A Dynamic Conditional Random Field Model for Joint Labeling of Object and Scene Classes

Christian Wojek and Bernt Schiele

Computer Science Department
TU Darmstadt
{wojek,schiele}@cs.tu-darmstadt.de

Abstract. Object detection and pixel-wise scene labeling have both been active research areas in recent years and impressive results have been reported for both tasks separately. The integration of these different types of approaches should boost performance for both tasks as object detection can profit from powerful scene labeling and also pixel-wise scene labeling can profit from powerful object detection. Consequently, first approaches have been proposed that aim to integrate both object detection and scene labeling in one framework. This paper proposes a novel approach based on conditional random field (CRF) models that extends existing work by 1) formulating the integration as a joint labeling problem of object and scene classes and 2) by systematically integrating dynamic information for the object detection task as well as for the scene labeling task. As a result, the approach is applicable to highly dynamic scenes including both fast camera and object movements. Experiments show the applicability of the novel approach to challenging real-world video sequences and systematically analyze the contribution of different system components to the overall performance.

1 Introduction

Today, object class detection methods are capable of achieving impressive results on challenging datasets (e.g. PASCAL challenges [1]). Often these methods combine powerful feature vectors such as SIFT or HOG with the power of discriminant classifiers such as SVMs and AdaBoost. At the same time several authors have argued that global scene context [2,3] is a valuable cue for object detection and therefore should be used to support object detection. This context-related work however has nearly exclusively dealt with static scenes. As this paper specifically deals with highly dynamic scenes we will also model object motion as an additional and important cue for detection.

Pixel-wise scene labeling has also been an active field of research recently. A common approach is to use Markov or conditional random field (CRF) models to improve performance by modeling neighborhood dependencies. Several authors have introduced the implicit notion of objects into CRF-models [4,5,6,7]. The interactions between object nodes and scene labels however are often limited to uni-directional information flow and therefore these models have not yet shown

D. Forsyth, P. Torr, and A. Zisserman (Eds.): ECCV 2008, Part IV, LNCS 5305, pp. 733–747, 2008.

the full potential of simultaneously reasoning about objects and scene. By formulating the problem as a *joint* labeling problem for object and scene classes, this paper introduces a more general notion of object-scene interaction enabling bidirectional information flow. Furthermore, as we are interested in dynamic scenes, we make use of the notion of dynamic CRFs [8], which we extend to deal with both moving camera and moving objects.

Therefore we propose a novel approach to jointly label objects and scene classes in highly dynamic scenes for which we introduce a new real-world dataset with pixel-wise annotations. Highly dynamic scenes are not only a scientific challenge but also an important problem, e.g. for applications such as autonomous driving or video indexing where both the camera and the objects are moving independently. Formulating the problem as a joint labeling problem allows 1) to model the dynamics of the scene and the objects separately which is of particular importance for the scenario of independently moving objects and camera, and 2) to enable bi-directional information flow between object and scene class labels.

The remainder of this paper is structured as follows. Section 2 reviews related work from the area of scene labeling and scene analysis in conjunction with object detection. Section 3 introduces our approach and discusses how object detection and scene labeling can be integrated as a joint labeling problem in a dynamic CRF formulation. Section 4 introduces the employed features, gives details on the experiments and shows experimental results. Finally, section 5 draws conclusions.

2 Related Work

In recent years, conditional random fields (CRFs) [9] have become a popular framework for image labeling and scene understanding. However, to the best of our knowledge, there is no work which explicitly models object entities in *dynamic* scenes. Here, we propose to model objects and scenes in a joint labeling approach on two different layers with different information granularity and different labels in a dynamic CRF [8].

Related work can roughly be divided into two parts. First, there is related work on CRF models for scene understanding, and second there are approaches aiming to integrate object detection with scene understanding.

In [10] Kumar&Hebert detect man-made structures in natural scenes using a single-layered CRF. Later they extend this work to handle multiple classes in a two-layered framework [5]. Kumar&Hebert also investigated object-context interaction and combined a simple boosted object detector for side-view cars with scene context of road and buildings on a single-scale database of static images. In particular, they are running inference separately on their two layers and each detector hypothesis is only modeled in a neighborhood relation with an entire region on the second layer. On the contrary, we integrate multi-scale objects in a CRF framework where inference is conducted jointly for objects and context. Additionally, we propose to model edge potentials in a consistent layout by exploiting the scale given by a state-of-the-art object detector [11]. Torralba *et al.* [7] use boosted classifiers to model unary and interaction potentials in order

to jointly label object and scene classes. Both are represented by a dictionary of patches. However, the authors do not employ an object detector for entire objects. In our work we found a separate object detector to be essential for improved performance. Also Torralba *et al.* use separate layers for each object and scene class and thus inference is costly due to the high graph connectivity, and furthermore they also work on a single-scale database of static images. We introduce a sparse layer to represent object hypotheses and work on dynamic image sequences containing objects of multiple scales. Further work on simultaneous object recognition and scene labeling has been conducted by Shotton *et al.* [6]. Their confusion matrix shows, that in particular object classes where color and texture cues do not provide sufficient discriminative power on static images – such as boat, chair, bird, cow, sheep, dog – achieve poor results. While their Texton feature can exploit context information even from image pixels with a larger distance, the mentioned object classes remain problematic due to the unknown object scale. Furthermore, He *et al.* [4] present a multi-scale CRF which contains multiple layers relying on features of different scales. However, they do not model the explicit notion of objects and their higher level nodes rather serve as switches to different context and object co-occurrences. Similarly, Verbeek&Triggs [12] add information about class co-occurrences by means of a topic model. Finally, several authors proposed to adopt the CRF framework for object recognition as a standalone task [13,14,15] without any reasoning about the context and only report results on static single-scale image databases.

Dynamic CRFs are exploited by Wang&Ji [16] for the task of image segmentation with intensity and motion cues in mostly static image sequences. Similarly, Yin&Collins [17] propose a MRF with temporal neighborhoods for motion segmentation with a moving camera.

The second part of related work deals with scene understanding approaches from the observation of objects. Leibe *et al.* [18] employ a stereo camera system together with a structure-from-motion approach to detect pedestrians and cars in urban environments. However, they do not explicitly label the background classes which are still necessary for many applications even if all objects in the scene are known. Hoiem *et al.* [3] exploit the detected scales of pedestrians and cars together with a rough background labeling to infer the camera's viewpoint which in turn improves the object detections in a directed Bayesian network. Contrary to our work, object detections are refined by the background context but not the other way round. Also, only still images are handled while the presence of objects is assumed. Similarly, Torralba [2] exploits filter bank responses to obtain a scene prior for object detection.

3 Conditional Random Field Models

The following section successively introduces our model. It is divided into three parts: the first reviews single layer CRFs, the second additionally models objects in a separate layer and the last adds the scene's and objects' dynamics.

We denote the input image at time t with $\mathbf{x}^t$, the corresponding class labels at the grid cell level with $\mathbf{y}^t$ and the object labels with $\mathbf{o}^t$.

3.1 Plain CRF: Single Layer CRF Model for Scene-Class Labeling

In general a CRF models the conditional probability of all class labels $\mathbf{y}^t$ given an input image $\mathbf{x}^t$. Similar to others, we model the set of neighborhood relationships N_1 up to pairwise cliques to keep inference computationally tractable. Thus, we model

$$\log(P_{pCRF}(\mathbf{y}^t|\mathbf{x}^t, N_1, \Theta)) = \sum_i \Phi(y_i^t, \mathbf{x}^t; \Theta_\Phi) + \sum_{(i,j)\in N_1} \Psi(y_i^t, y_j^t, \mathbf{x}^t; \Theta_\Psi) - \log(Z^t) \tag{1}$$

Z^t denotes the so called partition function, which is used for normalization. N_1 is the set of all spatial pairwise neighborhoods. We refer to this model as *plain CRF*.

Unary Potentials. Our unary potentials model local features for all classes C including scene as well as object classes. We employ the joint boosting framework [19] to build a strong classifier $H(c, \mathbf{f}(x_i^t); \Theta_\Phi) = \sum_{m=1}^{M} h_m(c, \mathbf{f}(x_i^t); \Theta_\Phi)$. Here, $\mathbf{f}(x_i^t)$ denotes the features extracted from the input image for grid point i. M is the number of boosting rounds and c are the class labels. h_m are *weak learners* with parameters Θ_Φ and are shared among the classes for this approach. In order to interpret the boosting confidence as a probability we apply a softmax transform [5]. Thus, the potential becomes:

$$\Phi(y_i^t = k, \mathbf{x}^t; \Theta_\Phi) = \log \frac{\exp H(k, \mathbf{f}(x_i^t); \Theta_\Phi)}{\sum_c \exp H(c, \mathbf{f}(x_i^t); \Theta_\Phi)} \tag{2}$$

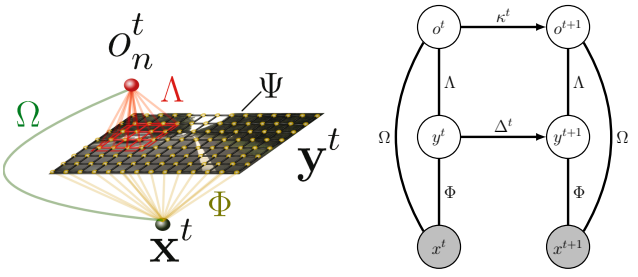
Edge Potentials. The edge potentials model the interaction between class labels at two neighboring sites y_i^t and y_j^t in a regular lattice. The interaction strength is modeled by a linear discriminative classifier with parameters $\Theta_\Psi = \mathbf{w}^T$ and depends on the difference of the node features $\mathbf{d}_{ij}^t := |\mathbf{f}(x_i^t) - \mathbf{f}(x_j^t)|$.

$$\Psi(y_i^t, y_j^t, \mathbf{x}^t; \Theta_\Psi) = \sum_{(k,l)\in C} \mathbf{w}^T \begin{pmatrix} 1 \\ \mathbf{d}_{ij}^t \end{pmatrix} \delta(y_i^t = k)\delta(y_j^t = l) \tag{3}$$

3.2 Object CRF: Two Layer Object CRF for Joint Object and Scene Labeling

Information that can be extracted from an image patch locally is rather limited and pairwise edge potentials are too weak to model long range interactions. Ideally, a complete dense layer of hidden variables would be added to encode possible locations and scales of objects, but since inference for such a model is computationally expensive we propose to inject single hidden variables $\mathbf{o}^t = \{o_1^t, \ldots, o_D^t\}$ (D being the number of detections) as depicted in figure 1(a). To instantiate those nodes any multi-scale object detector can be employed.

The additional nodes draw object appearance from a strong spatial model and are connected to the set of all corresponding hidden variables $\{\mathbf{y}^t\}_{o_n^t}$ whose



(a) (b)

Fig. 1. (a) Graphical model for the *object CRF*; note that different edge colors denote different potentials; (b) Graphical model for our full *dynamic CRF*; observed nodes are grey, hidden variables are white, for the sake of readability we omit the spatial layout of $\mathbf{y}^t$ with the corresponding edge potential Ψ

evidence $\{\mathbf{x}^t\}_{o_n^t}$ support the object hypotheses. The new nodes' labels in this work are comprised of $O = \{object, background\}$; but the extension to multiple object classes is straight forward. Thus, we introduce two new potentials into the CRF model given in equation (1) and yield the *object CRF*:

$$\log(P_{oCRF}(\mathbf{y}^t, \mathbf{o}^t | \mathbf{x}^t, \Theta)) = \log(P_{pCRF}(\mathbf{y}^t | \mathbf{x}^t, N_2, \Theta)) + \sum_n \Omega(o_n^t, \mathbf{x}^t; \Theta_\Omega) + \sum_{(i,j,n)\in N_3} \Lambda(y_i^t, y_j^t, o_n^t, \mathbf{x}^t; \Theta_\Psi) \tag{4}$$

Note that $N_2 \subset N_1$ denotes all neighborhoods where no object is present in the scene, whereas N_3 are all inter-layer neighborhoods with hypothesized object locations. Ω is the new unary object potential, whereas Λ is the inter-layer edge potential.

Unary Object Potentials. To define object potentials we use a state-of-the-art object detector. More specifically, we use a sliding window based multi-scale approach [11] where a window's features are defined by $\mathbf{g}(\{\mathbf{x}^t\}_{o_n^t})$ and classified with a linear SVM, the weights being $\mathbf{v}$ and b being the hyperplane's bias. To get a probabilistic interpretation for the classification margin, we adopt Platt's method [20] and fit a sigmoid with parameters s_1 and s_2 using cross validation.

$$\Omega(o_n^t, \mathbf{x}^t; \Theta_\Omega) = \log \frac{1}{1 + \exp(s_1 \cdot (\mathbf{v}^T \cdot \mathbf{g}(\{\mathbf{x}^t\}_{o_n^t}) + b) + s_2)} \tag{5}$$

Consequently, the parameters are determined as $\Theta_\Omega = \{\mathbf{v}, b, s_1, s_2\}$.

Inter-Layer Edge Potentials. For the inter-layer edge potentials we model the neighborhood relations in cliques consisting of two underlying first layer nodes y_i^t, y_j^t and the object hypothesis node o_n^t. Similar to the pairwise edge

potentials on the lower layer, the node's interaction strength is modeled by a linear classifier with weights $\Theta_\Lambda = \mathbf{u}$.

$$\Lambda(y_i^t, y_j^t, o_n^t, \mathbf{x}^t; \Theta_\Lambda) = \sum_{(k,l)\in C; m\in O} \mathbf{u}^T \begin{pmatrix} 1 \\ \mathbf{d}_{ij}^t \end{pmatrix} \delta(y_i^t = k)\delta(y_j^t = l)\delta(o_n^t = m) \quad (6)$$

It is important to note, that the inter-layer interactions are anisotropic and scale-dependent. We exploit the scale given by the object detector to train different weights for different scales and thus can achieve real multi-scale modeling in the CRF framework. Furthermore, we use different sets of weights for different parts of the detected object enforcing an object and context consistent layout [15].

3.3 Dynamic CRF: Dynamic Two Layer CRF for Object and Scene Class Labeling

While the additional information from an object detector already improves the classification accuracy, temporal information is a further important cue. We propose two temporal extensions to the framework introduced so far. For highly dynamic scenes – such as the image sequences taken by a driving car, which we will use as an example application to our model, it is important to note that objects and the remaining scene have different dynamics and thus should be modeled differently. For objects we estimate their motion and track them with a temporal filter in 3D space. The dynamics for the remaining scene is mainly caused by the camera motion in our example scenario. Therefore, we use an estimate of the camera's ego motion to propagate the inferred scene labels at time t as a prior to time step $t + 1$.

Since both – object and scene dynamics – transfer information forward to future time steps, we employ directed links in the corresponding graphical model as depicted in figure 1(b). It would have also been possible to introduce undirected links, but those are computationally more demanding. Moreover, those might not be desirable from an application point of view, due to the backward flow of information in time when online processing is required.

Object Dynamics Model. In order to model the object dynamics we employ multiple extended Kalman filters [21] – one for each object. For the dynamic scenes dataset which we will use for the experimental section the camera calibration is known and the sequences are recorded from a driving car. Additionally, we assume the objects to reside on the ground plane. Consequently, Kalman filters are able to model the object position in 3D coordinates. Additionally, the state vector contains the objects' width and speed on the ground plane as well as the camera's tilt and all state variables' first derivative with respect to time.

For the motion model we employ linear motion dynamics with the acceleration being modeled as system noise which proved sufficient for the image sequences used below. The tracks' confidences are given by the last associated detection's score. Hence, we obtain the following integrated model:

$$\log(P_{tCRF}(\mathbf{y}^t, \mathbf{o}^t | \mathbf{x}^t, \Theta)) = \log(P_{pCRF}(\mathbf{y}^t | \mathbf{x}^t, N_2, \Theta)) + \sum_n \kappa^t(o_n^t, \mathbf{o}^{t-1}, \mathbf{x}^t; \Theta_\kappa) + \sum_{(i,j,n)\in N_3} \Lambda(y_i^t, y_j^t, o_n^t, \mathbf{x}^t; \Theta_\Lambda) \quad (7)$$

where κ^t models the probability of an object hypothesis o_n^t at time t given the history of input images. It replaces the previously introduced potentials for objects Ω. The parameter vector consists of the detector's parameters and additionally of the Kalman filter's dynamics $\{A, W\}$ and measurement model $\{H_t, V_t\}$ and thus $\Theta_\kappa = \Theta_\Omega \cup \{A, W, H_t, V_t\}$.

Scene Dynamic Model. In the spirit of recursive Bayesian state estimation under the Markovian assumption, the posterior distribution of $\mathbf{y}^{t-1}$ is used as a prior to time step t. However, for dynamic scenes the image content needs to be transformed to associate the grid points with the right posterior distributions. In this work we estimate the projection Q from $\mathbf{y}^t$ to $\mathbf{y}^{t+1}$ given the camera's translation and calibration (Θ_{Δ^t}). Thus, we obtain an additional unary potential for $\mathbf{y}^t$.

$$\Delta^t(y_i^t, \mathbf{y}^{t-1}; \Theta_{\Delta^t}) = \log(P_{tCRF}(y^{t-1}_{Q^{-1}(i)} | \Theta)) \quad (8)$$

The complete *dynamic CRF* model including both object and scene dynamics as depicted in figure 1(b) then is

$$\log(P_{dCRF}(\mathbf{y}^t, \mathbf{o}^t, \mathbf{x}^t | \mathbf{y}^{t-1}, \mathbf{o}^{t-1}, \Theta)) = \log(P_{tCRF}(\mathbf{y}^t, \mathbf{o}^t | \mathbf{x}^t, \Theta)) + \sum_i \Delta^t(y_i^t, \mathbf{y}^{t-1}; \Theta_{\Delta^t}) \quad (9)$$

3.4 Inference and Parameter Estimation

For inference in the undirected graphical models we employ sum-product loopy belief propagation with a parallel message update schedule. For parameter estimation we take a piecewise learning approach [22] by assuming the parameters of unary potentials to be conditionally independent of the edge potentials' parameters. While this no longer guarantees to find the optimal parameter setting for Θ, we can learn the model much faster as discussed by [22].

Thus, prior to learning the edge potential models we train parameters Θ_Φ, Θ_Ω for the unary potentials. The parameter set Θ_κ for the Kalman filter is set to reasonable values by hand.

Finally, the edge potentials' parameter sets Θ_Ψ and Θ_Λ are learned jointly in a maximum likelihood setting with stochastic meta descent [23]. As proposed by Vishwanathan *et al.* we assume a Gaussian prior with meta parameter σ on the linear weights to avoid overfitting.

4 Experiments

To evaluate our model's performance we conducted several experiments on two datasets. First, we describe our features which are used for texture and location

based classification of scene labels on the scene label CRF layer. Then we introduce features employed for object detection on the object label CRF layer. Next, we briefly discuss the results obtained on the Sowerby database and finally we present results on image sequences on a new dynamic scenes dataset, which consist of car traffic image sequences recorded from a driving vehicle under challenging real-world conditions.

4.1 Features for Scene Labeling

Texture and Location Features. For the unary potential Φ at the lower level as well as for the edge potentials Ψ and inter-layer potentials Λ we employ texture and location features. The texture features are computed from the 16 first coefficients of the Walsh-Hadamard transform. This transformation is a discrete approximation of the cosine transform and can be computed efficiently [24,25] – even in real-time (e.g. on modern graphics hardware). The features are extracted at multiple scales from all channels of the input image in CIE *Lab* color space. As a preprocessing step, a and b channels are normalized by means of a gray world assumption to cope with varying color appearance. The L channel is mean-variance normalized to fit a Gaussian distribution with a fixed mean to cope with global lighting variations. We also found that normalizing the transformation's coefficients according to Varma&Zisserman [26] is beneficial. They propose to L_1-normalize each filter response first and then locally normalize the responses at each image pixel. Finally, we take the mean and variance of the normalized responses as feature for each node in the regular CRF lattice. Additionally, we use the grid point's coordinates within the image as a location cue. Therefore, we concatenate the pixel coordinates to the feature vector.

HOG. In the experiments described below we employ a HOG (Histogram of Oriented Gradients) detector [11] to generate object hypotheses. HOG is a sliding window approach where features are computed on a dense grid. First, histograms of gradient orientation are computed in *cells* performing interpolation with respect to the gradient's location and with respect to the magnitude. Next, sets of neighboring cells are grouped into overlapping *blocks*, which are normalized to achieve invariance to different illumination conditions. Our front and rear view car detector has a window size of 20×20 pixels. It is trained on a separate dataset of front and rear car views containing 1492 positive instances from the LabelMe database [27] and 178 negative images.

4.2 Results

Sowerby Dataset. The Sowerby dataset is a widely used benchmark for CRFs, which contains 7 outdoor rural landscape classes. The dataset comprises 104 images at a resolution of 96×64 pixels. Following the protocol of [5] we randomly selected 60 images for training and 44 images for testing. Some example images with inferred labels are shown in figure 2. However, this dataset does neither contain image sequences nor cars that can be detected with an object detector

Table 1. Comparison to previously reported results on the Sowerby dataset

	Pixel-wise accuracy	
	Unary classification	plain CRF model
He *et al.* [4]	82.4%	89.5%
Kumar&Hebert [5]	85.4%	89.3%
Shotton *et al.* [6]	85.6%	88.6%
This paper	84.5%	91.1%

and thus we can only compare our *plain CRF* model (equation 1) with previous work on this set.

The experiments show that our features and CRF parameter estimation is competitive to other state-of-the-art methods. Table 1 gives an overview of previously published results and how those compare to our model (see figure 3). While the more sophisticated Textons features [6] do better for unary classification, our CRF model can outperform those since our edge potentials are learned from training data. For this dataset we use a grid with one node for each input pixel, while the Gaussian prior σ was set to 1.25. The Walsh-Hadamard transform was run on the input images at the aperture size of 2, 4, 8 and 16 pixels. Moreover, we used a global set of weights for the isotropic linear classifiers of the edge potentials, but distinguish between north-south neighborhood relations and east-west neighborhood relations.

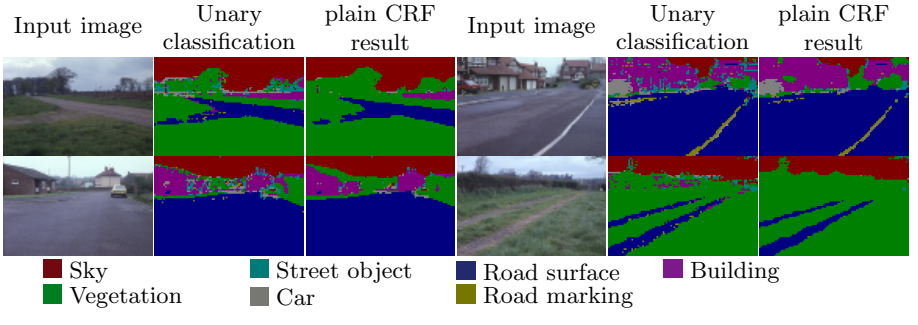


Fig. 2. *Sowerby* dataset example results

Dynamic Scenes Dataset. To evaluate our *object* and *dynamic CRF* we set up a new *dynamic scenes* dataset with image sequences consisting of overall 1936 images[1]. The images are taken from a camera inside a driving car and mainly show rural roads with high dynamics of driving vehicles at an image resolution of 752×480 pixels. Cars appear at all scales from as small as 15 pixels up to 200 pixels. The database consists of 176 sequences with 11 successive images each. It is split into equal size training and test sets of 968 images.

[1] The dataset is available at `http://www.mis.informatik.tu-darmstadt.de`.

To evaluate pixel level labeling accuracy the last frame of each sequence is labeled pixel-wise, while the remainder only contains bounding box annotations for the frontal and rear view car object class. Overall, the dataset contains the eight labels *void*, *sky*, *road*, *lane marking*, *building*, *trees & bushes*, *grass* and *car*. Figure 3 shows some sample scenes. For the following experiments we used 8×8 pixels for each CRF grid node and texture features were extracted at the aperture sizes of 8, 16 and 32 pixels.

We start with an evaluation of the unary classifier performance on the scene class layer. Table 2 lists the pixel-wise classification accuracy for different variations of the feature. As expected location is a valuable cue, since there is a huge variation in appearance due to different lighting conditions. Those range from bright and sunny illumination with cast shadows to overcast. Additionally, motion blur and weak contrast complicate the pure appearance-based classification. Further, we observe that normalization [26] as well as multi-scale features are helpful to improve the classification results.

Table 2. Evaluation of texture location features based on overall pixel-wise accuracy; Multi-scale includes feature scales of 8, 16 and 32 pixels, Single-scale is a feature scale of 8 pixels; note that these number do not include the CRF model – adding the *plain CRF* to the best configuration yields an overall accuracy of 88.3%

		Normalization			
		on		off	
		multi-scale	**single-scale**	**multi-scale**	**single-scale**
Location	on	82.2%	81.1%	79.7%	79.7%
	off	69.1%	64.1%	62.3%	62.3%

Next, we analyze the performance of the different proposed CRF models. On the one hand we report the overall pixel-wise accuracy. On the other hand the pixel-wise labeling performance on the car object class is of particular interest. Overall, car pixels cover 1.3% of the overall observed pixels. Yet, those are an important fraction for many applications and thus we also report those for our evaluation.

For the experiments we used anisotropic linear edge potential classifiers with 16 parameter sets, arranged in four rows and four columns. Moreover, we distinguish between north-south and east-west neighborhoods. For the inter-layer edge potentials we trained different weight sets depending on detection scale (discretized in 6 bins) and depending on the neighborhood location with respect to the object's center.

Table 3 shows recall and precision for the proposed models. Firstly, the employed detector has an equal error rate of 78.8% when the car detections are evaluated in terms of precision and recall. When evaluated on a pixel-wise basis the performance corresponds to 60.2% recall. The missing 39.8% are mostly due to the challenging dataset. It contains cars with weak contrast, cars at small scales and partially visible cars leaving the field of view. Precision for the detector evaluated on pixels is 37.7%. Wrongly classified pixels are mainly around the objects and on structured background on which the detector obtains false detections.

Table 3. Pixel-wise recall and precision for the pixels labeled as *Car* and overall accuracy on all classes

	No objects			**With object layer**			**Including object dynamics**		
	Recall	Precision	Acc.	Recall	Precision	Acc.	Recall	Precision	Acc.
CRF	50.1%	57.7%	88.3%	62.9%	52.3%	88.6%	70.4%	57.8%	88.7%
dyn. CRF	25.5%	44.8%	86.5%	75.7%	50.8%	87.1%	78.0%	51.0%	88.1%

Let us now turn to the performance of the different CRF models. Without higher level information from an object detector *plain CRFs* in combination with texture-location features achieve a recall of 50.1% with a precision of 57.7%. The recognition of cars in this setup is problematic since CRFs optimize a global energy function, while the car class only constitutes a minor fraction of the data. Thus, the result is mainly dominated by classes which occupy the largest regions such as sky, road and trees.

With higher level object information (*object CRF*) recall can be improved up to 62.9% with slightly lower precision resulting from the detector's false positive detections. However, when objects are additionally tracked with a Kalman filter, we achieve a recall of 70.4% with a precision of 57.8%. This proves that the object labeling for the car object class leverages from the object detector and additionally from the dynamic modeling by a Kalman filter.

Additionally, we observe an improvement of the overall labeling accuracy. While *plain CRFs* obtain an accuracy of 88.3%, the *object CRF* achieves 88.6% while also including object dynamics further improves the overall labeling accuracy to 88.7%. The relative number of 0.4% might appear low, but considering that the database overall only has 1.3% of car pixels, this is worth noting. Thus, we conclude that not only the labeling on the car class is improved but also the overall scene labeling quality.

When the scene dynamics are modeled additionally and posteriors are propagated over time (*dynamic CRF*), we again observe an improvement of the achieved recall from 25.5% to 75.7% with the additional object nodes. And also the objects' dynamic model can further improve the recall to 78.0% correctly labeled pixels. Thus, again we can conclude that the CRF model exploits both the information given by the object detector as well as the additional object dynamic to improve the labeling quality.

Finally, when the overall accuracy is analyzed while the scene dynamic is modeled we observe a minor drop compared to the static modeling. However, we again consistently observe that the object information and their dynamics allow to improve from 86.5% without object information to 87.1% with *object CRFs* and to 88.1% with the full model.

The consistently slightly worse precision and overall accuracy for the dynamic scene models need to be explained. Non-car pixels wrongly labeled as car are mainly located at the object boundary, which are mainly due to artifacts of the

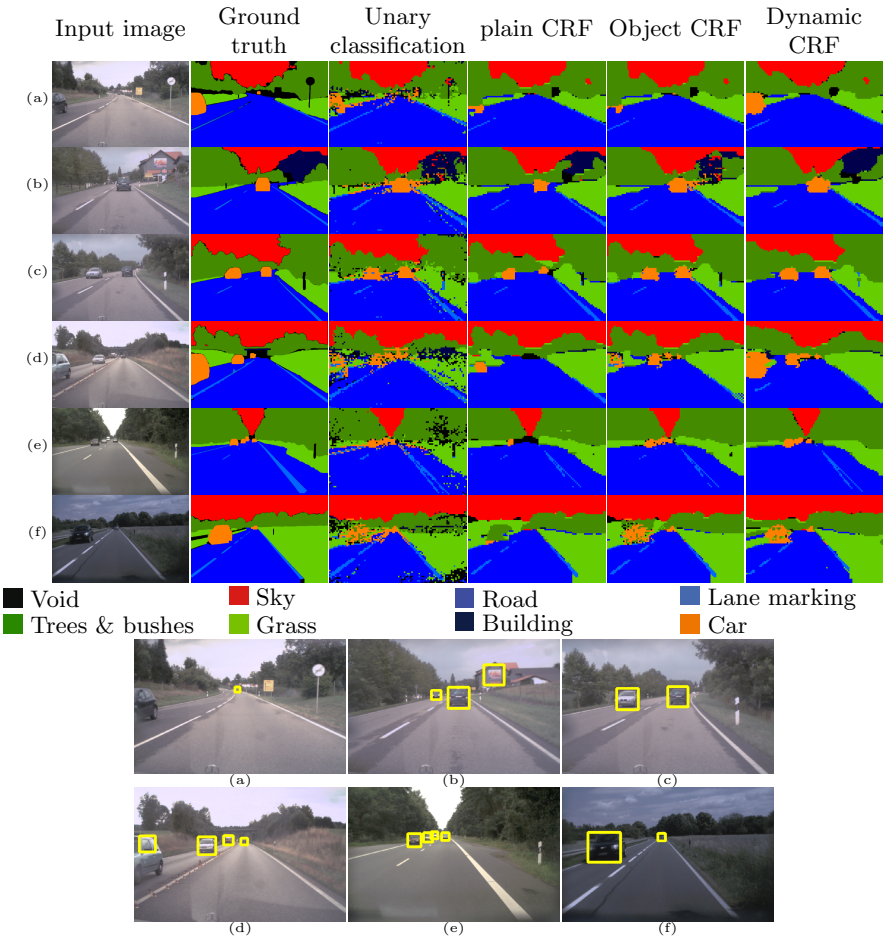


Fig. 3. *Dynamic scenes* dataset example result scene labels and corresponding detections in left-right order (best viewed in color); note that detections can be overruled by the texture location potentials and vice versa

scene label forward propagation. Those are introduced by the inaccuracies of the speedometer and due to the inaccuracies of the projection estimation.

A confusion matrix for all classes of the *dynamic scenes* database can be found in table 4. Figure 3 shows sample detections and scene labelings for the different CRF models to illustrate the impact of the different models and their improvements. In example (d) for instance the car which is leaving the field of view is mostly smoothed out by a *plain CRF* and *object CRF*, while the *dynamic CRF* is able to classify almost the entire area correctly. Additionally, the smaller cars which get smoothed out by a *plain CRF* are classified correctly by the *object* and *dynamic CRF*. Also note that false object detections as in example (c) do not result in a wrong labeling of the scene.

Table 4. Confusion matrix in percent for the *dynamic scenes* dataset; entries are row-normalized

True class	Fraction Inferred	Sky	Road	Lane marking	Trees & bushes	Grass	Building	Void	Car
Sky	10.4%	**91.0**	0.0	0.0	7.7	0.5	0.4	0.3	0.1
Road	42.1%	0.0	**95.7**	1.0	0.3	1.1	0.1	0.5	1.3
Lane marking	1.9%	0.0	36.3	**56.4**	0.8	2.9	0.2	1.8	1.6
Trees & bushes	29.2%	1.5	0.2	0.0	**91.5**	5.0	0.2	1.1	0.4
Grass	12.1%	0.4	5.7	0.5	13.4	**75.3**	0.3	3.5	0.9
Building	0.3%	1.6	0.2	0.1	37.8	4.4	**48.4**	6.3	1.2
Void	2.7%	6.4	15.9	4.1	27.7	29.1	1.4	**10.6**	4.8
Car	1.3%	0.3	3.9	0.2	8.2	4.9	2.1	2.4	**78.0**

5 Conclusions

In this work we have presented a unifying model for joint scene and object class labeling. While CRFs greatly improve unary pixel-wise classification of scenes they tend to smooth out smaller regions and objects such as cars in landscape scenes. This is particularly true when objects only comprise a minor part of the amount of overall pixels. We showed that adding higher level information from a state-of-the-art HOG object detector ameliorates this shortcoming. Further improvement – especially when objects are only partially visible – is achieved when object dynamics are properly modeled and when scene labeling information is propagated over time. The improvement obtained is bidirectional, on the one hand the labeling of object classes is improved, but on the other hand also the remaining scene classes benefit from the additional source of information.

For future work we would like to investigate how relations between different objects such as partial occlusion can be modeled when multiple object classes are detected. Additionally, we seek to improve the ego-motion estimation of the camera to further improve the performance. This will also allow us to employ motion features in the future. Finally, we assume that the integration of different sensors such as radar allow for a further improvement of the results.

Acknowledgements. This work has been funded, in part, by Continental Teves AG. Further we thank Joris Mooij for publically releasing libDAI and BAE Systems for the Sowerby Dataset.

References

1. Everingham, M., Zisserman, A., Williams, C., van Gool, L.: The pascal visual object classes challenge results. Technical report, PASCAL Network (2006)
2. Torralba, A.: Contextual priming for object detection. IJCV, 169–191 (2003)

3. Hoiem, D., Efros, A.A., Hebert, M.: Putting objects in perspective. In: CVPR (2006)
4. He, X., Zemel, R.S., Carreira-Perpiñán, M.Á.: Multiscale conditional random fields for image labeling. In: CVPR (2004)
5. Kumar, S., Hebert, M.: A hierarchical field framework for unified context-based classification. In: ICCV (2005)
6. Shotton, J., Winn, J., Rother, C., Criminisi, A.: Textonboost: Joint appearance, shape and context modeling for multi-class object recognition and segmentation. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3951. Springer, Heidelberg (2006)
7. Torralba, A., Murphy, K.P., Freeman, W.T.: Contextual models for object detection using boosted random fields. In: NIPS (2004)
8. McCallum, A., Rohanimanesh, K., Sutton, C.: Dynamic conditional random fields for jointly labeling multiple sequences. In: NIPS Workshop on Syntax, Semantics (2003)
9. Lafferty, J.D., McCallum, A., Pereira, F.C.N.: Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In: ICML (2001)
10. Kumar, S., Hebert, M.: Discriminative random fields: A discriminative framework for contextual interaction in classification. In: ICCV (2003)
11. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: CVPR (2005)
12. Verbeek, J., Triggs, B.: Region classification with markov field aspect models. In: CVPR (2007)
13. Quattoni, A., Collins, M., Darrell, T.: Conditional random fields for object recognition. In: NIPS (2004)
14. Kapoor, A., Winn, J.: Located hidden random fields: Learning discriminative parts for object detection. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3954. Springer, Heidelberg (2006)
15. Winn, J., Shotton, J.: The layout consistent random field for recognizing and segmenting partially occluded objects. In: CVPR (2006)
16. Wang, Y., Ji, Q.: A dynamic conditional random field model for object segmentation in image sequences. In: CVPR (2005)
17. Yin, Z., Collins, R.: Belief propagation in a 3D spatio-temporal MRF for moving object detection. In: CVPR (2007)
18. Leibe, B., Cornelis, N., Cornelis, K., Van Gool, L.: Dynamic 3D scene analysis from a moving vehicle. In: CVPR (2007)
19. Torralba, A., Murphy, K.P., Freeman, W.T.: Sharing features: Efficient boosting procedures for multiclass object detection. In: CVPR (2004)
20. Platt, J.: Probabilistic outputs for support vector machines and comparison to regularized likelihood methods. In: Smola, A.J., Bartlett, P., Schoelkopf, B., Schuurmans, D. (eds.) Advances in Large Margin Classifiers, pp. 61–74 (2000)
21. Kalman, R.E.: A new approach to linear filtering and prediction problems. Transactions of the ASME–Journal of Basic Engineering 82, 35–45 (1960)
22. Sutton, C., McCallum, A.: Piecewise training for undirected models. In: 21th Annual Conference on Uncertainty in Artificial Intelligence (UAI 2005) (2005)
23. Vishwanathan, S.V.N., Schraudolph, N.N., Schmidt, M.W., Murphy, K.P.: Accelerated training of conditional random fields with stochastic gradient methods. In: ICML (2006)
24. Hel-Or, Y., Hel-Or, H.: Real-time pattern matching using projection kernels. PAMI 27, 1430–1445 (2005)

25. Alon, Y., Ferencz, A., Shashua, A.: Off-road path following using region classification and geometric projection constraints. In: CVPR (2006)
26. Varma, M., Zisserman, A.: Classifying images of materials: Achieving viewpoint and illumination independence. In: Tistarelli, M., Bigun, J., Jain, A.K. (eds.) ECCV 2002. LNCS, vol. 2359. Springer, Heidelberg (2002)
27. Russell, B.C., Torralba, A., Murphy, K.P., Freeman, W.T.: LabelMe: A database and web-based tool for image annotation. IJCV 77, 157–173 (2008)