

DOC: Deep OCclusion Estimation from a Single Image

Peng Wang^{1(✉)} and Alan Yuille^{1,2}

¹ University of California, Los Angeles, USA
jerryking234@gmail.com

² John Hopkins University, Baltimore, USA
alan.1.yuille@gmail.com

Abstract. In this paper, we propose a deep convolutional network architecture, called DOC, which detects object boundaries and estimates the occlusion relationships (i.e. which side of the boundary is foreground and which is background). Specifically, we first represent occlusion relations by a binary edge indicator, to indicate the object boundary, and an occlusion orientation variable whose direction specifies the occlusion relationships by a left-hand rule, see Fig. 1. Then, our DOC networks exploit local and non-local image cues to learn and estimate this representation and hence recover occlusion relations. To train and test DOC, we construct a large-scale instance occlusion boundary dataset using PASCAL VOC images, which we call the PASCAL instance occlusion dataset (PIOD). It contains 10,000 images and hence is two orders of magnitude larger than existing occlusion datasets for outdoor images. We test two variants of DOC on PIOD and on the BSDS ownership dataset and show they outperform state-of-the-art methods typically by more than 5AP. Finally, we perform numerous experiments investigating multiple settings of DOC and transfer between BSDS and PIOD, which provides more insights for further study of occlusion estimation.

1 Introduction

Humans are able to recover the occlusion relationships of objects from single images. This has long been recognized as an important ability for scene understanding and perception [4, 15]. As shown on the left of Fig. 1, we can use occlusion relationships to deduce that the person is holding a dog, because the person's hand occludes the dog and the dog occludes the person's body. Electrophysiological [18] and fMRI [13] studies suggest that occlusion relationships are detected as early as visual area V2. Biological studies [9] also suggest that occlusion detection can require feedback from higher level cortical regions, indicating that long-range context and semantic-level knowledge may be needed. Psychophysical studies show that there are many cues for occlusion including edge convexity [23], edge-junctions, intensity gradients, and texture [35].

Electronic supplementary material The online version of this chapter (doi:[10.1007/978-3-319-46448-0_33](https://doi.org/10.1007/978-3-319-46448-0_33)) contains supplementary material, which is available to authorized users.



Fig. 1. Left: Occlusion boundaries represented by orientation θ (the red arrows), which indicates occlusion relationship using the “left” rule where the left side of the arrows is foreground. Right: More examples from our Pascal instance occlusion dataset (PIOD). (Color figure online)

Computer vision researchers have also used similar cues for estimating occlusion relations. A standard strategy is to apply machine learning techniques to combine cues like convexity, triple-points, geometric context, image features like HOG, and spectral features, e.g. [5, 20, 37, 45]. These methods, however, mostly rely on hand-crafted features and have only been trained on the small occlusion datasets currently available. But in recent years, fully convolutional deep convolutional neural networks (FCN) [29] that exploit local and non-local cues, and trained on large datasets, have been very successful for related visual tasks such as edge detection [49] and semantic segmentation [6]. In addition, visualization of deep networks [30, 51] show that they can also capture and exploit the types of visual cues needed to estimate occlusion relations.

This motivates us to apply deep networks to estimate occlusion relationships, which requires constructing a large annotated occlusion dataset. This also requires making design choices such as how to represent occlusion relations and what type of deep network architecture is best able to capture the local and non-local cues required. We represent occlusion relations by a per-pixel representation with two variables: (i) a binary edge variable to indicate if a pixel is on a boundary, and (ii) a continuous-valued occlusion orientation variable (at each edge pixel) in the tangent direction of the edge whose direction indicates the occlusion relationship using the left rule (i.e. the region to the left of the edge is in front of the region to the right). Our DOC network architecture is based on recent fully convolutional networks [29] and is multi-scale so that it can take into account local and non-local image cues. More specifically, we design two versions of DOC based on [6, 49] respectively.

To construct our dataset, we select PASCAL VOC images [12] where many of the object boundaries have already been annotated [7, 16]. This simplifies our annotation task since we only have to label the occlusion orientation variable specifying border ownership. Our Pascal Instance Occlusion Dataset (PIOD) consists of 10,000 images and is two orders of magnitude larger than existing ones such as the BSDS border ownership [37] (200 images) and GeoContext [20] (100 images). We note that the NYU depth dataset [41] (1449 indoor images) can also be used to test occlusion relations, but restricted to indoor images.

This paper makes two main contributions: (1) We design a new representation and corresponding loss for FCN architecture showing that it performs well and is computationally efficient (0.6s/image). (2) We create a large occlusion boundary dataset over the PASCAL VOC images, which is a new resource for studying occlusion. We will release our models, code and dataset.

2 Related Work

In computer vision, studying occlusion relations has often been confined to multiview problems such as stereo and motion [2, 17, 42, 44, 48]. In these situations multiple images are available and so occlusion can be detected by finding pixels which have no correspondence between images [3, 14].

Inferring occlusion relations from a single image is harder. Early work restricted to simple domains, e.g. blocks world [38] and line drawings [8] using a variety of techniques ranging from algebraic [43] to the use of markov random fields (MRF) for capturing non-local context [39, 50]. The 2.1D sketch [34] is a mid-level representation of images involving occlusion relations, but it was conceptual and served to draw attention to the importance of this task.

Research on detecting occlusion relations in natural images was stimulated by the construction of the BSDS border ownership dataset [37]. Computer vision methods typically addressed this problem using a two stage approach. For example, [37] used the Pb edge detector [33] to extract edge features and then used a MRF to determine foreground and background. This was followed up [24] who used a richer set of occlusion cues. Other work by [20] introduced the use of explicit high-level cues including semantic knowledge (e.g., sky and ground) and introduced a new dataset GeoContext for this purpose. Note that in this paper we do not use explicit high-level cues although these might be implicitly captured by the deep network. Recently, [45] used multiple features (e.g., HOG) joint with structure random forest (SRF) [10] and geometric grouping cues (for non-local context) to recover the boundaries and foreground background simultaneously. Maire et al. [31, 32] also designed and embed the border ownership representation into inference the segmentation depth ordering.

Occlusion relations can also be addressed using techniques which estimate 3D depth from single images. These methods typically use either MRF (to capture non-local structure) [19, 26, 40], deep learning [11], or combinations of both [25, 27, 46]. These studies do not explicitly attempt to estimate occlusion, but it can be deduced by detecting the depth discontinuities in the estimated depth map. To train these methods, however, requires annotated 3D data which is hard to obtain for outdoor images, such as those in PASCAL VOC. Hence these methods are most suitable for indoor studies, e.g., on the NYU depth dataset [41].

Our method builds on the fully convolutional network literature and, in particular, recent work on edge detection [49] and semantic segmentation [6] which exploit multi-scale and capture local and non-local cues. We also handle network downsampling by combining the “hole” algorithm [6] and deconvolution [29].



Fig. 2. Four ways to represent occlusion relations, see Sect. 3.1. Consider the boundary contour joining triple points 1 and 3 where border ownership changes at the midpoint 2. The background is to the left of the border between points 1 and 2, but it is on the right of the border between 2 and 3. The “left” rule uses the occlusion orientation on the contour, see arrow, to indicate border ownership (i.e. the left side of the arrow specifies the side of foreground). In panels (a) and (b), the triples points 1, 3 and the junction 2 are explicitly represented. Panel (a), see [37], uses binary variables e_{12}, e_{23} to specify border ownership, while panel (b) includes explicit semantic knowledge [20] where regions are represented by their semantic types, e.g., t_1, t_2 . In panels (c) and (d), the representation is pixel-based. $\mathbf{L}_{o,q}, \mathbf{L}_o$ represent occlusions in terms of a boundary indicator and an occlusion orientation variable (the dense arrows) using the “left” rule to indicate border ownership (i.e. which side is foreground). The difference is that in panel (c), see [45], $\mathbf{L}_{o,q}$ quantizes the occlusion orientation to take 8 values. In contrast, in panel (d), \mathbf{L}_o allows it to take continuous values.

3 The DOC Network

This section describes our DOC deep network. Designing this network requires addressing two main issues: (1) specifying a representation for occlusion relations and a loss function, (2) a deep network architecture that captures the local and non-local cues for detecting occlusion. We now address these issues in turn (Fig. 2).

3.1 Occlusion Relations: Representation and Loss Functions

Representing Occlusion Relations. We represent occlusion relations using an edge map to represent the boundaries between objects (and background) and an orientation variable to indicate the depth ordering across the boundary. We first review existing methods for representing occlusion to motivate our choice and clarify our contribution.

Methods for representing occlusion relations can be roughly classified into four types as shown in Fig. 3. The first two types, panels (a) and (b), represent triple points and junctions explicitly (we defined junctions to be places where border ownership changes). The third and fourth types, panels (c) and (d) use a pixel-based representation with a pair of label indicating boundary and occlusion orientation. The representations in panels (a) and (b) were used in [20, 37] respectively. A limitation of computer vision models which uses these types of representations is that performance is sensitive to errors in detecting

triple points and junctions. The representation in panel (c) enables the use of pixel-based methods which are more robust to failures to detect triple points and junctions [45]. But it quantizes the occlusion orientation variable into 8 bins, which can be problematic because two very similar orientations can be treated as being different (if they occur in neighboring bins). Hence we propose the representation in panel (d) where the occlusion orientation variable is continuous. This pixel-based representation is well suited for deep networks using local and non-local cues and regression to estimate the continuous orientation variable.

Loss Functions for Occlusion Relations. Given an image \mathbf{I} we assign a pair of labels, $\mathbf{l} = \{e, \theta\}$, to each pixel. Here $e \in \{1, 0\}$ is a binary indicator variable with $e = 1$ meaning that the pixel is located on a boundary. $\theta \in (-\pi, \pi]$ is an occlusion orientation variable defined at the boundaries, i.e. when $e = 1$, which specifies the tangent of the boundary and whose direction indicates border ownership using the “left” rule, see Figs. 3 (d) and 1 left. If $e = 0$, we set $\theta = \text{nan}$ and do not use these points for the occlusion loss computation.

For training, we denote the set of training data by $\mathcal{S} = \{(\mathbf{I}_i, \mathcal{L}_i)\}_{i=1}^N$, where N is the number of training images, and $\mathcal{L}_i = \{\mathbf{L}_{ei}, \mathbf{L}_{oi}\}$ are the ground truth annotations, where \mathbf{L}_{ei} specifies the boundary and \mathbf{L}_{oi} the occlusion orientation. Our goal is to design a DCNN that can learn a mapping function parameterized by \mathbf{W} , i.e. $f(\mathbf{I}_i : \mathbf{W})$, that can estimate the ground truth \mathcal{L}_i .

To learn the parameters \mathbf{W} , we define a loss function:

$$l_{doc}(\mathcal{S} : \mathbf{W}) = \frac{1}{N} \left(\sum_i l_e(\mathbf{I}_i, \mathbf{L}_{ei} : \mathbf{W}) + \sum_i l_o(\mathbf{I}_i, \mathbf{L}_{oi} : \mathbf{W}) \right) \quad (1)$$

where $l_e(\mathbf{I}, \mathbf{L}_e : \mathbf{W})$ is the loss for the boundaries, and $l_o(\mathbf{I}, \mathbf{L}_o : \mathbf{W})$ is the loss for the occlusion orientations. The boundary loss is the balanced sigmoid cross entropy loss, which is the same as the HED edge detector [49].

The occlusion orientation loss function strongly penalizes wrong directions (i.e. errors in border ownership using the “left” rule) but only weakly penalizes the tangent direction, as illustrated in Fig. 3. Let θ_j and θ_j^* respectively denote the occlusion orientation groundtruth and the estimation. Then the loss is:

$$l_o(\mathbf{I}, \mathbf{L}_o : \mathbf{W}) = - \sum_{j:e_j=1} \log P(\theta_j^* | \theta_j, \mathbf{W})$$

where, $P(\theta_j^* | \theta_j, \mathbf{W}) = \frac{1}{Z} \begin{cases} \mathbf{1} & : |\theta_j - \theta_j^*|_1 \in [0, \delta] \cup [2\pi - \delta, 2\pi + \delta] \\ \text{Sigmoid}(\alpha(f(|\theta_j - \theta_j^*|_1))) & : \text{otherwise} \end{cases}$

$$f(|\theta_j - \theta_j^*|_1) = \begin{cases} \pi/2 - |\theta_j - \theta_j^*|_1 & : |\theta_j - \theta_j^*|_1 \in [0, \pi] \\ |\theta_j - \theta_j^*|_1 - \pi & : |\theta_j - \theta_j^*|_1 \in (\pi, 2\pi] \\ 3\pi/2 - |\theta_j - \theta_j^*|_1 & : |\theta_j - \theta_j^*|_1 \in (2\pi, +\infty) \end{cases} \quad (2)$$

where $|x|_1$ is the absolute value of x . Z is the normalizing constant. This loss function has two hyper parameters α and δ , where α is a scale factor for the sigmoid function, which controls the strength at direction inverting points. δ controls a non-penalizing range when the θ_j^* is close enough to θ_j .

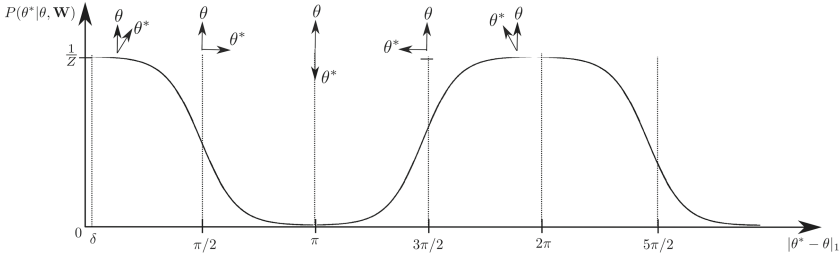


Fig. 3. The orientation probability $P(\theta_j^* | \theta_j, \mathbf{W})$ as a function of the difference between the predicted and ground truth orientation, i.e. θ^* and θ in the figure.

3.2 The Network Architecture

We experimented two DOC architectures, DOC-HED and DOC-DMLFOV, which are based respectively on the holistic-nested edge detector network (HED) [49] and the deeplab multi-scale large field of view DMLFOV network [6]. We choose these networks because: (1) Both exploit local and non-local information and have multi scale outputs (important for occlusion). (2) Both were state-of-the-art on their assigned tasks (and remain highly competitive). HED for detecting edges in the BSDS dataset [1], and DMLFOV for PASCAL semantic segmentation. Also they use different features, for edges or regions, which makes them interesting to compare. Here we refer readers to our supplementary materials or original papers for detailed network architectures.

Two Streams and up Sampling. To adapt HED and DMLFOV to estimate occlusion relations we modify them in two ways: (1) For pixel-based tasks, requiring precise localization of boundaries and estimation of occlusion orientation, we need to up sample the network outputs, to correct for low-resolution caused by max pooling (particularly important for DMLFOV which addressed the less precise task of semantic segmentation). To achieve this we combine the “hole” algorithm [7] with deconvolution up-sampling [29]. (2) To adapt HED and DMLFOV to work on the occlusion representation, see previous section, we adopt a two stream network (encouraged by prior work [47] when using deep networks to address two tasks simultaneously). For estimating the boundaries we keep the original network structure. For estimating the occlusion orientation, which requires a large range of context, we combine outputs only at higher levels of the network (experiments shown that low-level outputs were too noisy to be useful). Thus, for the DOC-HED network, we drop the side output predictions before “conv3” (as in Fig. 4), and for DOC-DMLFOV we drop the predictions (also from side outputs) before “conv3”.

Training Phase. We train DOC-HED and DOC-DMLFOV using the pixel-based representations described in the previous section. They are trained on

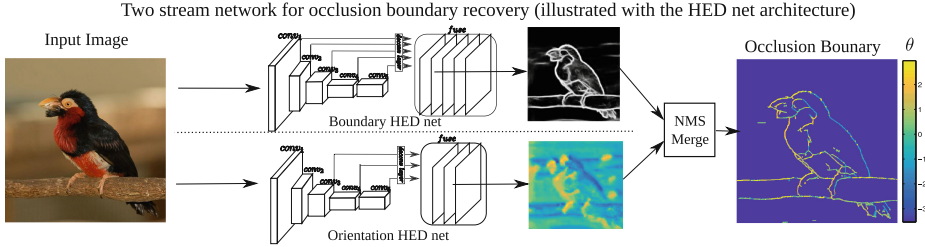


Fig. 4. For inference, we first apply a two stream network (shown for HED) to predict pixel-wise boundaries and the occlusion orientations respectively. Then, we apply non-maximum suppression (NMS) to the boundaries, merge the two predictions, and recover the occlusion boundaries.

both the BSDS border ownership dataset [37] and on a new dataset, based on PASCAL VOC, which we will describe in the next section.

Testing Phase. Given an input image, DOC outputs a boundary map and an occlusion orientation map (from the two streams). To combine the results, we first perform non-maximum suppression (NMS) on the boundary map, using the method as [10]. Then we obtain the occlusion orientation for each edge pixel (i.e. pixel that we have classified as boundary) from the orientation map. Finally we adjust the orientation estimation to ensure that neighboring pixels on the curve have similar orientations. More specifically, we align the orientation to the tangent line estimated from the boundary map since we trust the accuracy of the predicted boundaries. Formally, at a pixel j , the predicted orientation and one direction of the tangent line are θ_j and θ_{tj} respectively. We set θ_j to be θ_{tj} if $|\theta_j - \theta_{tj}| \bmod 2\pi \in [0, \pi/2) \cup (3\pi/2, 2\pi]$, and to the reverse direction of θ_{tj} otherwise. Finally, motivated by the observation that the results are more reliable if the boundary and orientation predictions are consistent, we take $c_{oj} = |\cos(|\theta_j - \theta_{tj}|)|_1$ as the confidence score for the occlusion orientation prediction at pixel j . Finally, given the predicted confidence score c_{ej} from the boundary network outputs, our final confidence score for the occlusion boundary at pixel j is defined to be $c_{ej} + c_{oj}$.

4 Pascal Instance Occlusion Dataset (PIOD)

A large dataset is of critical for training and evaluating deep network models. The BSDS border ownership dataset [37] helped pioneer the study of occlusion relations on natural images but is limited because it only contains 200 images, and hence it may not be able to capture the range of occlusion relations that happen in natural images (our experiments will address how well models trained on one dataset transfer to another).

We choose to annotate occlusion on the PASCAL VOC dataset because it contains well-selected images, and other researchers have already annotated the

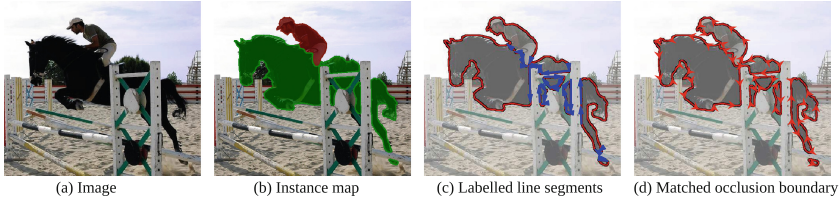


Fig. 5. The annotation process of our PIOD. Given an image, we provide two annotated maps, i.e. (b) the semantic instance map and (c) the generated boundary map. An annotator needs to supplement the boundary map with directed line segments following the “left” rule. We assume the objects occlude background by default, so the annotator only needs to label the boundaries violating this rule or between adjacent instances. Finally, we match the labelled line segments to all the boundaries as shown in (d).

boundaries for 20 object instances [7, 16]. These object boundary annotations are very reliable because the annotators were given clear instructions and consistency checks were performed. Hence our annotation task reduces to annotating border ownership by specifying the directions of the occlusion orientation. Our strategy is to annotate the directions of line segments to specify occlusion orientations, or boundary ownership, using the “left” rule. We do this by a two stage process, as shown in Fig. 5. The annotators are asked to label directed straight line segments which lie close to the object boundaries and whose directions specify the border ownership. The second stage is performed by an algorithm which matches the directed line segments to the annotated boundaries. The idea is that the first stage can be done quickly, since the line segments do not have to lie precisely on the edges, while the second stage gives an automated way to exploit the existing boundary annotations [7, 16].

Stage 1: Annotate with Directed Line Segments. For each image, the annotator is given two annotation maps: (i) the boundary map, and (ii) the semantic instance map [7, 16]. We assume the object is occluding the background, so we only annotate the boundaries between any two adjacent object instances and the boundaries where objects are occluded by background. For each boundary segment, the annotator draw a directed line segment close to the boundary whose direction indicates the occlusion orientation based on the “left” rule.

Stage 2: Matching Directed Line Segments to Object Boundaries. To associate the directed line segments to the boundary map, we developed a matching tool which maps the annotated line segments to the boundaries of all object instances. Our ground truth occlusion boundaries are then represented by a set of boundary fragments, similar to [20]. Each fragment is associated with a start and end point of a directed line segment. Finally, we convert this representation to an occlusion orientation map where each pixel on the object boundary is assigned an occlusion orientation value indicating the local occlusion direction. This process is shown in Fig. 5, where we give images with our labelled results overlaid.

Finally, we produce a frequency statistics of the object occlusion relationships and visualize it as a matrix, which we show in the supplementary materials due to space limit. It helps us to observe object interactions in PIOD.

5 Experiments

We experimented with our DOC approach on the BSDS ownership dataset [37] and our new PASCAL instance occlusion dataset (PIOD). As mentioned before, these datasets differ by size (PIOD is two orders of magnitude bigger) and boundary annotations (PIOD contains only the boundaries of the 20 PASCAL objects while BSDS includes internal and background edges).

In this section, we first propose a more reliable criteria for occlusion boundary evaluation than that used by [37, 45], which was also questioned by previous work [24] (see Sect. 5.1). Then, we conduct extensive experiments with the DOC networks as described in Sect. 3.2. These show that DOC significantly outperforms the state-of-the-art [45]. Both DOC-HED and DOC-DMLFOV perform well, so we perform experiments on both PIOD and on the BSDS ownership data to gain insights about the network architectures for future research. We also study transfer between the two datasets, and other issues.

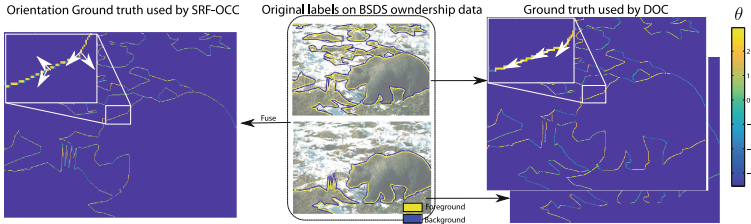


Fig. 6. Center: the two ground truth maps for each image in the BSDS ownership data. Left: limitation of the orientation map generated by SRF-OCC [45] for occlusion evaluation. In the white rectangle, the white arrows show the quantized ground truth orientation at corresponding pixels, which is not smooth or intuitively correct. Also, at bottom right, label inconsistent ground truth edges are discarded when fusing multiple maps. Right: our proposed multiple ground truth occlusion orientation maps for learning and evaluation.

Implementation Details. For the orientation loss function in Eq. (2), we set $\alpha = 4$ and $\delta = 0.05$ respectively, chosen using the validation set. For learning both networks, DOC-HED and DOC-DMLFOV, we used the deep supervision strategy [49], with the learning rate and stage-wise training the same as for HED and DMLFOV respectively. We initialized the models using versions of HED and DMLFOV released by the authors.

For learning on the BSDS ownership dataset, we followed the HED strategy and use adaptive input size for training and testing by setting the “batchsize” to 1 and “itersize” to 10. When learning on PIOD, since the number of images is very large, to save training time, we resize all the input images to 386×386 by keeping the aspect ratio and padding with zeros. We set the “batchsize” to 15 and “itersize” to 2. For both datasets, we augment each image as proposed by HED. We implement all our models based on the published parsenet [28] fork of Caffe [22], which includes both the “hole” algorithm and deconvolution. We also merge the implemented input and cross entropy loss layers from the code released by HED.

5.1 Evaluation Criteria

Specifying a criterion for evaluating occlusion relations is not easy. The problem is that it involves two tasks: detecting boundaries and specifying border ownership. One proposed criteria [37] computes the percentage of the pixels for which the occlusion relations are estimated correctly. But this criteria was criticized [24] because it depends on the selected pixel matching method (between the estimates and the groundtruth boundaries) and the choice of threshold for the edge detector. e.g., a high threshold for the edge detector will detect fewer boundaries but may label their border ownership more accurately. Another criteria was proposed by [45], who released evaluation code. But, see Fig. 6, we found two problems that may lead to unreliable results. The first is that they quantize the occlusion orientation angle to take 8 values which can lead to errors, see the white rectangle on the left of Fig. 6. This quantization problem is enhanced because the orientation was computed based on a local pixel-wise gradient (relying on a pair of neighbored pixels with 8 connections). The second problem is they evaluate on the BSDS ownership dataset which combines boundary maps from different annotators but without checking for consistency [21], which may bias the evaluation since the error cases due to label inconsistency are dropped.

To address these two problems, we first propose to compute the orientation based on a local boundary fragment of length 10 pixels, as used by [20], yielding a smoother and intuitively more reasonable ground truth orientation for evaluation, see right of Fig. 6. Secondly, for evaluating the occlusion relations, we propose a new criteria called the *Occlusion accuracy w.r.t. boundary recall Curve*, which we refer to as the *AOR curve*. This adapts edge detection and occlusion, which was similar in spirit with the PRC curve [36] for depth ordering.

Formally, given the occlusion boundary estimation result with threshold t , we find the correctly detected boundary pixels and their corresponding ground truth pixels by matching them to a ground truth map by the standard edge correspondence method [1]¹. Then for each pixel i on the estimated boundaries, its predicted occlusion orientation θ_i^* is compared to the corresponding ground truth orientation θ_i . We keep the match if $|\theta_i - \theta_i^*| \in [0, \pi/2) \cup (3\pi/2, 2\pi]$, but

¹ We use the toolbox from the BSDS benchmark website.

drop it as a false positive otherwise. After matching all the pixels we obtain two values: (i) the recall rate $R_e(t)$ of the ground truth boundary, and (ii) the accuracy $A_o(t)$ of occlusion orientation prediction given the recalled boundaries.

By varying the threshold t , we can summarize the relationship between $R_e(t)$ and $A_o(t)$ by a curve comparing the accuracy of border ownership as a function of the amount of boundary recalled (i.e. each point on the curve corresponds to a value of the threshold t). In our experiments, we draw the curves to uniformly sample 33 thresholds. For the AOR curve, the accuracy at high recall is most important since more test data used for evaluation yields more reliable indication for the model’s ability. We will release our developed evaluation code and ground truth for reproducing all our results.

5.2 Performance Comparisons

We extensively compare our deep occlusion (DOC) approaches with different settings and configurations of the HED [49] and DMLFOV [6] networks. We also compare DOC-HED and DOC-DMLFOV to the state-of-the-art occlusion recovering algorithm [45] which we refer to as SRF-OCC (it uses structured random forests). In Fig. 7, we see almost all our models outperform SRF over both datasets over 6%, showing the effectiveness of our approach.

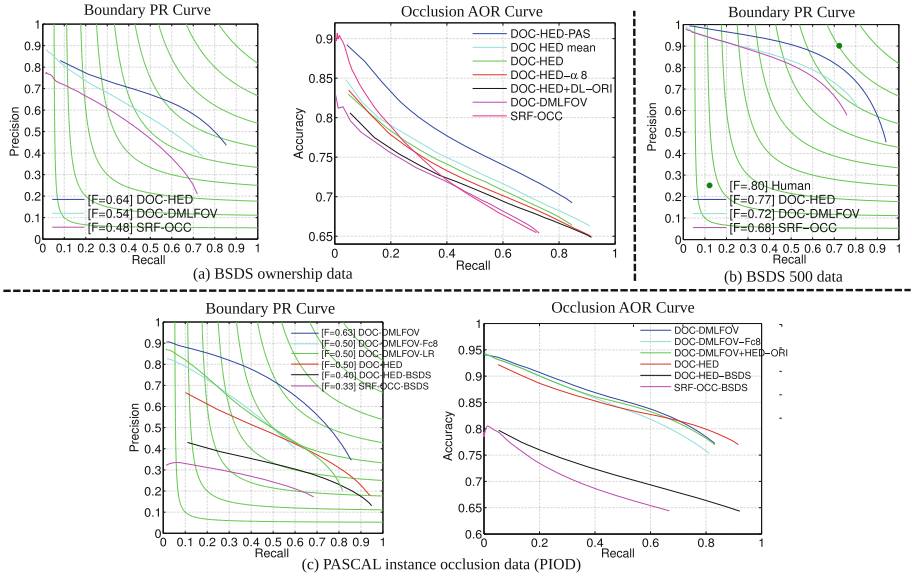


Fig. 7. Quantitative comparison on BSDS ownership data. SRF-OCC [45] is the baseline model. In (b), we show the edge detection performance on BSDS 500 testing data with models trained from the BSDS ownership data (with only 100 images). This shows the DOC-HED model we trained are comparable to those in the HED paper (best viewed in color). Details are in Sect. 5.2.

BSDS Ownership Data. The BSDS ownership dataset contains 100 training images and 100 testing images. We evaluate our deep networks on this dataset although its small size makes them challenging to train. The edge detection comparisons, see left of Fig. 7(a), show that DOC-HED performs best, DOC-DMLFOV is the runner up and SRF-OCC performs less well.

Observe that the results for DOC-HED are not as good as that reported for HED by [49] when trained and tested on the full BSDS dataset. So we evaluated our trained DOC-HED model over the standard BSDS 200 test images and give the results in Fig. 7(c), showing performance very similar to HED (Fusion-output). We think the difference is due to three reasons. Firstly, in order to give fair comparisons to SRF-OCC we train on 100 image only (unlike 300 for HED). Secondly, the images in BSDS ownership data are a non-randomly selected subset of the full BSDS dataset, where the images were chosen to study occlusion and edges inside are harder to detect. Thirdly, each image in this data only uses two ground truth annotations which might introduce labeling noise [21].

On the right of Fig. 7(a), we give results for occlusion relations using our AOR curve. Trained on just 100 images, and tested with single scale image input, the DOC-HED network (green line) performs best, outperforming SRF-OCC when the edge recall rate is higher than 0.3, and the margin goes above 4% at high recall rate of 0.7. The relatively weak performance of the DOC-DMLFOV network (pink line) is probably because it is a more complex network than HED and does not have enough data in BSDS ownership to train it properly. Its performance is lower than DOC-HED network, but is still competitive with SRF-OCC for recall above 0.7. Finally, we investigate transfer by pre-training DOC-HED-PAS (blue line) on PIOD and then fine-tuning it on BSDS ownership data. This improves performance by another 3%, yielding an average improvement of 6% over the SRF-OCC model on the BSDS ownership dataset. This illustrates the advantages of having more data when training deep networks, as well as the ability to transfer models trained on PASCAL to BSDS. Finally, we give visualization results in Fig. 8(a), illustrating that our DOC model recovers better semantic boundaries.

PASCAL Instance Occlusion Dataset (PIOD). PIOD contains 10,100 images, and we take 925 images from the VOC 2012 validation set for testing. We show performance for semantic edge detection at the left of Fig. 7(c). Note there is a difference with BSDS which includes many low-level edges, while PIOD contains only object boundaries. The figure shows that DOC DMLFOV provides the best performance, presumably because it captures strong long-range context, while DOC-HED performs comparatively weaker in this case. In addition, we study transfer from BSDS ownership to PIOD and show that DOC-HED-BSDS (i.e. trained on BSDS) outperforms SRF-OCC-BSDS, but both perform much worse than the deep networks trained on PIOD.

For estimating occlusion relations, see right of Fig. 7(c), DOC-DMLFOV performs best, but only a little better than DOC-HED (i.e. by around 1.5%) and worse than DOC-HED for recall higher than 0.78. This is because, for the

boundaries which are correctly estimated, DOC-HED also gives accurate occlusion orientation estimates.

We also evaluated the ability of SRF-OCC and DOC-HED models when trained only on BSDS. As shown in the figure, DOC-HED-BSDS outperforms SRF-OCC-BSDS significantly on PIOD by a margin of 5% and is higher at every level of recall, showing better ability of deep networks despite the small amount of training data. Some examples visualizing our results are shown in Fig. 8(b). Notice that many of the false positives in the DOC predictions are intuitively correct but were not labelled. The deep networks trained on PIOD data do much better than those trained on BSDS.

Additional Comparisons on the Two Datasets.

Tuning of α . Recall that α is the parameter controlling the sharpness of the occlusion orientation term in Eq. (2). As shown in the right of Fig. 7(a) (DOC-HED- α 8), if we set α to 8 then performance drops slightly because it only weakly penalizes the closeness between θ and θ^* , We found the optimal value to be 4, and fixed this in the experiments.

Scales of input images. On the right of Fig. 7(a) (DOC-HED mean), we show the results from averaging three images scale ([0.5, 1.0, 1.5]) outputs from the DOC-HED network. But multi-scale only gave marginal improvement. This suggests that for boundary detection, multi-scale networks and multi-scale input contain similar information.

Multi-scales network vs. Single scale network. We compared the final fusion output (DOC DMLFOV) vs. single side output (from the “fc8” layer) based on the DOC DMLFOV network over PIOD. As shown on the left of Fig. 7(c) (DOC DMLFOV-Fc8), single side output gives much weaker performance for boundary detection since it localizes the edges worse compared to multi-scale. On the right of Fig. 7(c), DOC DMLFOV-Fc8 performs well but is still weaker than DOC DMLFOV for occlusion recovery. This shows, as expected, that high level features contribute most to the occlusion orientation estimation.

High resolution vs. Low resolution loss. Unlike the original loss based on down-sampled ground truth used by DMLFOV for training semantic segmentation [6], our loss is computed using Deconv from the label map at the original image resolution. At left of Fig. 7(c), the low resolution model (DMLFOV-LR) drops both boundary detection and occlusion orientation.

Replacing the boundary detector network stream. As the AOR curve performs a joint evaluation of boundary detection and border ownership, we must see how DOC-HED and DOC DMLFOV perform on each individual task. We already compared them for boundary detection, so we now switch the occlusion network.

In Fig. 7(a) (DOC-HED+DL-ORI), we use DOC-HED for boundary detection but DOC-DMLFOV for the occlusion orientation. This gives a performance drop of 2% compared to DOC-HED. This shows, for dataset with internal edges like BSDS, DOC-HED also outperforms DOC-DMLFOV on occlusion prediction. In Fig. 7(c) (DOC-DMLFOV+HED-ORI), we apply the same strategy but use DOC-DMLFOV for boundaries and DOC-HED for occlusion orientation,

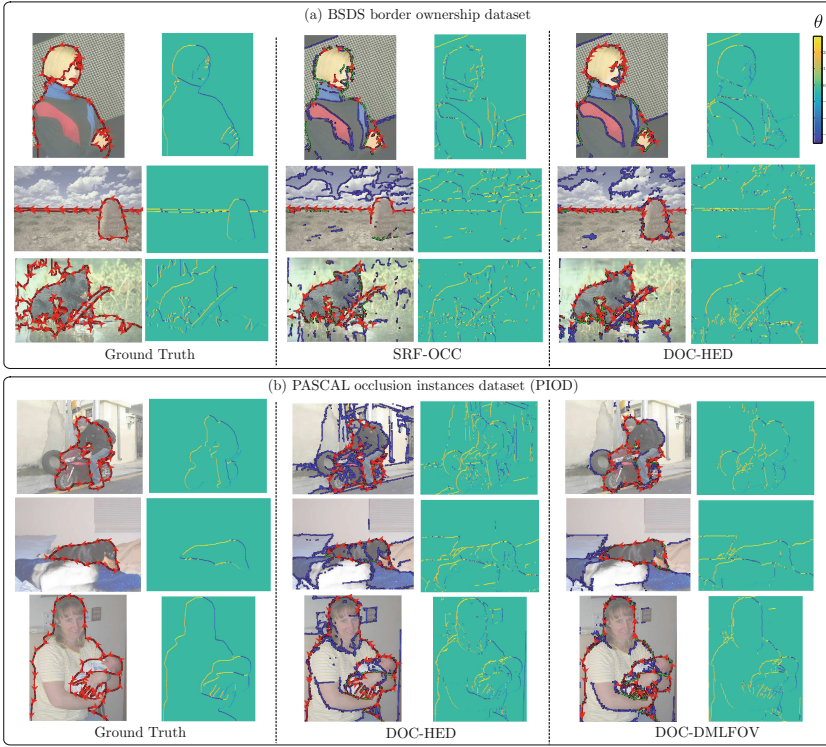


Fig. 8. Qualitative comparisons (best viewed in color). At the left side of each column, we show algorithm results compared with ground truth. The “red” pixels with arrows are correctly labelled occlusion boundaries, the “green” pixels are correctly labeled boundaries but incorrect occlusion, and the “blue” pixels are false positive boundaries. At the right of each column, we show the occlusion boundaries by a 2.1D relief sculpture. In the figure, the foreground regions are raised (embossed). (a) Comparisons on the BSDS ownership data between SRF-OCC [45] and DOC-HED. (b) Comparisons on PIOD between DOC-HED and DOC-DMLFOV. Note for some images, some internal occlusion boundaries are recovered (although they are not labelled correct), e.g., the tire on the bike and the woman’s right arm. This that DOC can generalize from boundaries to some internal edges. We give more examples in Fig. 1 of the supplementary material. (Colour figure online)

giving a result close to that from DOC-DMLFOV. This shows when training on the large dataset PIOD, DOC-HED (smaller network) can perform as well as DOC-DMLFOV for occlusion estimation. These experiments show DOC-HED performs well in general for occlusion estimation.

6 Conclusion and Future Work

In this paper, we designed an end-to-end deep occlusion network (DOC) for estimating occlusion relations. We gave two variants, DOC-HED and

DOC-DMLFOV, and show that they both give big improvements over state-of-the-art methods. We also constructed a new dataset PIOD for studying occlusion relations which is two orders of magnitude larger than comparable datasets. We show that PIOD enables better training and testing of deep networks for estimating occlusion relations. We also show good transfer from PIOD to the smaller BSDS border ownership dataset, but that methods trained on BSDS border ownership are sub-optimal on PIOD. Our results show that DOC-HED and DOC-DMLFOV have complementary strengths which can be combined in future work. We hope that our PIOD dataset will serve as a resource to stimulate research in this important research area.

Acknowledgment. This work is supported by NSF award CCF-1317376. and NSF STC award CCF-1231216. We thank Lingxi Xie, Zhou Ren for paper reading and useful advice.

References

1. Arbelaez, P., Maire, M., Fowlkes, C., Malik, J.: Contour detection and hierarchical image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **33**(5), 898–916 (2011)
2. Ayvaci, A., Raptis, M., Soatto, S.: Sparse occlusion detection with optical flow. *Int. J. Comput. Vis.* **97**(3), 322–338 (2012)
3. Belhumeur, P.N., Mumford, D.: A bayesian treatment of the stereo correspondence problem using half-occluded regions. In: 1992 IEEE Computer Society Conference on Proceedings of Computer Vision and Pattern Recognition, CVPR 1992, pp. 506–512. IEEE (1992)
4. Biederman, I.: On the semantics of a glance at a scene. In: Kubovy, M., Pomerantz, J.R. (Eds.) *Perceptual Organization*, pp. 213–263 (1981)
5. Calderero, F., Caselles, V.: Recovering relative depth from low-level features without explicit t-junction detection and interpretation. *Int. J. Comput. Vis.* **104**(1), 38–68 (2013)
6. Chen, L., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L.: Semantic image segmentation with deep convolutional nets and fully connected CRFS. In: CoRR abs/1412.7062 (2014)
7. Chen, X., Mottaghi, R., Liu, X., Fidler, S., Urtasun, R., Yuille, A.L.: Detect what you can: Detecting and representing objects using holistic models and body parts. In: CVPR, pp. 1979–1986 (2014)
8. Cooper, M.C.: Interpreting line drawings of curved objects with tangential edges and surfaces. *Image Vision Comput.* **15**(4), 263–276 (1997)
9. Craft, E., Schütze, H., Niebur, E., Heydt, R.: A neural model of figure-ground organization. *J. Neurophysiol.* **97**(6), 4310–4326 (2007)
10. Dollár, P., Zitnick, C.L.: Fast edge detection using structured forests. *IEEE Trans. Pattern Anal. Mach. Intell.* **37**(8), 1558–1570 (2015)
11. Eigen, D., Fergus, R.: Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. arXiv preprint [arXiv:1411.4734](https://arxiv.org/abs/1411.4734) (2014)
12. Everingham, M., Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The pascal visual object classes (VOC) Challenge. *IJCV* **88**(2), 303–338 (2010)

13. Fang, F., Boyaci, H., Kersten, D.: Border ownership selectivity in human early visual cortex and its modulation by attention. *J. Neurosci.* **29**(2), 460–465 (2009)
14. Geiger, D., Ladendorff, B., Yuille, A.: Occlusions and binocular stereo. In: Sandini, G. (ed.) *ECCV 1992*. LNCS, vol. 588, pp. 425–433. Springer, Heidelberg (1992)
15. Gibson, J.: The perception of surface layout: a classification of types. Unpublished Purple Perils essay (1968)
16. Hariharan, B., Arbelaez, P., Bourdev, L.D., Maji, S., Malik, J.: Semantic contours from inverse detectors. In: *ICCV*, pp. 991–998 (2011)
17. He, X., Yuille, A.: Occlusion boundary detection using pseudo-depth. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) *ECCV 2010, Part IV*. LNCS, vol. 6314, pp. 539–552. Springer, Heidelberg (2010)
18. Heydt, R., Macuda, T., Qiu, F.T.: Border-ownership-dependent tilt aftereffect. *JOSA A* **22**(10), 2222–2229 (2005)
19. Hoiem, D., Efros, A.A., Hebert, M.: Recovering surface layout from an image. *IJCV* **75**(1), 151–172 (2007)
20. Hoiem, D., Efros, A.A., Hebert, M.: Recovering occlusion boundaries from an image. *IJCV* **91**(3), 328–346 (2011)
21. Hou, X., Yuille, A., Koch, C.: Boundary detection benchmarking: beyond f-measures. In: *CVPR*, pp. 2123–2130 (2013)
22. Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., Guadarrama, S., Darrell, T.: Caffe: convolutional architecture for fast feature embedding. arXiv preprint [arXiv:1408.5093](https://arxiv.org/abs/1408.5093) (2014)
23. Kanizsa, G., Gerbino, W.: Convexity and symmetry in figure-ground organization. *Vision and Artifact*, pp. 25–32 (1976)
24. Leichter, I., Lindenbaum, M.: Boundary ownership by lifting to 2.1d. In: *ICCV*, pp. 9–16 (2009)
25. Li, B., Shen, C., Dai, Y., van den Hengel, A., He, M.: Depth and surface normal estimation from monocular images using regression on deep features and hierarchical crfs. In: *CVPR*, pp. 1119–1127 (2015)
26. Liu, B., Gould, S., Koller, D.: Single image depth estimation from predicted semantic labels. In: *CVPR*, pp. 1253–1260 (2010)
27. Liu, F., Shen, C., Lin, G., Reid, I.D.: Learning depth from single monocular images using deep convolutional neural fields. In: *CoRR abs/1502.07411* (2015)
28. Liu, W., Rabinovich, A., Berg, A.C.: ParseNet: Looking wider to see better. In: *CoRR abs/1506.04579* (2015)
29. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: *CVPR*, pp. 3431–3440 (2015)
30. Mahendran, A., Vedaldi, A.: Understanding deep image representations by inverting them. In: *CVPR*. pp. 5188–5196 (2015)
31. Maire, M.: Simultaneous segmentation and figure/ground organization using angular embedding. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) *ECCV 2010, Part II*. LNCS, vol. 6312, pp. 450–464. Springer, Heidelberg (2010)
32. Maire, M., Narihira, T., Yu, S.X.: Affinity CNN: Learning pixel-centric pairwise relations for figure/ground embedding. In: *Computer Vision and Pattern Recognition (CVPR)* (2016)
33. Martin, D.R., Fowlkes, C., Malik, J.: Learning to detect natural image boundaries using local brightness, color, and texture cues. *IEEE Trans. Pattern Anal. Mach. Intell.* **26**(5), 530–549 (2004)
34. Nitzberg, M., Mumford, D.: The 2.1-d sketch. In: *ICCV*, pp. 138–144 (1990)
35. Palmer, S.E., Ghose, T.: Extremal edge a powerful cue to depth perception and figure-ground organization. *Psychol. Sci.* **19**(1), 77–83 (2008)

36. Visa, G.P., Salembier, P.: Precision-recall-classification evaluation framework: application to depth estimation on single images. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014, Part I. LNCS, vol. 8689, pp. 648–662. Springer, Heidelberg (2014)
37. Ren, X., Fowlkes, C.C., Malik, J.: Figure/ground assignment in natural images. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3952, pp. 614–627. Springer, Heidelberg (2006)
38. Roberts, L.G.: Machine Perception of Three-Dimensional Solids. Outstanding Dissertations in the Computer Sciences. Garland Publishing, New York (1963)
39. Saund, E.: Logic and MRF circuitry for labeling occluding and thinline visual contours. In: NIPS, pp. 1153–1159 (2005)
40. Saxena, A., Sun, M., Ng, A.Y.: Make3d: learning 3d scene structure from a single still image. *IEEE Trans. Pattern Anal. Mach. Intell.* **31**(5), 824–840 (2009)
41. Silberman, N., Hoiem, D., Kohli, P., Fergus, R.: Indoor segmentation and support inference from RGBD images. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012, Part V. LNCS, vol. 7576, pp. 746–760. Springer, Heidelberg (2012)
42. Stein, A.N., Hebert, M.: Occlusion boundaries from motion: low-level detection and mid-level reasoning. *IJCV* **82**(3), 325–357 (2009)
43. Sugihara, K.: An algebraic approach to shape-from-image problems. *Artif. intell.* **23**(1), 59–95 (1984)
44. Sundberg, P., Brox, T., Maire, M., Arbelaez, P., Malik, J.: Occlusion boundary detection and figure/ground assignment from optical flow. In: CVPR, pp. 2233–2240 (2011)
45. Teo, C.L., Fermüller, C., Aloimonos, Y.: Fast 2d border ownership assignment. In: CVPR, pp. 5117–5125 (2015)
46. Wang, P., Shen, X., Lin, Z., Cohen, S., Price, B.L., Yuille, A.L.: Towards unified depth and semantic prediction from a single image. In: CVPR, pp. 2800–2809 (2015)
47. Wang, P., Shen, X., Lin, Z.L., Cohen, S., Price, B.L., Yuille, A.L.: Joint object and part segmentation using deep learned potentials. In: CoRR abs/1505.00276 (2015)
48. Weinzaepfel, P., Revaud, J., Harchaoui, Z., Schmid, C.: Learning to detect motion boundaries. In: CVPR, pp. 2578–2586 (2015)
49. Xie, S., Tu, Z.: Holistically-nested edge detection. In: CoRR abs/1504.06375 (2015)
50. Yu, S.X., Lee, T.S., Kanade, T.: A hierarchical Markov random field model for figure-ground segregation. In: Figueiredo, M., Zerubia, J., Jain, A.K. (eds.) EMM-CVPR 2001. LNCS, vol. 2134, pp. 118–133. Springer, Heidelberg (2001)
51. Zeiler, M.D., Fergus, R.: Visualizing and understanding convolutional networks. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014, Part I. LNCS, vol. 8689, pp. 818–833. Springer, Heidelberg (2014)