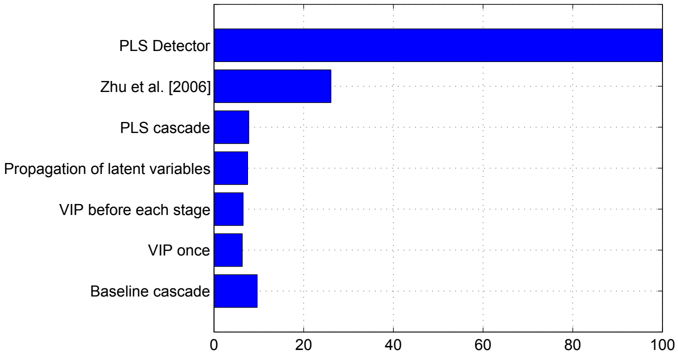

Fig. 7. Percentage of projections performed by each method, normalized by the number of projection required by the PLS detector.

IV. Conclusions

In this work, we proposed two novel optimization approaches to reduce the computational cost of pedestrian detection. The first optimization is based on random filtering approach to discard a large number of detection windows, which is further improved by the application of a regression to correct the window location to fit the persons in the image. This approach can be applied as a early step of any sliding window based detector. Compared to the application of a detector method alone, our experimental evaluation demonstrated that accurate results at a reduced computation cost may be achieved by our method even when a large number of detection windows are discarded.

The second optimization approach addresses the computational cost of the Partial Least Squares (PLS) Detector [8] by proposing the usage of a rejection cascade based on PLS. This method allows reducing the computational cost by discarding less promising samples earlier. In order to discard even more samples in earlier stages of the cascade, we proposed the use of the PLS-based feature sorting method VIP and to improve the detection rate, a latent variable propagation scheme is employed. Results showed that the combination of VIP and propagation of latent variables is promising due to the significant reduction on the number of projections, even when compared to a well-known cascade approach [7].

Scientific Publications and Awards

During the development of this work, we were awarded as one of the best works at the Seminar Week of the Graduate Program in Computer Science at Universidade Federal de Minas Gerais. I also coautored a paper by Schwartz et al. [10], published in Elsevier Neurocomputing Journal (Qualis A1), which was used in this Master's thesis. In addition, we published three technical papers as results of the Master's thesis. The following list provides references to these documents.

- Melo, V., Leão, S., Campos, M., Menotti, D., and Schwartz, W. (2013). *Fast pedestrian detection based on a Partial Least Squares Cascade*. In IEEE International Conference on Image Processing **(Qualis A1)** [12].
- Melo, V., Leão, S., and Schwartz, W. (2013). *Pedestrian Detection Optimization Based on Random Filtering*. In Workshop of Works in Progress (WIP) at Conference on Graphics, Patterns and Images (SIBGRAPI) [14].
- Melo, V., Leão, S., Menotti, D., and Schwartz, W. (2014). *An Optimized Sliding Window Approach to Pedestrian Detection*. In International Conference on Pattern Recognition **(Qualis A1)** [15].

An extension of the Master's thesis is currently being prepared for submission to a journal. In this extension, we propose an approach for generation of pedestrian detection locations, built on top of random filtering, in which we use a pedestrian detector to rank the windows selected by random filtering, and then we select a percentage of the high scoring ones to create regions more promising of containing pedestrians.

Acknowledgment

The authors would like to thank FAPEMIG, CAPES, and CNPq for the financial support.

References

[1] F. Porikli, F. Bremond, S. Dockstader, J. Ferryman, A. Hoogs, B. Lovell, S. Pankanti, B. Rinner, P. Tu, and P. Venetianer, "Video Surveillance: Past, Present, and Now the Future," *Signal Processing Magazine*, pp. 190–198, 2013.

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

[3] P. Dollar, C. Wojek, B. Schiele, and P. Perona, "Pedestrian Detection: An Evaluation of the State of the Art," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 34, pp. 743–761, 2012.

[4] R. Benenson, M. Mathias, R. Timofte, and L. Van Gool, "Pedestrian Detection at 100 Frames per Second," in *IEEE Intl. Conference on Computer Vision and Pattern Recognition*, 2012.

[5] M.-M. Cheng, J. Warrell, W.-Y. Lin, S. Zheng, V. Vineet, and N. Crook, "Efficient salient region detection with soft image abstraction," in *IEEE Intl. Conference on Computer Vision*, 2013, pp. 1–8.

[6] P. Viola and M. Jones, "Rapid Object Detection using a Boosted Cascade of Simple Features," in *IEEE Intl. Conference on Computer Vision and Pattern Recognition*, 2001.

[7] Q. Zhu, S. Avidan, M.-C. Yeh, and K.-T. Cheng, "Fast Human Detection using a Cascade of Histograms of Oriented Gradients," in *IEEE Intl. Conference on Computer Vision and Pattern Recognition*, 2006, pp. 1491–1498.

[8] W. Schwartz, A. Kembhavi, D. Harwood, and L. Davis, "Human Detection Using Partial Least Squares Analysis," in *IEEE Intl. Conference on Computer Vision*, 2009.

[9] B. Schölkopf and A. J. Smola, *Learning with kernels: support vector machines, regularization, optimization and beyond*. MIT press, 2002.

[10] W. R. Schwartz, V. H. C. de Melo, H. Pedrini, and L. S. Davis, "A Data-Driven Detection Optimization Framework," *Neurocomputing*, 2013.

[11] R. Rosipal and N. Kramer, "Overview and Recent Advances in Partial Least Squares," *Lecture Notes in Computer Science*, vol. 3940, pp. 34–51, 2006.

[12] V. H. C. Melo, S. Leao, M. Campos, D. Menotti, and W. R. Schwartz, "Fast pedestrian detection based on a partial least squares cascade," in *Image Processing (ICIP), 2013 20th IEEE International Conference on*, 2013, pp. 4146–4150.

[13] N. Dalal and B. Triggs, "Histograms of Oriented Gradients for Human Detection," in *IEEE Intl. Conference on Computer Vision and Pattern Recognition*, 2005, pp. 886–893.

[14] V. H. C. Melo, S. Leão, and W. R. Schwartz, "Pedestrian detection optimization based on random filtering," in *Workshop of Works in Progress (WIP) in SIBGRAPI (XXVI Conference on Graphics, Patterns and Images)*, August 2013, pp. 1–4.

[15] V. H. C. Melo, S. Leão, D. Menotti, and W. R. Schwartz, "An Optimized Sliding Window Approach to Pedestrian Detection," in *IAPR International Conference on Pattern Recognition*, 2014, pp. 4346–4351.