Hierarchical Support Vector Random Fields: Joint Training to Combine Local and Global Features

Paul Schnitzspan, Mario Fritz, and Bernt Schiele

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

Abstract. Recently, impressive results have been reported for the detection of objects in challenging real-world scenes. Interestingly however, the underlying models vary greatly even between the most successful approaches. Methods using a global feature descriptor (e.g. [1]) paired with discriminative classifiers such as SVMs enable high levels of performance, but require large amounts of training data and typically degrade in the presence of partial occlusions. Local feature-based approaches (e.g. [2–4]) are more robust in the presence of partial occlusions but often produce a significant number of false positives. This paper proposes a novel approach called hierarchical support vector random field that allows 1) to combine the power of global feature-based approaches with the flexibility of local feature-based methods in one consistent multi-layer framework and 2) to automatically learn the tradeoff and the optimal interplay between local, semi-local and global feature contributions. Experiments show that both the combination of local and global features as well as the joint training result in improved detection performance on challenging datasets.

1 Introduction

The first goal of this paper is to propose a novel hierarchical framework that effectively combines the power of global feature-based models with the flexibility of local feature-based representations. The second goal is to derive an efficient and effective procedure to jointly train all model parameters in order to automatically learn the tradeoff and the interdependence of the different layers of the novel hierarchical model as well as between the local and global feature contributions. To achieve this, this paper leverages the ability of CRFs [5] to model neighborhood dependencies not only between local image features, but also between object sub-parts and parts using a multi-layer CRF. On the top-layer we incorporate a global object detector while on the layers below we employ smaller apertures in terms of object-parts and local features or sub-parts. The layers are connected via intra-layer potentials to benefit from simultaneous bottom-up and top-down propagation schemes. This allows to set up a joint and hierarchical model of local and global discriminative methods that augments CRFs to a multi-layer model with powerful unary classifiers.

D. Forsyth, P. Torr, and A. Zisserman (Eds.): ECCV 2008, Part II, LNCS 5303, pp. 527–540, 2008.

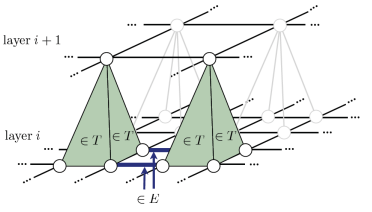


Fig. 1. Illustration of the model architecture. Two layers are connected via the ternary cliques T. The alternation between binary cliques E and ternary cliques T is key to the computationally feasability while a high degree of of interconnectedness is introduced.

The contributions of this paper are the following. First, we extend classical one-layer CRFs to multi-layer CRFs while maintaining computational tractability. Second, this work shows how to integrate local, semi-local and global information in a powerful model. Third, we extend CRFs to a consistent framework, which allows to jointly train the parameters of nonlinear classifiers and the CRF parameters. Fourth, we experimentally show the contributions of the various components of the model on challenging datasets. The paper is structured as follows. First, we refer to related work. In section 2 we introduce our multi-layer model, the respective potential functions and the parameters to be optimized. In section 3 we explain how we apply the model to object detection and verification. Finally, in section 4 we evaluate various aspects of our work on two different datasets.

Related Work. Related work [4, 6–9] addressing the combination of global and local features showed promising results and specifically improved performance compared to making use of only one type of features - either local or global. Especially the idea of [8] of integrating localized features of growing apertures in one model is relevant to our goals since meaningful object parts can be determined while ensuring global consistency. In our work we exploit a similar approach in terms of evidence aggregation, but additionally we are able to learn local neighborhood dependencies and address bi-directional interactions between entire objects and object parts. In contrast to [4, 7, 9] where global and local features are matched independently from each other, we combine bottom-up and top-down cues simultaneously showing improved performance.

Concerning CRF literature, [6, 10] proposed multi-layer CRFs to account for global consistency and due to that showed improved performance. In [6] a global scene potential is introduced to assert consistency of local regions. Thereby, the authors were able to benefit from integrating the context of a given scene. However, their model works with global priors set in advance and only uses learned local classifiers. Rather than to rely on priors alone, in our work, all parameters of the layers are trained jointly. In [10] a tree-structured CRF is proposed based on previously inferred segmentations of images. Thereby, the authors neglect direct local neighborhood dependencies, which our model learns jointly with long range dependencies.

[11–13] introduced two-layer CRFs, where the top layer deploys one node, which superimposes objects. That idea is especially interesting for object detection since the superimposed node manages local deformations of objects and therefore asserts consistency of object instances. Our work goes beyond those approaches by introducing multiple layers of evidence aggregation, which not only guarantees consistency of object instances, but also deploys different levels of information (from local to global) to our model.

CRF-based approaches like [11, 13–15] showed that incorporating powerful unary classifiers in CRFs is key to the overall performance, but those unary classifiers were trained independently from local neighborhood potentials. In contrast, we investigate the joint optimization of all aspects of our model and show that this joint optimization leads to improved performance compared to independent training. Taskar et al. [16] transformed the original problem introduced by Lafferty et al. [5] into its equivalent exponentially sized dual formulation but the latter lacks the intuitive nature of the primal problem and they only evaluated it on an OCR task. Due to the modularity of our approach we reduce the problem size while keeping the intuitive nature of the original formulation.

2 Hierarchical Support Vector Random Fields (hSVRF)

While global detectors have been shown to achieve impressive results in object detection for unoccluded object instances, part-based approaches tend to be more successful in dealing with partial occlusion. Since adjacent regions in images are not independent from each other, CRFs model these dependencies directly by introducing binary clique potentials. However, standard CRFs work on a very local level and long range dependencies are not addressed explicitly in simple one-layer models. Therefore, our approach incorporates SVMs and multiple layers of CRFs in one consistent framework in order to combine local neighborhood and long range dependencies. In the following we will describe how we set up the multi-layer model step by step starting from the simple one-layer case.

2.1 One-Layer CRF Model

We overlay the image X with a grid of nodes where each node is linked to the evidence in the image via unary ψ and binary ϕ potential functions. We denote the set of grid nodes by $y_i \in Y$ in which each y_i is associated to a certain region x_i in the image X. $e_{ij} = (y_i, y_j) \in E$ refers to the binary cliques connecting two adjacent nodes y_i and y_j. Each node $y_i \in Y$ will be assigned a label from $\{0, \ldots, p\}$ which indicates the parts of an object $\{1, \ldots, p\}$ or background $\{0\}$. We denote the set of all labels by $\mathcal{Y}$. Therefore, the factorization of the conditional probability distribution of the nodes Y given the image can be written as

$$p(Y|X) = \frac{1}{Z} \prod_{y_i \in Y} \psi(y_i, x_i) \prod_{e_{ij} \in E} \phi(y_i, y_j, x_i, x_j) \ . \tag{1}$$

Here, Z refers to a normalization factor called the partition function.

2.2 Multilayer CRF Model

As motivated before, one-layer CRFs act at a very local level and represent a single view on the data typically represented with unary and binary potentials. In order to overcome those local restrictions, we introduce multiple layers $l \in \{1, \ldots, N_L\}$ with associated unary potentials ψ^l and binary potentials ϕ^l, to enhance the model by evidence aggregation on a local ($l = 1$) to a global level ($l = N_L$). Different numbers of parts are deployed to different layers $\{0, \ldots, p^l\}$. We propose a connectivity between the layers as displayed in Figure 1, which provides a high degree of interconnectedness and yet results in a computationally tractable model, which is highly desirable for both inference and training. The key to this is the alternation between binary cliques $e_{ij} \in E^l$ and ternary cliques $t_{ijk} \in T^l$ that omit the introduction of higher (higher than third) order cliques. The conditional distribution for this multi-layer model resolves into:

$$p(Y|X) = \frac{1}{Z} \prod_{l=1}^{L} \left[\prod_{y_i \in Y^l} \psi^l(y_i, x_i) \prod_{e_{ij} \in E^l} \phi^l(y_i, y_j, x_i, x_j) \right] \quad \text{(intra-layer)}$$
$$\prod_{l=1}^{L-1} \prod_{t_{ijk} \in T^l} \theta^l(y_i, y_j, y_k, x_i, x_j, x_k) \quad \text{(inter-layer)} \tag{2}$$

where additional to the one-layer notation $\theta^l(\cdot, \cdot, \cdot, \cdot, \cdot, \cdot)$ denotes the ternary clique potentials that connect layer l to layer $l + 1$ using third-order cliques. T^l describes the set of all ternary cliques between layer l and layer $l + 1$ (see Figure 1 for illustration).

This model combines different views on the data by layer-specific potentials and the hierarchical structure accounts for longer range dependencies.

2.3 Potentials

As described in Eq. 1 and 2 the conditional probabilities factor into unary potentials ψ^l, binary potentials ϕ^l and additional ternary potentials θ^l required for the multi-layer model. Due to the flexibility of CRFs, the layer-specific feature functions $f^l(X)$, $g^l(X)$ and $h^l(X)$ for the unary, binary and ternary potentials respectively can be chosen freely. Those deployed in the experiments are detailed in section 3.1.

Unary Potentials. The discriminative power in the unary potentials is key to the overall performance of the CRF. In some cases, a CRF using less powerful classifiers such as the commonly used logistic regression can even be outperformed by an SVM employing no connectivity at all [14].

Therefore, we build our unary potentials on SVMs to leverage previous results on robust large margin classification. We adapt the one-against-all strategy which results in training one SVM for each class. $f^l(\cdot)$ refers to the feature function for the node features and ρ_c^l denotes the offset. Then, the potential of node y_i being of class c is defined as

$$\psi^l(y_i = c, x_i; \beta_c^l, \zeta_c^l, \rho_c^l) = \exp\left(\sum_{j \in \zeta_c^l} \left(\beta_c^l\right)_j K(f_j^l(X), f_i^l(X)) + \rho_c^l \right) . \tag{3}$$

where ζ_c^l indexes the set of support vectors for class c and layer l and $\left(\beta_c^l\right)$ refers to model parameters to be optimized. We used RBF kernels to define the kernel function $K(f_j^l(X), f_i^l(X)) = \exp\left(-\gamma \left\| f_j^l(X) - f_i^l(X) \right\|^2\right)$ with bandwidth parameter γ. Note, that this approach employs multiclass one-against-all SVMs.

Binary and Ternary Potentials. We define the binary and ternary potentials using a linear classification model, which is a popular choice in the CRF literature. For the binary potentials we set

$$\phi^l(y_i = c_1, y_j = c_2, x_i, x_j; u^l) = \exp\left(\left(u_{c_1 c_2}^l\right)^T g_{ij}^l(X)\right) \quad . \tag{4}$$

where y_i and y_j are two adjacent nodes and c_1 and c_2 refer to any label from $\{0, \ldots, p^l\}$. $g^l(\cdot)$ denotes the feature function for the binary potentials of layer l. $u_{c_1 c_2}^l$ refers to the parameters to be trained. The ternary potentials are defined as

$$\theta^l(y_i = c_1, y_j = c_2, y_k = c_3, x_i, x_j, x_k; v^l) = \exp\left(\left(v_{c_1 c_2 c_3}^l\right)^T h_{ijk}^l(X)\right) \quad . \tag{5}$$

where y_i, y_j, y_k belong to one three-wise connected clique t_{ijk} and $c_1, c_2 \in \{0, \ldots, p^l\}$ and $c_3 \in \{0, \ldots, p^{l+1}\}$. $h^l(\cdot)$ denotes the feature function for the ternary potentials at layer l. $v_{c_1 c_2 c_3}^l$ refers to the parameters to be optimized.

2.4 Parameter Learning and Inference

In contrast to CRF literature like [11, 13–15], we jointly optimize all model parameters. Given M training images $X^m, m = \{1, \ldots, M\}$ we optimize the conditional log-likelihood $\mathcal{L}(\beta, u, v) = \sum_{m=1}^{M} \log P\left(Y^m | X^m; \beta, u, v\right)$ via gradient descent for binary and ternary clique potentials. The unary potentials are trained with Newton optimization.

This joint training is facilitated by the primal SVM training proposed by Chapelle [17] that showed competitive results compared to common quadratic programming in the dual formalism. We make use of that idea and incorporate primal SVM training in the CRF framework.

Primal SVM Training. As described in [17] the constraints of the original primal optimization problem can be integrated with a loss function in the objective function, yielding an unconstrained optimization problem. As long as this loss function is differentiable with respect to the model parameters, the optimization can be solved by Newton optimization. Originally, the non-differentiable hinge loss is used for SVM training in the dual, but [17] showed competitive results using the differentiable quadratic loss or the Huber loss (a differentiable approximation of the hinge loss). The primal optimization problem for kernel SVMs is denoted by:

$$\min_{\beta_c^l} F = \min_{\beta_c^l} \left(\sum_{i,j \in \zeta_c^l} \left(\beta_c^l\right)_i \left(\beta_c^l\right)_j K(f_i(X), f_j(X)) + C \sum_{i=1}^{n} L\left(y_i, S_c^l\left(f_i^l(X)\right)\right) \right) \quad . \tag{6}$$

where L denotes a suitable loss function and C the penalty term. The target function $S_c^l(\cdot)$ is of the form (representer theorem [18]):

$$S_c^l\left(f_i^l(X)\right) = \sum_{j \in \zeta_c^l} \left(\beta_c^l\right)_j K(f_j^l(X), f_i^l(X)) + \rho_c^l \ . \tag{7}$$

where $f_i^l(x)$ denotes a feature vector to be classified. $K(\cdot,\cdot)$ denotes the kernel function and $\left(\beta_c^l\right)$ refers to the parameters to be optimized (note that these are not Lagrange multipliers). We consider the differentiable quadratic loss:

$$L(y_i, S_c^l(f_i(X))) = \left(\max\left\{0; 1 - (\delta(y_i, c))\left(S_c^l\left(f_i^l(X)\right)\right)\right\}\right)^2 \ . \tag{8}$$

where $\delta(y_i, c) \in \{-1, 1\}$ refers to whether y_i belongs to class c (=1) or not (=-1).

[17] proposed to optimize the parameters $\left(\beta_c^l\right)$ with Newton optimization:

$$\beta_c^l \leftarrow \beta_c^l - \eta \left(H^l\right)^{-1} \frac{\partial F}{\partial \beta_c^l} \ . \tag{9}$$

where η denotes the learning rate and the Hessian H^l equals $2\left(\frac{1}{C}K + KI^0K\right)$ with kernel matrix K. I^0 is a diagonal matrix, where the entries are 1 for $\beta_c^l > 0$ and 0 otherwise. The number of non-zero entries equals the number of support vectors. In order to update the offset ρ_c^l the Hessian can be augmented by an additional row and column and the offset term can be concatenated with the parameters β_c^l (see [17] for details).

Joint Training of hSVRF. In order to account for joint training of the hSVRF parameters, we adapt the loss function $L(\cdot,\cdot)$ to consider unary SVM classifications as well as joint CRF classifications , which respects the entire multi-layer model. In that sense, object evidence, local neighborhood dependencies as well as longer range dependencies are taken into account to optimize the unary parameters. We achieve this by adapting the loss function to consider the belief of node y_i belonging to class c inferred with Loopy Belief Propagation [19].

$$L(y_i, b_c(y_i), S_c^l(f_i(X))) = \left[\max\left\{0; (1 - \delta(y_i, c)b_c(y_i))\left(1 - \delta(y_i, c)S_c^l\left(f_i^l(X)\right)\right)\right\}\right]^2 \tag{10}$$

where the belief $b_c(y_i)$ of node y_i belonging to class c ranges between -1 and 1. Whenever the CRF votes for the wrong class ($(1 - \delta(y_i, c)b_c(y_i)) > 1$) the original primal SVM loss function is amplified for calculating the Newton step. Otherwise ($(1 - \delta(y_i, c)b_c(y_i)) < 1$) the impact of the original primal loss function on the Newton step is reduced. Note, the Hessian is not affected by our changes in the loss function and $\frac{\partial F}{\partial \beta_c^l}$ can be computed similar to [17].

The parameters of the binary and ternary clique potentials can be optimized via gradient descent. Similar to [12], the gradient with respect to the binary parameters $\{u_{c_1c_2}^l\}$ of layer l can be expressed as

$$\frac{\partial \mathcal{L}}{\partial u_{c_1c_2}^l} = \sum_{e_{ij} \in E} \left(\delta(y_i, c_1)\delta(y_j, c_2) - b_{c_1c_2}(y_i, y_j)\right) g_{ij}^l(x) \ . \tag{11}$$

(a) (b)

Fig. 2. (a) Three-layer instantiation of our model. The evidence aggregation is sketched: Starting from local information like fragments of a wheel over whole wheels to entire objects at the top layer. (b) Example of the part assignment of training data (left: training image; middle: part assignment of middle layer; right: part assignment of bottom layer). Colors encode assignments of parts; dark blue indicates background.

where $\delta(\cdot,\cdot)$ refers to the Kronecker-delta and $b_{c_1c_2}(\cdot,\cdot)$ denotes the pairwise belief of two adjacent nodes belonging to class c_1 and c_2.

Analogously, the gradient with respect to the ternary clique parameters $v^l_{c_1c_2c_3}$ of layer l can be written as

$$\frac{\partial \mathcal{L}}{\partial v^l_{c_1c_2c_3}} = \sum_{t_{ijk}\in T} \left(\delta(y_i,c_1)\delta(y_j,c_2)\delta(y_k,c_3) - b_{c_1c_2c_3}(y_i,y_j,y_k)\right) h^l_{ijk}(x) \quad . \tag{12}$$

where $b_{c_1c_2c_3}(\cdot,\cdot,\cdot)$ denotes the ternary beliefs of three connected nodes.

This concept for updating the parameters of our model alternates between the max margin notation of SVM training and the max likelihood formalism of CRFs. Although we show performance improvements with our scheme, this might be a restriction, since one unique optimization scheme is desirable. In future work we will investigate algorithms to overcome this restriction.

We use quadratic programming (the common SVM training) decoupled from the CRF to initialize the parameters β^l. The dual support vector coefficients α^l_c and parameters β^l_c are connected via $\left(\beta^l_c\right)_j = y_j \left(\alpha^l_c\right)_j$ as described in [17]. Given the starting solution for β^l we start the joint optimization by Newton optimization for unary classifiers and gradient descent for binary and ternary clique parameters.

Inference. Given the parameters (β^l), (u^l) and (v^l) we seek to infer probabilities of the nodes belonging to the different classes. Loopy Belief Propagation (LBP) [19] infers beliefs, that one node y belongs to class c while respecting the pairwise and three-wise dependencies of adjacent nodes.

3 Application to Computer Vision Tasks

To support our claims about the benefits of the local to global CRF model and the presented joint optimization, we evaluate the approach on two challenging computer vision tasks: object detection and hypothesis verification. But first we describe in detail how the method is adapted to the specific settings and show

how to obtain part annotations for the training phase. We consider a 3-layer instantiation of the presented model as visualized in Figure 2(a) and detailed below.

3.1 Feature Functions

Until now, we have not defined the feature functions $f^l(\cdot)$, $g^l(\cdot)$, $h^l(\cdot)$, that are specific to each layer in the CRF we propose. They link the potentials to the actual image evidence and account for local neighborhood and long range dependencies. We build on the concept of computing histograms of oriented gradients, that has been shown to be very successful on a local level, describing interest points [20], as well as on a global level [1], describing full objects in a holistic manner. However, due to the generality of our work, any suitable feature function can be deployed to our model.

Unary Potential Feature Functions. We calculate histograms of oriented gradients for a grid of non-overlapping 8×8 pixel regions and concatenate 4 neighboring histograms of gradients to one block descriptor as described in [1]. This results in a 36 dimensional feature for each node, that we define to be the unary feature function on the first level $f^1(\cdot)$. For the higher levels $f^2(\cdot), \ldots, f^L(\cdot)$ we successively double the number of considered blocks in horizontal and vertical directions until on the highest level, we encode the full object as in [1]. As illustrated in Figure 2(a), the motivation behind this scheme is to aggregate evidence for an object class from different spatial localities ranging from fragments (e.g. fragment of a wheel), parts (e.g. whole wheel) to a holistic view on the object (e.g. whole motorbike).

Binary Potential Feature Functions. Intuitively, binary potentials are responsible for modeling local dependencies by supporting or inhibiting label propagation to the neighboring nodes. In computer vision, simple pixel-based gradient-based measures are often used to inhibit propagation across potential object borders [15]. Our approach goes beyond that by taking into account the change in the gradient orientation histograms between the neighboring nodes.

$$g_{ij}^l(X) = \left(\left| f_i^l(X) - f_j^l(X) \right| \ , \ 1 \right)^T \ . \tag{13}$$

Here, we extended each difference by an offset for being capable eliminating small isolated regions.

Ternary Potential Feature Functions. Similar to the binary potentials, ternary potentials encode local dependencies, too. But furthermore, they act as a link between layers, facilitating propagation of information across locality and position in our model. Due to the computational tractability of the hierarchy we can propagate object evidence across layers and thereby manage efficient bottom-up and top-down reasoning during inference.

To allow the ternary potential to assess the compatibility of a particular labeling of a three-wise connected clique, we define the ternary potential feature function to be the stacked pairwise difference of the feature vector associated to

the 3 relevant nodes. Since higher level nodes involve more HOG blocks and are higher dimensional than lower level ones, we calculate the average over connected blocks (denoted by operation $avg(\cdot)$) in order to fit the dimension of lower level nodes.

$$h^l_{ijk}(x) = \left(\mid f^l_i(x) - f^l_j(x) \mid , \mid f^l_i(x) - avg(f^{l+1}_k(x)) \mid , \mid f^l_j(x) - avg(f^{l+1}_k(x)) \mid , 1 \right)^T. \tag{14}$$

where nodes i and j are on layer l and node k is on layer $l+1$.

3.2 Part Assignment

For optimizing the conditional log-likelihood during training, ground truth part labels are required for each training instance in order to be able to train the multiclass potentials of our model. While the labeling for the top (object) layer is given by a bounding box annotation or segmentation of the objects, the part annotation on the lower layers is not obvious. Inspired by [21] we obtain part labels in a data driven way by applying k-means clustering across images to infer part annotations. Instead of mere spatial clustering, we append to the image coordinates the features described in 3.1. In this fashion the importance is on the cluster appearance and the 2 coordinate dimensions act as regularization for the clustering to maintain a rough spatial layout. Despite the simple data-driven approach, we obtain a sensible partitioning of our training instances, that exposes appearance-based though well localized assignment of parts as exemplified in Fig. 2(b).

3.3 Object Detection and Verification

In this paragraph we show how we infer object locations of one object class. As described in section 2.4, LBP yields a label assignment across layers taking into account beliefs that nodes are associated with parts (bottom and middle layer), object (top layer) or background (all layers). Given a test image we could initialize our model at every pixel location for being able to infer all possible object hypotheses. However, to reduce computational effort we first deploy the bottom and middle layer of our model. This step produces a part map of the whole image while respecting the dependencies of the bottom and middle layer. From the training set we know possible part constellations and we search for those constellations in the part map of test images to infer hypotheses of object locations. This approach resembles the ISM voting of [4] despite that the evidence of parts of our model are inferred simultaneously and therefore the parts interact with each other. Thus, the evidence of one sub-part conditions the constellation of direct neighbors via links in the bottom layer and via the middle layer it also affects the evidence of further image regions. This step generates initial hypotheses that still need to be validated by the complete model. Since the part-based approach of the lower layers showed to yield a high recall, this approach makes sense as we first search for possible locations and then infer the complete

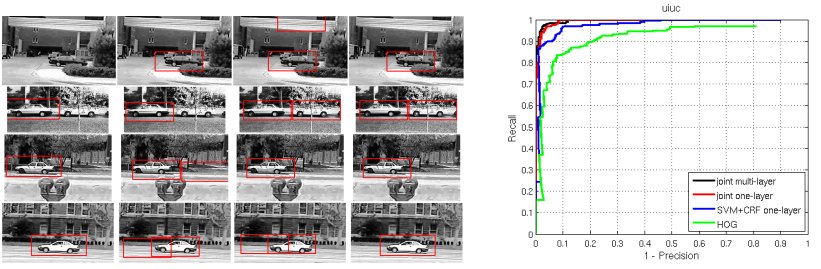


Fig. 3. (a) Examples on the UIUC dataset. The columns show results at EER of HOG detector, one-layer separate training, one-layer joint training and multi-layer model. (b) UIUC detection performance of the different aspects.

model for the hypothized bounding boxes. The approach of coupling generative models with a discriminative verification stage has been shown to be fruitful [22]. In this spirit, we address a hypothesis verification task by only infering our model at hypothesized bounding boxes. LBP simultaneously infers beliefs of all nodes of our model. Since at the top layer we only deploy one node, we can directly use the belief of that node belonging to the object class as a score. For the layers underneath we compute object probabilities similar to the ISM [4] part voting scheme as described above and multiply them to the global belief. Thereby we distinguish between left and right facing objects and consider the maximum of the deduced scores.

4 Experiments

In all experiments we used SVM^{light} [23] for initial SVM training. Training the model took approximately 12 hours while we were able to infer 15 hypotheses per second.

Object Detection. For the detection task we evaluated our model on the UIUC single scale car dataset. We trained the whole model on 250 bounding boxes containing cars and 200 negative crops. This experiment contains performance measurements of i) only the global object detector, ii) the one-layer model of our approach while training the SVM and CRF parameters separately, iii) the one-layer model while training the parameters jointly and iv) the complete multi-layer model. For the part labeling during training we deployed k-means clustering with 8 means for the bottom layer and 4 means for the middle layer. For the one-layer model we used 8 means in the clustering step. The detection performance was evaluated on the 170 UIUC test images. Figure 3(b) compares the different aspects described in the previous sections. Both the joint training and the multi-layer approach consistently improved the performance. Especially note the large performance gap between the complete model (97,5% in equal error rate) and the HOG detector (87,0% in EER). Figure 3(a) shows some example

Table 1. Results of the detection task on the UIUC car dataset

Method	EER
Multi-layer	97,5
One-layer part-based joint training	96,0
One-layer part-based without joint training	93,0
Global object detector [1]	87,0
Mutch & Lowe [24]	99,9
Leibe et al. [4]	97,5
Hoiem & Winn [11]	approx. 93,5
Winn & Shotton [13]	approx. 92,9

images where the HOG detector can not detect all cars due to partial occlusion and the one-layer models infer false positives, while the multi-layer model detects all cars correctly. These results expose the benefits of joint training and integration of local to global information. Our model successfully learns the tradeoff between global vs. local object detection and improves the performance of both ideas by combining powerful global descriptors and flexible local feature approaches. Further note the performance improvement between training the SVM independently from the other CRF parameters (93,0%) and training them jointly (96,0%) for the one-layer model. This evaluation highlights the advantage of training all model parameters jointly as proposed in section 2. In Tab. 1 we compare our model to the state-of-the-art in object detection on this dataset. As it can be seen, we achieve competitive results compared to other well performing models. Only Mutch and Lowe [24] outperform our model while we obtained the same performance as [4]. Further, we outperform the CRF-based approaches of Hoiem and Winn [11] and Winn and Shotton [13].

Verification of HOG Detector Hypothesis. For the hypothesis verification task we evaluated our model on the PASCAL 2006 motorbikes dataset [25] containing challenging multiscale, partially occluded and multiview instances. Since we want to explore the combination of an initial detector with our model acting as a verification stage, we trained a HOG detector on the provided training set and generated initial hypotheses on the test set. We set the parameters to allow for high recall at the drawback of more false positives. We also trained our joint multi-layer model on the training set and calculated the score of our approach on the hypotheses of the HOG detector (see Figure 4(a)). Thereby, our multi-layer model achieved 43,7% in average precision (the common performance measure of [25]) improving the state-of-the-art by 4,7%. Note in particular that we outperformed the global HOG detector, that reported an average precision of 39%, which emphasizes the benefit of combining global and local features. The next best performance for the motorbikes is 37,1% achieved by the approach of [26] which we outperform by 6,6%. Furthermore, our model shows a high performance improvement (more than 10% in average precision) compared to the remaining approaches.Particularly, the high precision for high scores of bounding boxes is promising; with no false positives 16% of all motorbikes are extracted while none of the other state-of-the-art approaches obtained such high recall at perfect precision.

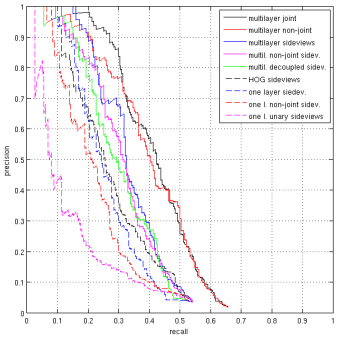
Fig. 5(a) shows precision-recall-curves from which the contributions of different aspects of our model to the overall performance gain can be deduced. Consistent

Method	AP	AP trained on sidev.
Multi-layer	43,7	36,0
Multi-layer not jointly trained	42,0	33,5
Decoupled multiple layers	-	32,2
One-layer model	-	27,7
One-layer not jointly trained	-	24,2
One-layer unary classifier	-	19,0
Dalal & Triggs [1]	39,0	30,0
Chum & Zisserman [26]	37,1	-
Laptev [27]	31,8	-
Viitaniemi & Laaksonen [28]	26,5	-
Shotton et al. [15]	17,8	-
Leibe et al. [4]	15,3	-

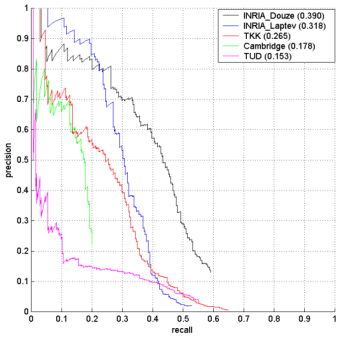
(a)

(b)

Fig. 4. (a) Results for the motorbike PASCAL06 challenge (AP = average precision). (b) Example images for detecting sideviews of motorbikes: (Left) one-layer part-based model; (middle) HOG sideviews; (right) joint multi-layer model.



(a)



(b)

Fig. 5. (a) PASCAL06 detection performance of our model. (b) State of the art approaches on the PASCAL06 motorbikes.

with the results obtained on the UIUC database, the jointly trained multi-layer model improves the performance to 43,7% while the non-jointly trained model with fixed SVM coefficients obtained 42% in average precision.

Verification of Generative ISM Object Detector Hypothesis. For further testing the different aspects of our model, we decided to test our discriminative model on hypotheses obtained by the ISM-model. Since the latter model was shown to yield promising results for the subset of left or right facing instances, we trained our model on sideviews of motorbikes, but evaluated the aspects on the complete multiview dataset. Overall the ISM model extracted 4238 hypotheses and achieved an average precision of 15,3%. We trained our model on 100 rightfacing and respective mirrored left views and 200 randomly cropped background images. As it can be seen in Fig. 4(a) our multi-layer model could improve the performance compared to other settings of our approach.

Concerning the average precision performance measure the jointly trained multi-layer model (36,0%) significantly improved the results of the non-jointly trained model with fixed SVM parameters (33,5%), completely decoupled layers (32,3%), the HOG detector trained on sideviews (30%) and the one-layer settings of our model: jointly trained SVM and CRF parameters (27,7%), fixed SVM parameters (24,2%) and the unary classifier (SVM) (19,0%).

Fig. 4(b) shows some example detections for training on sideviews. Partially occluded objects can not be detected by the global detector, while the part based approach and our multi-layer model infer them correctly. Furthermore, false detections of the part based approach can be removed by the global detector for correct detections of our multi-layer model. However, the rear view of the motorbike (third row) can not be detected correctly due to the focus on sideviews. The measured improvements for joint training and the multi-layer approach are consistent with respect to both tested databases.

5 Conclusion

We present a novel multi-layer CRF which combines the power of global object detectors and flexible local feature approaches. Our model successfully learns the tradeoff between local and global feature contributions for improved performance. Furthermore, we show how SVM classifiers can be incorporated into this multi-layer CRF framework and how training can be performed jointly. Experiments show that performance improves consistently. Finally, we outperform the state-of-the-art on the challenging PASCAL06 motorbike detection task. In future, we will investigate to use more layers in our model and deploy different tractable hierarchies to it, which can be easily done due to the generality of our work. We will also explore different features for evidence aggregation.

Acknowledgments. We thank Stefan Roth for helpful discussions and Joris Mooij for making libDAI available online. This work has been funded, in part, by the GRK 1362 of the DFG and the EU project CoSy (IST-2002-004250).

References

1. Dalal, N., Triggs, B.: Histograms of Oriented Gradients for Human Detection. In: CVPR (2005)
2. Felzenszwalb, P.F., Huttenlocher, D.P.: Pictorial Structures for Object Recognition. IJCV (2005)
3. Fergus, R., Zisserman, A., Perona, P.: Object class recognition by unsupervised scale invariant learning. In: CVPR 2003 (2003)
4. Leibe, B., Seemann, E., Schiele, B.: Pedestrian Detection in Crowded Scenes. In: CVPR (2005)
5. Lafferty, J., McCallum, A., Pereira, F.: Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In: ICML (2001)
6. He, X., Zemel, R., Carreira-Perpinan, M.: Multiscale conditional random fields for image labeling. In: CVPR 2004 (2004)

7. Heisele, B., Ho, P., Wu, J., Poggio, T.: Face recognition: Component-based versus global approaches. In: CVIU 2003 (2003)
8. Lazebnik, S., Schmid, C., Ponce, J.: Beyond Bag of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories. In: CVPR (2006)
9. Zhang, D., Li, S.Z., Gatica-Perez, D.: Real-Time Face Detection Using Boosting in Hierarchical Feature Spaces. In: ICPR 2004 (2004)
10. Reynolds, J., Murphy, K.: Figure-ground segmentation using a hierarchical conditional random field. In: CRV 2007 (2007)
11. Hoiem, D., Rother, C., Winn, J.: 3D Layout CRF for Multi-View Object Class Recognition and Segmentation. In: CVPR 2007 (2007)
12. Kumar, S., August, J., Hebert, M.: Discriminative Random Fields. In: IJCV 2006 (2006)
13. Winn, J., Shotton, J.: The Layout Consistent Random Field for Recognizing and Segmenting Partially Occluded Objects. In: CVPR (2006)
14. Lee, C.H., Greiner, R., Schmidt, M.: Support Vector Random Fields for Spatial Classification. In: Jorge, A.M., Torgo, L., Brazdil, P.B., Camacho, R., Gama, J. (eds.) PKDD 2005. LNCS (LNAI), vol. 3721, pp. 121–132. Springer, Heidelberg (2005)
15. 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)
16. Taskar, B., Guestrin, C., Koller, D.: Max margin markov networks. In: NIPS 2003 (2003)
17. Chapelle, O.: Training a Support Vector Machine in the Primal. In: Neural Computation (2007)
18. Kimeldorf, G.S., Wahba, G.: A correspondence between bayesian estimation on stochastic processes and smoothing by splines. Annals of Math. Stat. (1970)
19. Pearl, J.: Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference. Morgan Kaufmann, San Francisco (1988)
20. Lowe, D.: Distinctive Image Features from Scale-Invariant Keypoints. In: IJCV 2004 (2004)
21. Bouchard, G., Triggs, B.: Hierarchical Part-Based Visual Object Categorization. In: CVPR (2005)
22. Fritz, M., Leibe, B., Caputo, B., Schiele, B.: Integrating Representative and Discriminant Models for Object Category Detection. In: ICCV (2005)
23. Joachims, T.: Making large-Scale SVM Learning Practical. In: Advances in Kernel Methods - Support Vector Learning (1999)
24. Mutch, J., Lowe, D.: Multiclass Object Recognition with Sparse, Localized Features. In: CVPR (2006)
25. Everingham, M., Zisserman, A., Williams, C.K.I., Van Gool, L.: The PASCAL Visual Object Classes Challenge 2006 (VOC 2006) Results (2006)
26. Chum, O., Zisserman, A.: An Exemplar Model for Learning Object Classes. In: CVPR (2007)
27. Laptev, I.: Improvements of object detection using boosted histograms. In: BMVC 2006 (2006)
28. Viitaniemi, V., Laaksonen, J.: Techniques for Still Image Scene Classification and Object Detection. In: ICANN (2006)