Vehicle Detection using Mixture of Deformable Parts Models: Static and Dynamic Camera

Leissi Castañeda Leon, Roberto Hirata Jr.
Institute of Mathematics and Statistics
University of São Paulo (USP)
São Paulo, Brazil
{leissicl,hirata}@ime.usp.br

***Abstract*—Vehicle detection in video is an important problem in Computer Vision because of the potential applications in security, vehicle traffic, driving assistance and so on. In this work, we used Mixture of Deformable Part Models (MDPM) for vehicle detection in video sequences obtained from static and dynamic cameras. The MDPM method was originally proposed by Felzenszwalb et al in the realm of object detection in images. We tested this method in the realm of video sequences for vehicle detection. We designed a set of experiments that explore the number of components of the mixture and the number of parts model. We performed a comparison study of symmetric and asymmetric MDPMs for vehicle detection. Our findings show that not only the MDPM performed well in vehicle detection in video, but also the best number of components and parts model confirmed the number suggested in Felzenzwalb et al's paper. Finally, the results show some differences between the symmetric and asymmetric MDPMs in vehicle video detection considering different scenarios.**

***Keywords*-Mixture of deformable part models; vehicle detection.**

I. Introduction

Vehicle detection is an important topic of research because of the several challenges yet to be solved, such as the capability of detect vehicles in movement, or not, in respect to a dynamic or static camera. Also, it is part of several real life applications like driving assistance, security and so on. For each application, a different and specific solution is usually expected and applied. This is why the state of the art in vehicle detection refers to solutions in specific domains.

The literature reports basically two groups of approaches: (i) model (of a vehicle) based approach, that is initialized a priori [1], [2], [3], [4], [5], [6]; and (ii) the approaches that only use information of video sequences, for instance: background/foreground subtraction or extraction [7], [8], [9], movement estimation [9], point features [10], [11], Markov random field [9] and etc. Both groups of approaches can use 2D or 3D information [12], [13], [14], [15].

Model based approaches are usually less subject to class variability, pose, illumination, occlusion and background variation [16]. The second group of approaches usually is not robust to deal with the challenges mentioned above. A recent model based method is the Mixture of Deformable Part Models (MDPM) [5]. It was introduced in the context of object detection in images and it was successfully used in the PASCAL challenge 2008 [17].

In this work, vehicle detection in videos is done by performing a matching between a vehicle model based on MDPM to the features collected, for each frame of the video, by sliding windows of Histogram Oriented Gradient (HOG). The principal contributions of this work are: the application of the MDPM method in the realm of video sequences, the design of a set of experiments that explore the number of components and part models in symmetric and asymmetric MDPMs,and a comparison study of performance of symmetric and asymmetric MDPMs for vehicle detection in video sequences obtained by static and dynamic cameras. Finally, considering that the MDPM is built using static images from a image dataset, the experiments show a considerable diversity of scenes in which it can be successfully applied.

The following sections discuss related work and give a general overview of the proposed method. In Section II, an overview of the MDPM is given. An application of MDPM for vehicle detection in video is presented in Section III. An experimental evaluation to explore the number of MDPMs parameters (mixture components and parts model) and an evaluation of symmetric and asymmetric MDPMs is discussed in Section IV. Finally, the directions for future work is given in the Conclusion (Section V).

A. Related work

Different methods for vehicle detection have been reported in the literature [16]. Recent literature reports that appearance-based models achieve good performance in challenging scenarios [18]. These models include edgelets [19], strip features [6], HOG features [20], [5], [21] and statistical learning of object parts [22]. A review of recent vision-based on-road vehicle detection systems, where the camera is mounted on the vehicle is presented in [23].

Object detection has made successfully use of methods based on HOG and support vector machines (SVM), introduced by Dalal and Triggs in [20]. One modification of HOG feature extraction method to treat with sensitive and insensitive features and the Latent SVM (LSVM) classifier used with the symmetric MDPM was proposed in [5]. The asymmetric MDPM case is described in [21].

The Deformable Part Model (DPM) presented by Felzenszwalb et al. [5], [21] demonstrates a state-of-the-art performance on difficult object detection benchmarks. Variants of it

have been proposed, such as [24], [25], [19] where a technique has been proposed to modeling the flexibility of objects such as people, cats and dogs. In some cases, the results obtained do not present better performance than the results of the state-of-the-art MDPM.

In respect to the datasets, different sets have been proposed for learning and evaluation of vehicle detection algorithms in image dataset. Two examples of such datasets that have images of cars as an object category are: the PASCAL VOC challenge [17] and Caltech [26]. One can also find datasets of vehicles images [18] and videos [27] obtained from urban surveillance or urban traffic. Finally, it is possible to find datasets of vehicles videos on the road, considering a dynamic camera [28]. It is denominated as dynamic camera because of the video sequences are obtained from a camera equipped in a vehicle in movement.

B. Technique overview

The approach has two parts: a training and a testing phase. The training phase refers to the construction of a vehicle model using the algorithm proposed in [5] and [21] and an image dataset as data to train it. The vehicle model is the input data to the testing phase, where the method performs a multi-scale vehicle detection based on the sliding window technique over each video frame.

II. Mixture of Deformable Part Models overview

In this section, we shortly review the algorithm used to construct MDPMs for the vehicle object. The algorithm basically consists of a feature extraction step, a model definition and a classifier induction. One can find a more complete description in [5], [21].

A. Feature extraction

The feature extraction refers to the 31 HOG features which are obtained from each image/frame by computing HOGs in blocks of 8×8 pixels. The HOG features include 9 bins for contrast insensitive characteristics, 18 bins for contrast sensitive characteristics and 4 texture gradients.

A map of characteristics is defined by the 31 HOG features computed over a image/frame. A *filter* is a rectangular template in a map of characteristics defined by an array of the 31 HOG features. Two filters, named *root* and *part*, are computed on a map of characteristics pyramid H. The root filter refers to a coarse representation of a vehicle and the part filter refers to a finer representation because it is obtained at twice the resolution of the root filter.

B. MDPM definition

A vehicle class is modeled by a MDPM, where the mixture is defined by m components $(M_1, ..., M_m)$. The model for the c-th component (M_c) is based in a star model of a pictorial structure and it is defined by linear filters (root and parts), by a set of permitted localizations to each part in respect to the root and by a cost of deformation to each part. Formally, it is a $n+2$-tuple as defined in 1,

$$M_c = \{F_0, (F_1, v_1, d_1), ..., (F_1, v_1, d_1), b\}, \quad (1)$$

where F_0 is the root filter, n is the number of parts and b is a real valued bias term, necessary to make a component comparable in a MDPM. Each part model is defined by a 3-tuple (F_i, v_i, d_i), where F_i is the i-th part filter, v_i is a bi-dimensional vector which specified the fixed position for part i relative to the root position, and d_i is a four dimensional vector which specify the coefficients of a quadratic function defining a deformation cost for each placement of the part i relative to v_i.

An object hypothesis h_{obj} specifies a c-th mixture component as well as the locations of both the root and part filters in the feature pyramid. Formally, $h_{obj} = (c, p_0, ..., p_n)$, where p_i encodes the 2D position and the level in the pyramid H for the filter i.

The score of a hypothesis, $score(h_{obj})$, is given by the scores of the filters at their locations minus a deformation cost that depends on the relative position of each part with respect to the root filter, plus the bias, as is shown in 2.

$$score(h_{obj}) = \sum_{i=0}^{n} F_i'.\phi(H, p_i) - \sum_{i=1}^{n} d_i.\phi_d(dx_i, dy_i) + b, \quad (2)$$

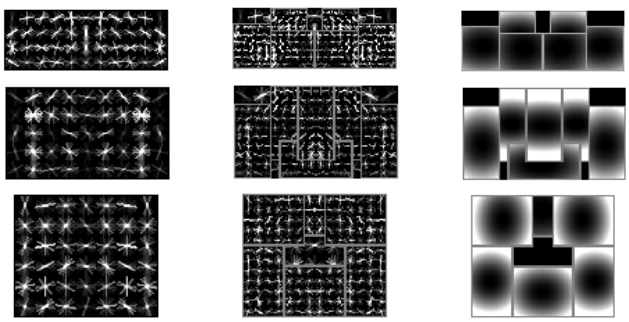
where $\phi(H, p_i)$ is a sub-window in the space-scale pyramid H with the upper left corner in p_i, $(dx_i, dy_i) = (x_i, y_i) - (2(x_0, y_0) + v_i)$ and $\phi_d(dx_i, dy_i) = (dx, dy, dx^2, dy^2)$ are deformation features.

C. Classifier induction

The training data consists of images with labeled bounding boxes, where a positive example defines a vehicle object and a negative example is created using random sub-windows of the images with non vehicle objects. The classifier needs to learn the model structure, filters and deformations costs. To the learning process, a latent SVM formulation is used, where the latent variables are the exact object location in the positive examples.

Training symmetric MDPM: Training begins by splitting the positive examples into nc groups of identical size. This procedure is complex because each group contains examples grouped based on their bounding boxes aspect ratio. Initialize the root filter by resizing the positive examples to the average aspect ratio dimension. Obtain the HOG features and train the filter using a linear SVM. Combine the root filters into a mixture without parts and train the model parameters. In this case, the latent variable is the root filter localization. Next, the filter parts are initialized to each component using a simple heuristic and a greedy process. Finally, the mixture is updated by training it with new data and using hard negatives examples with a cache.

Fig. 1 shows a symmetric MDPM of 3 components and 6 model parts. Each row represents a component, the first column shows the root filter (in coarse resolution), the second column shows the part filters (in finer resolution) and the third column shows the spatial model for each part, that is black center is a least cost.



(a) Root filter (b) Part filters (c) Spatial model

Fig. 1: Symmetric MDPM of 3 components and 6 model parts.

Training asymmetric MDPM: The training phase here is similar to the one exposed before. The difference is that, for each positive example in a given group, another example is generated by flipping it vertically. Each group is now clustered in two subgroups: left view and right view. A cropping and resizing process is applied to each image before obtaining the HOG features and the clustering.

The clustering algorithm is a variant of the know k-means with the restriction that an example and its flipped counterpart are not in the same cluster. Randomly and repeatedly, an example and its flipped counterpart are selected and assigned to a cluster. After all images have been assigned to a cluster, a local search method is used to improve the clustering.

This local search method is made by repeatedly select an example and its counterpart and check if swapping their clusters can reduce the total sum of squared distances (SSD) from the examples to their assigned center of the cluster. Then, the next steps are similar to the used in symmetric mixtures.

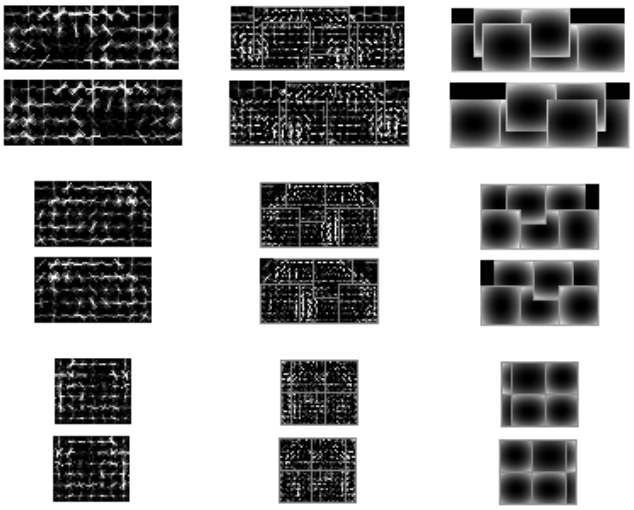
Fig. 2 shows an asymmetric MDPM of 3 components and 6 model parts. But the final asymmetric MDPM has 6 components because of the bilateral asymmetry (flipped counterpart) that allows each component to represent left or right vehicle poses.

III. MDPM TO VEHICLE DETECTION

The process defined here is to construct the vehicle MDPMs and detect vehicles objects, i.e., testing phase, in a video sequence. The first part was obtained following the process described in the previous section using the annotated image database. We then use a sliding window technique to detect vehicles objects over each video frame with the MDPMs. The system detects vehicles computing a score for each MDPM and applying a threshold, that was obtained in the training phase, to the scores obtained in the matching process for each video frame.

The matching process refers to defining an overall score for each root location based on best placement of parts. That is, a high score root locations defined detections. Formally, it is defined as:

$$score(p_0) = \max_{p_1,\ldots,p_n} score(h_{obj}). \tag{3}$$



(a) Root filter (b) Part filters (c) Spatial model

Fig. 2: Asymmetric MDPM of 3 components and 6 model parts.

Finally, a post-processing is applied to delete some bounding boxes that are overlapping. Basically, we delete a bounding box if the overlapping area, a_o, of two bounding boxes B_1 and B_2 is larger than, or equal, to 0.5 i.e.:

$$a_o = \frac{\text{area}(B_1 \cap B_2)}{\text{area}(B_1 \cup B_2)} \geq 0.5. \tag{4}$$

The principal problem is to find the number of mixture components and the number of model parts of a MDPM.

The symmetric case, as well as the use of two components with six model parts, was suggested in [21], a three mixture components with eight model parts for the asymmetric case were used. It is worth to note that there is no explanation for these choices. We present experiments in the Section IV-A that show the number of components and parts which can be used.

IV. EXPERIMENTAL EVALUATION

To validate our approach using MDPMs, a series of experiments have been done in image and video datasets. The process is implement and executed in MATLAB R2009b and C++ in a GNU/Linux operational system.

A. MDPM: number of components and parts model

We proceeded to determined the parameters of the MDPMs by model selection. For this, we defined $MC = \{2, 3, 4\}$ to be the number of mixture components, and $MP = \{4, 6, 8, 10\}$ to be the number of model parts. We then build a model for each combination ($MC \times MP$) and tested this model in the test set of the image dataset.

Dataset: The MDPM is created using the PASCAL VOC 2007 dataset [29]. This dataset is composed of a set of images, a set of annotations (bounding boxes) and, in addition, standard procedures for evaluation. The dataset is partitioned in 50% to train/validation and 50% to test. The PASCAL VOC 2007 dataset (classification/detection task) for the class car has

713 images (1250 objects to train/validate) and 721 images (1201 objects) to test.

Metric: The metric used to evaluate object detection is based on the Average Precision (AP) of its precision-recall curve across a test set. The precision is given by the fraction of predicted bounding boxes that are correct detections and recall is the fraction of detections obtained. The objective is to have a high recall and precision values, that is, values close to 1 in the range of $[0, 1]$. Therefore, in practice, for each image, the predicted bounding boxes are computed and also a confidence value for the prediction. The predicted bounding boxes are ordered in respect to the confidence value in a decreasing order and the precision and recall values are calculated for the first test image, then for the first two test images and so on until they are calculated for the entire test set. Finally, the AP is obtained by 5 where $pr_{interp}(\tilde{r})$ is an interpolated precision that takes the maximum precision over all recalls greater than r.

$$AP = \frac{1}{11} \sum_{r \in 0,.1,.2,...,1} \left[\max_{\tilde{r}:\tilde{r} \geq r} \quad pr_{interp}(\tilde{r}) \right]. \qquad (5)$$

A correct detection is a true positive if the overlap area (a_s) between the predicted bounding box (B_p) and the annotated ground truth (B_{gt}) is more than 50%, as defined in 4, otherwise it is a false positive. True positive detections can end up being false positive detections if multiple detections overlap with the ground truth. In this case, only one of them, the one with the best confidence, is considered a true positive, the others will be considered false positives.

For our purposes, two objectives are defined in our experiments: the first is only to apply a detection process (Base) and the second one is to apply a bounding boxes prediction process (BB); both are defined in [5].

Results: Table I shows the results obtained for the symmetric and asymmetric MDPMs. The titles are self described, for instance, 2M 4P means 2 components of the mixture and 4 parts of the model. We also have as headers, symmetric and asymmetric MDPMs, precision, recall and AP for each objective defined. The overall APs best results were obtained using a mixture of 3 components and using 6, 8 and 10 parts which are in bold in the table. We select these three best model parameters and use them in the detections in the video datasets.

We also compare our best results obtained using PASCAL VOC 2007 test set in respect to the obtained by the original work [5], [21] and by the different techniques in the literature such [30], [31], [32], [33] in the same test set. This is shown in Table II where we observe that the technique outperforms other approaches on this kind of data and an equivalent result that was obtained in the original work.

TABLE I: Precision, recall, AP obtained modifying the symmetric and the asymmetric MDPMs parameters.

		Base			BB		
	MDPM's	**Prec**	**Recall**	**AP**	**Prec**	**Recall**	**AP**
Symmetric	**2M 4P**	0.024	0.668	0.449	0.024	0.669	0.482
	2M 6P	0.023	0.664	0.464	0.024	0.679	0.502
	2M 8P	0.020	0.671	0.471	0.020	0.674	0.495
	2M 10P	0.022	0.651	0.463	0.023	0.667	0.488
	3M 4P	0.020	**0.677**	0.444	0.021	**0.689**	0.468
	3M 6P	0.028	0.657	0.462	0.028	0.670	0.486
	3M 8P	0.017	**0.678**	**0.481**	0.018	**0.690**	**0.503**
	3M 10P	0.019	**0.684**	**0.490**	0.020	**0.702**	**0.511**
	4M 4P	**0.062**	.637	0.458	**0.063**	0.641	0.486
	4M 6P	**0.061**	.629	0.463	**0.063**	0.646	0.485
	4M 8P	**0.051**	.655	0.473	**0.052**	0.667	0.497
	4M 10P	0.043	0.655	**0.487**	0.043	0.666	**0.505**
Asymmetric	**1M 4P**	0.020	0.700	0.496	0.021	0.681	0.517
	1M 6P	0.025	0.701	0.525	0.027	0.685	0.544
	1M 8P	0.022	0.702	0.515	0.023	0.692	0.545
	1M 10P	0.022	0.704	0.519	0.023	0.687	0.543
	2M 4P	0.020	0.712	0.521	0.022	0.700	0.546
	2M 6P	0.029	0.720	0.538	0.030	0.709	0.568
	2M 8P	0.032	0.722	0.537	0.034	0.711	0.564
	2M 10P	**0.035**	0.714	0.537	**0.036**	0.699	0.559
	3M 4P	0.028	**0.735**	0.534	0.029	**0.724**	**0.572**
	3M 6P	0.033	**0.732**	**0.548**	0.035	**0.714**	**0.579**
	3M 8P	**0.035**	**0.731**	**0.549**	**0.037**	**0.719**	**0.583**
	3M 10P	**0.048**	**0.732**	**0.545**	**0.050**	0.710	0.568
	4M 4P	0.033	0.719	0.539	0.035	0.709	0.571

TABLE II: Comparison of the best AP obtained using PASCAL VOC 2007 test set.

Method	**AP**
Symmetric MDPM	0.511
Asymmetric MDPM	**0.583**
UofCTTIUCI [5]	0.516
UoCTTI [21]	**0.596**
MKL [30]	0.506
Active Masks [31]	0.540
NUS-Context [32]	0.560
Superposition potential (SP) [33]	0.539
SP+HOG-bundle potentials [33]	0.529

B. Vehicle detection considering a static camera

The second experiment tests the method on a video dataset which were obtained considering a static camera. Symmetric and asymmetric MDPMs of 3 components and 6, 8 and 10 parts are tested in the experiment. Two video datasets that represent different traffic situations have been selected and the performance of the MDPMs was analysed to detect vehicles in that situations. Because we do not have the ground truth of each vehicle in the videos sequences, we show the results obtained and visually analyse the results.

Dataset 1: The first dataset represents a highway traffic situation [27]. The frame resolution is 320 x 240 pixels. The dataset was created with the objective of classify the traffic as: *heavy*, *medium* and *light*, based on the time or number of frames in which a specific car appears. The dataset has been acquired in different weather and times of the day such as: normal weather, clear and/or sunny day and in a rainy day.

The dataset consists of nine different videos which are detailed in Table III. They represent the three different classes in three different situations. For the medium and heavy traffic class in a rainy day situation, the camera has drops on the lenses In addition, the area of the road where most of the

TABLE III: The videos from traffic used on the experiments results.

Class	ID	Filename	Weather	Frames
Ligth	V1	$cctv052x2004080516x01640$	overcast	48
	V2	$cctv052x2004080613x00015$	clear	53
	V3	$cctv052x2004080606x01821$	rain	47
Medium	V4	$cctv052x2004080516x01638$	overcast	53
	V5	$cctv052x2004080516x01644$	clear	53
	V6	$cctv052x2004080618x00079$	rain	53
Heavy	V7	$cctv052x2004080516x01646$	overcast	49
	V8	$cctv052x2004080517x01654$	clear	53
	V9	$cctv052x2004080618x00080$	rain	53

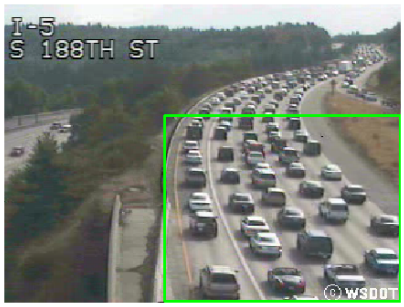


Fig. 3: The selected area to apply the detection process.

traffic is presented has been masked to make the detection process more precise, see Fig.3 where the green bounding box represents this area.

Figure 4 represents some results for the first dataset. For each video, the first three columns was obtained using symmetric MDPMs and the last three columns using asymmetric MDPM. We observe that using symmetric MDPMs, in general, better results were obtained. The description of results is defined by considering the MDPM that has the least number of false positives (FP) and also the least number of false negatives (FN) over all videos. For videos V1 e V2, they have the least FP and the least FN using a symmetric MDPM of 3 mixture components and 6 part models (3M 6P). For videos V4, V8 e V9, they have the least FP and the least FN using a symmetric MDPM of 3M 10P. For V3, it has the least FP with an asymmetric MDPM of 3M 10P and least FN with a symmetric MDPM of 3M 6P but, in general, it has better results using symmetric MDPM of 3M 6P. For V5, a the least FP was obtained with an asymmetric MDPM of 3M 6P and the least FN and in general, better results were obtained using symmetric MDPM of 3M 6P. For V6 better results were obtained using symmetric MDPM of 3M 8P and for V7 using symmetric MDPM of 3M8P. We saw that better results was obtained using symmetric MDPMs. A possible reason for that is because the vehicles in the video are in a symmetric view (i.e. back view).

In respect to the clear day videos, the principal reason to have more false negative was the shadows generated by each vehicle, this can be see in the results for the video: V5. Finally, observing the results, the video where better results were obtained is V7 using a symmetric MDPM of 3M 8P.

Dataset 2: The second set is represented by video: $AVSS\_PV\_EVAL.avi$ from [34]. This video was created to analyse parked vehicles, therefore there is no bounding box for each car present in the video sequence. The video has 6586 frames with a resolution of 720 x 576 pixels.

Figure 5 shows some examples of the results obtained for the second dataset [34]. The results show the bounding boxes obtained using symmetric (first three columns) and asymmetric (last three columns) MDPM of 3 components and 6, 8, 10 parts. The frames 2241, 3196, 4241, 5261, 6411 are presented.

We observe that good results were obtained for the different views of the vehicles that appear in the scene. In some cases, the vehicles of a side view are better detected with an asymmetric MDPM (see the second row in Fig. 5). However, the better results were obtained with symmetric MDPMs. This suggests that we could consider both types of MDPM, in order to improve the results.

C. Vehicle detection considering a dynamic camera

In this experiment, we test the method based on MDPM over the video dataset which were obtained considering a dynamic camera without any ground plane information.

Dataset: In this case, the video dataset used is [28]. This data was captured by Nico and Kurt Cornelis at KU Leuven and it is available for non-commercial research use. The sequence contains 1175 image pairs acquired with a stereo camera mounted on top of a moving vehicle and recorded over a distance of approximately 500 meters. Each frame has a low resolution of 380 x 288 pixels. The dataset have the annotation of all cars that were within a distance of 50 meters and visible by at least 40-50%, that is, it contains 77 (sufficiently visible) static and 4 moving cars. In addition, the dataset contains information of the external camera calibration and ground plane estimated by Structure from Motion [28]. We select the data generated by the left camera.

Results: The metric used to evaluate detection in video is the same as used in PASCAL VOC, that is: recall, precision and average precision (AP). We believe that the PASCAL metric gives a reliable measure of performance. In addition, we present the false positive per image (FPPI) measure that is obtained by the rate of $FP/\#TotalFrames$, where FP are the false positives.

The results of this experiment are shown in Fig. 6, were the green bounding boxes represent the ground truth and red bounding boxes the detections, and in Fig. 7. As one can see from the plot, detection reaches a level of about 40.24% recall and 1.88 false positives per image (FPPI) at this level of recall using an asymmetric MDPM of 3 mixture components and 8 parts model. While in [14], it reaches a recall of 50% and yields 1.3 FPPI at this level of recall and in the most recent experiment showed in [13] it yields with 4 FPPI also at 50% of recall without using ground plane information.

We observe that the technique has equivalent results as the one obtained in [13], [14], even considering that we are not

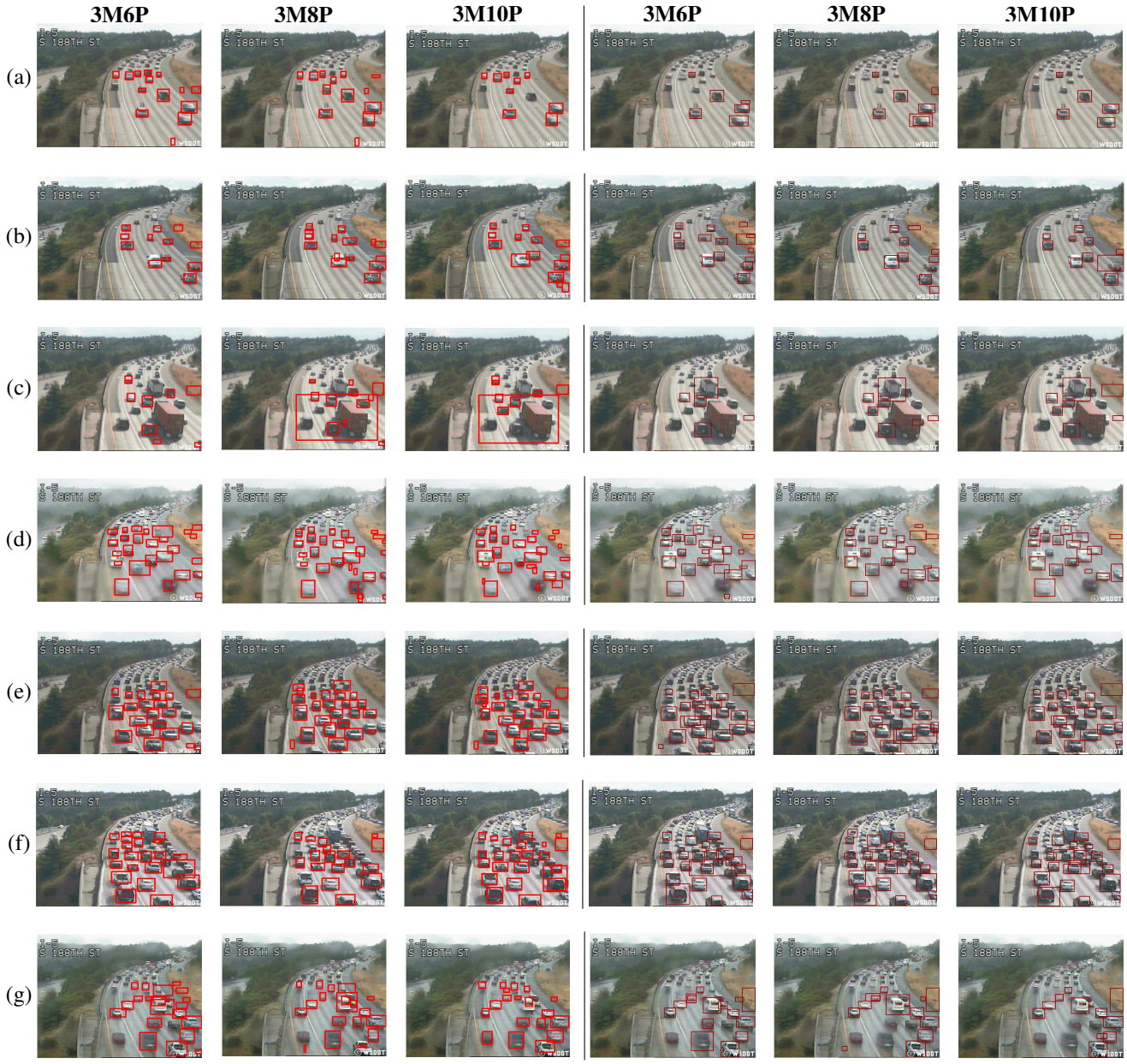


Fig. 4: Examples of results for videos of the class LIGHT: (a)V1, (b)V2, MEDIUM: (c)V5, (d)V6 and HEAVY: (e)V7, (f)V8, (g)V9; using symmetric (first three columns) and asymmetric (last three columns) MDPM.

consider the ground plane and camera information. We believe that with this information, we can get better results, i.e., less false positives. In respect to the false positives, we observe that there are bad detections such as windows of buildings and also because of the technique detect little cars that do not have ground truth, the authors define the ground truth only to the cars that are at a specific distance from the camera position.

V. Conclusion

In this paper we presented a vehicle detection approach in videos using MDPM obtained from image datasets. Several experiments have been made and we even confirmed the number of mixtures and part models suggested by Felzenzwalb et al.

The method proposed by Felzenszwalb et al has been tested to different video sequences and the results are good mainly considering that the models have been created by images and the low quality of some of the video sequences (of the static camera). It is important to notice that we do not separate the images by their view, i.e. left, right, rear, to train the MDPMs. This could be interesting to obtain a more discriminative MDPM but we did not explored this approach in this paper. Considering the high intraclass-view variability

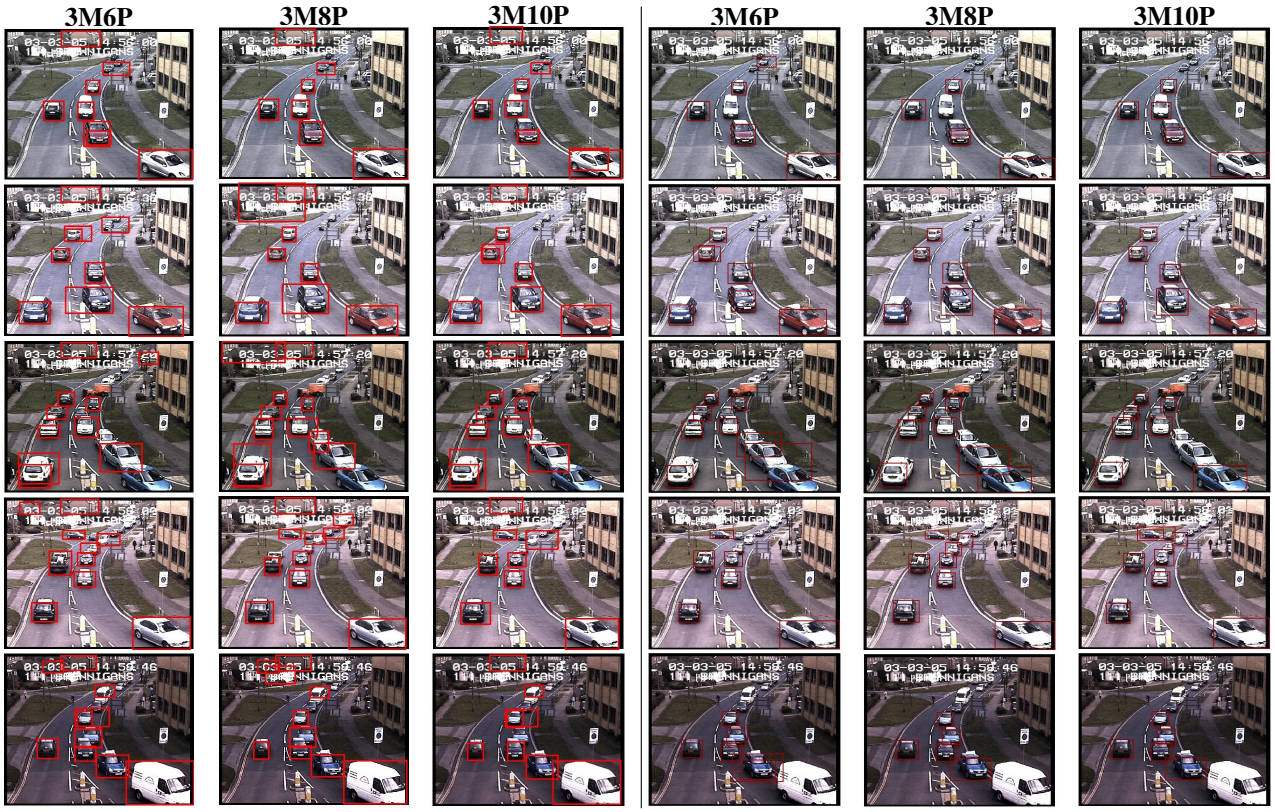


Fig. 5: Examples of results using symmetric (first three columns) and asymmetric (last three columns) MDPM of 3 components and 6, 8, 10 parts using o AVSS video [34].

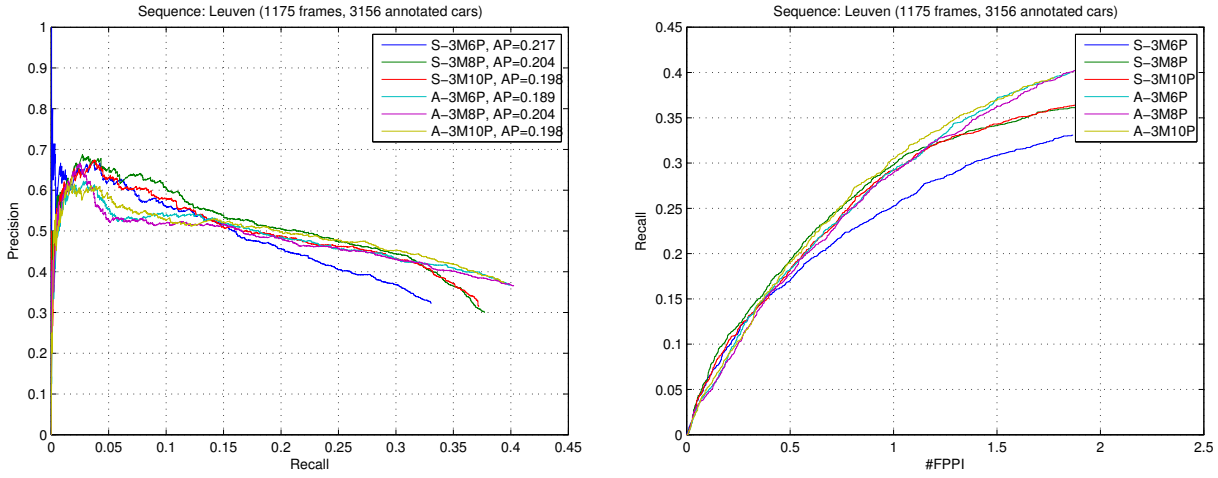


Fig. 7: Results over symmetric and asymmetric MDPMs of 3 components and 6, 8, 10 parts model. The left plot shows the Precision vs Recall curve and AP value. The right plot shows the Recall vs False Positive Per Image (FPPI).

of PASCAL dataset, this must have affected negatively the performance [35].

For the case of video sequences obtained from a dynamic camera, the results obtained are comparable to [13], [14], in some cases with less false positives per images. We plan to extend the application to this video sequence but considering the camera and ground plane information in future work. Future work could includes also the combination of both types of MDPM in order to improve the performance of the detections as was discussed before. We also plan to reduce the

Fig. 6: Examples of results over [28] using an asymmetric MDPM of 3 components and 8 parts.

size of part filters and consequently increment their number in the model in order to avoid consider the background as part of the foreground.

ACKNOWLEDGMENT

The authors acknowledge FAPESP and CNPq for partially funding this work.

REFERENCES

[1] P. Constantine and P. Tomaso, "A trainable system for object detection," *International Journal of Computer Vision*, vol. 38, no. 1, pp. 15–33, 2000.

[2] R. Fergus, P. Perona, and A. Zisserman, "Object class recognition by unsupervised scale-invariant learning," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2, 2003, pp. II/264–II/271.

[3] N. Dalal, "Finding people in images and videos," Ph.D. dissertation, Institut National Polytechnique de Grenoble, july 2006.

[4] B. Leibe, A. Leonardis, and B. Schiele, "Robust object detection with interleaved categorization and segmentation," *International Journal of Computer Vision*, vol. 77, no. 1-3, pp. 259–289, 2008.

[5] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 9, pp. 1627–1645, 2010.

[6] W. Zheng and L. Liang, "Fast car detection using image strip features," in *2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops 2009*, 2009, pp. 2703–2710.

[7] S.-C. S. Cheung and C. Kamath, "Robust techniques for background subtraction in urban traffic video," S. Panchanathan and B. Vasudev, Eds., vol. 5308, no. 1. SPIE, 2004, pp. 881–892.

[8] D. R. Magee, "Tracking multiple vehicles using foreground, background and motion models," *Image Vision Comput.*, vol. 22, no. 2, pp. 143–155, 2004.

[9] S. Kamijo, K. Ikeuchi, and M. Sakauchi, "Vehicle tracking in low-angle and front-view images based on spatio-temporal markov random field model," in *In Proceedings of the 8th World Congress on Intelligent Transportation Systems (ITS*, 2001.

[10] Z. Kim, "Real time object tracking based on dynamic feature grouping with background subtraction," in *CVPR*, 2008.

[11] N. Saunier and T. Sayed, "A feature-based tracking algorithm for vehicles in intersections," in *CRV*, 2006, p. 59.

[12] P. Sudowe and B. Leibe, *Efficient use of geometric constraints for sliding-window object detection in video*, ser. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2011, vol. 6962 LNCS.

[13] N. Cornelis, B. Leibe, K. Cornelis, and L. Van Gool, "3d urban scene modeling integrating recognition and reconstruction," *International Journal of Computer Vision*, vol. 78, no. 2-3, pp. 121–141, 2008.

[14] B. Leibe, N. Cornelis, K. Cornelis, and l. Van Gool, "Integrating recognition and reconstruction for cognitive traffic scene analysis from a moving vehicle," in *DAGM-Symposium*, 2006, pp. 192–201.

[15] B. Leibe, K. Schindler, N. Cornelis, and L. Van Gool, "Coupled object detection and tracking from static cameras and moving vehicles," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 10, pp. 1683–1698, 2008.

[16] G. Wang, D. Xiao, and J. Gu, "Review on vehicle detection based on video for traffic surveillance," in *Automation and Logistics, 2008. ICAL 2008. IEEE International Conference on*, sept. 2008, pp. 2961 –2966.

[17] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (voc) challenge," *International Journal of Computer Vision*, vol. 88, no. 2, pp. 303–338, Jun. 2010.

[18] R. Feris, J. Petterson, B. Siddiquie, L. Brown, and S. Pankanti, "Large-scale vehicle detection in challenging urban surveillance environments," in *Proceedings of the 2011 IEEE Workshop on Applications of Computer Vision (WACV)*, ser. WACV '11. IEEE Computer Society, 2011, pp. 527–533.

[19] B. Wu and R. Nevatia, "Cluster boosted tree classifier for multi-view, multi-pose object detection," in *ICCV*, 2007, pp. 1–8.

[20] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proceedings - 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR*, 2005, pp. I: 886–893.

[21] P. F. Felzenszwalb, R. B. Girshick, and D. McAllester, "Discriminatively trained deformable part models, release 4."

[22] H. Schneiderman and T. Kanade, "Statistical method for 3d object detection applied to faces and cars," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 1, 2000, pp. 746–751.

[23] Z. Sun, G. Bebis, and R. Miller, "On-road vehicle detection: A review," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 5, pp. 694–711, 2006.

[24] P. F. Felzenszwalb, R. B. Girshick, and D. McAllester, "Cascade object detection with deformable part models," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2010, pp. 2241–2248.

[25] Y. . Lin, T. . Liu, and C. . Fuh, *Fast object detection with occlusions*, ser. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2004, vol. 3021.

[26] G. Griffin, A. Holub, and P. Perona, "Caltech-256 object category dataset," California Institute of Technology, Tech. Rep.

[27] A. B. Chan and N. Vasconcelos, "Probabilistic kernels for the classification of auto-regressive visual processes," in *Proceedings - 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2005*, vol. I, 2005, pp. 846–851.

[28] B. Leibe, N. Cornelis, K. Cornelis, and L. Van Gool, "Dynamic 3d scene analysis from a moving vehicle," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2007.

[29] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results."

[30] A. Vedaldi, V. Gulshan, M. Varma, and A. Zisserman, "Multiple kernels for object detection," in *Proceedings of the International Conference on Computer Vision*, September 2009.

[31] Y. Chen, L. Zhu, and A. Yuille, "Active mask hierarchies for object detection," in *Proceedings of the 11th European conference on Computer vision: Part V*, ser. ECCV'10, 2010, pp. 43–56.

[32] Z. Song, Q. Chen, Z. Huang, Y. Hua, and S. Yan, "Contextualizing object detection and classification," *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, pp. 1585–1592, 2011.

[33] R. Mottaghi, "Augmenting deformable part models with irregular-shaped object patches," *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, 2012.

[34] "i-lids dataset for avss 2007."

[35] R. J. López-Sastre, T. Tuytelaars, and S. Savarese, "Deformable part models revisited: A performance evaluation for object category pose estimation," in *ICCV Workshops*, 2011, pp. 1052–1059.