A Fast, Modular Scene Understanding System using Context-Aware Object Detection

Cesar Cadena, Anthony Dick and Ian D. Reid

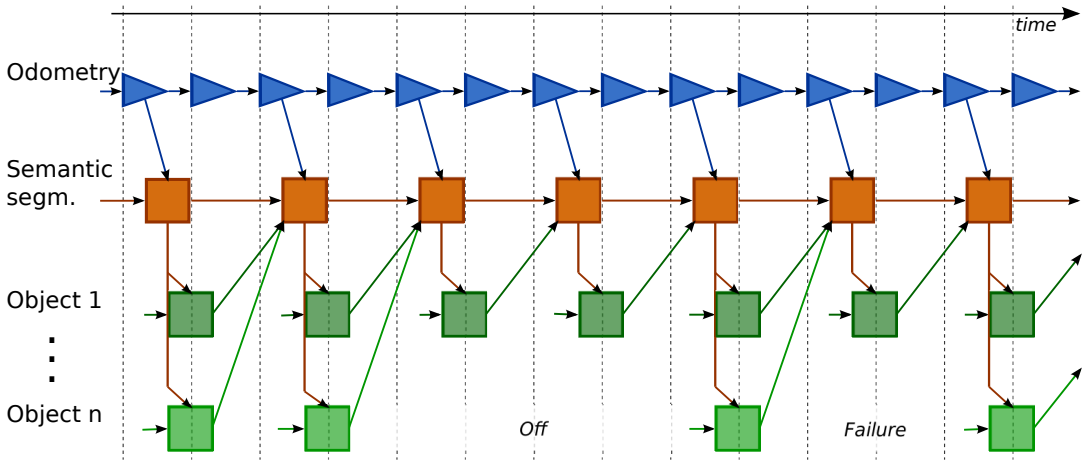


Fig. 1: Proposed system in this paper. The semantic segmentation (SS) thread is used to improve the performance of any specific object detector which itself feedback to SS the detected object to improve the generic object class with this new evidence. Any number of different object detectors can be turned off or on seamlessly at any time during operation. Figures in this paper are best viewed in color.

***Abstract*— We propose a semantic scene understanding system that is suitable for real robotic operations. The system solves different tasks (semantic segmentation and object detections) in an opportunistic and distributed fashion but still allows communication between modules to improve their respective performances. We propose the use of the semantic space to improve specific out-of-the-box object detectors and an update model to take the evidence from different detection into account in the semantic segmentation process. Our proposal is evaluated with the KITTI dataset, on the object detection benchmark and on five different sequences manually annotated for the semantic segmentation task, demonstrating the efficacy of our approach.**

I. Introduction

Semantic scene understanding is a fundamental requirement to enhance the autonomy and robustness of robotic operations. Depending on the task at hand a robot needs different levels of scene understanding, such as a geometric reconstruction, a segmentation of the scene into different semantic components, or detecting specific objects of interest. In this work we focus on the perceptual tasks of semantic segmentation and detection of different specific objects.

The authors are with the Department of Computer Science, at the University of Adelaide, Adelaide, SA 5005, Australia.
{cesar.cadena, anthony.dick, ian.reid}@adelaide.edu.au

We are extremely grateful to the Australian Research Council for funding this research through project DP130104413, the ARC Centre of Excellence for Robotic Vision C E140100016, and through a Laureate Fellowship FL130100102 to IDR.

Commonly in the past these different perception tasks were approached independently. However recently a number of works have demonstrated that holistic approaches that combine different scene understanding methods can outperform the independent methods [7,12,13,19,20,29,30,32].

Although solving for all components of the scene holistically is arguably the correct thing to do, these approaches typically suffer from some important drawbacks that prevent their use in practical robotic operations. Most holistic approaches are based on solving the same fixed set of perception tasks at each frame. For instance, suppose that a holistic model is learnt to carry out semantic segmentation and to detect 3 different objects (e.g. birds, cars and hydrants). If during operation only car detections are required, the holistic system must either learn a new model or continue to redundantly detect birds and hydrants. Training a holistic model for every possible combination of subtasks is infeasible.

Contribution: In contrast to a holistic approach, we claim that different perception tasks should be treated as different (software) modules that can be activated or deactivated at will without impairing the rest of the system. But at the same time, they should communicate their outputs to improve the performance of each individual task.

A schematic of the system proposed in this paper is shown in Fig. 1, where we have a dedicated thread for semantic segmentation into coarse classes and multiple threads for different specific object detectors. The semantic segmentation sends information to the object detectors, and the

object detectors feedback their evidence to the semantic segmentation system. Our main contribution is two fold:

- We use the semantic segmentation as context to improve the specific object detection;
- We propose a principled update model to take into account the evidence of detected objects in the semantic segmentation.

Our approach is robust to failures in individual software components, and is flexible and modular in the number and type of components it uses.
In the next section, we review the relevant related work, then in Sections III and IV we present in detail our main contributions. In Section V we describe the experiments in urban real environments, and finally, in Section VI we present discussions and conclusions of the presented work.

II. Related Work

In the computer vision community a large variety of approaches has been proposed to obtain a semantic segmentation of the scene typically using a Markov random field framework [8,19,21,26,28,31]. With the exception of few [8,19], most commonly the graph structure is induced by super-pixels obtained by over-segmentation algorithms on individual pixels, or by the stixels representation [26]. All of these approaches contain a combination of textural "stuff" classes (e.g. ground, sky) and object or "thing" classes (e.g. car, trash bin).

On the other hand, the proposed systems of [2,10,15,25] segment the scene into coarser classes (e.g. horizontal, vertical, mix and sky). We opt for this kind of approach as it is valid in many general scenarios and useful enough for basic robotic operation. Note that any known or unknown finer class is contained within a coarse class.

Different approaches have been proposed employing context information to improve object detectors [14,16,22] In [14] the context is learnt in terms of relationships between regions and the object of interest in an unsupervised manner using appearance. [16] integrates the reasoning about viewpoint and geometry (surface orientation) in the object detection. [22] incorporates context using a deformable part based model that accounts for appearance as well as for all other semantic classes. Instead of learning a latent SVM model on all possible finer classes, we evaluate the compatibility of the detections with our coarse semantic representation in the so called "semantic space" [24]. To our knowledge, this is the first time that the semantic space is used as a context for a task other than scene recognition/classification.

In the robotics community there has likewise been a trend towards holistic interpretations of the scene. For example, the work of [30] uses a hierarchical semantic labeling model to solve the semantic segmentation during a robotic task. Although the hierarchical approach is simplified and adapts the labels to the task at hand, their system still requires minutes to obtain the labeling for one frame to plan the next movement of the robot, while the environment is assumed to be static.

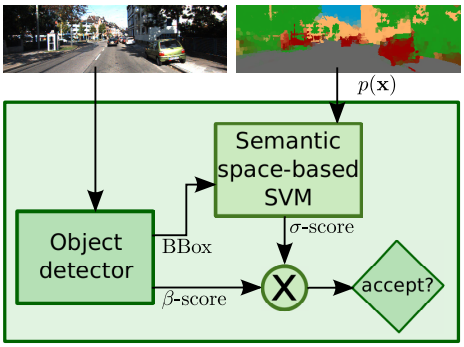


Fig. 2: Block diagram for the evaluation of the semantic context on the candidate bounding boxes. The decision is taken after re-scoring the bounding box score β-score with the semantic context score σ-score. On the top we show an example of the inputs to this subsystem, the original image and the probability map over the semantic classes with color code: ■*ground*, ■*objects*, ■*building* and ■*vegetation* (the intensity of each color is proportional to the probability of each class).

III. Using Context to Improve Specific Object Detectors

In this section we describe our first contribution, the use of the semantic segmentation to provide context for specific object detection, Fig. 2.

Given an image I, and the pixel-wise probability map of belonging to coarser classes $p(\mathbf{x})$, we want to improve the discriminative power of any specific object detector on I. In this work we will use five coarse classes: {*ground, building, vegetation, sky, object*}, where the class *object* represents the general object class. The specific object detector can be implemented by any out-of-the-box state of the art method. These specific objects are a subset of the coarse class *object*.

A. Semantic Space as Context

For a candidate detection of an specific object, we want to know if it is compatible with the probability distribution over the semantic classes where its bounding box is located. To evaluate this compatibility we use high level semantic features that can be represented in a low dimensional space, the *semantic space* [24]. The semantic space is defined as the space formed by the probability simplex $\mathbb{P}^{M-1}$ of M semantic classes, see Fig. 3(top). A vector $\mathbf{s}$ in this space for the bounding box can be computed by averaging over its pixelwise probability map[1].

Ideally, the context should include the global nature of the scene, for instance, a sea scene context helps to discard car detection candidates. For the present work we constrain ourselves to urban environments and use only the local context of the candidate detection. To capture a variety of possible context layouts, we use two overlapping grids based on the bounding box and expanding it by 66% in each spatial coordinate. The total number of cells is n_c=41 in the configuration shown in Fig. 4(right), 25 from the black cells and 16 from the red cells. We compute $\mathbf{s}_i$ for each

[1]Alternative options to the average are possible, e.g. computing the normalized histogram over the maximum a posteriori assignment or by max-pooling kind of operation. We have tested those options, however they did not improve the results.

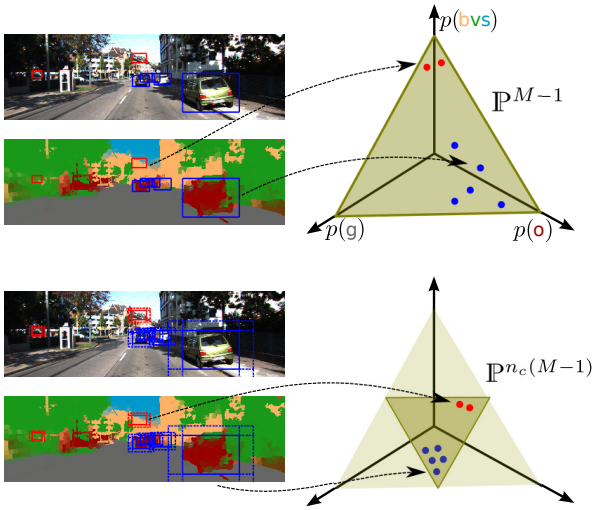


Fig. 3: Top: Representation of bounding boxes in semantic space (or simplex) through a vector $\mathbf{s} \in \mathbb{P}^{M-1}$, $\mathbf{s}$ is a vector of marginal probabilities of the box belonging to *ground* (g), *object* (o), *building* (b), *vegetation* (v) or *sky* (s). Bottom: The same representation is obtained for each cell in a n_c-cell grid centered on the candidate detection, then the n_c vectors are concatenated and re-normalized obtaining a vector $\mathbf{s} \in \mathbb{P}^{n_c(M-1)}$ which encodes the co-occurrence of different semantic classes around the candidate. The semantic space is simplified for illustration purposes.

cell i, concatenate and re-normalize them, to obtain a vector $\mathbf{s} \in \mathbb{P}^{n_c(M-1)}$ which represents the context of the candidate detection, see Fig. 3(bottom).

One way to compute the distance between two points in the probabilistic simplex is by using a geodesic distance. In general geodesic distances are hard to compute, but in the case of the probability simplex it is possible to exploit the isomorphism $F : \mathbb{P}^{M-1} \to \mathbb{S}_+^{M-1}$ by $\mathbf{s} \mapsto 2\sqrt{\mathbf{s}}$, where $\mathbb{S}_+^{M-1}$ is the positive subset in the sphere of radius 2 in $\mathbb{R}^M$ [17,18]. In the same way we can map $\mathbf{s} \in \mathbb{P}^{n_c(M-1)}$ to $\mathbb{S}_+^{n_c(M-1)}$. A distance between two points on the sphere can be computed as the length of the arc that connects them, which in fact corresponds to the Bhattacharya distance:

$$D(\mathbf{s}, \mathbf{s}^*) = 2 \arccos(\langle \sqrt{\mathbf{s}}, \sqrt{\mathbf{s}^*} \rangle) \tag{1}$$

We use the semantic kernel directly from the geodesic distance as used in [18]:

$$k(\mathbf{s}, \mathbf{s}^*) := -D(\mathbf{s}, \mathbf{s}^*) \tag{2}$$

This kernel is then used in a SVM classifier to re-score the initial bounding box score. Fig. 2 depicts the block diagram of this process. An image is processed by a specific object detector which outputs the bounding boxes (BBox) and their confidence scores (β). We assume β-score $\in [0, 1]$. With the location of the BBox-es and the semantic probability map $p(\mathbf{x})$ we compute the vector $\mathbf{s}$ for the n_c-cells grid. Vector $\mathbf{s}$ is then evaluated in the SVM classifier with kernel from Eq. 2, obtaining the SVM score (σ-score $\in [0, 1]$) of belonging to the positive class (true detection) given the semantic context. As β and σ are computed independently, the final acceptance decision is based on the product of β-score by σ-score.

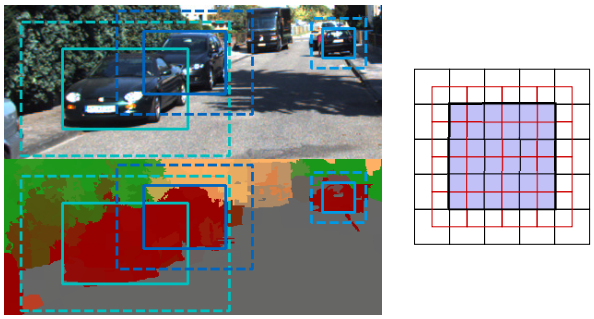


Fig. 4: On the left, original image and the corresponding semantic probability map. We show three bounding box for car (solid lines) and the context we want to capture (dashed lines). On the right, structure of the grid used over the bounding box (shaded). Though simple, the n_c cells grid obtained for each detection captures different meaningful configuration in the semantic space.

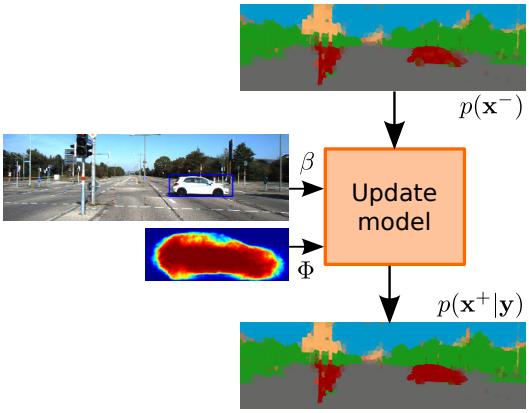


Fig. 5: The update model receives the initial probability map $p(\mathbf{x}^-)$ and the evidence from the object detector to obtain the posterior probability distribution over the coarse classes $p(\mathbf{x}^+|\mathbf{y})$.

IV. Using Object Detectors to Improve Generic Object Class

In this section we describe our second contribution, a principled way to update the semantic probability map with the detections found in image I.

Given as inputs the location (BBox) and confidence (β) of specific objects detected on the current scene, we want to update the pixelwise semantic probability map $p(\mathbf{x})$ in the corresponding object location. We illustrate this process in Fig. 5. For different objects, or instances of an object, we can assume that we have access to some information about the 2D shape of that instance, encoded as a shape prior Φ. The shape prior encourages the update to be consistent with the specific instance.

A. Update Model

For each pixel x we have $p(\mathbf{x}^-)$, the initial probability that it belongs to each coarse class, obtained from semantic segmentation. We also have the indicator variable $\mathbf{y}$ which is 1 when x lies within a detected object's bounding box, and 0 otherwise. We can approximate the posterior of the semantic probability per pixel $p(\mathbf{x}^+|\mathbf{y})$ by:

$$p(\mathbf{x}^+|\mathbf{y} = 1) \propto p(\mathbf{y} = 1|\mathbf{x}^-)p(\mathbf{x}^-) \tag{3}$$

$$p(\mathbf{x}^+|\mathbf{y}=0) = p(\mathbf{x}^-) \tag{4}$$

The intuition behind Eq. 4 is that if there are no specific objects occupying the pixel location, then there is no evidence to modify the semantic probability. Note that the fact of not detecting one specific object does not mean negative evidence to the general *object* class as this class contains all other possible objects. If there is a specific object occupying the pixel location we increase the semantic probability of the general *object* class as we have more evidence in its favour, Eq. 3.

We can further expand Eq. 3 to model the performance of the detector as follows:

$$\begin{aligned} p(\mathbf{y}=1|\mathbf{x}^-) \quad = \quad & \overbrace{p(\mathbf{y}=1|\mathbf{e}=1)}^{\text{true positive rate}} p(\mathbf{e}=1|\mathbf{x}^-) \\ & + \underbrace{p(\mathbf{y}=1|\mathbf{e}=0)}_{\text{false positive rate}} p(\mathbf{e}=0|\mathbf{x}^-) \end{aligned} \tag{5}$$

where variable $\mathbf{e}$ models object existence. The true and false positive rates, TPR and FPR respectively, can be obtained at training time using a validation set.

We model the conditional probability distribution that the specific object exists $p(\mathbf{e}|\mathbf{x}^-)$ by:

$$\begin{aligned} p(\mathbf{e}=1|\mathbf{x}^-) = & \begin{cases} \frac{1+\beta\Phi_x}{2} & \text{if } \mathbf{x}^- = object \\ \frac{1-\beta\Phi_x}{2} & \text{if } \mathbf{x}^- \neq object \end{cases} \\ p(\mathbf{e}=0|\mathbf{x}^-) = & \quad 1 - p(\mathbf{e}=1|\mathbf{x}^-) \end{aligned} \tag{6}$$

where Φ_x the shape prior evaluated at the pixel location and β is the confidence of the specific object detector.

The conditional probability of existence of the specific object $p(\mathbf{e}=1|\mathbf{x}^-)$ is proportional to the confidence of the specific detector β and a shape prior Φ for that specific object. If the confidence of the detector tends to zero $\beta \to 0$, or if the average shape of the object does not cover the current pixel location $\Phi_x \approx 0$, the conditional probability $p(\mathbf{e}=1|\mathbf{x}^-)$ becomes a uniform distribution as the evidence in favour of the existence of the specific object at the pixel location decreases. In the opposite case, high confidence from the detector $\beta \to 1$, and the evaluated pixel inside the average shape $\Phi_x \approx 1$, the conditional probability would be one for $\mathbf{x}^-$ in the general *object* class and zero otherwise.

As final consideration, a pixel could obtain evidence from detectors for different objects, for example, a pixel belonging to a person occluding a car could be contained in the bounding box for pedestrians as well as for cars. In this case, we carry out the update using the independent opinion pool [1]:

$$p(\mathbf{x}^+|\mathbf{y}_{1\cdots n}=1) \propto \left(\prod_j p(\mathbf{y}_j=1|\mathbf{x}^-) \right) p(\mathbf{x}^-) \tag{7}$$

with n different object detections.

B. Tracking

We are interested in a continuous operation in a dynamic environment where we are gathering data in a sequential way. As the scene changes by the robot's ego-motion or by the dynamic of other objects (e.g. people), the detectors can miss true detections in a given frame because of hard occlusions. To incorporate a temporal consistency in those cases we add a simple Kalman filter tracker for each detection with a constant velocity motion model in the image space.

V. Experimental Setup and Results

The popular KITTI dataset [11] is used to demonstrate the improvements and advantages of our proposal. First, we present our results in an object detection benchmark where we demonstrate that our approach to incorporate the semantic context in the detections outperforms out-of-the-box detectors. Next, we show the improvements on the coarse semantic segmentation when taking into account the evidence from specific object detectors.

A. Improving Object Detectors

We evaluated the specific object detectors and the improvement by our semantic context using the KITTI object detection benchmark [11]. A dataset with 7481 training images and 7518 testing images is provided. We divide the original training images in a training set (4000 images) and a validation set (3481 images). The dataset provides the evaluation code used for the benchmark to compute Precision-Recall curves. The Average Precision (AP) is used as a single metric of comparison.

1) Specific Object Detector: We perform the evaluation on the benchmark for two specific objects, Pedestrian and Car, and select as out-of-the-box detector the aggregate channel features (ACF) proposed by [6].[2] ACF is a very efficient object detector with competitive accuracy in detection [6]. We train ACF on the training set for one model of pedestrian and eight models for cars centred on eight orientations every 45 degrees.

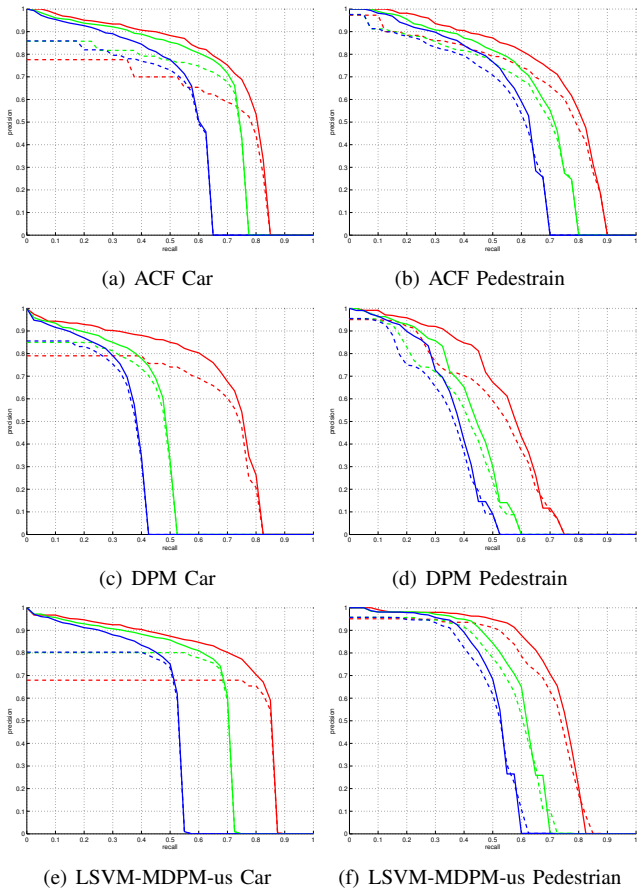
2) Semantic space-based SVM: We use the semantic segmentation system proposed in [2][3]. With this system we obtain a pixelwise probability map for five coarse classes: $\{$*ground, buildings, vegetation, sky, objects*$\}$.

The semantic segmentation was obtained for the training set and the semantic features ((25+16) cells x 5 classes = 205 dimensional vector) are efficiently computed using integral images on each class. We use libSVM [4] for the SVM-classifiers, one for each specific object. We use the ground truth bounding boxes as the positive samples and the negative ones are harvested from false positives of ACF in the training set.

3) Validation: We report the results on the validation set in terms of AP for the baseline detector alone and with the re-scoring by the semantic context in Table I, ACF and ACF+SC respectively, for Car detection and for Pedestrian

[2]Available from "Piotr's Matlab Toolbox" [5]

[3]Available online at `http://cs.adelaide.edu.au/~cesar/research.html`



(a) ACF Car (b) ACF Pedestrain

(c) DPM Car (d) DPM Pedestrain

(e) LSVM-MDPM-us Car (f) LSVM-MDPM-us Pedestrian

Fig. 6: Precision-Recall curves for the out-of-the-box specific object detectors (dashed lines) and their improvements by using the semantic context (solid lines) on validation set. Line colors follow the benchmark convention for the level of difficulty: red (Easy), green (Moderate), blue (Hard).

detection. In Figs. 6(a) and 6(b) we compare the Precision-Recall curves generated by the baseline (dashed line) and by using the semantic context (solid lines) for different difficulty levels, where a consistent improvement in precision is observed in all cases.

To demonstrate the modularity of our approach we evaluate the same (no-retraining) semantic space-based SVM with other baseline detectors. The first one is the popular deformable part-based model (DPM) [9] version 5[4]. We use the pre-trained pedestrian and car models and run the code directly on the validation set. The second baseline is also a DPM based one (LSVM-MDPM-us) as reported in the benchmark by [12]. The pre-trained models for this detector are available with the KITTI object detection benchmark. We expect a better results for this baseline with respect to the previous ones as LSVM-MDPM-us was trained with all the original training images, covering our validation set.

In Table I and Figs. 6(c-f) we observe that by using the semantic context we consistently outperform every baseline, even when it was trained for a different baseline (ACF) detector.

[4]Code and pre-trained models available at `http://www.cs.berkeley.edu/~rbg/latent/index.html`

	Easy	Moderate	Hard
Car			
ACF	56.83%	59.30%	48.84%
ACF + SC	71.42%	65.64%	53.56%
DPM	56.97%	39.57%	31.82%
DPM + SC	66.38%	42.68%	34.07%
LSVM-MDPM-us	57.41%	55.72%	42.27%
LSVM-MDPM-us + SC	74.59%	62.30%	47.12%
Pedestrian			
ACF	67.14%	59.64%	52.27%
ACF + SC	72.96%	63.60%	55.61%
DPM	48.79%	37.99%	33.05%
DPM + SC	54.24%	42.10%	36.34%
LSVM-MDPM-us	67.16%	56.00%	48.48%
LSVM-MDPM-us + SC	71.07%	58.77%	50.36%

TABLE I: Average precision on validation set.

	Easy	Moderate	Hard
Car			
ACF	55.89%	54.74%	42.98%
ACF + SC	69.11%	58.66%	45.95%
Pedestrian			
ACF	44.49%	39.81%	37.21%
ACF + SC	51.53%	44.49%	40.38%

TABLE II: Average precision on testing set.

4) Benchmark: ACF and the semantic context module were further trained with all training images and submitted to the benchmark. Both results can be found under ACF and ACF+SC method names on the KITTI Object Detection Evaluation website at `http://www.cvlibs.net/datasets/kitti/evalobject.php`, where further details about the parameters for training are also available[5].

In Table II we reproduce the reported results by the evaluation system. The improvements by using the semantic context with respect to the raw detector in the benchmark are consistent with those obtained in the validation set, an improvement of 13(easy), 4(moderate) and 3(hard) percentage points for the Car detection benchmark, and 7(easy), 5(moderate) and 3(hard) percentage points for the Pedestrian detection benchmark.

A joint inference (holistic) scene understanding system by [12] modified the LSVM-MDPM-us baseline and reported their improvements under the method name LSVM-MDPM-sv in the benchmark website. As a comparison their relative improvement is 3(easy), 2(moderate) and 3(hard) for Car, and 1(easy), 2(moderate) and 1(hard) for Pedestrian. In the validation set we outperform this relative improvement for the same baseline with our approach.

[5]Actually, we have tried to submit other baselines (+ SC) but it was not possible due to benchmark's policies.

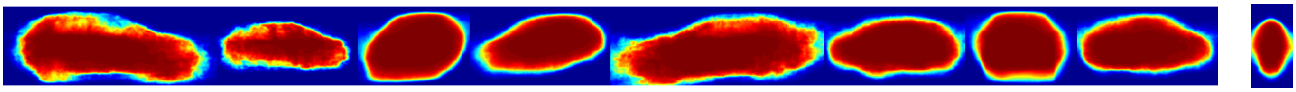


Fig. 7: Shape priors Φ, for eight orientation (models) of cars and for one model of pedestrian (rightmost).

Object	TPR	FPR
Pedestrian	0.6	0.32
Car	0.8	0.17

TABLE III: True Positive Rate selected for each object and its corresponding False Positive Rate as computed in the validation set.

B. Improving the Semantic Segmentation

We evaluate the improvement on semantic segmentation accuracy on KITTI using 70 manually labelled images as ground truth made available by [27], with a split of 45 for training and 25 for testing. The labeling is mapped to the 5 coarser classes as described in [2]. We further extend the testing set by manually labeling other 132 images in four other sequences[6]. Resulting in a testing set of 157 images from 5 different sequences.

1) Semantic Segmentation: We use the semantic segmentation system for single frame proposed by [2] and augmented with the recursive inference for video sequences as described by [3]. In short, the system performs the semantic segmentation in a given time k using a graphical model over a superpixel over-segmentation. Then, using the odometry information, the marginal semantic probabilities are propagated to $k+1$ as observed nodes in the graphical model.

After obtaining the semantic probability map in time k, $p(\mathbf{x}_k^-)$, we update it with the evidence gathered from the specific object detectors.

2) Object Detector, Shape Prior and Statistics: Our update model requires the candidate bounding box location and its score (BBox, β), the true positive rate (TPR = $p(\mathbf{y} = 1|\mathbf{e} = 1)$, the false positive rate (FPR = $p(\mathbf{y} = 1|\mathbf{e} = 0)$, and the shape prior ($\Phi$) of the specific objects we are detecting.

From our previous evaluation we select ACF+SC as object detector. In the object validation set we compute the FPR and thresholds for the TPR required for each object, Table III, without distinguish among the three difficulty levels. Here, we consider a true detection if the Jaccard coefficient (or VOC-PASCAL overlap) is greater or equal than 0.5.

The shape prior is obtained in the following way. The semantic probability map is computed for the 4000 images in the object training set, then we take into account only the probability to belong to the general *object* class inside the ground truth bounding boxes for each object's model. Once all the evidence is properly re-scaled and accumulated, the final shape prior Φ is obtained by evaluate a logistic function on it. Fig. 7 depicts the shape prior that we have obtained by this procedure.

[6]Corresponding to KITTI's raw sequences: 2011_09_26_drive_0052 (26 images), 2011_09_26_drive_0017 (58 images), 2011_09_26_drive_0020 (15 images) and 2011_09_28_drive_0039 (33 images).

	Precision	Recall	F_1-score
Baseline [2,3]	89.4	85.2	87.2
Baseline + Ped. detection	89.3	86.8	88.0
Baseline + Car. detection	89.1	86.5	87.8
Baseline + Ped. & Car det.	89.0	88.1	88.6
Bsl. + Ped & Car det. + tracking	88.9	88.3	88.6

TABLE V: *Object* segmentation results in pixel-wise percentage.

3) Tracking: The tracking system is implemented to track the candidate bounding boxes of the raw detector, before the semantic context evaluation. A candidate (track) is deleted if it is rejected by the semantic context module or if it has been unobserved for two consecutive frames.

4) Evaluation: We report the recall accuracy for the 5 coarse classes, as well as the class average and global accuracy, in Table IV. Given that the object class in this setting contains all possible objects the ratio of pixels that belong to cars or pedestrians with respect to the general *object* class is very low, the changes in the accuracy for the general *object* class due to any change in cars or pedestrian are also relatively small. Because of this, we report a second set of results in Table V where we take into account only the pixels inside of the final accepted specific object detections and compute the precision, recall and F_1-score for the general *object* class.

The results of the semantic segmentation system without updating with the specific object detectors are reported in Tables IV and V as our baseline method. By updating the semantic probability map using the evidence from the detections (Baseline + Ped. & Car detection) the recall is increased in 3 percentage points while the precision decreased in less than 1 percentage point. After including the object tracking system the recall is further increased. The F_1-score has always increased meaning that the gain in recall overcomes the slight decrement in precision. This decrement is due to the shape prior doesn't align perfectly with the specific object, then we are prone to update, for example, the ground under a car, between the tires, to the general *object* class. In Fig. 8 we show some examples of the improvement in the semantic segmentation.

The system increases the accuracy of the semantic segmentation even when we disable one of the detectors, see Tables IV and V.

C. Timing

The experiments were carried out with a research implementation of our approach in Matlab and C++ on a IntelCore i7-4820K CPU @ 3.7GHz ×8 and 31.4GB of RAM, with GPU GeForce GTX 680/PCIe/SSE2. The computational cost is detailed in Table VI. We provide the timing for the sake of

	ground	*objects*	*building*	*vegetation*	*sky*	Average	Global
Baseline [2,3]	90.3	61.6	83.2	90.9	85.5	82.3	81.7
Baseline + Pedestrian detection	90.2	**62.1**	83.2	90.9	85.5	82.4	81.8
Baseline + Car detection	90.2	**61.9**	83.2	90.9	85.5	82.3	81.8
Baseline + Ped. & Car detection	90.2	**62.5**	83.2	90.9	85.5	82.4	81.9
Baseline + Ped. & Car det. + tracking	90.2	**62.5**	83.2	90.9	85.5	82.4	81.9

TABLE IV: Semantic segmentation recall accuracy in pixel-wise percentage. Note that the specific object detectors do not affect the accuracy of other coarse classes (see text for further details) while improving the *object* class by a small amount.

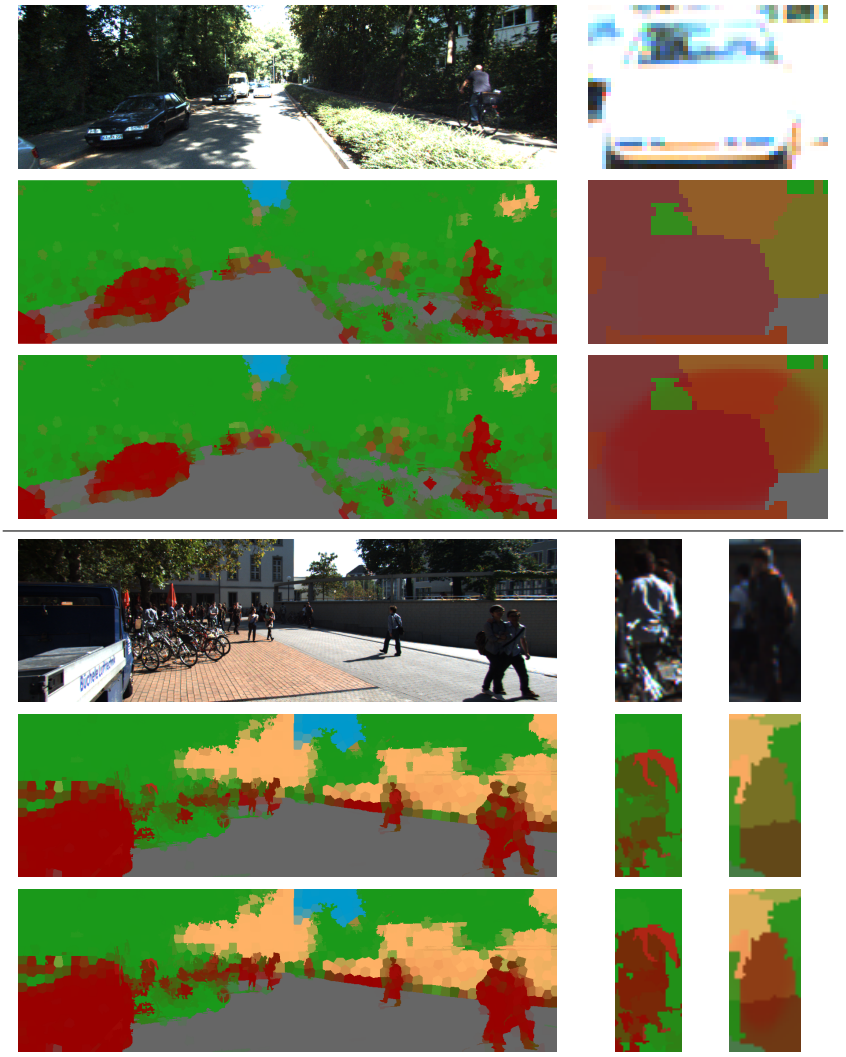


Fig. 8: We show some examples of the achieved improvement by updating the semantic segmentation with the detected objects as described in Section V-B. In each case the first row corresponds to the original image, second row is the semantic segmentation alone $p(\mathbf{x}^-)$, and the third one corresponds to the updated model $p(\mathbf{x}^+|\mathbf{y})$. On the right we zoom in some of the detections. Please note the higher probability of belonging to the general object class after the update.

completeness, however these numbers should be taken only as reference given that the components are not yet in a fully integrated system. For the specific object detection we use Piotr's Matlab Toolbox [5] running the pedestrian detection on only one core and the car detection on all eight cores, one per model.

VI. Discussion

In this paper we have presented a modular scene understanding system as an alternative to holistic models. Our approach to the scene understanding is driven by robotics operation where important aspects for software components such as robustness, efficient management, flexibility and task-relevant criteria must to be taken into account.

We have demonstrated that a coarse semantic interpretation of a scene is of value as context for different specific

Component	Runtime	Implementation
Semantic Segmentation	176ms/frame	C++/GPU
Specific Obj. detection		
Pedestrian	148ms/frame	Matlab/mex
Car	130ms/frame/model	Matlab/mex
Context Evaluation		
Pedestrian		
Feat. and Kernel	23ms/frame	Matlab
libSVM	41ms/frame	mex
Car		
Feat. and Kernel	24ms/frame	Matlab
libSVM	58ms/frame	mex
Sem. Segmen. Update	30ms/frame	Matlab

TABLE VI: Detailed computational timing per component.

object detectors. We have used the so called semantic space to strengthen or weaken candidate detections leading to a boost in performance over different out-of-the-box state of the art object detectors. We will carry out a more in-depth study in the optimal resolution and configuration of the grid to extract the context that can potentially lead to even better improvements.

The specific object detector modules can be turned on/off at any time without changing any particular components in the semantic segmentation thread. In cases where we turn on the software modules for specific object detectors we use their evidence about detections to update the coarse semantic probability map of the scene.

As future work, we want to enrich our scene understanding system, taking advantage of its modularity, with other perceptual tasks, like 3D shape alignment/recovery and more sophisticated tracking systems [23].

REFERENCES

[1] J.O. Berger. *Statistical decision theory and Bayesian analysis*. Springer, 1985.

[2] C. Cadena and J. Košecká. Semantic segmentation with heterogeneous sensor coverages. In *Proc. IEEE Int. Conf. Robotics and Automation*, Hong Kong, China, June 2014.

[3] C. Cadena and J. Košecká. Recursive Inference for Prediction of Objects in Urban Environments. In *International Symposium on Robotics Research*, Singapore, December 2013.

[4] C.C. Chang and C.J Lin. Libsvm: A library for support vector machines. *ACM Trans. Intell. Syst. Technol.*, 2(3), May 2011.

[5] P. Dollár. Piotr's Image and Video Matlab Toolbox (PMT). `http://vision.ucsd.edu/~pdollar/toolbox/doc/index.html`.

[6] P. Dollár, R. Appel, S. Belongie, and P. Perona. Fast feature pyramids for object detection. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 2014.

[7] J. Dong, Q. Chen, S. Yan, and A. Yuille. Towards unified object detection and semantic segmentation. In *Computer Vision–ECCV 2014*, pages 299–314. Springer, 2014.

[8] C. Farabet, C. Couprie, L. Najman, and Y. LeCun. Learning hierarchical features for scene labeling. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, PP(99):1–1, 2013.

[9] P.F. Felzenszwalb, R.B. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 32:1627–1645, 2010.

[10] A. Flint, D. Murray, and I.D. Reid. Manhattan scene understanding using monocular, stereo, and 3d features. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 2228–2235. IEEE, 2011.

[11] A. Geiger, P. Lenz, and R. Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[12] A. Geiger, C. Wojek, and R. Urtasun. Joint 3d estimation of objects and scene layout. In *Advances in Neural Information Processing Systems*, 2011.

[13] G. Heitz, S. Gould, A. Saxena, and D. Koller. Cascaded classification models: Combining models for holistic scene understanding. In *Advances in Neural Information Processing Systems 21*, pages 641–648, 2009.

[14] G. Heitz and D. Koller. Learning spatial context: Using stuff to find things. In *Computer Vision ECCV 2008*, pages 30–43, 2008.

[15] D. Hoiem, A.A. Efros, and M. Hebert. Geometric context from a single image. In *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, pages 654–661. IEEE, 2005.

[16] D. Hoiem, A.A. Efros, and M. Hebert. Putting objects in perspective. *International Journal of Computer Vision*, 80(1):3–15, 2008.

[17] A. Jencova. Geodesic distances on density matrices. *Journal of Mathematical Physics*, 45(5):1787–1794, 2004.

[18] R. Kwitt, N. Vasconcelos, and N. Rasiwasia. Scene recognition on the semantic manifold. In *Computer Vision–ECCV 2012*, pages 359–372. Springer, 2012.

[19] L. Ladický, P. Sturgess, K. Alahari, C. Russell, and P.H.S. Torr. What, Where and How Many? Combining Object Detectors and CRFs. In *Computer Vision - ECCV 2010*. Springer Berlin Heidelberg, 2010.

[20] C. Li, A. Kowdle, A. Saxena, and T. Chen. Towards holistic scene understanding: Feedback enabled cascaded classification models. In *Advances in Neural Information Processing Systems 23*, pages 1351–1359, 2010.

[21] B. Micusik, J. Košecká, and G. Singh. Semantic parsing of street scenes from video. *The International Journal of Robotics Research*, 31(4):484–497, 2012.

[22] R. Mottaghi, X. Chen, X. Liu, S. Fidler, R. Urtasun, and A. Yuille. The role of context for object detection and semantic segmentation in the wild. In *CVPR*, 2014.

[23] V.A. Prisacariu and I.D. Reid. Pwp3d: Real-time segmentation and tracking of 3d objects. *International journal of computer vision*, 98(3):335–354, 2012.

[24] N. Rasiwasia and N. Vasconcelos. Scene classification with low-dimensional semantic spaces and weak supervision. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–6, June 2008.

[25] S.H. Raza, M. Grundmann, and I. Essa. Geometric context from videos. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3081–3088, June 2013.

[26] T. Scharwächter, M. Enzweiler, U. Franke, and S. Roth. Stixmantics: A medium-level model for real-time semantic scene understanding. In *Computer Vision–ECCV 2014*, pages 533–548. Springer, 2014.

[27] S. Sengupta, E. Greveson, A. Shahrokni, and P.H.S. Torr. Urban 3D Semantic Modelling Using Stereo Vision. In *Proc. IEEE Int. Conf. Robotics and Automation*, 2013.

[28] G. Singh and J. Košecká. Nonparametric scene parsing with adaptive feature relevance and semantic context. In *Computer Vision and Pattern Recognition (CVPR), IEEE Conference on*, 2013.

[29] C. Wojek and B. Schiele. A dynamic conditional random field model for joint labeling of object and scene classes. In *Computer Vision ECCV 2008*, pages 733–747, 2008.

[30] C. Wu, I. Lenz, and A. Saxena. Hierarchical semantic labeling for task-relevant rgb-d perception. In *Robotics: Science and Systems (RSS)*, 2014.

[31] J. Xiao and L. Quan. Multiple view semantic segmentation for street view images. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 686 –693, oct. 2009.

[32] J. Yao, S. Fidler, and R. Urtasun. Describing the scene as a whole: Joint object detection, scene classification and semantic segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 702 –709, june 2012.