Explicit Occlusion Modeling for 3D Object Class Representations

M. Zeeshan Zia[1], Michael Stark[2], and Konrad Schindler[1]

[1] Photogrammetry and Remote Sensing, ETH Zürich, Switzerland
[2] Stanford University and Max Planck Institute for Informatics

Abstract

Despite the success of current state-of-the-art object class detectors, severe occlusion remains a major challenge. This is particularly true for more geometrically expressive 3D object class representations. While these representations have attracted renewed interest for precise object pose estimation, the focus has mostly been on rather clean datasets, where occlusion is not an issue. In this paper, we tackle the challenge of modeling occlusion in the context of a 3D geometric object class model that is capable of fine-grained, part-level 3D object reconstruction. Following the intuition that 3D modeling should facilitate occlusion reasoning, we design an explicit representation of likely geometric occlusion patterns. Robustness is achieved by pooling image evidence from of a set of fixed part detectors as well as a non-parametric representation of part configurations in the spirit of poselets. *We confirm the potential of our method on cars in a newly collected data set of inner-city street scenes with varying levels of occlusion, and demonstrate superior performance in occlusion estimation and part localization, compared to baselines that are unaware of occlusions.*

1. Introduction

In recent years there has been a renewed interest in 3D object (class) models for recognition and detection. This trend has lead to a fruitful confluence of ideas from object detection on one side and 3D computer vision on the other side. State-of-the-art methods are not only capable of view-point invariant object categorization, but also give an estimate of the object's 3D pose [28, 21], and the locations of its parts [20, 26]. Some go as far as estimating 3D wireframe models and continuous pose from single images [40, 19, 39].

Still, viewpoint-invariant detection and modeling is far from being solved, and several open research questions remain. Here, we focus on the problem of (partial) occlusion by other scene parts. Knowing the detailed part-level occlusion pattern of an object is valuable information both for the object detector itself and for higher-level scene models that use the object class model. In fact, 3D object detection under severe occlusions is still a largely open problem. Most detectors [5, 9] break down at occlusion levels of ≈ 20%.

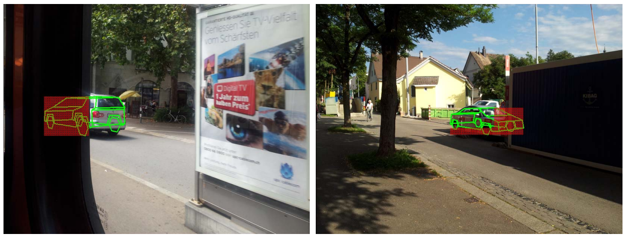


Figure 1. Fully automatic 3D shape, pose, and occlusion estimation.

However, when working with an explicit 3D representation of an object class, it should in principle be possible to estimate that pattern. Addressing self-occlusion is rather straight-forward with a 3D representation [37, 40], since it is fully determined by the object shape and pose. On the other hand, inter-object occlusion is much harder to model, because it introduces relatively many additional unknowns (the occlusion states of all individual regions/parts of the object). Some part-based models resort to a data-driven strategy: every individual part can be occluded or unoccluded, and that latent state is estimated together with the object shape and pose [20, 12].

Such a model has two weaknesses: first, it does not make any assumptions about the nature of the occluder, and can therefore lead to rather unlikely occlusion patterns (*e.g.* arbitrarily scattered small occluders). And second, it will have limited robustness, require careful tuning, and be hard to adapt to different scenarios. The latter is due to the tendency to simply label any individual part as occluded whenever it does not fit the evidence, and the associated brittle trade-off between the likelihood of occlusion and the uncertainty of the image evidence.

We argue that in many scenarios a per-part occlusion model is unnecessarily general. Rather, one can put a strong prior on the co-occurrence of part occlusions, because most occluders are compact objects, and all one needs to know about them is the (also compact) projection of their outline onto the image plane. We therefore propose to restrict the possible occluders to a small finite set that can be explicitly enumerated, and to estimate the type of occluder and its location during inference. The very simple, but powerful intuition behind this is that *when restricted to compact regions inside the object's bounding box, the number of possible occlusion patterns is in fact very small.* Still such an occluder model is more general than one that only truncates the bounding box from left, right, above or below (*e.g.* [35, 6]) or at image boundaries [32], *c.f.* Fig. 2. *E.g.*, the proposed model can represent a vertical pole occluding the middle of the object, a frequent case in urban scenarios.

The contribution described in this paper is a viewpoint-invariant method for detailed reconstruction of severely occluded objects in monocular images. To obtain a complete framework for detection and reconstruction, the novel method is initialized with a variant of the *poselets* framework [2] adapted to the needs of our 3D object model. The object representation itself has three parts: a deformable shape model in the form of an active shape model defined over local object parts, an appearance model which integrates evidence from detectors for the parts as well as their configurations, and an occlusion model in the form of a set of occlusion masks. Experiments on images with strong occlusions show that the model can correctly infer even large occluders, and enables monocular 3D modeling in situations where representations without occlusion model fail.

2. Related work

In the early days of computer vision, 3D object models with a lot of geometric detail [27, 3, 22, 30] commanded a lot of interest, but unfortunately failed to tackle challenging real world imagery. Most current object class detectors provide coarse outputs in the form of 2D or 3D bounding boxes along with classification into a discrete set of viewpoints [38, 28, 21, 9, 24, 29, 33, 25, 13]. Recently, there has been renewed interest in providing geometrically more detailed outputs, with different degrees of geometric consistency across viewpoints [20, 40, 37, 26, 15, 39]. Such models have the potential to enhance high-level reasoning about objects and scenes, *e.g.* [16, 7, 34, 14, 36].

Unfortunately occlusion, which is one of the most challenging impediments to visual object class modeling, has largely remained untouched in the context of such fine-grained object models. Recent attempts at occlusion reasoning in 2D object recognition include modeling the visibility/occluder mask [10, 35, 32, 11, 17], training detectors for occluded objects in specific frequently found configurations [31], using depth and/or motion cues [6, 23], asserting an "occluder part" when part evidence is missing [12], applying RANSAC to choose a subset of parts [20], encoding occlusion states using local mixtures [15], and using a large number of partial object detectors which cluster together to give the full object [2], without explicit occluder modeling.

Fixed global object models have been known to give good results for fully visible object recognition [5], often outperforming part-based models. However, part-based models have unsurprisingly been found preferable for occlusion invariant detection [2, 12]; in fact even when "global" models are extended to cope with occlusions [35, 17] they are divided into many local cells, which are effectively treated as parts with fixed relative locations. Part-based 3D object models with strong geometric constraints as [20, 40] are thus strong candidates for part-level occlusion reasoning: they can cope with locally missing evidence, but still ensure the relative part placement always corresponds to a plausible global shape. On the downside, these are computationally fairly expensive models, therefore their evaluation on images in [20] is limited to a small bounding box around the object of interest. We thus propose a two-layer model, where objects are first detected with a variant of the *poselet* method [2] to obtain a rough localization and pose; then a detailed shape, pose and occlusion mask are inferred with an explicit 3D model as in [40, 39], which also includes the additional clues for part placement afforded by the preceding detector. Note that the two layers go together well, since spatially compact occluders will leave configurations of adjacent object parts ("poselets") visible.

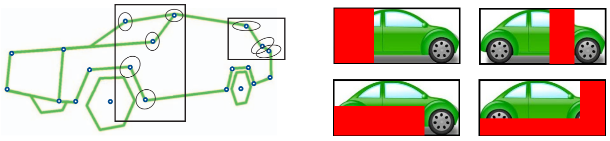
3. Model

We propose to split 3D object detection and modeling into two layers. The first layer is a representation in the spirit of the *poselet* framework [2], *i.e.* a collection of viewpoint-dependent part $configurations$ tied together by relatively loose geometric constraints. The purpose of this layer is to find, in a large image, approximate 2D bounding boxes with rough initial estimates of the objects' pose. The part-based structure enables the model to deal with partial occlusion, and provides evidence for visible $configurations$ that can be used in the second layer.

The second layer is a 3D active shape model based on local $parts$, augmented with a collection of explicit occlusion masks. The ASM tightly constrains the object geometry to plausible shapes, and thus can more robustly predict object shape when parts are occluded, respectively predict the locations of the occluded parts. The model also includes the activations of the $configurations$ from the first layer as additional evidence, tying the two layers together.

3.1. Parts and part configurations

We start the explanation with the local appearance model. The atomic units of our representation are $parts$, which are small square patches located at salient points of the object. The patches are encoded with densely sam-



(a) (b)

Figure 2. (a) Two larger part $configurations$ comprising of multiple smaller $parts$, as well as their relative distributions, (b) a few example occlusion masks.

pled shape-context descriptors [1], and a multi-class Random Forest is trained to recognize them. The classifier is viewpoint-invariant, meaning that one class label includes views of a part over all poses in which the part is visible [39]. This marginalization over viewpoints speeds up part detection (which is the bottleneck of the method) by an order of magnitude[1] compared to individual per-viewpoint classifiers [1, 29, 40], while we did not observe a perfomance drop at the system level in spite of visibly blurrier part likelihoods. Additionally, the classifier also has a background class, which will be used for normalization (*c.f.* Sec.3.4). Like [29, 40, 39] we exploit the fact that with modern descriptors the part classifier can be trained mostly on synthetic renderings of 3D CAD models rather than on real data, which massively reduces the annotation effort.

The basic unit of the first layer are larger part $configurations$ ranging in size from 25% to 60% of the full object extent. These are defined in the spirit of *poselets*: Small sets of the local $parts$ described above are chosen and clustered (with standard k-means) according to the parts' spatial layout. The advantage of this clustering is that it discovers when object portions have high variability in appearance, *e.g.* the rear portion of sedans *vs.* hatchbacks as seen in a side view. To account for the spatial variability *within* a $configuration$, a single component DPM detector [9] is trained for each configuration. We found that for these detectors real training data is needed, thus they are trained on annotated training images.

3.2. Geometric model

As explained earlier, we employ different geometric models for the initial detection and the subsequent 3D modeling. The first layer follows the philosophy of the ISM/poselet method. For each configuration the mean offset from the object centroid as well as the mean relative scale are stored during training, and at test time detected $configurations$ cast a vote for the object center and scale. These votes are then combined via greedy agglomerative clustering, similar to [2]. After non-maximum suppression, the output of the first layer consists of a set of approximate 2D bounding boxes, each with a coarse pose estimate (quantized to 8 canonical viewpoints) and a list of activated $configurations$.

The second layer utilizes a more explicit representation of global object geometry that is better suited for estimating detailed 3D object shape and pose. In the tradition of *active shape models* we learn a deformable 3D wireframe from annotated 3D CAD models, like in [40, 39]. The wireframe model is defined through an ordered collection of n vertices in 3D-space, chosen at salient points on the object surface in a fixed topological layout. Following standard point-based shape analysis [4] the object shape and variability are represented as the sum of a mean wireframe μ and deformations along r principal component directions $\mathbf{p}_j$. The geometry parameters s_k determine the amount of deviation from the mean shape (in units of standard deviation σ_j along the respective directions): $\mathbf{X}(\mathbf{s}) = \mu + \sum_{k=1}^{r} s_k \sigma_k \mathbf{p}_k + \epsilon$. The $parts$ described above are defined as small windows around the 2D projection of such a vertex ($\approx 10\%$ in size of the full object width). The parts cover the full extent of the represented object class, thus they allow for fine-grained estimation of 3D geometry and continuous pose, as well as for detailed reasoning about occlusion relations. We point out once more that these parts are viewpoint-independent, *i.e.* a part covers the appearance of a vertex over the entire viewing sphere.

3.3. Explicit occluder representation

While the first layer contains only implicit information about occluders (in the form of supposedly visible, but undetected $configurations$), the second layer includes an explicit occluder representation. Occluders are assumed to block the view onto a spatially connected region of the object. Due to the object being modeled as a sparse collection of parts, occluders can only be distinguished if the visibility of at least one part changes, which further reduces the space of possible occluders. Thus, one can well approximate the set of all occluders by a discrete set of occlusion masks a (for convenience we denote the empty mask which leaves the object fully visible by a_0). Fig. 2(b) shows exemplary occlusion masks.

With that set, we aim to explicitly recover the occlusion pattern during second-layer inference, by selecting one of the masks. All parts falling inside the occlusion mask are considered occluded, and consequently their detection scores are not considered in the objective function (Sec. 3.4). Instead, they are assigned a fixed low score, corresponding to a weak uniform prior that prefers parts to be visible and counters the bias to “hide behind the occluder”.

Occlusion of parts is modeled by indicator functions $o_j(\mathbf{s}, \boldsymbol{\theta}, a)$, where j represents the part index, $\mathbf{s}$ represents the object geometry (3.2) and $\boldsymbol{\theta}$ the viewpoint. The set of masks a_i act as a prior that specifies which parts occlusions can co-occur. For completeness we mention that object self-

[1] Also, training is two orders of magnitude faster.

occlusion is modeled with the same indicator variables, but does not require separate treatment, since it is completely determined by shape and pose.

3.4. Shape, pose, and occlusion estimation

During inference, we attempt to find instances of the 3D shape model and of the occlusion mask that best explain the observed image evidence.

Recall that we wish to estimate an object's 3D pose (5 parameters, assuming no in-plane rotation), geometric shape (7 ASM shape parameters), and an occluder index (1 parameter). Taken together, we are faced with a 13-dimensional search problem, which would be prohibitively expensive even for a moderate image size. We therefore first cut down the search space in the first layer with a simpler and more robust object detection step, and then fit the full model locally at a small number of (candidate) detections.

First layer inference starts by detecting instances of our part $configurations$ in the image with the corresponding DPM detectors. Each detected configuration casts an associated vote for the full object 2D location and scale $\mathbf{q} = (q_x, q_y, q_s)$, and for the pose $\boldsymbol{\theta} = (\theta_{az}, \theta_{el})$. At this point, the azimuth angle is restricted to a small set of discrete steps and the elevation angle is fixed, both to be refined in the second layer. The votes are clustered with a greedy agglomerative clustering scheme as in [2] to obtain detection hypotheses $\mathcal{H}$, each with a list of contributing configurations $\{l_1 \dots l_p\}$ that voted for the object's presence.

Part location prediction from first layer. Since the configurations are made up of multiple parts confined to a specific layout with little spatial variability (Sec. 3.1), their detected instances l_i already provide some information about the part locations in image space. The means $\boldsymbol{\mu}_{ij}$ and covariances $\boldsymbol{\sigma}^2_{ij}$ of the parts' locations within a configuration's bounding box are estimated from the training data, and v_{ij} are binary flags indicating which parts j are found within the $configuration$ l_i. Fig. 2(a) illustrates two such larger $configurations$, whose detection can be used to predict the location of the constituent parts as gaussian distributions with the respective means and covariances relative to the bounding box of the configuration.[2]

Second layer objective function. After evaluating the first layer of the model we are left with a sparse set of (putative) detections, such that we can afford to evaluate a relatively expensive objective function. We denote an object instance by $\mathbf{h} = (\mathbf{s}, f, \boldsymbol{\theta}, \mathbf{q}, a)$, comprising of shape parameters $\mathbf{s}$ (eqn. 3.2), camera focal length f, viewpoint parameters for azimuth and elevation $\boldsymbol{\theta}$, and translation and scale parameters in image space $\mathbf{q}$. The projection matrix P that maps the 3D vertices $\mathbf{X}_j(\mathbf{s})$ to image points $\mathbf{x}_j$ is assumed to depend only on $\boldsymbol{\theta}$, and $\mathbf{q}$, while f is fixed, assuming similar perspective effects for all images: $\mathbf{x}_j = \mathsf{P}(f, \boldsymbol{\theta}, \mathbf{q})\mathbf{X}_j(\mathbf{s})$.

Fitting the model amounts to finding a MAP-estimate of the objective function $\mathcal{L}(\mathbf{h})$:

$$\hat{\mathbf{h}} = \arg\max_h \left[\mathcal{L}(\mathbf{h})\right] \ , \tag{1}$$

$$\mathcal{L}(\mathbf{h}) = \max_{\varsigma} \left[\frac{1}{\sum_{j=1}^m o_j(\mathbf{s}, \boldsymbol{\theta}, a_0)} \sum_{j=1}^m \Big(\mathcal{L}_v + \mathcal{L}_o + \mathcal{L}_c\Big)\right] . \tag{2}$$

The factor $1/\sum_{j=1}^m o_j(\mathbf{s}, \boldsymbol{\theta}, a_0)$ normalizes for the varying number of self-occluded parts at different viewpoints. For each potentially visible part there are three terms: $\mathcal{L}_v$ is the evidence $S_j(\varsigma, \mathbf{x}_j)$ for part j if it is visible, found by looking up the detection score at image location $\mathbf{x}_j$ and scale ς. Part likelihoods are normalized with the background score $S_b(\varsigma, \mathbf{x}_j)$, as in [33]. $\mathcal{L}_o$ assigns a fixed likelihood c to the part, if it lies under the occlusion mask. $\mathcal{L}_c$ measures how well the part j is predicted by the larger $configurations$.

$$\mathcal{L}_v = o_j(\mathbf{s}, \boldsymbol{\theta}, a) \log \frac{S_j(\varsigma, \mathbf{x}_j)}{S_b(\varsigma, \mathbf{x}_j)} \ , \tag{3}$$

$$\mathcal{L}_o = \big(o_j(\mathbf{s}, \boldsymbol{\theta}, a_0) - o_j(\mathbf{s}, \boldsymbol{\theta}, a)\big) c \ , \tag{4}$$

$$\mathcal{L}_c = \frac{o_j(\mathbf{s}, \boldsymbol{\theta}, a)}{p} \sum_{i=1}^p v_{ij} \log\left(1 + \lambda \mathcal{N}(\mathbf{x}_j; \boldsymbol{\mu}_{ij}, \boldsymbol{\sigma}^2_{ij})\right) \ . \tag{5}$$

Second layer inference. The objective (2) is a mixed discrete-continuous function which is neither convex nor smooth, and thus cannot be easily maximized. We find an approximate MAP-estimate $\hat{\mathbf{h}}$ with sample-based stochastic hill-climbing. Specifically, we maintain a set of weighted samples (particles), each corresponding to a distinct set of values in the space of object hypotheses $\{\mathbf{s}, \boldsymbol{\theta}, \mathbf{q}, a\}$. Particles are iteratively updated, by re-sampling individual parameters from independent Gaussians centered at the current values, similar to [18]. In our scheme the variances of these Gaussians are gradually reduced according to a fixed annealing schedule. Other than the remaining parameters, the mask indices a are discrete and have no obvious ordering. To define similarity between them we sort the set of masks w.r.t. the Hamming distance from the current one, then we sample the offset in this ordering from a Gaussian.

The inference is initialized at the location, scale and pose returned by the first layer, while the initial shape parameters are chosen randomly and the occlusion mask is set to a_0.

4. Experiments

In the following, we evaluate the performance of our approach in detail, focusing on its ability to recover fine-grained, part-level accurate object shape and accompanying occlusion estimates. In particular, we quantify the ability of our method to localize entire objects (Sect. 4.3), to localize their constituent parts (Sect. 4.5), and to estimate occluded object portions (in the form of part occlusion labels), for varying levels of occlusion (Sect. 4.4).

[2] In practice it is beneficial to only use configurations whose part predicitons are sufficiently accurate, as determined by cross-validation.

The free parameters for (4) and (5) are estimated by cross-validation on the 3D Object Classes [28], for which part level annotations are publicly available [40]. The set of 288 occlusion masks has been generated automatically and pruned manually to exclude very unlikely masks.

4.1. Data set

As a testbed we have collected a novel, challenging data set of inner-city street scenes. It consists of 101 images of resolution 2 mega-pixels, showing street scenes with cars, with occlusions ranging from 0% to $> 60\%$ of the bounding box as well as the parts. Although there are several publicly available car datasets, none of them is suitable for our purposes, sinc we found that part detector performance deteriorates significantly for objects smaller than 60 pixels in height. Some datasets do not contain occluded cars (*e.g.* 3D Object Classes [28], EPFL Multiview Cars [24]); others do, but have rather low resolution (Ford Campus Vision and Lidar, Pascal VOC [8]), which makes them unsuitable for detailed geometric model fitting – and also seems unrealistic, given today's omnipresent high-resolution cameras. We further opted for taking the pictures ourselves, in order to avoid the strong bias of internet search towards high-contrast, high-saturation images. Figures 4,5 show example images from the new data set.[3]

4.2. Model variants and baselines

We evaluate and compare the performance of the following competing models: *(i)* a naive baseline without 3D estimation, which places a fixed canonical 3D car (the mean of our active shape model) inside the detected first-layer bounding box in the estimated (discrete) pose. *(ii)* the ASM model of [40, 39], which corresponds to the second layer of our model without any form of occlusion reasoning (*i.e.* assuming that all parts are visible except for self-occlusions), and without using the part $configurations$ from the first layer. *(iii)* the proposed model, including prediction of occluders, but not using the $configurations$ during second-layer inference. *(iv)* our full model with occluder prediction and leveraging additional evidence from $configurations$ for second-layer inference.

4.3. Object localization

We commence by verifying that our first layer, *i.e.* a combination of DPM *configuration* detectors and *poselet*-style voting, is competitive with alternative algorithms for detecting objects in 2D. To that end we compare our first layer, trained on a dataset comprising of around 1000 full car images downloaded from the internet, with the original poselet implementation [2] pre-trained on Pascal VOC [8] (training code for [2] is not publicly available). We also include the deformable part model (DPM, [9]), both trained on the same 1000 car images (using default parameters), as well as the pre-trained model (on Pascal VOC [8]), as a popular state-of-the-art reference. Unfortunately neither of these implementations directly outputs a viewpoint.

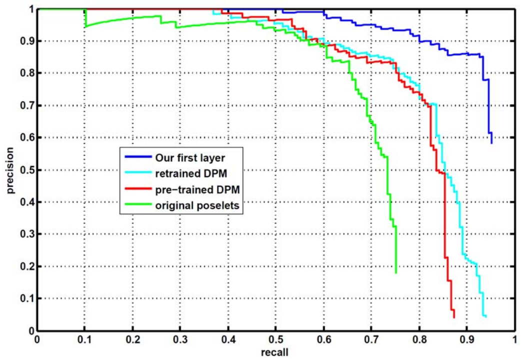


Figure 3. Object detection acuracy of different 2D detectors.

Protocol. We follow the classical object detection protocol of Pascal VOC [8], plotting precision *vs*. recall for 50% intersection-over-union between predicted and ground truth bounding box.

Results. Precision-recall curves are shown in Fig. 3. We observe that the original poselets [2] already perform reasonably well on our data (67% AP). The pre-trained DPM [9] improves the results to 76% AP , and the retrained model, to 79% AP. Our first layer outperforms both by a significant margin, achieving 88% AP, which we consider a solid basis for the subsequent 3D inference. In particular we point out that the combination of a strong part detector with Hough-style voting reaches high recall (up to 95%) at reasonable precision. The fact that only few instances are irrevocably lost in the first layer confirms that splitting into a coarse detection layer and a detailed modeling layer is a viable approach (see Tab. 1).

	Full dataset	$< 80\%$ visibility	$< 60\%$ visibility
Total cars	165	96	48
Detected	147	85	42

Table 1. First-layer detection results (bounding box and 1D pose). Subsequent second-layer results are given for the detected instances (line "detected").

4.4. Occlusion estimation

We proceed by evaluating how well our model can distinguish between occluded and unoccluded parts. Note that while this ability is potentially also useful for further reasoning about the occluder, its primary importance here lies in the 3D object modeling itself: a good estimate of the part-level occlusion state is necessary in order not to mistakenly use evidence from background structures, and hence forms

[3] The data set, along with all training data and annotations, pre-trained models, and source code is made available at http://www.igp.ethz.ch/photogrammetry/downloads.

the basis for recovering the objects' 3D extent and shape.

Protocol. The predicted part occlusions are evaluated as two-class classification: we first remove all self-occluded $parts$, and then compare occlusion labeling o_j induced by the estimated occluder a_i to ground truth annotations.

Results. Tab. 2 shows the percentages of correctly inferred part occlusions. First, we observe that the acuracy decreases with increasing occlusion level, matching our intuition. Baseline 1 is obviously not applicable, since it offers no possibility to decide about part-level occlusion. To make the second baseline comparable, which also does not make occlusions explicit, we place a threshold (equal in value to c used in the likelihood (2)) on part detection scores and call parts with too low scores occluded. Although that heuristic works surprisingly well, our occlusion inference outperforms the baseline by significant margins (4.5 – 5.9%) for all levels of occlusion. Additionally using the active $configurations$ from the first layer during inference boosts classification performance by a further 1.2 – 3.0%. We point out that the additional evidence provided by the larger $configurations$ is most beneficial at high levels of occlusion, and that even for heavily occluded vehicles that are only 30 – 60% visible, 83.1% of the part occlusions are correctly predicted.

	Full dataset	< 80% visibility	< 60 % visibility
baseline 1	—	—	—
baseline 2 [40, 39]	79.5%	76.7%	75.6%
w/o configurations (ours)	84.4%	82.6%	80.1%
w/ configurations (ours)	**85.6%**	**84.7%**	**83.1%**

Table 2. Part-level occlusion prediction (percentage of correctly classified parts). See text for details.

4.5. Part localization

The primary goal of our occlusion model is better 3D object modeling: we wish to correctly predict objects' spatial extent, shape and pose, to support higher-level tasks such as monocular depth estimation, free-space computation and physically plausible, collision-free scene understanding. To quantify the ability to recover 3D extent and shape, we assess how well individual parts of the 3D geometric model can be localized. Since we have no 3D ground truth, part localization accuracy is measured in the 2D image plane by comparing to manual annotations.

Protocol. We follow the common evaluation protocol of human body pose estimation and report the average percentage of correctly localized parts, using a relative threshold adjusted to the size of the car. The threshold is set to 20 pixels for a car of size 500×170 pixels, *i.e.* $\approx 4\%$ of the total length.

Results. Part localization results for different levels of ocluſion are given in Tab. 3. We make the following observations. First, baseline 1 performs poorly, *i.e.* the bounding box and pose predictions of a 2D detector and/or a rigid average car are insufficient. Second our occlusion-aware approach outperforms the 3D-ASM of [40, 39] without occlusion modeling by 2.5% on the entire dataset, and that margins increase to 5.3% for the heavily occluded cars. Third, adding evidence form $configurations$ brings only a small improvement for the full dataset, but the improvement is more pronounced for heavier occlusions. Finally, we manage to sucessfully localize $> 80\%$ of the parts even at occlusion levels of 40% or more.

Fig. 5 shows qualitative examples, highlighting the differences between the naive baseline 1, the baseline approach without occlusion modeling [40, 39], and the two evaluated variant of our model. Clearly, the fits without occlusion model are severely disturbed in the presence of even moderate occlusion. Our approach without $configurations$ seems to perform as well as the full model when it comes to predicting the occluder, but is slightly more prone to mistakes concerning the overall object shape (e.g., rows a, b). Figure 4 shows further qualitative results of the full model.

	Full dataset	< 80% visibility	< 60 % visibility
baseline 1	32.0%	33.6%	39.7%
baseline 2 [40, 39]	80.0%	75.6%	74.5%
w/o configurations (ours)	82.5%	80.0%	79.8%
w/ configurations (ours)	**82.7%**	**80.7%**	**83.5%**

Table 3. Part localization accuracy (percentage of correctly localized parts). See text for details.

5. Conclusion

We have explored the problem of occlusion in the context of geometric, part-based 3D object class representations for object detection and modeling. We have proposed a two-layer model, consisting of a robust, but coarse 2D object detector, followed by a detailed 3D model of pose and shape. The first layer accumulates votes from view-point dependent part $configurations$, such that it can tolerate quite large degrees of occlusion, but does not explicitly detect them. The second layer combines an explicit deformable 3D shape model over smaller $parts$ with evidence from the first-level $configurations$, as well as with an explicit occlusion model in the form of a collection of possible occlusion masks. Although that representation of occlusion is rather simple, experiments on detecting and modeling cars in a dataset of street scenes have confirmed the model to correctly estimate both the occlusion pattern and the car shape and pose even under severe occlusion, clearly outperforming a baseline that is agnostic about occlusions.

Acknowledgements. This work has been supported by the Max Planck Center for Visual Computing and Communication.

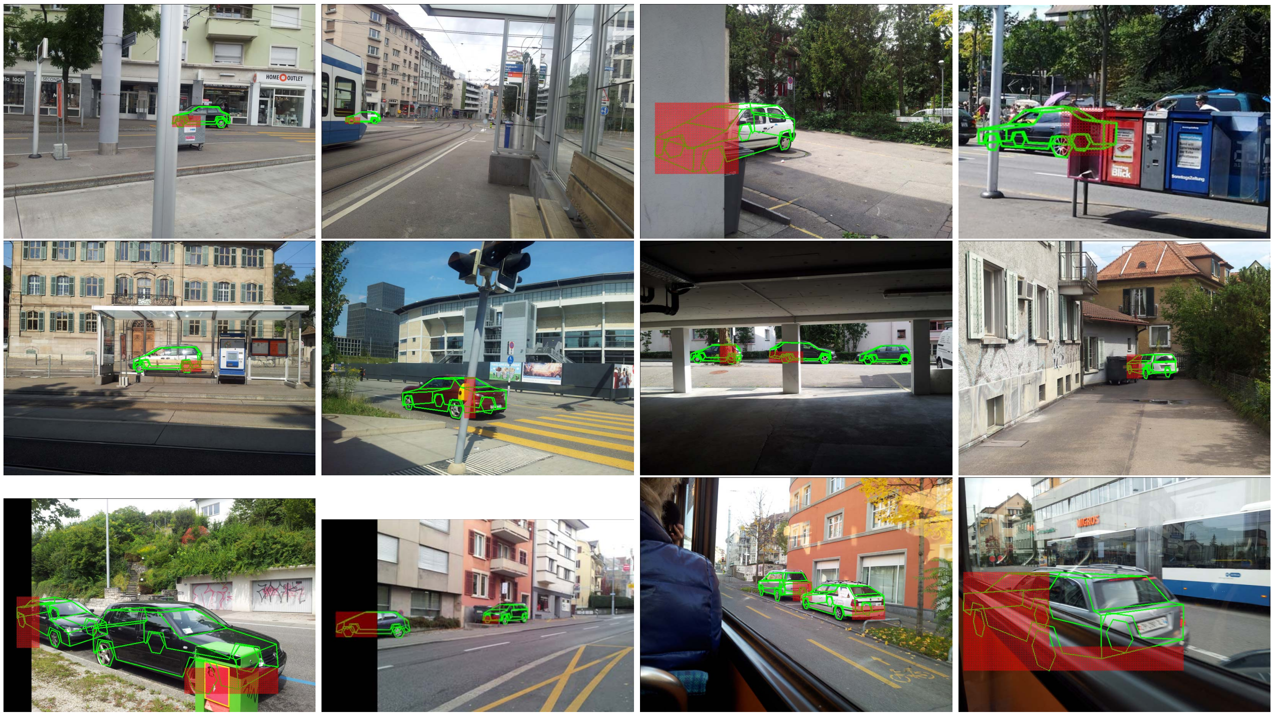


Figure 4. Example detections using our full system.

References

[1] M. Andriluka, S. Roth, and B. Schiele. Pictorial structures revisited: People detection and articulated pose estimation. *CVPR 2009*.

[2] L. Bourdev and J. Malik. Poselets: Body part detectors trained using 3d human pose annotations. *ICCV 2009*.

[3] R. A. Brooks. Symbolic reasoning among 3-d models and 2-d images. *Artificial Intelligence*, 17(1-3), 1981.

[4] T. F. Cootes, C. J. Taylor, D. H. Cooper, and J. Graham. Active shape models, their training and application. *CVIU*, 61(1), 1995.

[5] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. *CVPR 2005*.

[6] M. Enzweiler, A. Eigenstetter, B. Schiele, and D. M. Gavrila. Multi-cue pedestrian classification with partial occlusion handling. *CVPR 2010*.

[7] A. Ess, B. Leibe, K. Schindler, and L. V. Gool. Robust multi-person tracking from a mobile platform. *PAMI*, 31(10), 2009.

[8] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (VOC) challenge. *IJCV*, 88(2), 2010.

[9] P. F. Felzenszwalb, R. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part based models. *PAMI*, 32(9), 2010.

[10] R. Fransens, C. Strecha, and L. V. Gool. A mean field EM-algorithm for coherent occlusion handling in MAP-estimation. *CVPR 2006*.

[11] T. Gao, B. Packer, and D. Koller. A segmentation-aware object detection model with occlusion handling. *CVPR 2011*.

[12] R. B. Girshick, P. F. Felzenszwalb, and D. McAllester. Object detection with grammar models. *NIPS 2011*.

[13] D. Glasner, M. Galun, S. Alpert, R. Basri, and G. Shakhnarovich. Viewpoint-aware object detection and pose estimation. *ICCV 2011*.

[14] V. Hedau, D. Hoiem, and D. Forsyth. Thinking inside the box: Using appearance models and context based on room geometry. *ECCV 2010*.

[15] M. Hejrati and D. Ramanan. Analyzing 3d objects in cluttered images. *NIPS 2012*.

[16] D. Hoiem, A. Efros, and M. Hebert. Putting objects in perspective. *IJCV*, 80(1), 2008.

[17] S. Kwak, W. Nam, B. Han, and J. H. Han. Learn occlusion with likelihoods for visual tracking. *ICCV 2011*.

[18] M. Leordeanu and M. Hebert. Smoothing-based optimization. *CVPR 2008*.

[19] M. J. Leotta and J. L. Mundy. Vehicle surveillance with a generic, adaptive, 3d vehicle model. *PAMI*, 33(7), 2011.

[20] Y. Li, L. Gu, and T. Kanade. Robustly aligning a shape model and its application to car alignment of unknown pose. *PAMI*, 33(9), 2011.

[21] J. Liebelt, C. Schmid, and K. Schertler. Viewpoint independent object class detection using 3D feature maps. *CVPR'08*.

[22] D. Lowe. Three-dimensional object recognition from single two-dimensional images. *Artificial Intelligence*, 31(3), 1987.

[23] D. Meger, C. Wojek, B. Schiele, and J. Little. Explicit occlusion reasoning for 3d object detection. *BMVC 2011*.

[24] M. Ozuysal, V. Lepetit, and P. Fua. Pose estimation for category specific multiview object localization. *CVPR 2009*.

[25] N. Payet and S. Todorovic. From contours to 3d object detection and pose estimation. *ICCV 2011*.

[26] B. Pepik, P. Gehler, M. Stark, and B. Schiele. 3DDPM - 3d deformable part models. *ECCV 2012*.

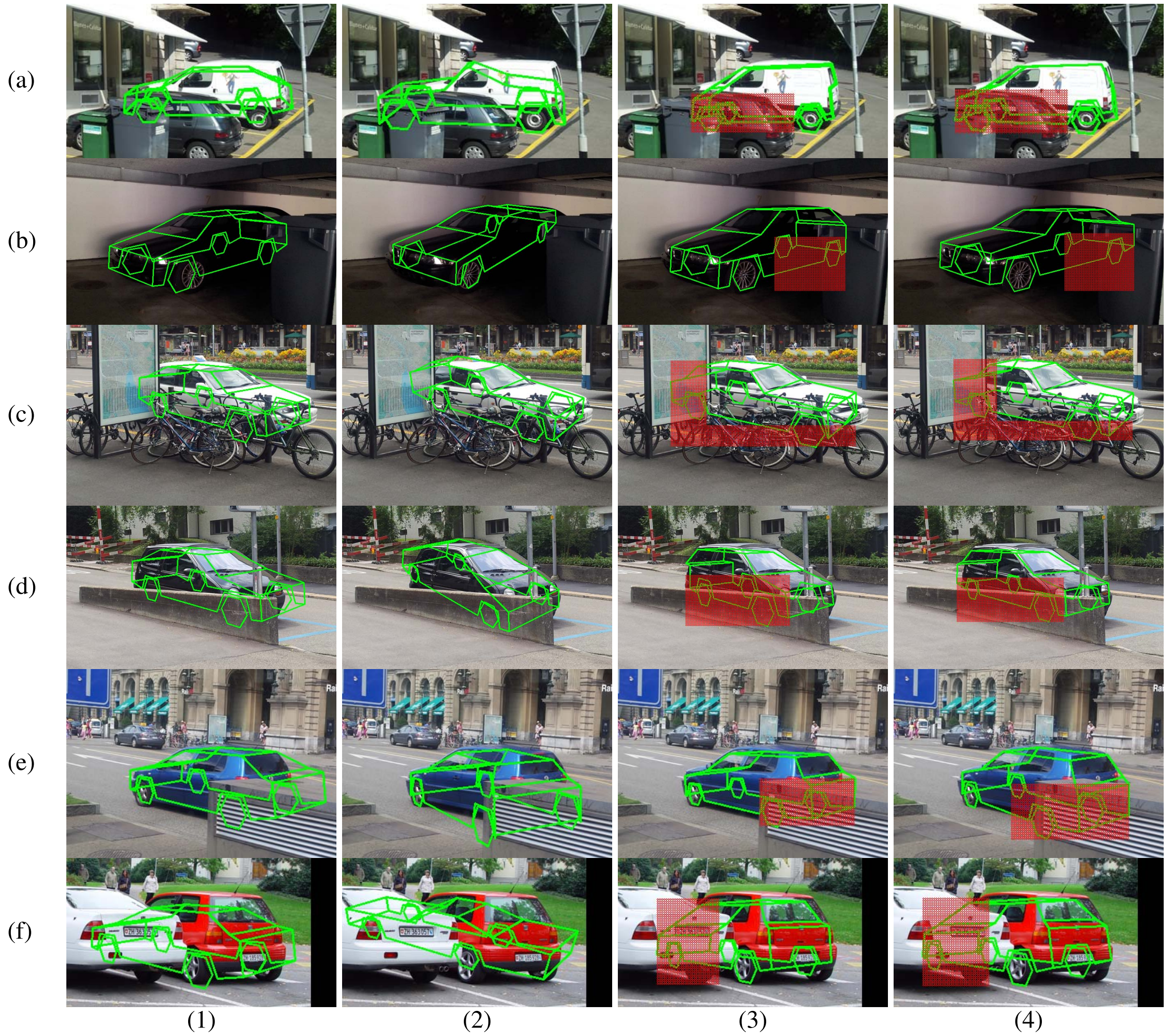


Figure 5. Comparing model fits: canonical car shape in detected bounding box *i.e*. baseline 1 (column 1), baseline 2 [40, 39] (column 2), without *poselets* (column 3), with *poselets* (column 4).

[27] L. G. Roberts. *Machine Perception of Three-Dimensional Solids*. PhD thesis, MIT, 1963.

[28] S. Savarese and L. Fei-Fei. 3D generic object categorization, localization and pose estimation. *ICCV 2007*.

[29] M. Stark, M. Goesele, and B. Schiele. Back to the future: Learning shape models from 3D CAD data. *BMVC 2010*.

[30] G. D. Sullivan, A. D. Worrall, and J. Ferryman. Visual object recognition using deformable models of vehicles. *IEEE Workshop on Context-Based Vision*, 1995.

[31] S. Tang, M. Andriluka, and B. Schiele. Detection and tracking of occluded people. *BMVC 2012*.

[32] A. Vedaldi and A. Zisserman. Structured output regression for detection with partial truncation. *NIPS 2009*.

[33] M. Villamizar, H. Grabner, J. Andrade-Cetto, A. Sanfeliu, L. V. Gool, and F. Moreno-Noguer. Efficient 3d object detection using multiple pose-specific classifiers. *BMVC 2011*.

[34] H. Wang, S. Gould, and D. Koller. Discriminative learning with latent variables for cluttered indoor scene understanding. *ECCV 2010*.

[35] X. Wang, T. Han, and S. Yan. An HOG-LBP human detector with partial occlusion handling. *ICCV 2009*.

[36] C. Wojek, S. Roth, K. Schindler, and B. Schiele. Monocular 3d scene modeling and inference: Understanding multi-object traffic scenes. *ECCV 2010*.

[37] Y. Xiang and S. Savarese. Estimating the aspect layout of object categories. *CVPR 2012*.

[38] P. Yan, S. Khan, and M. Shah. 3D model based object class detection in an arbitrary view. *ICCV 2007*.

[39] M. Z. Zia, M. Stark, B. Schiele, and K. Schindler. Detailed 3d representations for object recognition and modeling. *PAMI*, 2013.

[40] M. Z. Zia, M. Stark, K. Schindler, and B. Schiele. Revisiting 3D geometric models for accurate object shape and pose. *3dRR 2011*.