2015 IEEE Intelligent Vehicles Symposium (IV)
June 28 - July 1, 2015. COEX, Seoul, Korea

Fast pixelwise road inference based on Uniformly Reweighted Belief Propagation

Mario Passani, J. Javier Yebes and Luis M. Bergasa

Abstract— **The future of autonomous vehicles and driver assistance systems is underpinned by the need of fast and efficient approaches for road scene understanding. Despite the large explored paths for road detection, there is still a research gap for incorporating image understanding capabilities in intelligent vehicles. This paper presents a pixelwise segmentation of roads from monocular images. The proposal is based on a probabilistic graphical model and a set of algorithms and configurations chosen to speed up the inference of the road pixels. In brief, the proposed method employs Conditional Random Fields and Uniformly Reweighted Belief Propagation. Besides, the approach is ranked on the KITTI ROAD dataset yielding state-of-the-art results with the lowest runtime per image using a standard PC.**

I. Introduction

Nowadays, visual robust recognition of the driving path is a key issue in the develop of autonomous vehicles, which need to reliably operate under complex traffic situations and naturalistic road environments [1], [2], [3]. Besides, the Intelligent Vehicles community is making great efforts to incorporate road scene understanding capabilities in ever more sophisticated Advanced Driver Assistance Systems (ADAS), e.g., lane departure warning [4] and adaptive cruise control [1], which are mainly focused on traffic safety.

Indeed, urban objects which attract a biggest interest from the driver's point of view, are always on the ground, excluding some traffic signs and traffic lights. Moreover, there is a pose relation between the objects in a road scene with respect to the road (e.g., vehicles, pedestrians, vegetation, buildings, sky, etc). As a consequence, road detection is an important research topic because it allows to impose geometrical constraints for the driving activities (autonomous or driver assistance) and for the detection of objects. Thus, it also improves the road scene understanding

On the other hand, the employment of monocular vision systems in ADAS and autonomous driving for the task of road detection is relatively inexpensive and easy to integrate. They capture the 3D scene as perceived by a driver, who performs a local navigation based on the scene features, while global way-points can still be established by other navigation techniques (e.g., GPS and pre-computed maps). Nonetheless, road segmentation using monocular vision is not a simple task, being specially challenging in rural areas and inner-cities due to the usual absence of lane markings [2].

**This project was supported in part by the MINECO Smart Driving Applications project(TEC2012-37104) and by the RoboCity2030 III-CM (S2013/MIT-2748) funded by Programas de I+D en la Comunidad de Madrid and cofunded by Structural Funds of the UE.

The authors are with the Department of Electronics, UAH. Alcalá de Henares, Spain. e-mail: mario.passani, javier.yebes, bergasa@depeca.uah.es

This paper presents a practical study of the pixelwise road segmentation combining harmoniously computer vision techniques and Probabilistic Graphical Models (PGMs). Particularly, the semantic labeling of the pixels in the road scene images relies upon Conditional Random Fields (CRFs) [5] and approximated marginal inference [6].

This robust probabilistic approach is able to work in road scene without markings. In addition, compared to other proposals in the state-of-the-art [7], this paper also contributes with a fast and efficient technique to perform road inference from monocular images. Our main goal is to run the algorithm under real time constraints, such that it could be embedded on-board vehicles. Therefore, our proposal is based in the following steps: Firstly a region of interest is selected based on an estimation of the horizon line. This is the image area where it is more likely to find the road. Secondly, we implement a fast pixelwise road inference based on the robust probabilistic approach mentioned above. Thirdly, inspired in the recent works about visual place recognition with low resolution scenes [8], our approach reduces the size of road images based on superpixels, resulting in miniaturized scenes [9] which are efficient to compute. Finally, morphological operations are applied to deal with scaling artifacts and misclassified pixels. An overview of this workflow is depicted in Fig. 1.

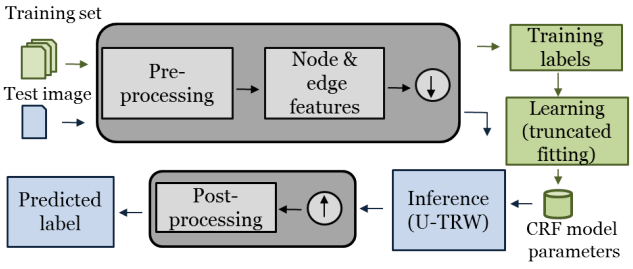


Fig. 1: Overall system workflow. Down and up arrows correspond with the downsample and upsample performing with superpixels.

The proposed methodology is validated on the public KITTI-ROAD benchmark [2], ranking the second in "Urban Multiple Marked lanes" and fourth in the global "Urban Road" category. Moreover, our contribution is the fastest approach in the ranking, achieving real time performance on a standard PC without the need for a dedicated GPU.

The remainder of the paper begins with an overview of related works on road segmentation. Next, Sections III to VI describe our methodology, which includes the pre- and post-processing stages, the learning and inference with

CRFs and the description of the employed visual features. Section VII provides some implementation details and Section VIII shows the experimental results on KITTI ROAD dataset.

II. State of the art

Road detection is a difficult task due to many reasons, including the absence of lane markings, variations in lighting conditions, different road surface materials, occlusions with other vehicles and objects, etc. Typically, the most used approach for segmenting marked roads is the localization of the markings [10], [11]. However, for unstructured roads and structured roads without remarkable boundaries, road segmentation must be addressed from a different perspective.

Methods based upon segmenting the road using color cues have been proposed [12]. However, they do not work well when the roads surfaces have slight color differences compared to the scene environment. Besides, they may also fail in situations with strong shadows and highly illuminated areas. On the other hand, color and texture features employed in [13] in conjunction with Artificial Neural Networks (ANN) are exposed to some limitations in appearance, i.e., roads can present aperiodic texture, which is hard to characterize.

Although the detection of the vanishing point [14], followed by the segmentation of the corresponding road area was probed to be robust against variations in illumination and the road type, this approach may fail with curved roads, heavy traffic and strong shadow edges.

Alternatively, Vitor et al. [15] create a set of probabilistic models using an adapted version of the Joint Boosting algorithm with Texton and Diston feature maps. Essentially, the method consists on a set of weak classifiers analyzing color and disparity information. Although this work achieves state-of-the-art results for the road segmentation, the classifier requires 2.5 minutes for each frame. SPRAY [16] also proposes a set of classifiers (boundary, road and marking) creating three confidence maps. However, it runs faster than the previous one and it yields road recognition improvements in the KITTI ROAD benchmark. Nonetheless, this real time performance is at the expense of using a powerful GPU.

Moreover, a recent work by R. Mohan [17] combines Deep Deconvolutional and Convolutional Neural Networks for the general task of scene parsing. Compared to different engineered features election methods, this is an alternative technique to automatically learn features directly from the images. This method is currently ranking first on KITTI ROAD benchmark. However, this approach is computationally intensive and it requires a GPU cluster to process the data.

Therefore, our contribution aims at carrying out an efficient semantic labeling of monocular images, with a particular emphasis on road segmentation. The computation time will be reduced during the inference stage, while, at the same time, relying on a complex and robust machine learning methodology, i.e., Conditional Random Fields.

III. Preprocessing of the road scenes

Assuming that the images are captured from an on-board camera in a moving vehicle, the road detection process can be bounded to a specific region of interest (ROI), due to physical and continuity constraints. This ROI is where the road is more likely to be found on the images.

Therefore, we propose to filter out the area of each road scene which is not expected to contain any road pixels based on the horizon line. This preprocessing step discards a large region of the image, reducing the computational cost. A rectangular mask will remove the pixels that are out of the expected region of interest (ROI), which contains the road. The height h of the ROI depends on the estimation of the horizon line of the urban scenes, while its width w equals the image width. To estimate the horizon line, a set of training images are employed to detect the vanishing points of the scenes. In particular, we use the locally adaptive soft-voting scheme proposed by Kong *et al* [14]. Then, a value for h is obtained, also leaving a certain margin about the mean height of the vanishing points.This idea is depicted in Fig. 2.

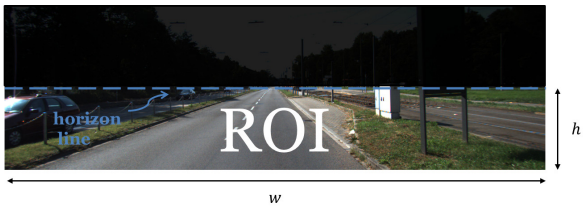


Fig. 2: Rectangular mask filters out non-road expected pixels and the ROI contains the road. The horizon line is estimated from a set of training images.

IV. Model learning and road inference

To detect the road pixels in urban and interurban images, our approach assigns semantic labels to the superpixels taken from the preprocessed following the technique published by the authors in [9]

To model the uncertainty associated to this classification process, a probabilistic approach is proposed, which relies upon Conditional Random Fields (CRF) [5]. This is a PGM that represents the conditional probability distribution $p(\mathbf{y}|\mathbf{x})$ where $\mathbf{x}$ is a vector denoting the observed data (visual features in our case) and $\mathbf{y}$ is a tuple of variables that have to be estimated (either latent or not). Hence, it is not needed to explicitly model $p(\mathbf{x})$ thanks to the conditional formulation. For the road segmentation case, the vector $\mathbf{y}$ is related to the array of output classes from the superpixels. Each of them takes its values from a set of labels $\mathcal{L}$ defining the two possible semantic classes: *road* and *off-road*.

A. CRF model

CRFs can be represented by undirected graphs, in which the nodes are random variables and the existence of a link between nodes defines conditional dependencies between the involved variables. Particularly, this paper presents a pairwise CRF model with a grid-like structure whose nodes

correspond to a lattice of superpixels in a 4-neighborhood on the ROI described in Section III. This structure allows us to encode spatial dependencies inside the images in an easy way, as can be seen in Fig. 3. At each node i an output random variable y_i is defined, taking a value from a set of labels $\mathcal{L} = \{0, 1\}$ corresponding to off-road and road respectively. Besides, the observable random variable x_i encodes the values of certain local features that will be exposed in Section V.

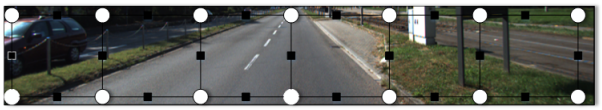


Fig. 3: Graph of the CRF model aligned with the ROI.

The probability distribution of the semantic labels vector $\mathbf{y}$ conditioned on the observed features $\mathbf{x}$ and a vector of unknown model parameters $\mathbf{w}$ can be factored in a product of unary (ψ_i) and pairwise ($\psi_\mathcal{C}$) positive factors [5], called potentials:

$$p(\mathbf{y}|\mathbf{x};\mathbf{w}) = \frac{1}{Z(\mathbf{x};\mathbf{w})} \prod_i \psi_i(y_i, \mathbf{x}; \mathbf{w_i}) \prod_\mathcal{C} \psi_\mathcal{C}(y_i, y_j, \mathbf{x}; \mathbf{w}_\mathcal{C}) \tag{1}$$

The first product in (1) is over all individual nodes i, while the second one is over the set of cliques $\mathcal{C}$ of order two, i.e., the edges in the graph which model neighboring superpixels interactions. $Z(\mathbf{x};\mathbf{w})$ is known as the partition function for normalization purposes.

Furthermore, the graphical model in Fig. 3 is not tree-structured, but instead contains cycles. As a consequence, the CRF presents two issues:

1) Exact inference is known to be NP-hard.
2) Learning is intractable and can also generate poor predictions when the model is misspecified.

The first issue can be solved performing approximate inference with some method like Loopy Belief Propagation [5] or Tree-Reweighted Belief Propagation [18]. In this paper, we employ the latter one because of its demonstrated effectiveness to reach the global minimum in binary image segmentation. Next, to solve the second issue, we propose to employ a recent work of J. Domke [6], which presents a parameter learning based upon approximate marginal inference instead the usual approach based on approximations of the likelihood.

Then, the automatic semantic labeling of the road scenes is divided into two basic stages: the off-line and more computationally demanding process of model learning and the online semantic inference with much less computational demand. They can be also referred to as training and testing stages from a classification point of view.

B. Inference

Inference will be treated in first place, before learning, because its engine is employed both during prediction and model training. As already indicated in previous section, our method for approximate inference is based on Tree-Reweighted Belief Propagation (TRW) [18]. With the aim of providing a fast and efficient approach for road detection, this inference algorithm has been implemented in C++ as we describe in this section.

Firstly, the main formulation of the message-passing approach is reproduced here for clarity. At each iteration, every node i of the graph sends a message $m_\mathcal{C}(y_i)$ to its neighbor $\mathcal{N}_i$ in the clique. Then, the message passing update is:

$$\begin{aligned} m_{i\to j}(y_j) &\propto \sum_i \psi_i(y_i, \mathbf{x}) \cdot \psi_{ij}^{\frac{1}{\rho_{ij}}}(y_j, y_i, \mathbf{x}) \cdot \eta \\ \eta &= \frac{1}{m_{j\to i}^{1-\rho_{ij}}(y_j)} \prod_{n\in\mathcal{N}_i\setminus j} m_{n\to i}^{\rho_{ni}}(y_i) \end{aligned} \tag{2}$$

where $\mathcal{N}_i$ is the set of neighbors of node i, coefficients ρ_{ij} are called *edge appearance probabilities* indicating the probability that a given edge appears in a spanning tree of the graph and $\propto$ means assigned after normalization.

After the messages have converged, each node can form an estimate of its local approximate marginal defined as,

$$\mu_i(y_i) \propto \psi_i(y_i) \prod_{n\in\mathcal{N}_i} m_{n\to i}^{\rho_{ni}}(y_i) \tag{3}$$

In particular, we use a simplified version of TRW, Uniformly Reweighted Belief Propagation [19] assigning a constant appearance probability to all edges, thus $\rho_{ij} = \rho \,\forall\, i, j$. It reduces the computational complexity being also an optimal choice for our graph structure. Besides, this simplified scheme turns out to outperform Belief Propagation (BP) in graphs with cycles. Also, note that in the special case of $\rho_{ij} = 1$, TRW simplifies into local BP.

Additionally, it must be noted that according to [20], the optimal value of ρ for graphs, satisfying certain symmetry conditions as in our case, can be approximated with the number of vertices ($|V|$) and edges ($|E|$) using (4).

$$\rho^* \approx \frac{|V| - 1}{|E|} \tag{4}$$

In general, it can be stated that this parameter tends to 0.5 for bigger grid graphs like ours. Although this is not necessarily the optimum, it is the largest number that leads to a convex inference problem.

C. Learning

The aim of learning is to select the optimal model from its feasible set, based on the training data, obtaining the vector of parameters $\mathbf{w}^*$. Using a loss-based approach [5], the learning formulation is as follows:

$$\mathbf{w}^* = \arg\min_\mathbf{w} \sum \Delta(\mathbf{y}, \mathbf{t}; \mathbf{w}) \tag{5}$$

where $\Delta(\mathbf{y}, \mathbf{t}; \mathbf{w})$ denotes a loss function measuring the similarity between the ground-truth training labels $\mathbf{t}$ and the estimated labels $\mathbf{y}$. Thus, the choice of parameters $\mathbf{w}$ influences the loss function through $\mathbf{y}$.

As for the impossibility to compute the true marginals in our not tree-structured graphical model, the loss function is defined with respect to the marginal predictions.

To minimize (5), the loss gradient has to be calculated, which can be very expensive, requiring many message-passing iterations. To reduce the expense of such training, we first use the truncated fitting alternative [6] to compute a predicted marginal using TRW. Indeed, five iterations are selected to truncate the message-passing. Then, we back-propagate predicted marginals making slight modifications of the parameters to reduce and adjust the loss. This process is repeated until the loss is lower than a configurable threshold.

Furthermore, the choice of the loss function is important because, on the one hand, it influences the accuracy, and on the other hand, the simpler the loss the easier the gradient is to compute. Particularly, we employ the conditional logistic loss [21] evaluated on the cliques and given by (6). This function ensures the consistency between the predicted marginals and joint distribution when using our CRF model.

$$\Delta = -\sum_{\mathcal{C}} \log \mu(\mathbf{y}_{\mathcal{C}}|\mathbf{x};\mathbf{w}) \tag{6}$$

V. Image features for the CRF potentials

Since our graphical model has a pairwise grid-like structure and according to the presented CRF formulation into unary and pairwise potentials in (1), two types of feature functions will be encoded in the model: node features and edge features. These can be described by exponential functions using log-linear combinations of features extracted from the observed data $\mathbf{x}$. This is due to the fact that the potentials are restricted to be positive. Thus:

$$\begin{cases} \psi_i = \exp(\mathbf{w_i}^T \cdot \mathbf{f}_i(\mathbf{x})) & \text{(7a)} \\ \psi_{\mathcal{C}} = \exp(\mathbf{w}_{\mathcal{C}}{}^T \cdot \mathbf{g}_{\mathcal{C}}(\mathbf{x})) & \text{(7b)} \end{cases}$$

In (7a) $\mathbf{f}_i$ is a vector-valued function that defines the features of node i; similarly in (7b), $\mathbf{g}_{\mathcal{C}}$ is another vector that maps the input to pairwise features of nodes in the clique $\mathcal{C}$. This idea is illustrated in Fig. 4. In both cases, we carry out a feature scaling or whitening to speed up and favor the gradient-descent loops in the learning phase. Basically, the procedure consists in scaling the features, subtracting the mean, and dividing by the standard deviation.

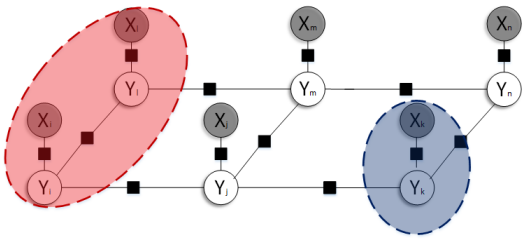


Fig. 4: Portion of the proposed graphical model. The node and edge features are overlaid in blue and red, respectively.

A. Node features

Several features are extracted to represent the visual appearance related to every node of the graph. They carry information about color, position, texture and shape that are concatenated into $\mathbf{f}_i$ to span a rich feature space.

Firstly, we propose to incorporate observations describing the color distribution in the scene. After testing different color spaces as RGB, HSV, Lab and grayscale, we empirically found that HSV yielded the lowest overall errors. Then, we experiment serializing the components hue and saturation, quite immune to scene illumination changes in a $(2k+1)$ squared patch around a pixel for the values $k = 0, 1, 2$. We found that the use of bigger patches did not improve road detection performance, but it raised computation time considerably. In particular, increasing the dimensionality from 2D ($k = 0$) to 50D ($k = 2$) produces a 2% increase in accuracy at a cost of approximately doubling the computation time. Thus, we decided to obtain a vector of two values for each node with the intensities for the components hue and saturation.

As already described in [9], we add a set of features which are summarized next. Two features, $\mathbf{f}_u$ and $\mathbf{f}_v$, account for the normalized position along the horizontal and vertical axes. Also, the texture of the roads is captured in the form of LBP descriptors. In particular, we employ $P = 4$ sampling points and a radius $R = 1$ pixel, obtaining a feature function $\mathbf{f_{LBP}}$ of $2^P = 16$ elements. Besides, the local appearance and shape is acquired as HOG vectors over a grid of non-overlapping 8×8 cells, with 9 orientation bins per cell, concatenating 4 cells to one block descriptor. After normalizations, a 36-dimensional vector $\mathbf{f_{HOG}}$ is obtained. According to our experiments $\mathbf{f_{HOG}}$ and $\mathbf{f_{LBP}}$ are the most discriminative features in terms of the overall accuracy.

As a result, the observation variables $\mathbf{X_i}$ associated to each node in the presented graphical model (see Fig. 3) correspond to 56-dimensional feature vectors ($D = 2{+}2{+}16{+}36 = 56$), which populate the potentials of the CRF formulation.

B. Edge features

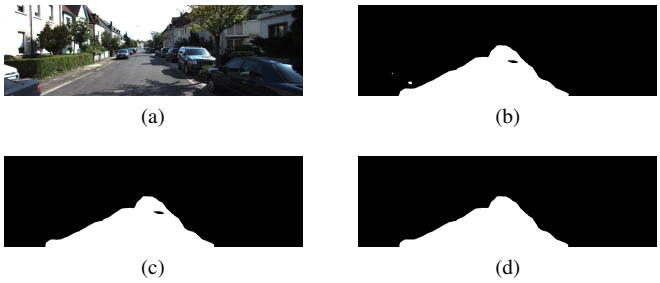
The relations between the adjacent nodes in the undirected graph are incorporated into the model by the edge features. In brief, they consist of [9]: a bias feature to capture any effects on the states of the random variables that are independent on the other features, and the discretized L-2 norm of the difference of hue and saturation intensities for two neighbor nodes. The resultant 11D vector is doubled in size and arranged differently depending on whether the edges are vertical or horizontal allowing us to separately parameterize vertical and horizontal edges.

VI. Post-processing

The output from the presented road segmentation approach and for every input image is a set of predicted pixels that are likely to belong to a road in the real world scenes. However, some of the pixels from output variable nodes ($\mathbf{y_i}$) in our proposed graphical model may be misclassified due to feature scaling artifacts. In particular, the presence of small specks classified as "road" in a large area corresponding to "off-road" and vice versa. An example of these small specks and holes is illustrated in Fig. 5.b.

To deal with these specific misclassifications, a morphological opening is done to eliminate the specks classified

like "road" and a morphological closing is performed to eliminate the specks classified like "off-road". In both cases, a rectangular structuring element of 15×15 size is employed. Fig. 5 shows some real examples on this regard.



(a) (b) (c) (d)

Fig. 5: Post-processing of a sample result from the pixelwise road inference: (a) input image (b) predicted pixel labels (c) removal of false positives using a morphological opening (d) removal of false negatives using a morphological closing.

VII. Implementation details

Since training and inference tasks are computationally expensive, we seek a practical approach to reduce the computation time. Besides, inspired on recent works of visual place recognition with low resolution images we also opt to study and test lower resolutions in the dataset employed in Section VIII. To do that, once the features have been computed on the original images, we reduce their resolution by using superpixels and we subsample the ground-truth labels accordingly to a percentage of the original size. Once the approximate marginals are calculated, we upsample them to the original resolution. With this approach, we shrink the space of hypotheses and achieve a speed-up both in training and testing stages, but more notable during inference, which is the application-oriented part of the algorithm.

For training we employ the Toolbox of Justin Domke [22], mainly developed in MatLab with some Mex files to speed up certain algorithms. Time reduction is not a priority for the training task. However, to meet real-time restrictions during inference and to build a code easier to integrate into Advance Driver Assistante Systems (ADAS), we implemented the preprocessing, the inference and the postprocessing stages in C++, employing two open-source libraries: *OpenCV* and *Eigen* for the vision and linear algebra operations.

VIII. Experiments

The public KITTI ROAD [2] dataset is employed for evaluation. It consists of 600 frames ($\approx 375 \times 1242$ pixels) extracted from several video sequences at a minimum spatial distance of 20 meters. Besides, it is split in three subsets, each representing a typical road scene category in inner cities: urban unmarked (UU), urban marked two-way road (UM) and urban marked multi-lane road (UMM), having each one a subset of training and test images. Besides, URBAN-ROAD quantifies the previous three categories in a single set of measurements.

Table I shows the effectiveness of our approach, abbreviated as PGM-ARS, to extract the road while varying the reduction percentage of the image resolution evaluated with 5-fold cross validation on the training images. All evaluations are performed in the called "bird eye view" due to is best suited for vehicle control [2] using the standard metrics precision and recall.

TABLE I: Road estimation results obtained for different sizes of the validation images. All results are in %

	UU		UM		UMM	
Img Res	**PRE**	**REC**	**PRE**	**REC**	**PRE**	**REC**
5 %	72.49	76.80	71.13	81.69	83.33	90.08
10 %	79.90	80.92	75.11	86.63	89.83	93.02
15 %	79.50	81.34	75.65	86.88	89.80	93.66
20 %	**82.75**	83.96	**84.44**	87.53	**90.07**	94.26
25 %	82.52	84.13	84.34	88.04	89.84	94.58
30 %	82.22	84.53	83.10	88.31	89.76	94.67
40 %	82.05	84.79	82.69	88.54	89.55	94.88
50 %	81.88	**85.24**	82.08	**88.72**	89.15	**94.92**

The highest precision values are obtained for 20% image resolution, whereas the recall values are slightly increased with the resolution. In fact, the gain between 20% and 50% rows is lower than 1.5% for all categories.

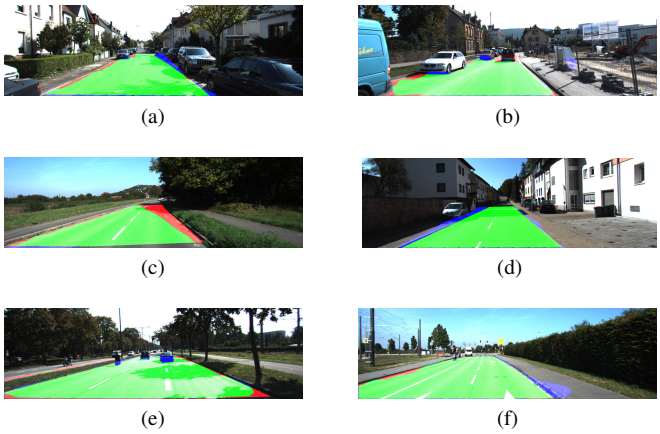
Therefore, there is not much improvement for increasing image size. Our explanation for this result is twofold. In first place, bigger images have more granular detail, partly reducing the intra-region similarity. Secondly, in lower-resolution models, there are fewer intermediate variables, facilitating the spread of messages and the CRF model convergence. These results validate our superpixels hypothesis as the optimal way for segmenting complex images with a fair time processing. Besides, due to memory limitations during training, we have not been able to test percentages over 50%.

According to the previous cross-validation experiments, we opt to reduce the images at 20% and then evaluate the road segmentation performance on the testing set. At this small resolution, inference is computationally efficient requiring less than 50 ms per image in a i7-4700MQ processor and without a big loss in accuracy. The results are shown in Table II using standard Kitti metrics where $MaxF$ represents the maximum F-measure (from precision and recall curves) and AP the average precision respectively [2].Fig. 6 illustrates some of the predicted samples. It can be observed that the precision values are degraded for UM and UU categories. This is explained by an increase of false positives in complex scenes in which parking lots, garage entrances and crossroads have a road appearance. Our algorithm classifies them as road, but they are "off-road" in the ground-truth.

Next, Table III depicts the most representative entries in the public KITTI ROAD benchmark. Our PGM-ARS proposal is placed among the state-of-the-art. We obtain similar values at much lower time costs, but ranking second in roads with multiple marked lanes (UMM_ROAD) and fourth in the general UM_ROAD category among monocular approaches to the date of April 2015. Besides, the proposals achieving higher accuracies require longer computation times

TABLE II: Road estimation results on the test set images

Benchmark	MaxF %	AP %	PRE %	REC %
UM_ROAD	81.20	69.82	78.32	84.30
UMM_ROAD	90.95	85.68	88.86	93.14
UU_ROAD	79.82	68.33	77.97	81.76
URBAN_ROAD	85.52	74.75	83.24	87.92



(a) (b) (c) (d) (e) (f)

Fig. 6: Examples of road detection images for the test set obtained from the public benchmark suite, red denotes false negatives, blue areas correspond to false positives and green represents true positives (a),(b) UU; (c),(d) UM; (e),(f) UMM

and more hardware resources compared to ours. It must be also noted that our approach uses monocular images and does not require stereo vision nor 3D points.

IX. Conclusions and Future Works

This paper has presented an efficient method for road detection in monocular images. It proposed a fast pixelwise road inference using a robust probabilistic graphical model, i.e. CRF and Uniformly Reweighted Belief Propagation. Different visual features have been selected to efficiently exploit the context and pixel dependencies in the road scene. The employment of superpixels from reduced images and the C++ implementation of the inference stage have contributed to achieve real-time performance, which may ease its integration into ADAS and autonomous driving systems. Experiments conducted on KITTI ROAD dataset have shown that the overall accuracy of our proposed PGM-ARS is among the state-of-the-art performance but achieving the lowest runtime per image in an standard CPU.

TABLE III: Comparison of KITTI URBAN-ROAD state-of-the-art

Method	Setting	MaxF	Runtime	Environment
DDN	Mono	92.55%	2 s	GPU @ 2.5 Ghz (Python + C/C++)
ProBoost	Stereo	87.21%	2.5 min	>8 cores @ 3.0 Ghz (C/C++))
SPRAY	Mono	86.33%	45 ms	NVIDIA GTX 580 (Python + OpenCL)
PGM-ARS	Mono	85.52%	50 ms	4 cores @ 2.1 GHz
RES3D-Velo	Laser	85.49%	0.36 s	1 core @ 2.5 Ghz (C/C++)

In the next future, our intent is to extend this work for multi-class road scene segmentation and to further study the difficult cases of UU and UM categories.

References

[1] A. B. Hillel, R. Lerner, D. Levi, and G. Raz, "Recent progress in road and lane detection: a survey," *Machine Vision and Applications*, vol. 25, no. 3, pp. 727–745, 2014.

[2] J. Fritsch, T. Kuehnl, and A. Geiger, "A New Performance Measure and Evaluation Benchmark for Road Detection Algorithms," in *IEEE Intelligent Transportations Systems Conference (ITSC)*, Oct 2013, pp. 1693–1700.

[3] N. Einecke and J. Eggert, "Block-matching Stereo with Relaxed Fronto-Parallel Assumption," in *IEEE Intelligent Vehicles Symposium (IV)*, June 2014, pp. 700–705.

[4] L. M. Bergasa, D. Almería, J. Almazán, J. J. Yebes, and R. Arroyo, "DriveSafe: an App for Alerting Inattentive Drivers and Scoring Driving Behaviors," in *IEEE Intelligent Vehicles Symposium (IV)*, June 2014, pp. 240–245.

[5] S. Nowozin and C. H. Lampert, "Structured Learning and Prediction in Computer Vision," *Foundations and Trends in Computer Graphics and Vision*, vol. 6, no. 3-4, pp. 185–365, 2011.

[6] J. Domke, "Learning Graphical Model Parameters with Approximate Marginal Inference," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 10, pp. 2454–2467, 2013.

[7] KITTI, "Road estimation benchmark," [web page] http://www.cvlibs.net/datasets/kitti/eval_road.php, 2014.

[8] R. Arroyo, P. F. Alcantarilla, L. M. Bergasa, J. J. Yebes, and S. Gámez, "Bidirectional Loop Closure Detection on Panoramas for Visual Navigation," in *IEEE Intelligent Vehicles Symposium (IV)*, Dearborn, USA, June 2014, pp. 1378–1383.

[9] M. Passani, J. J. Yebes, and L. M. Bergasa, "CRF-based semantic labeling in miniaturized road scenes," in *IEEE Intelligent Transportations Systems Conference (ITSC)*, Oct 2014, pp. 1902–1903.

[10] J. McCall and M. Trivedi, "Video-based Lane Estimation and Tracking for Driver Assistance: Survey, System, and Evaluation," *IEEE Transactions on Intelligent Transportation Systems*, vol. 7, no. 1, pp. 20–37, March 2006.

[11] M. Felisa and P. Zani, "Robust Monocular Lane Detection in Urban Environments," in *IEEE Intelligent Vehicles Symposium (IV)*, June 2010, pp. 591–596.

[12] H. Yinghua, W. Hong, and Z. Bo, "Color-Based Road Detection in Urban Traffic Scenes," *IEEE Intelligent Transportations Systems Conference (ITSC)*, vol. 5, no. 4, pp. 309–318, Dec 2004.

[13] P. Y. Shinzato and D. F. Wolf, "A Road Following Approach Using Artificial Neural Networks Combinations," *Journal of Intelligent and Robotic Systems*, vol. 62, no. 3-4, pp. 527–546, 2011.

[14] H. Kong, J.-Y. Audibert, and J. Ponce, "Vanishing Point Detection for Road Detection," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2009, pp. 96–103.

[15] G. B. Vitor, A. C. Victorino, and J. . V. Ferreira, "A Probabilistic Distribution Approach for the Classification of Urban Roads in Complex Environments," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2014, pp. 1–6.

[16] T. T. Kühnl, F. Kummert, and J. Fritsch, "Spatial ray features for real-time ego-lane extraction," in *IEEE Intelligent Transportations Systems Conference (ITSC)*, Sept 2012, pp. 288–293.

[17] R. Mohan, "Deep Deconvolutional Networks for Scene Parsing," *arXiv preprint arXiv:1411.4101*, 2014.

[18] M. J. Wainwright, T. S. Jaakkola, and A. S. Willsky, "Tree-Reweighted Belief Propagation Algorithms and Approximate ML Estimation by Pseudo-Moment Matching," in *International Conference on Artificial Intelligence and Statistics (AISTATS)*, Jan 2003.

[19] H. Wymeersch, federico Penna, and V. Savic, "Uniformly Reweighted Belief Propagation for Estimation and Detection in Wireless Networks," *IEEE Transactions on Wireless Communications*, vol. 11, no. 4, pp. 1587–1595, April 2012.

[20] M. J. Wainwright, T. S. Jaakkola, and A. S. Willsky, "A New Class of Upper Bounds on the Log Partition Function," in *Conference on Uncertainty in Artificial Intelligence (UAI)*, 2002, pp. 536–543.

[21] S. Kakade, Y. W. Teh, and S. T. Roweis, "An Alternate Objective Function for Markovian Fields," pp. 275–282, 2002.

[22] J. Domke, "Graphical Models/CRF toolbox," [web page] http://users.cecs.anu.edu.au/~jdomke/JGMT/, 2013.