Pedestrian detection based on LIDAR-driven sliding window and relational parts-based detection

Luciano Oliveira and Urbano Nunes

***Abstract*— The most standard image object detectors are usually comprised of one or multiple feature extractors or classifiers within a sliding window framework. Nevertheless, this type of approach has demonstrated a very limited performance under datasets of cluttered scenes and real life situations. To tackle these issues, LIDAR space is exploited here in order to detect 2D objects in 3D space, avoiding all the inherent problems of regular sliding window techniques. Additionally, we propose a relational parts-based pedestrian detection in a probabilistic non-iid framework. With the proposed framework, we have achieved state-of-the-art performance in a pedestrian dataset gathered in a challenging urban scenario. The proposed system demonstrated superior performance in comparison with pure sliding-window-based image detectors.**

I. Introduction

Much research has been developed in the last decade with the goal of designing high-performance image-based object detectors. In fact, many approaches have been proposed to tackle this problem, ranging from pure sliding-window-based detectors [1], [2], [3], [4] up to stereoscopic methods [8], [9]. Although much has evolved all these years, perception systems for pedestrian detection is still an open problem, and perfection seems to be far.

Let us consider a very standard pedestrian detector, that is, that one formed by one or more feature extractors and/or one or more classifiers within a sliding-window framework, according to Fig. 1. By sliding fixed-size windows on several octaves of the input image (scales or resized version of the input image) in horizontal and vertical directions, it is possible to detect a previously trained object throughout many scales and orientations. Objects detected in more than one octave are commonly clustered by means of a non-maxima suppression method in order to have just one window per object in the final result image. In each window slid, a feature vector is gathered in order to lately feed a classifier that, after training, provides the estimation of being or not the object of interest. Having overlapped windows regulated by the size of the stride, all these steps represents, indeed, a brute force search with the aim of detecting objects in the image.

One of the first detection systems, based on Haar-like features, was proposed by Papageorgiou and Poggio [1]. That type of feature was used as an input into a support vector machine (SVM) classifier, reaching almost a perfect detection performance in a data set with very few images. Another earlier detection system was proposed by Wohler et al. [2], and was based on a time delay neural network (TDNN). Instead of inputting single feature vectors, TDNN is fed by a temporal sequence of raw frames, allowing the extraction of shapes in a trainable type of convolutional neural network. Viola and Jones [3], motivated by Papageorgiou and Poggio's system, improved the former system in terms of computational load, speeding up not only the way of extracting the Haar-like features, by means of an integral image, but also the classification of those features by using an adaboost classifier. Although, Haar-like features with adaboost classifier have become the very choice for standard vision systems in many applications, those features usually present quite a poor performance for pedestrian detection in cluttered scenarios. The reason for that is that Haar-like features often fail to capture object representation when the contrast between object and background presents just small variations. To cope with this limitation on discriminating small object-background contrast, Dalal and Triggs [4] pro-

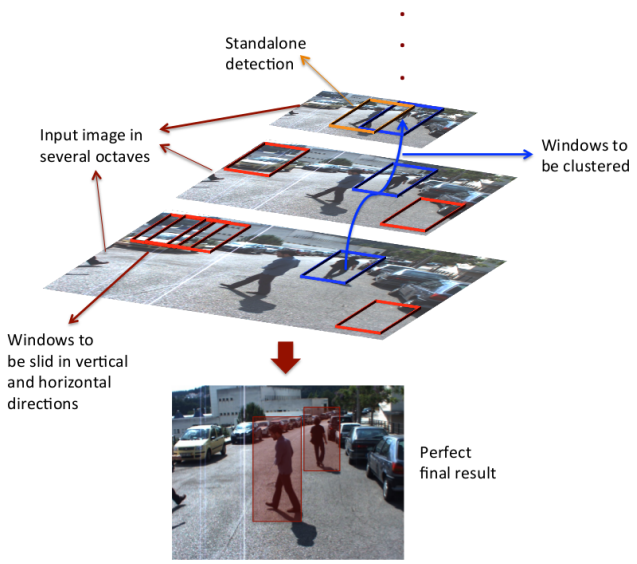


Fig. 1: The brute force characteristic of image sliding window. Given the input image, the input image is resized through several octaves (scales); in each octave, a fixed-size window is slid in horizontal and vertical direction. Parameters as stride and window size particularly influence the detection quality.

This work was partially supported by FCT-Portugal, under grant PTDC/EEA-AUT/113818/2009, and Fundação de Amparo a Pesquisa do Estado da Bahia (FAPESB-Brazil), under grant 6858/2011. Luciano Oliveira is with Intelligent Vision Research Lab, Federal University of Bahia, Brazil, and Urbano Nunes is with Institute of Systems and Robotics, Department of Electrical and Computer Engineering, University of Coimbra, Portugal {lrebouca@ufba.br, urbano@isr.uc.pt}.

posed the use of a histogram of oriented gradient (HOG) feature extractor based on the scale-invariant feature transform (SIFT) descriptor, proposed by Lowe [12], having this classified by an SVM. Dalal and Triggs' HOG descriptor is applied densely, in contrast to the SIFT descriptor, presenting superior performance in comparison to Haar-like features in many works [4], [6], [7].

Although many aspects of the pedestrian detection task were improved with each proposed method aforementioned, recently Dóllar et al [5] have showed that, for a huge number of frames (+250,000), HOG/SVM presents a poor performance in comparison with that one reported in [4]. It was explained not only by the biased analysis made by Dalal and Triggs [4] by assuming a false positive per image instead of false positive rate, but also the use of huge number of images to assess the performance of the classifier. Additionally, we could say that some aspects of that sliding-window method, such as the granularity of the stride, number of octaves and size of the sliding window might have influenced to the difference of performance between the two works mentioned.

Especially taking into consideration object detectors in perception frameworks for intelligent transportation systems (ITS), and many other areas, sliding-window approaches bring an additional drawback: many regions of the image which presents low probability of having an object must be thoroughly scrutinized all the time in search of an object. It is straightforward to notice that it should avoid for many systems to run on-the-fly. To overcome this problem, some methods aims to constraint image search by object saliency analysis [10], for example.

In our proposed method, instead of sliding windows in image space, windows are slid in LIDAR space, posteriorly backprojecting these windows into the image plane, considering proper sensor calibration. By considering an HOG/SVM detector (although it could be any other detector) for the upper and lower part of the person body, the final decision to be or not to be a pedestrian is made by a set of first-order rules, which will lately be evaluated by a Markov random field (MRF) [13]. While 2D windows are being slid in LIDAR space, it avoids two issues: to rescale images, and also the choice of finding a fine grain stride which ultimately can cause false alarm or miss detections, as they occur in a pure sliding window process. Yet, with our approach, we are also able to constraint the object search by carrying it out bellow the horizon line, that is, only in the more probable image regions, also avoiding the need of resizing the input image into several octaves.

II. The Proposed Approach

In the past, we have tried two novel approaches for pedestrian detection using only LIDAR, or LIDAR and vision: a standalone LIDAR detector based on a novel clustering strategy and using a template matching method [14], and a semantic fusion between a LIDAR and an ensemble of image detectors [15]. Figure 2 illustrates the timeline of the proposals, including the current one.

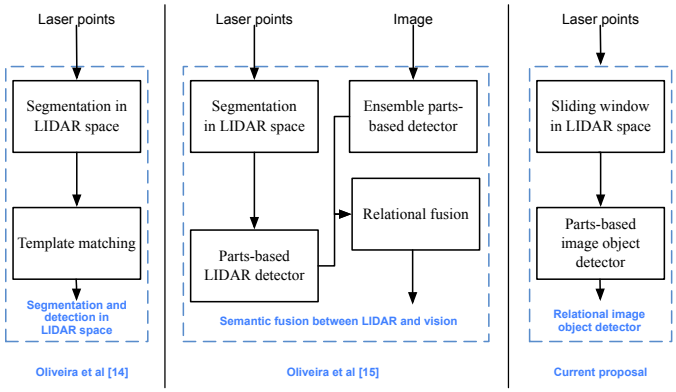


Fig. 2: Timeline of our proposals in pedestrian detection using LIDAR, LIDAR and vision, and sliding window in LIDAR space.

The earlier system reached 75% of hit rate (HR) at 0.51 of false alarm rate (FAR) over our gathered datasets. In the latter, with the same datasets, we were successful on increasing the HR by almost 18 percentage points with respect to a full image detector, keeping the same FAR. Also considering the datasets used in both past works, here, we are interested on investigating a new variant of LIDAR-vision integration: an image detector that relies on LIDAR geometry and relational integration among parts-based detectors. The rationale for our new system is not only to overcome the limitations presented in standard sliding-window-based image pedestrian detectors, but also to alleviate the computational cost of the semantic fusion system. In the next sections, details of our current proposed approach is given.

III. Using LIDAR space to sliding image windows

To explore the LIDAR space, a setup was prepared with the electric vehicle depicted in Fig. 3, called ISRobotCar, which was driven through our campus to gather the datasets. In the travelled trajectory, several pedestrians (actors) had been walking in front of the vehicle in several manners, individually or in groups.

The description of the datasets used can be found in [15]. Images are characterized by shadow covered areas, many degrees of object occlusion, illumination changes, many cars in the margins of the road, and just few frames without the presence of pedestrians. Each image was manually annotated to include all objects in the human field of view (hard annotation) only considering objects in the range of 2 up to 20 meters (information given by the LIDAR). A LIDAR-vision registration procedure, proposed by Zhang and Pless [16], was performed in order obtain an extrinsic calibration between the sensors (in LIDAR-camera direction). After sensor calibration, virtual windows with size of 1.0m $\times$ 1.8m were slid onto horizontal and vertical directions in LIDAR sensing space, with a stride of 0.20m, ranging over 2m up to 20m in front of the vehicle.

This LIDAR-driven sliding window approach brings some advantages: (i) it saves a lot of computational load in image

Fig. 3: ISRobotCar [11] equipped with a camera and a LIDAR to gather the datasets. Images were achieved with a special threaded acquisition as an attempt to guarantee a perfect sensor synchronization. Each sensor acquisition was carried out in a core of a dual-core.

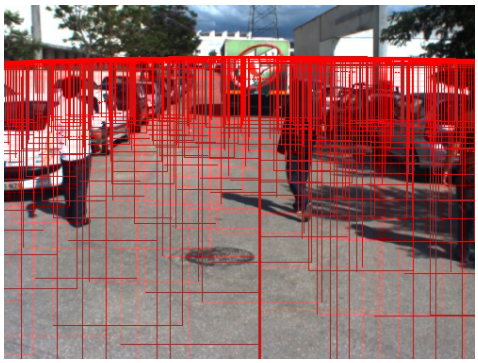


Fig. 4: Windows slid in LIDAR space and projected into the image plane.

resizing, (ii) it avoids to slide windows in the entire image (in practice, windows are slid constrained for the vanishing line), (iii) it also avoids typical problems in the sliding window procedure in image plane, like the granularity of window stride, which affects directly the number of false alarm and miss detection rate, (iv) it keeps the estimated distance from the vehicle for each window after the projection; particularly, the latter one brings benefits for an independent fusion of both sensors [15]. At the end, we had not only a time saving, but also a more accurate way of image object search.

Figure 4 illustrates the windows slid in LIDAR space, and backprojected into the image plane. The projected windows touch the ground and do not overcome the vanishing line, which is particular interesting for the detectors in the image plane to avoid regions with low probability of a pedestrian to be found. Our detector is performed then for each projected window and in only one step (without resizing). Still, each image window is normalized to the size of 54×108 px, and multiple windows are pruned by a regular non-maxima suppression procedure. Next, details of the proposed relational parts-based detector is given.

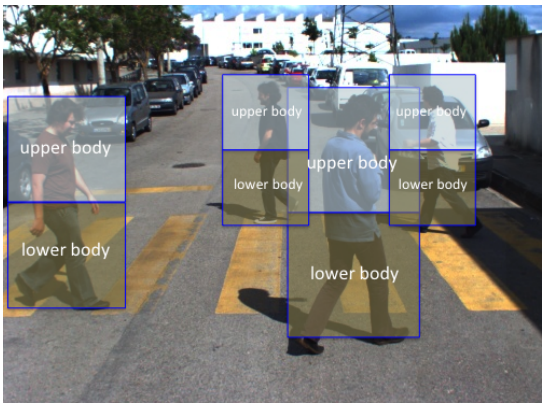


Fig. 5: Parts-based detector. On each person body part (upper and body), an HOG/SVM classifier is performed.

IV. Object-centered contextual detection

On each image window, after size normalization, an histogram of oriented gradient feature (or any other type of feature) vector is extracted for two image body parts. Each part has then a size of 54×54 px. Each HOG block follows the configuration proposed by Dalal and Triggs concerning the number of cells, bins, sizes of the cells and the block [4]. Figure 5 illustrates the parts-based object window.

After extracting the same feature pattern in the training dataset, two support vector machines (SVM) were trained. Considering the separating hyperplane of an SVM as $f(X) = sign(< W \cdot X > + b)$, where $X \in \mathbb{R}$ is the input vector, $W \in \mathbb{R}^N$ is a weight vector and $b \in \mathbb{R}$ is the bias component that adjusts the hyperplane $f(X)$ to better separate X. The confidence score, s_c, which are obtained by the Euclidean distance between the input vector (feature vector) and the separating hyperplane $f(x)$, is transformed into a probabilistic score, $P(s_c)$, following the approach in [17], such that

$$P(s_c) = \frac{1}{1 + exp(-\varrho_1 s_c + \varrho_2)}, \quad (1)$$

where ϱ_1 and ϱ_2 are usually obtained in the training process (in practice, ϱ_1 usually approaches to 1, while ϱ_2 approaches to 0); $s_c \in [0, \infty]$, and $P(s_c) \in [0, 1]$.

The probabilistic score provides the necessary normalized information about the classification confidence, which will be incorporated in the first-order logic (FOL) rules.

The first idea of markov logic network (MLN) [13] is to soften the pure FOL constraints, that is, when a world violates a formula, it becomes less probable, but not impossible. To this end, each formula in a knowledge base (KB) has a weight which represents how strong a formula is for a world. The higher the weight is, the bigger the probability for a world to be satisfied. The main goal of MLN is thus to unify the fundamental advantages of FOL and MRF, dealing at the same time with complexity and uncertainty (see the Appendix for more details about MLN foundations).

Table I summarizes the FOL rules used in the the last stage inference. The rationale of these rules is to deal

TABLE I: First-order formulas used in the last stage object inference.

No.	First-order logic	Description
1	$\forall$p, Person(p)	Query over pedestrian candidate p
2	$\forall$p, UpperHighScore(p)	Upper body of pedestrian candidate p with confidence score greater than 0.5
3	$\forall$p, LowerHighScore(p)	Lower body of pedestrian candidate p with confidence score greater than 0.5
4	$\forall$p, View(p,b)	Pedestrian p with a body part b
5	$\forall$p, View(p,"upper") $\wedge$ View(p,"lower") $\wedge$ UpperHighScore(p) $\wedge$ LowerHighScore(p) $\Rightarrow$ Person(p)	If an upper or a lower part of a pedestrian candidate p is viewed with a high score, it is a pedestrian
6	$\forall$p, View(p,"upper") $\wedge$ UpperHighScore(p) $\Rightarrow$ Person(p)	If only an upper part of a pedestrian candidate p is viewed with a high score, it is a pedestrian
7	$\forall$p, View(p,"lower") $\wedge$ LowerHighScore(p) $\Rightarrow$ Person(p)	If only a lower part of a pedestrian candidate p is viewed with a high score, it is a pedestrian
8	$\forall$p, !View(p,"upper") $\wedge$!View(p,"lower") $\Rightarrow$!Person(p)	If any of the body parts is not seen, then it is not a pedestrian

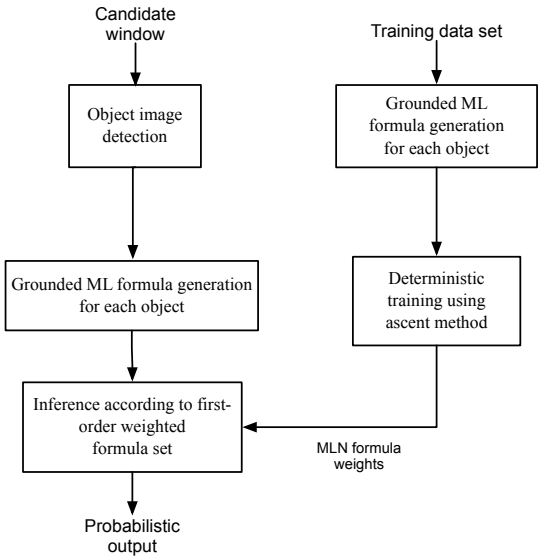
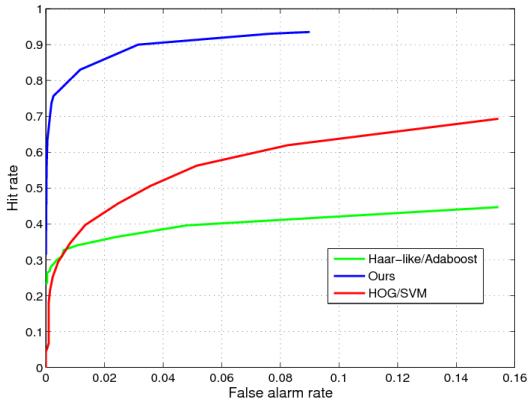


Fig. 6: Contextual detection. Either in the training or in the inference stages, a set of grounded ML formulas must be generated according to the FOL formulas in Table I. After training, weights are given to each one of the FOL formulas in order to perform inference over the grounded formulas generated by the rules. At the end, a probabilistic output is provided for each pedestrian candidate (projected LIDAR windows).



(a) ROC curves showing comparative results between Haar-like/Adaboost, HOG/SVM and our detector.

Detector	Hit rate (HR)
Ours	86%
HOG/SVM	42%
Haar-like/Adaboost	35%

(b) Operating points of the detectors in ROC curves, at FAR = 0.02

Fig. 7: Experimental evaluation over the datasets.

with not only the uncertainty of the HOG/SVM (or any other), but also occlusion situations defined by low expected probabilistic confidence score of the upper or lower image part. A pedestrian candidate p is represented by a unique sequential number (generated in grounded ML formulas) as objects are detected.

As mentioned before, these formulas are structured as a grounded MLN. Formulas 1 to 4 represents the definitions of the predicates used in formulas 5 to 8. Formula 5 treats the case of a whole view in pedestrian candidate, while formulas 6 and 7 attempts to deal with occlusion. Formula 8 defines the case that the candidate p is not a pedestrian. It is noteworthy that after training each formula will have weights in each predicated, which will influence the final inference. For each example to be classified in the testing dataset, a final score will be given, providing the information to be or not to be a pedestrian. For each example in the training stage, the predicates will be generate according to its specific meaning in Table I. Figure 6 summarizes the framework of the contextual detection, centered in the object.

Indeed, the proposed formulation to the problem of pedestrian detection (or any other object) brings twofold advantages: (i) it treats occlusion situations as a matter of learning from the data, since FOL rules will receive weights which will define how occlusion is being seen in practice, (ii) the whole framework is flexible enough to have plug-ins of any other detector (the more accurate it is, the better), providing the proposed method to span a great spectrum of applications. Next, experimental analysis is made in order to verify system performance.

V. Experimental Evaluation

To assess the performance of the proposed method, the vehicle equipped with a camera and a LIDAR (see Fig. 3) was driven through our campus. A special software was designed in order to guarantee sensor synchronization with parallelism in sensor data acquisition (as mentioned in Section III. A dataset with 2,157 frames was collected to

test the system, while 3,333 frames were annotated and used for training, with images gathered in a different scenario. The training data set was used either for the HOG/SVM parts-based detector or to train the FOL rules for MLN last stage detection. Particularly, the FOL rules were generated after training the HOG/SVM parts-based detector, since they would be useful in the definition of the rules. For instance, only after performing HOG/SVM in the training data set, we were able to establish the weights for the formulas (seen in Table I), like "UpperHighScore", "LowerHighScore", and so forth.

Receiver operating characteristic (ROC) curves were built in order to show the comparative performance of our detector in contrast to HOG/SVM and Haar-like/Adaboost [3] standard detectors. Figure 7a shows the comparative results. Considering a false alarm rate (FAR) of 2%, we had an HR of approximately 86% with our method, 42% with HOG/SVM and 35% with Haar-like/Adaboot. The operating points of all detectors, at FAR = 2%, are summarized in Table 7b.

In fact, it was experimentally demonstrated that there was a significant raise in the number of pedestrian hits with the idea of object-centered contextual detector, while keeping the false alarm rate in a very low and safe condition. The higher value of hit rate (HR), in our method, can be explainable by the lots of occlusion situation presented in the data set, which, in turn, is not detected by standard sliding-window based detectors. On the other hand, the idea of exploring LIDAR space helped with avoiding many false alarms above the vanishing line, which aided our proposed detector. One fact that drew attention was the performance of Haar-like/Adaboost, which was very poor. One possible answer could reside in the low contrast between people and background, in many situations.

Figure 8 shows examples of our detector in practice in a relatively crowded scene with a high level of occlusion and successful detections, in the first row; in the last row, a very high occlusion and examples of detection fails.

VI. Conclusion

In this paper, a new paradigm of detection was presented. For that, LIDAR space was exploited in order to avoid the rescaling process of regular sliding window method. Yet, an object-centered contextual detector was designed with a relational parts-based detector, and modelled by means of an MLN. This latter aided the detector to deal with occlusion, raising the performance of the detector in a challenging dataset. This latter benefit was clear, specially in comparison with two monolithic sliding-window-based detectors.

The proposed approach brought fourfold advantages: (i) it saves a lot of computational load of image rescaling, as it happens in regular sliding window, (ii) it implicitly constraints the object search below the vanishing line and the ground, (iii) it also avoids typical problems in the sliding window procedure in image plane, like the definition of window stride, which affects directly the number of false alarm and miss detection rate, and, finally, (iv) it keeps the estimated distance from the vehicle for each window after the projection.

For future work, we are investigating how to incorporate multiple detections (multiple objects of interest) inside MLN without increasing the number of detection passes.

Appendix

An MRF is a joint distribution of a set of variables $X = \{x_n\}_{n=1}^{N}$, written as

$$P(X = x) = \frac{1}{Z} \prod_{C} \phi_C(x_C)\,, \tag{2}$$

where $\phi_C(x_c)$ is a potential function over the cliques of the graph, and Z is a normalization constant, given by

$$Z = \sum_{x} \prod \phi_C(x_C)\,. \tag{3}$$

The choice of potential functions is not restricted to those that have a specific probabilistic interpretation as marginal or conditional distributions, since the partition constant Z can be used to normalize $P(X = x)$, appropriately. Considering those initial ideas, the main definition of an MLN can be given as

Definition 1. A first-order MLN is a set of pairs $(\mathcal{Q}_i, w_i)$, where $\mathcal{Q}_i$ is a formula in FOL and w_i is a real-number weight. Each $\mathcal{Q}_i$ is a node of a MRF.

Given different constants, a first-order MLN will produce different MRFs, which are called ground MLNs. Each ground MLN varies in size but keeps regularities in structure and parameters, given by the first-order MLN. Rather than defining the ground MLN as the form (2), and because we are restricted to potential functions which are strictly positive, it is more convenient to write it as a Boltzmann distribution (exponential representation), such that

$$P(X = x) = \frac{1}{Z} exp\left(\sum_{i} w_i \eta_i(x_i) \right)\,, \tag{4}$$

where Z is now equal to $\sum_{x \in X} exp(\sum_i w_i \eta_i(x_i))$, $\eta_i(x)$ is the number of true groundings of $\mathcal{Q}_i$ in x_i, which is, in turn, the state of the ith atom in $\mathcal{Q}_i$.

The weights of an MLN can be learnt or hand-crafted. Weight learning can be performed generative or discriminatively. In our case, we have used a discriminative approach, based on a voted perceptron weighted satisfiability solver, which is demonstrated to outperform generative approaches [18]. Discriminative learning is performed by using ascent method over the gradients of the conditional log-likelihood, given by

$$\frac{\partial}{\partial w_i} \log P_w(\psi|\varepsilon) = n_i(\psi, \varepsilon) - E_w[n_i(\psi, \varepsilon)]\,, \tag{5}$$

where ψ is a query predicate, ε is an evidence predicate, $E_w[n_i(\psi, \varepsilon)]$ is the expectation over the number of true groundings of formula i according to the MLN, approximated by a maximum a posteriori (MAP) inference. A

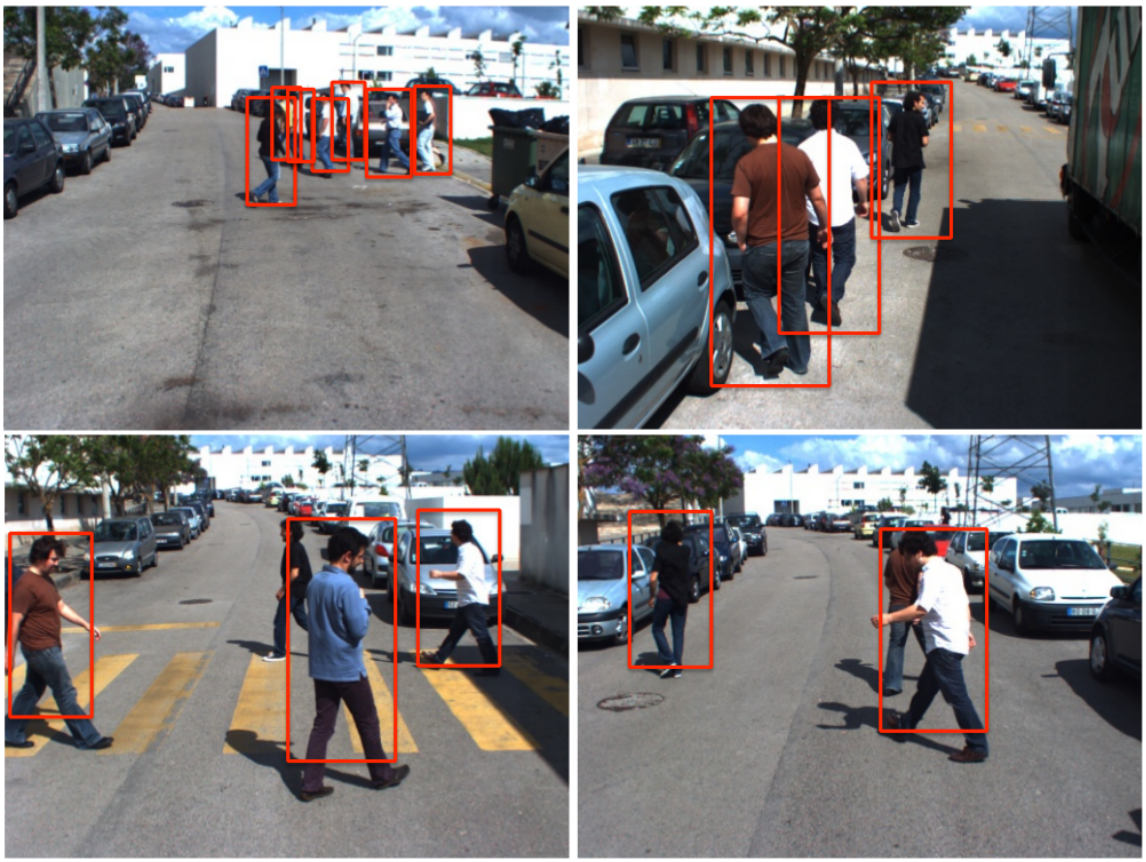


Fig. 8: Examples of the detection in practice: first row, a relatively crowded scene with a high level of occlusion and perfect results; last row, very high occlusion with fails.

training data set was used to learn the weights, w_i, of our MLN.

After training an MLN, inference is performed by a combination of Markov chain Monte Carlo (MCMC) and SampleSAT algorithms, called MC-SAT [19].

REFERENCES

[1] C. Papageorgiou, and T. Poggio, *A trainable system for object detection*, in International Journal of Computer Vision, pp. 15–33, 2000.

[2] C. Wohler, J. Aulanf, T. Portner and U. Franke, *A time delay neural network algorithm for real-time pedestrian recognition*, in International Conference on Intelligent Vehicles, pp. 247–252, 1998.

[3] P. Viola and M. Jones, *Rapid object detection using a boosted cascade*, in IEEE International Conference on Computer Vision and Pattern Recognition, pp. 511-518, 2001.

[4] N. Dalal and B. Triggs, *Histograms of oriented gradients for human detection*, in IEEE International Conference on Computer Vision and Pattern Recognition, pp. 886-893, 2005.

[5] P. Dollar, C. Wojek, B. Schiele and P. Perona, *Pedestrian detection: A benchmark*, in IEEE International Conference on Computer Vision and Pattern Recognition, pp. 304–311, 2009.

[6] L. Oliveira, G. Monteiro, P. Peixoto and U. Nunes, *Towards a Robust Vision-Based Obstacle Perception with Classifier Fusion in Cybercars*, in Computer Aided System Theory - EUROCAST, LNCS, vol. 4739, pp.1089–1096, 2007.

[7] H-X. Jia and Y-J. Zhang, *Fast Human Detection by Boosting Histograms of Oriented Gradients*, in International Conference on mage and Graphics, pp. 683-688, 2007.

[8] B. Leibe, N. Cornelis, K. Cornelis and L. Van Gool, *Dynamic 3D Scene Analysis from a Moving Vehicle*, in IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–8, 2007.

[9] G. Grubb, A. Zelinsky, L. Nilsson and M. Rilbe, *3D vision sensing for improved pedestrian safety*, in IEEE Intelligent Vehicles Symposium, pp. 19–24, 2004.

[10] G. Silva, L. Schnitman and L. Oliveira, *Multi-scale spectral residual analysis to speed up image object detection*, Conference on Graphics, Pattern and Image (SIBGRAPI), 2012.

[11] M. Silva, F. Moita, U. Nunes, L. Garrote, H. Faria and J.P. Ruivo, *ISRobotCar: The Autonomous Electric Vehicle Project*, in IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2012.

[12] D. Lowe, *Object recognition from local scale-invariant features*, in International Conference on Computer Vision, pp. 1150–1157, 1999.

[13] M. Richardson and P. Domingos, *Markov Logic Networks*, in Machine Learning, vol. 62, pp. 107–136, 2006.

[14] L. Oliveira and U. Nunes, *Context-aware pedestrian detection using LIDAR*, in IEEE Intelligent Vehicles Symposium, pp. 773–778, 2010.

[15] L. Oliveira; U. Nunes; P. Peixoto; M. Silva and F. Moita, *Semantic Fusion of Laser and Vision in Pedestrian Detection*, in Pattern Recognition, pp. 3648-3659, vol. 43, issue 10, 2010.

[16] Q. Zhang and R. Pless, *Extrinsic calibration of a camera and laser range finder (improves camera calibration)*, in IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 2301–2306, 2004.

[17] J. Platt, *Probabilistic outputs for support vector machines and comparison to regularize likelihood methods*, in Advances in Large Margin Classifiers, eds. A. Smola, P. Bartlett, B. Schoelkopf and D. Schuurmans, pp. 61–74, 2000.

[18] S. Parag and P. Domingos, *Discriminative training of Markov logic networks*, in National Conference on Artificial Intelligence, pp. 868–873, 2005.

[19] H. Poon and P. Domingos, *Sound and efficient inference with probabilistic and deterministic dependencies*, in National Conference on Artificial Intelligence, pp. 458–463, 2006.