Occlusion Reasoning for Object Detection under Arbitrary Viewpoint

Edward Hsiao, *Student Member, IEEE,* and Martial Hebert, *Member, IEEE*

Abstract—We present a unified occlusion model for object instance detection under arbitrary viewpoint. Whereas previous approaches primarily modeled local coherency of occlusions or attempted to learn the structure of occlusions from data, we propose to explicitly model occlusions by reasoning about 3D interactions of objects. Our approach accurately represents occlusions under arbitrary viewpoint without requiring additional training data, which can often be difficult to obtain. We validate our model by incorporating occlusion reasoning with the state-of-the-art LINE2D and Gradient Network methods for object instance detection and demonstrate significant improvement in recognizing texture-less objects under severe occlusions.

Index Terms—occlusion reasoning, object detection, arbitrary viewpoint

1 Introduction

Occlusions are common in real world scenes and are a major obstacle to robust object detection. While texture-rich objects can be detected under severe occlusions with distinctive local features, such as SIFT [1], many man-made objects have large uniform regions. These texture-less objects are characterized by their contour structure, which are often ambiguous even without occlusions. Instance detection of texture-less objects compounds this ambiguity by requiring recognition under arbitrary viewpoint with severe occlusions as shown in Fig. 1. While much research has addressed each component separately (texture-less objects [2]–[5], arbitrary viewpoint [6]–[8], occlusions [9]–[11]), addressing them together is extremely challenging. The main contributions of this paper are (i) a concise model of occlusions under arbitrary viewpoint without requiring additional training data and (ii) a method to capture global visibility relationships without combinatorial explosion.

In the past, occlusion reasoning for object detection has been extensively studied [11]–[13]. One common approach is to model occlusions as regions that are inconsistent with object statistics [10], [14], [15] and to enforce local coherency with a Markov Random Field [16] to reduce noise in these classifications. While assuming that any inconsistent region is an occlusion is valid if occlusions happen uniformly over an object, it ignores the fact there is structure to occlusions for many objects. For example, in real world environments, objects are usually occluded by other objects resting on the same surface. Thus it is often more likely for the bottom of an object to be occluded than the top of an object [17].

Recently, researchers have attempted to learn the structure of occlusions from data [9], [18]. With enough data, these methods can learn an accurate model of occlusions. However, obtaining a broad sampling of occluder objects is usually difficult, resulting in biases to the occlusions of a particular dataset. This becomes more problematic when considering object detection under arbitrary view [6], [19], [20]. Learning approaches need to learn a new model for each view of an object and require the segmentation of the occluder for each training image. This is intractable, especially when recent studies [6] have claimed that approximately 2000 views are needed to sample the view space of an object. A key contribution of our approach is to represent occlusions under arbitrary viewpoint without requiring additional annotated training data of occlusion segmentations. We demonstrate that our approach accurately models occlusions by using only information about the distribution of object dimensions in an environment and the size of the object of interest, and that learning occlusions from data does not give better performance.

Fig. 1: Example detections of (*left*) cup and (*right*) pitcher under severe occlusions.

Researchers have shown in the past that incorporating 3D geometric understanding of scenes [21], [22] improves the performance of object detection systems. Following these approaches, we propose to reason about occlusions by explicitly modeling 3D interactions of objects. For a given environment, we compute physical statistics of objects in the scene and represent an occluder

• *The authors are with The Robotics Institute, Carnegie Mellon University, 5000 Forbes Ave., Pittsburgh, PA, 15213.*
E-mail: {ehsiao,hebert}@cs.cmu.edu

as a probabilistic distribution of 3D blocks. The physical statistics need only be computed once for a particular environment and can be used to represent occlusions for many objects in the scene. By reasoning about occlusions in 3D, we effectively provide a unified occlusion model for different viewpoints of an object as well as different objects in the scene.

We incorporate occlusion reasoning with object detection by: (i) a bottom-up stage which hypothesizes the likelihood of occluded regions from the image data, followed by (ii) a top-down stage which uses prior knowledge represented by the occlusion model to score the plausibility of the occluded regions. We combine the output of the two stages into a single measure to score a candidate detection.

The focus of this paper is to demonstrate that a relatively simple model of 3D interaction of objects can be used to represent occlusions effectively for instance detection of texture-less objects under arbitrary view. Recently, there has been significant progress in simple and efficient template matching techniques [6], [23] for instance detection. These approaches work extremely well when objects are largely visible, but degrade rapidly when faced with strong occlusions in heavy background clutter. We incorporate our occlusion reasoning with two state-of-the-art gradient-based template matching methods, LINE2D [6] and Gradient Network (GN) [5], and demonstrate significant improvement in detection performance on the challenging CMU Kitchen Occlusion Dataset (CMU_KO8) [24].

2 Related Work

Occlusion reasoning has been widely used in many areas from object recognition to segmentation and tracking. While the literature is extensive, there has been comparatively little work on modeling occlusions from different viewpoints and using 3D information until recently. In the following, we review current techniques for occlusion reasoning and broadly classify them into four categories. We begin by discussing classical approaches which use object statistics, part-based models and multiple images, and then discuss more recent approaches on incorporating 3D information.

2.1 Inconsistent Object Statistics

Occlusions are commonly modeled as regions which are inconsistent with object statistics. Girshick *et al.* [14] use an occluder part in their grammar model when all parts cannot be placed. Wang *et al.* [10] use the scores of individual HOG filter cells, while Meger *et al.* [15] use depth inconsistency from 3D sensor data to classify occlusions. To reduce noise in occlusion classifications, local coherency of regions is often enforced [16]. Our approach hypothesizes occlusions as regions which are inconsistent with object statistics, and then performs higher level reasoning about the likelihood of the resulting occlusion pattern.

2.2 Multiple Images

When multiple images in a sequence are provided, adjacent frames are often used to disambiguate the object from the occluders. The location of objects is typically passed to subsequent frames to identify potential occlusions. For example, Shu *et al.* [25] pass the information of occluded parts to the following frame to penalize them when detecting. Ess *et al.* [26] keep tracks alive and extrapolate the state of occluded objects using an Extended Kalman Filter. Xing *et al.* [27] using a temporal sliding window to generate a set of reliable tracklets and use them to disambiguate fully occluded objects. Kowdle *et al.* [28] use motion cues and a smooth motion prior in a Markov Random Field framework to segment the scene into depth layers. Our approach differs from these methods in that we directly operate on a single image.

2.3 Part-based Models

While global object templates work well for detecting objects that are unoccluded, they quickly degrade when occlusions are present. A common representation is to separate an object into a set of parts, so that the overall detector is more robust to individual sections being occluded. Tang *et al.* [29] leverage the fact the occlusions often form characteristic patterns and extend the Deformable Parts Model (DPM) [30] for joint person detection. Vedaldi and Zisserman [31] decompose the HOG descriptor into small blocks which can selectively switch between either an object descriptor or an occlusion descriptor. For human pose estimation, Sigal and Black [32] encode occlusion relationships between body parts using hidden binary variables. Shu *et al.* [25] examine the contribution of each part using a linear SVM and adapts the classifier to use unoccluded parts which maximize the probability of detection. Wu and Nevatia [33] assign the responses of multiple part detectors into object hypotheses that maximize the joint likelihood. Given the visibility confidences of the parts on an object, our approach reasons about its likelihood in the real world.

2.4 3D Reasoning

More recently with the advent of Kinect [34] and more affordable 3D sensors, there has been increasing work on introducing 3D information for occlusion reasoning. Having 3D data provides richer information of the world, such as depth discontinuities and object size. Wojek *et al.* [35] combine object and part detectors based on their expected visibility using a 3D scene model. Contemporary with this work, Pepik *et al.* [36] leverage fine-grained 3D annotated urban street scenes to mine distinctive, reoccurring occlusion patterns. Detectors are then trained for each of these patterns. Zia *et al.* [37] model occlusions on a 3D geometric object class model by enumerating a small finite of occlusion patterns. Wang *et al.* [38] build a depth-encoded context model

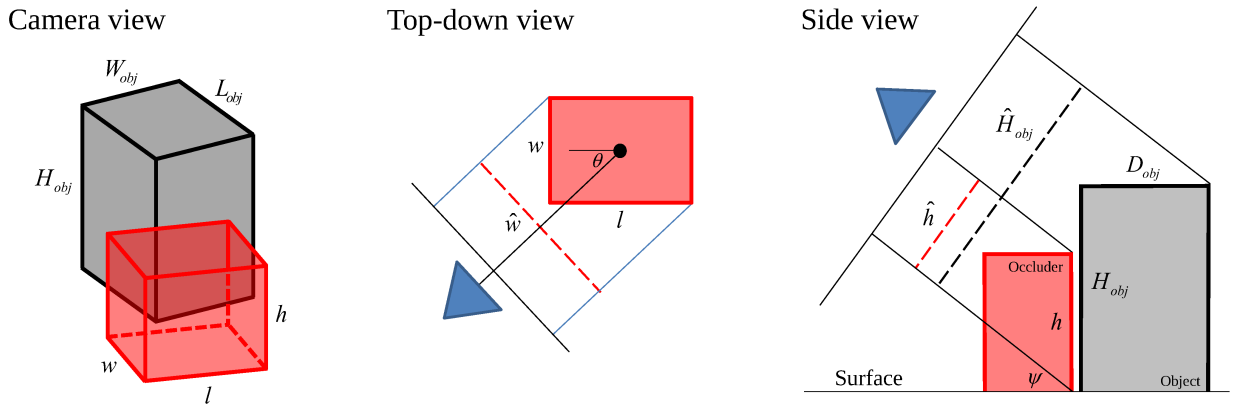


Fig. 2: Occlusion model. (*left*) Example camera view of an object (gray) and occluder (red). (*middle*) Projected width of occluder, $\hat{w}$, for a rotation of θ. (*right*) Projected height of occluder, $\hat{h}$, and projected height of object, $\hat{H}_{obj}$, for an elevation angle of ψ. Notice that $\hat{h}$ is the projection of h onto the image plane since this is the maximum height that can occlude the object, while $\hat{H}_{obj}$ is the apparent height of the object silhouette. An occluder needs a projected height of $\hat{h} \geq \hat{H}_{obj}$ to fully occlude the object.

using RGB-D information and extend Hough voting to include both the object location and its visibility pattern. Our approach uses the key idea of reasoning about occlusions in 3D, but works directly on a 2D image.

3 Occlusion Model

Occlusions in real world scenes are often caused by a solid object resting on the same surface as the object of interest. In our model, we approximate occluding objects by their 3D bounding box and demonstrate how to compute occlusion statistics of an object under different camera viewpoints, c, defined by an elevation angle ψ and azimuth θ.

Let $\mathcal{X} = \{X_1, ..., X_N\}$ be a set of N points on the object with their visibility states represented by a set of binary variables $\mathcal{V} = \{V_1, ..., V_N\}$ such that if $V_i = 1$, then X_i is visible. For occlusions O_c under a particular camera viewpoint c, we want to compute occlusion statistics for each point in $\mathcal{X}$. Unlike other occlusion models which only compute an occlusion prior $P(V_i|O_c)$, we propose to also model the global relationship between visibility states, $P(V_i|\mathcal{V}_{-i}, O_c)$ where $\mathcal{V}_{-i} = \mathcal{V} \backslash V_i$. Through our derivation, we observe that $P(V_i|O_c)$ captures the classic intuition that the bottom of the object is more likely to be occluded than the top. More interesting is $P(V_i|\mathcal{V}_{-i}, O_c)$ which captures the structural layout of an occlusion. The computation of these two occlusion properties both reduce to integral geometry [39] (an entire field dedicated to geometric probability theory).

We make a couple of approximations to tractably derive the occlusion statistics. Specifically, since objects which occlude each other are usually physically close together, we approximate the objects to be on the same support surface and we approximate the perspective effects over the range of object occlusions to be negligible.

3.1 Representation under different viewpoints

The likelihood that a point on an object is occluded depends on the angle the object is being viewed from. Most methods that learn the structure of occlusions from data [9] require a separate occlusion model for each view of every object. These methods do not scale well when considering detection of many objects under arbitrary view.

In the following, we propose a unified representation of occlusions under arbitrary viewpoint of an object. Our method requires only the statistics of object dimensions, which is obtained once for a given environment and can be shared across many objects for that environment.

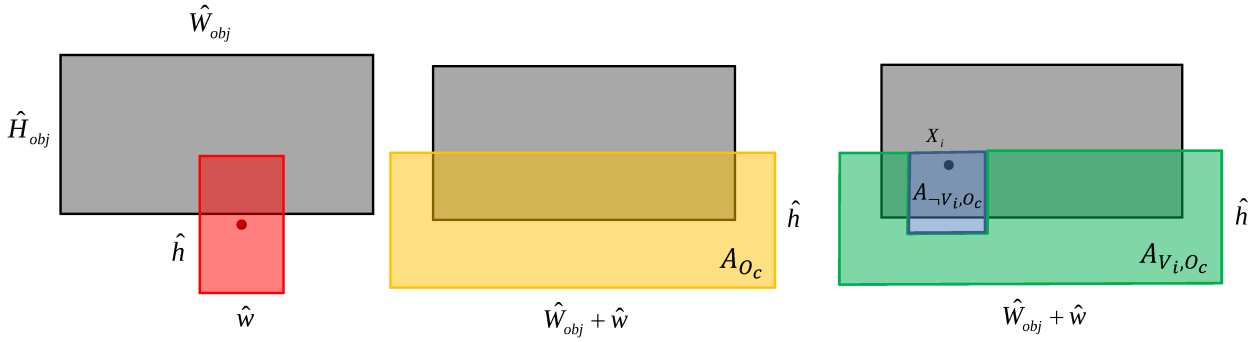
The representation we propose is illustrated in Fig. 2. For a specific viewpoint, we represent the portion of a block that can occlude the object as a bounding box with dimensions corresponding to the projected height $\hat{h}$ and the projected width $\hat{w}$ of the block.

The object of interest, on the other hand, is represented by its silhouette in the image. Initially, we derive our model using the bounding box of the silhouette with dimensions $\hat{H}_{obj}$ and $\hat{W}_{obj}$, and then relax our model to use the actual silhouette (Section 3.4).

First, we compute the projected width $\hat{w}$ of an occluder with width w and length l as shown by the top-down view in Fig. 2. In our convention, $\hat{w} = w$ for an azimuth of $\theta = 0$. Using simple geometry, the projected width is:

$$\hat{w}(\theta) = w \cdot |\cos\theta| + l \cdot |\sin\theta|. \quad (1)$$

Since θ is unknown for an occluding object, we obtain a distribution of $\hat{w}$ assuming all rotations about the vertical axis are equally likely. The distribution of $\hat{w}$ over $\theta \in [0, 2\pi]$ is equivalent to the distribution over any $\frac{\pi}{2}$ interval. Thus, the distribution of $\hat{w}$ is computed by transforming a uniformly distributed random variable on $[0, \frac{\pi}{2}]$ by (1). The resulting probability density of $\hat{w}$ is



(a) (b) (c)

Fig. 3: Computation of the occlusion prior. (*a*) We consider the center positions of a block (red) which occlude the object. The base of the block is always below the object, since we assume they are on the same surface. (*b*) The set of positions is defined by the yellow rectangle which has area A_{O_c}. (*c*) The set of positions which occlude the object while keeping X_i visible is defined by the green region which has area A_{V_i,O_c}.

given by:

$$p_{\hat{w}}(\hat{w}) = \begin{cases} \frac{2}{\pi}\left(1 - \frac{\hat{w}^2}{w^2+l^2}\right)^{-\frac{1}{2}}, & w \leq \hat{w} < l \\ \frac{4}{\pi}\left(1 - \frac{\hat{w}^2}{w^2+l^2}\right)^{-\frac{1}{2}}, & l \leq \hat{w} < \sqrt{w^2+l^2}. \end{cases} \quad (2)$$

The full derivation of this density is provided in Appendix A.

Next, we compute the projected height $\hat{h}$ of an occluder as illustrated by the side view of Fig. 2. We define $\hat{h}$ to be the projection of h on the image plane as this corresponds to the maximum height that can occlude the object given our assumptions. Blocks with different width and length, but the same height will have the same occlusion of the object vertically. Thus, for an elevation angle ψ and occluding block with height h, the projected height $\hat{h}$ is:

$$\hat{h}(\psi) = h \cdot \cos\psi. \quad (3)$$

The projected height of the object, $\hat{H}_{obj}$, is slightly different in that it accounts for the apparent height of the object silhouette. An object is fully occluded vertically only if $\hat{h} \geq \hat{H}_{obj}$. To compute $\hat{H}_{obj}$, we need the distance, D_{obj}, from the closest edge to the farthest edge of the object. Following the computation of the projected width $\hat{w}$, we have $D_{obj}(\theta) = W_{obj} \cdot |\sin\theta| + L_{obj} \cdot |\cos\theta|$. The projected height of the object at an elevation angle ψ is then given by:

$$\hat{H}_{obj}(\theta, \psi) = H_{obj} \cdot |\cos\psi| + D_{obj}(\theta) \cdot |\sin\psi|. \quad (4)$$

Finally, the projected width of the object $\hat{W}_{obj}$ is computed using the aspect ratio of the silhouette bounding box.

3.2 Occlusion Prior

Given the representation derived in Section 3.1, we want to compute a probability for a point on the object being occluded. Many systems which attempt to address occlusions assume that they occur randomly and uniformly across the object. However, recent studies [17] have shown that there is structure to occlusions for many objects.

We begin by deriving the occlusion prior using an occluding block with projected dimensions $(\hat{w}, \hat{h})$ and then extend the formulation to use a probabilistic distribution of occluding blocks of different sizes. The occlusion prior specifies the probability $P(V_i|O_c)$ that a point on the object $X_i = (x_i, y_i)$ is visible given an occlusion of the object. This involves estimating the area, A_{O_c}, covering the set of block positions that occlude the object (shown by the yellow region in Fig. 3b), and estimating the area, A_{V_i,O_c}, covering the set of block positions that occlude the object while keeping X_i visible (shown by the green region in Fig. 3c). The occlusion prior is then just a ratio of these two areas:

$$P(V_i|O_c) = \frac{A_{V_i,O_c}}{A_{O_c}}. \quad (5)$$

From Fig. 3b, a block (red) will occlude the object if its center is inside the yellow region. The area of this region, A_{O_c}, is:

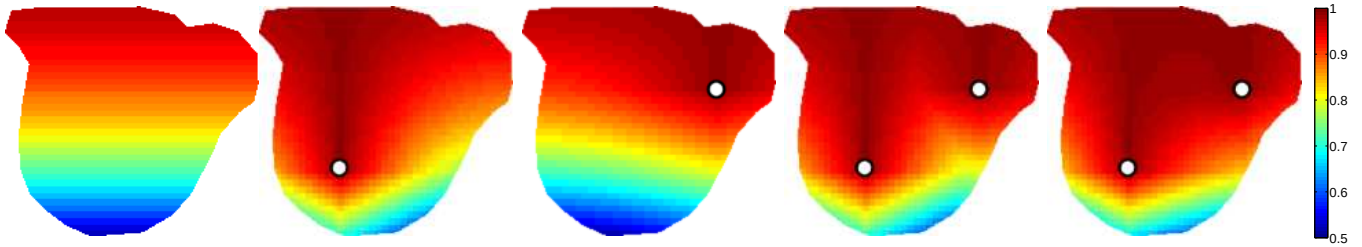
$$A_{O_c} = (\hat{W}_{obj} + \hat{w}) \cdot \hat{h}. \quad (6)$$

Next, from Fig. 3c, this region can be partitioned into a region where the occluding block occludes X_i (blue) and a region which does not (green). A_{V_i,O_c} corresponds to the area of the green region and can be computed as:

$$A_{V_i,O_c} = \hat{W}_{obj} \cdot \hat{h} + \hat{w} \cdot \min(\hat{h}, y_i). \quad (7)$$

The derivation is provided in Appendix B.

Now that we have derived the occlusion prior using a particular occluding block, we extend the formulation to a distribution of blocks of different sizes. Let $p_{\hat{w}}(\hat{w})$ and $p_{\hat{h}}(\hat{h})$ be distributions of $\hat{w}$ and $\hat{h}$ respectively. To simplify notation, we define $\mu_{\hat{w}} = \mathbb{E}_{p_{\hat{w}}(\hat{w})}[\hat{w}]$ and $\mu_{\hat{h}} = \mathbb{E}_{p_{\hat{h}}(\hat{h})}[\hat{h}]$ to be the expected width and height of the occluders under these distributions, and define



(a) (b) (c) (d) (e)

Fig. 4: Example of (*a*) occlusion prior $P(V_i|O_c)$, (*b,c*) conditional likelihood $P(V_i|V_j, O_c)$ and $P(V_i|V_k, O_c)$ given two separate points X_j and X_k individually, (*d*) approximate conditional likelihood $P(V_i|V_j, V_k, O_c)$ from (12), and (*e*) explicit conditional likelihood $P(V_i|V_j, V_k, O_c)$ from (10).

$\beta_y(y_i) = \int \min(\hat{h}, y_i) \cdot p_{\hat{h}}(\hat{h})\ d\hat{h}$. The average areas, A_{O_c} and A_{V_i,O_c}, are then given by:

$$A_{O_c} = (\hat{W}_{obj} + \mu_{\hat{w}}) \cdot \mu_{\hat{h}}, \tag{8}$$

$$A_{V_i,O_c} = \hat{W}_{obj} \cdot \mu_{\hat{h}} + \mu_{\hat{w}} \cdot \beta_y(y_i). \tag{9}$$

This derivation assumes that the distribution $p_{\hat{w},\hat{h}}(\hat{w}, \hat{h})$ can be separated into $p_{\hat{w}}(\hat{w})$ and $p_{\hat{h}}(\hat{h})$. For household objects, we empirically verified that this approximation holds. In practice, the areas are computed by discretizing the distributions and Fig. 4(a) shows an example occlusion prior. Fig. 5 shows how the distribution changes under different camera viewpoints. Our model is able to capture that the top of the object is much less likely to be occluded when viewed from a higher elevation angle than from a lower one. This is because the projected height of occluders is shorter the higher the elevation angle.

3.3 Occlusion Conditional Likelihood

Most occlusion models only account for local coherency and the prior probability that a point on the object is occluded. Ideally, we want to compute a global relationship between all visibility states $\mathcal{V}$ on the object. While this is usually infeasible combinatorially, we show how a tractable approximation can be derived in the following section.

Let $\mathcal{X}_{\mathcal{V}_{\text{-}i}}$ be the visible subset of $\mathcal{X}$ according to $\mathcal{V}_{\text{-}i}$. We want to compute the probability $P(V_i|\mathcal{V}_{\text{-}i}, O_c)$ that a point X_i is visible given the visibility of $\mathcal{X}_{\mathcal{V}_{\text{-}i}}$. Following Section 3.2, the conditional likelihood is given by:

$$P(V_i|\mathcal{V}_{\text{-}i}, O_c) = \frac{A_{V_i,\mathcal{V}_{\text{-}i},O_c}}{A_{\mathcal{V}_{\text{-}i},O_c}}. \tag{10}$$

This computation involves estimating the areas, $A_{\mathcal{V}_{\text{-}i},O_c}$ covering the set of block positions that occlude the object while keeping $\mathcal{X}_{\mathcal{V}_{\text{-}i}}$ visible, and $A_{V_i,\mathcal{V}_{\text{-}i},O_c}$ covering the set of block positions that occlude the object while keeping both X_i and $\mathcal{X}_{\mathcal{V}_{\text{-}i}}$ visible.

We first consider the case where we condition on one visible point, X_j (i.e., $\mathcal{X}_{\mathcal{V}_{\text{-}i}} = \{X_j\}$). To compute $P(V_i|V_j, O_c)$, we already have A_{V_j,O_c} from (9), so we just need A_{V_i,V_j,O_c}. The computation follows from Section 3.2, so we omit the details and just provide the results below. The detailed derivation is provided in Appendix C. If we let $\beta_x(x_i, x_j) = \int \min(\hat{w}, |x_i - x_j|) \cdot p_{\hat{w}}(\hat{w})\ d\hat{w}$, then:

$$\begin{aligned} A_{V_i,V_j,O_c} = {} & (\hat{W}_{obj} - |x_i - x_j|) \cdot \mu_{\hat{h}} \\ & + \left(\int_0^{|x_i - x_j|} (|x_i - x_j| - \hat{w}) \cdot p_{\hat{w}}(\hat{w})\ d\hat{w} \right) \cdot \mu_{\hat{h}} \\ & + \beta_x(x_i, x_j) \cdot \beta_y(y_i) + \mu_{\hat{w}} \cdot \beta_y(y_j). \end{aligned} \tag{11}$$

We can generalize the conditional likelihood to k visible points (i.e., $|\mathcal{X}_{\mathcal{V}_{\text{-}i}}| = k$) by counting as above, however, the number of cases increases combinatorially. We make the approximation that the point $X_j \in \mathcal{X}_{\mathcal{V}_{\text{-}i}}$ with the highest conditional likelihood $P(V_i|V_j, O_c)$ provides all the information about the visibility of X_i. This observation assumes that given V_j, the visibility of X_i is independent of the visibility of all other points (i.e., $V_i \perp \{\mathcal{V}_{\text{-}i} \backslash V_j\} | V_j$) and allows us to compute the global visibility relationship $P(V_i|\mathcal{V}_{\text{-}i}, O_c)$ without combinatorial explosion. The approximation of $P(V_i|\mathcal{V}_{\text{-}i}, O_{\text{-}i})$ is then:

$$P(V_i|\mathcal{V}_{\text{-}i}, O_c) \approx P(V_i|V_j^*, O_c), \tag{12}$$

$$V_j^* = \underset{V_j \in \mathcal{V}_{\text{-}i}}{\operatorname{argmax}} P(V_i|V_j, O_c). \tag{13}$$

For example, Fig. 4(d,e) shows the approximate conditional likelihood and the exact one for $|\mathcal{X}_{\mathcal{V}_{\text{-}i}}| = 2$. Fig. 5 shows how the distribution changes under different camera viewpoints.

3.4 Arbitrary object silhouette

The above derivation can easily be relaxed to use the actual object silhouette. The idea is to subtract the area, A_s, covering the set of block positions that occlude the object bounding box but not the silhouette from the areas described in Sections 3.2 and 3.3. Fig. 7 shows two example block positions. An algorithm to compute A_s is provided in Appendix D. The occlusion prior and conditional likelihood are then given by:

$$P(V_i|O_c) = \frac{A_{V_i,O_c} - A_s}{A_{O_c} - A_s}, \tag{14}$$

$$P(V_i|\mathcal{V}_{\text{-}i}, O_c) = \frac{A_{V_i,\mathcal{V}_{\text{-}i},O_c} - A_s}{A_{\mathcal{V}_{\text{-}i},O_c} - A_s}. \tag{15}$$

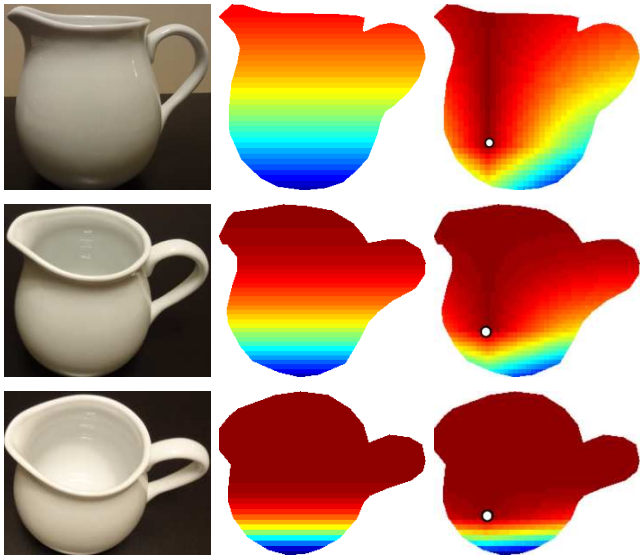
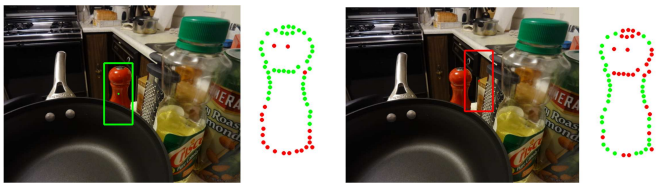


Fig. 5: The occlusion prior and conditional distribution under different camera viewpoints. We show the (*left*) model viewpoint, (*middle*) occlusion prior and (*right*) occlusion conditional likelihood.



(a) True positive (b) False positive

Fig. 6: Examples of occlusion hypotheses. (*a*) For a true detection, the occluded points (red) are consistent with our model. (*b*) For a false positive, the top of the object is hypothesized to be occluded while the bottom is visible, which is highly unlikely according to our model.

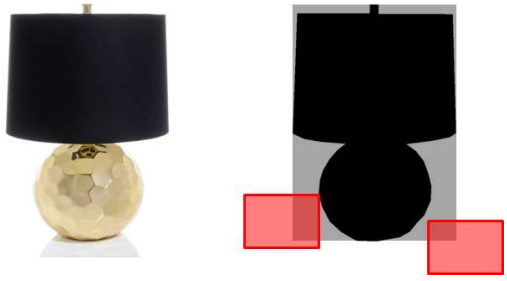


Fig. 7: Using an arbitrary object silhouette. (*left*) Object. (*right*) Two blocks in red which occlude the object bounding box in gray, but not the silhouette in black.

4 Object Detection

Given our occlusion model from Section 3, we augment an object detection system by (i) a bottom-up stage which hypothesizes occluded regions using the object detector, followed by (ii) a top-down stage which measures the consistency of the hypothesized occlusion with our model. We explore using the occlusion prior and occlusion conditional likelihood for scoring and show in our evaluation that both are informative for object detection. Initially, we assume that the hypothesized visibility labeling $\mathcal{V}^{\mathbf{z}}$ is provided for a sliding window location $\mathbf{z}$. We describe in detail in Section 5.3 how to obtain $\mathcal{V}^{\mathbf{z}}$ for different instances of algorithms.

Given $\mathcal{V}^{\mathbf{z}}$, we want a metric of how well the occluded regions agree with our model. Intuitively, we should penalize points that are hypothesized to be occluded by the object detector but are highly likely to be visible according to our occlusion model. From this intuition, we propose the following detection score:

$$\text{score}_f(\mathcal{V}^{\mathbf{z}}) = \frac{1}{N}\sum_{i=1}^{N} V_i^{\mathbf{z}} - f(\mathcal{V}^{\mathbf{z}}), \tag{16}$$

where $f(\mathcal{V})$ is a penalty function for occlusions. A higher score indicates a more confident detection, and for detections with no occlusion, the score is 1. For detections with occlusion, the penalty $f(\mathcal{V})$ is higher the more occluded points which are inconsistent with the model. In the following, we propose two penalty functions, $f_{\text{OPP}}(\mathcal{V})$ and $f_{\text{OCLP}}(\mathcal{V})$, based on the occlusion prior and occlusion conditional likelihood of Section 3.

4.1 Occlusion Prior Penalty

The occlusion prior penalty (OPP) gives high penalty to locations that are hypothesized to be occluded but have a high prior probability $P(V_i^{\mathbf{z}}|O_c)$ of being visible. Intuitively, once the prior probability drops below some level λ, the point should be considered part of a valid occlusion and should not be penalized. This corresponds to a hinge loss function $\Gamma(P,\lambda) = \max\left(\frac{P-\lambda}{1-\lambda}, 0\right)$. The linear penalty we use is then:

$$f_{\text{OPP}}(\mathcal{V}) = \frac{1}{N}\sum_{i=1}^{N}\left[(1-V_i)\cdot\Gamma\left(P(V_i|O_c),\lambda_p\right)\right]. \tag{17}$$

4.2 Occlusion Conditional Likelihood Penalty

The occlusion conditional likelihood penalty (OCLP), on the other hand, gives high penalty to locations that are hypothesized to be occluded but have a high probability $P(V_i|\mathcal{V}_{\text{-}i}, O_c)$ of being visible given the visibility labeling of all other points $\mathcal{V}_{\text{-}i}$. Using the same penalty function formulation as the occlusion prior penalty, we have that:

$$f_{\text{OCLP}}(\mathcal{V}) = \frac{1}{N}\sum_{i=1}^{N}\left[(1-V_i)\cdot\Gamma\left(P(V_i|\mathcal{V}_{\text{-}i}, O_c),\lambda_c\right)\right]. \tag{18}$$

5 Evaluation

In order to evaluate our occlusion model's performance for object instance detection, two sets of experiments were conducted; the first for a single view of an object and the second for multiple views of an object. While in practice, one would only detect objects under multiple

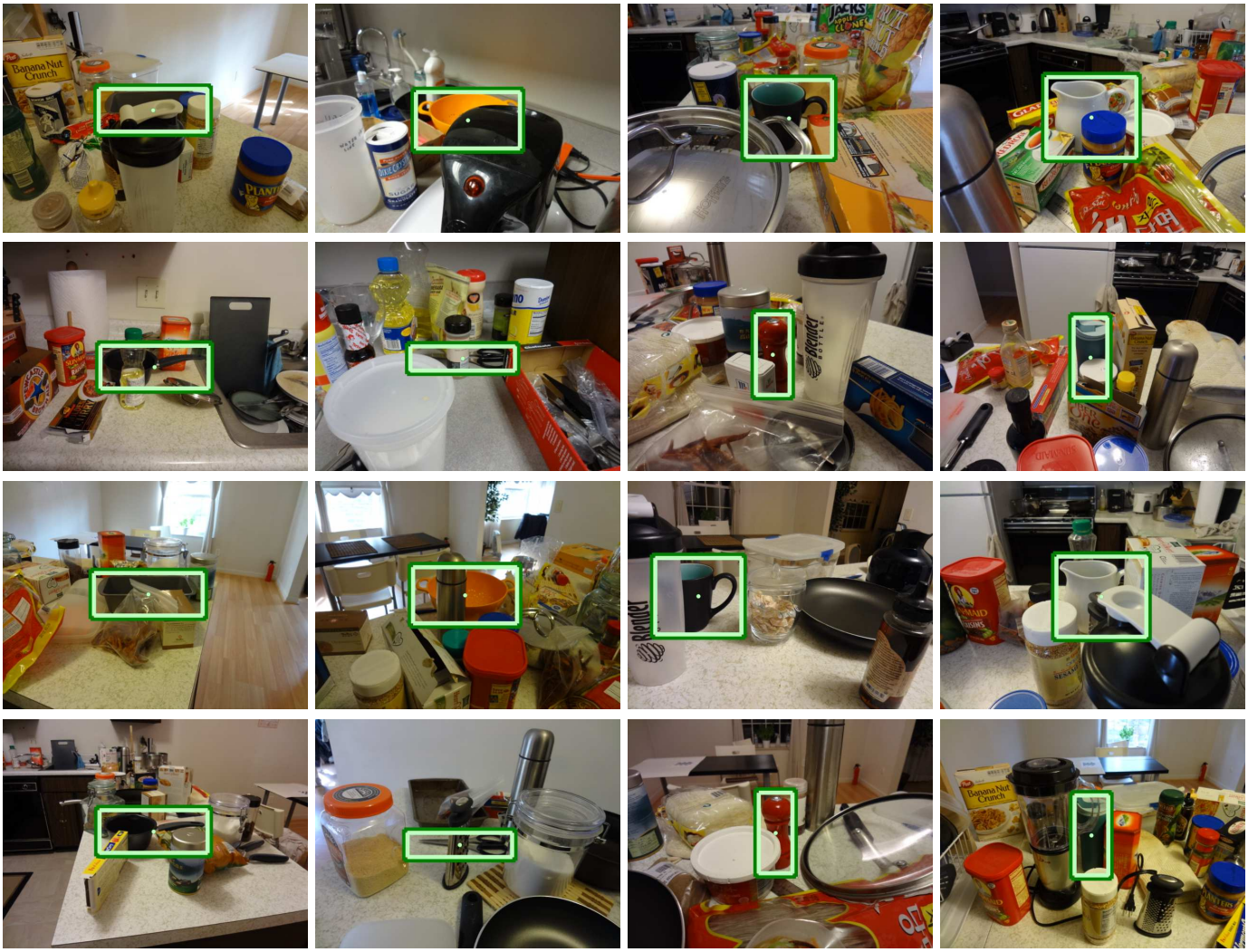


Fig. 8: Example detection results under severe occlusions in cluttered household environments.

views, it is important to tease apart the effect of occlusion from the effect of viewpoint.

In each set of experiments, we explore the benefits of (i) using only the bottom-up stage and (ii) incorporating prior knowledge of occlusions with the top-down stage. When evaluating the bottom-up stage, we hypothesize the occluded region and consider the score of only the visible portions of the detection. This score is equivalent to the first term of (16).

The parameters of our occlusion model were calibrated on images not in the dataset and were kept the same for all objects and all experiments. The occlusion parameters were set to $\lambda_p = 0.5$ and $\lambda_c = 0.95$. We show in Section 5.8 that our model is not sensitive to the exact choice of these parameters.

5.1 CMU Kitchen Occlusion Dataset (CMU_KO8)

Many object recognition algorithms work well in controlled scenes, but fail when faced with real-world conditions exhibiting strong viewpoint and illumination changes, occlusions and clutter. Current datasets for object detection under multiple viewpoints either contain objects on simple backgrounds [40] or have minimal occlusions [6], [8]. For evaluation under a more natural setting, the dataset we collected consists of common household objects in real, cluttered environments under various levels of occlusion. Our dataset contains 1600 images of 8 objects and is split evenly into two parts; 800 for a single view of an object and 800 for multiple views of an object. The single-view part contains ground truth labels of the occlusions and Fig. 9 shows that our dataset contains roughly equal amounts of partial occlusion (1-35%) and heavy occlusions (35-80%) as defined by [17], making this dataset very challenging.

For multiple-view evaluation, we focus our viewpoint variation to primarily the elevation angle as occlusions patterns for different azimuth angles are similar. While it may be harder to recognize certain objects for certain azimuth angles, we are focused only on the relative performance change with using an occlusion model. For our experiments, we use 25 model images for each object which is the same sampling density as [6]. Each model image was collected with a calibration pattern to ground truth the camera viewpoint (ψ, θ) and to rectify the object

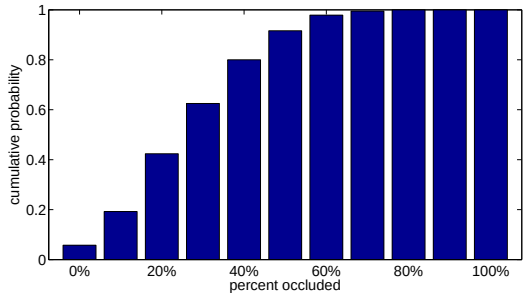


Fig. 9: Dataset occlusion statistics. Our dataset contains roughly equal amount of partial occlusions (1-35%) and heavy occlusions (35-80%).

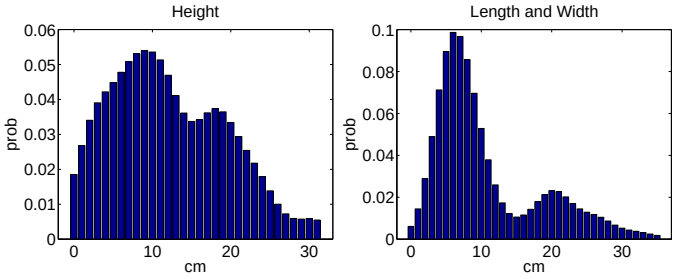


Fig. 10: Distribution of (*left*) heights and (*right*) length and width of occluders in household environments.

silhouette to be upright. The test data was collected by changing the camera viewpoint and the scene around a stationary object. A calibration pattern was used to ground truth the position of the object.

5.2 Validity of Occlusion Model

To derive the occlusion probabilities, we approximated occluder objects to be bounding boxes which are on the same surface as the object. While this approximation is consistent with the occlusion types observed by Dollar *et al.* in [17], we further validate the approximation on our dataset. Given the groundtruth occlusion labels in the dataset, we consider an occluded pixel to be consistent with the approximation if there are no un-occluded object pixels below it. From Fig. 11, for 80% of the images, over 90% of the occluded pixels are consistent.

5.3 Algorithms

We validate our approach by incorporating occlusion reasoning with two state-of-the-art methods for instance detection under arbitrary viewpoint, LINE2D [6] and Gradient Networks (GN) [5]. For fair comparison, we use the same M sampled edge points x_i for all the methods. These points are specified relative to the template center, $\mathbf{z}$. We give a brief description of the algorithms in our comparison below.

5.3.1 LINE2D [6]

The LINE2D method is a current state-of-the-art system for instance detection under arbitrary viewpoint. It represents an object by a template of sampled edge points,

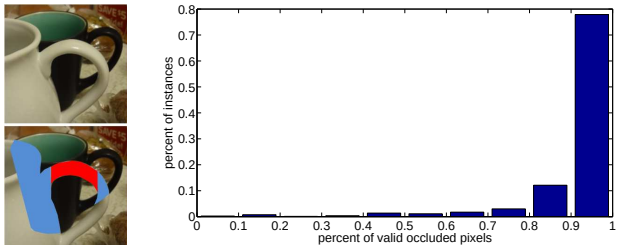


Fig. 11: Validity of occlusion model. (*left*) We show in blue, the occluded pixels which satisfy our approximation, and in red, those that do not. (*right*) For each object instance in the dataset, we evaluate the percentage of occluded pixels which satisfy our approximation that they can be explained by a bounding box with a base lower than the object. For 80% of the images, over 90% of the occluded pixels can be explained with our model.

each with a quantized orientation. For every scanning window location, a similarity score is computed between the gradient of each model point and the image. In [6], the gradient orientations are quantized into 8 orientation bins and the similarity for point x_i is the cosine of the smallest quantized orientation difference, $\Delta\theta_i$, between its orientation and the image orientations in a 7×7 neighborhood of $x_i + \mathbf{z}$. The score of a window is $\sum_{i=1}^{M} \cos(\Delta\theta_i)$. We kept all other parameters for LINE2D the same as [6]. We tested our implementation on a subset of the dataset provided by the authors of [6] and observed negligible difference in performance.

5.3.2 robust-LINE2D (rLINE2D) [24]

Since the LINE2D method returns continuous values, we need to threshold it to obtain the occlusion hypothesis. We consider a point to be occluded if the image gradient and the model gradient have different quantized orientations. Fig. 6 shows example occlusion hypotheses. This produces a binary descriptor for each scanning window location. The score of a window is $\sum_{i=1}^{M} \delta(\Delta\theta_i = 0)$ where $\delta(t) = 1$ if t is true. We refer to this method as robust-LINE2D (rLINE2D). In the evaluation, we show that simply binarizing the descriptor with rLINE2D significantly outperforms LINE2D in cluttered scenes with severe occlusions.

5.3.3 Gradient Networks (GN) [5]

The LINE2D and rLINE2D methods only consider local gradient information when matching the shape. Since these approaches do not account for edge connectivity, this often results in misclassification of occlusion in cluttered regions where the local gradient orientation matches accidentally. To address this issue, our Gradient Networks method captures contour connectivity directly on low-level image gradients. For each image pixel, the algorithm estimates the probability that it matches a template shape. Our results in [5] show significant improvement in shape matching and object detection in natural scenes with severe clutter and occlusions.

TABLE 1: Single view. Average precision.

Object	LINE2D	rLINE2D	rLINE2D+OPP	rLINE2D+OCLP	GN	GN+OPP	GN+OCLP
bakingpan	0.12	0.23	0.27	0.36	0.44	0.51	**0.61**
colander	0.26	0.60	0.66	0.74	0.76	0.75	**0.80**
cup	0.25	0.64	0.65	0.75	0.84	0.84	**0.89**
pitcher	0.14	0.66	0.69	0.76	0.71	0.64	**0.77**
saucepan	0.12	0.51	0.52	0.53	0.85	0.83	**0.85**
scissors	0.09	0.21	0.21	0.26	0.42	0.41	**0.45**
shaker	0.05	0.36	0.48	0.59	0.47	0.52	**0.64**
thermos	0.24	0.56	0.68	0.68	0.73	**0.79**	0.76
Mean	0.16	0.47	0.52	0.58	0.65	0.66	**0.72**

TABLE 2: Multiple view. Average precision.

Object	LINE2D	rLINE2D	rLINE2D+OPP	rLINE2D+OCLP	GN	GN+OPP	GN+OCLP
bakingpan	0.06	0.12	0.12	0.15	0.78	0.61	**0.79**
colander	0.19	0.55	0.58	0.67	0.72	0.71	**0.79**
cup	0.13	0.44	0.46	0.50	0.85	0.86	**0.87**
pitcher	0.08	0.22	0.32	0.34	0.53	0.60	**0.67**
saucepan	0.09	0.40	0.42	0.41	**0.90**	0.88	0.89
scissors	0.05	0.20	0.20	0.26	0.72	0.68	**0.81**
shaker	0.08	0.16	0.25	0.25	0.41	0.50	**0.52**
thermos	0.08	0.30	0.44	0.48	0.66	0.74	**0.81**
Mean	0.09	0.30	0.35	0.38	0.70	0.70	**0.77**

The GN algorithm returns a shape similarity $\Omega^{\mathbb{S}}$ for each pixel given the template $S(\mathbf{z})$. For fair comparison, we apply a 7×7 max spatial filter (i.e., equivalent to LINE2D) to $\Omega^{\mathbb{S}}$ resulting in $\hat{\Omega}^{\mathbb{S}}$. The template score at $\mathbf{z}$ is then $\sum_{i=1}^{M} \hat{\Omega}^{\mathbb{S}}(x_i+\mathbf{z})$. We use the soft shape model with normalized gradient magnitudes for the edge potential, and both color and orientation for the appearance. Since the similarity measure $\Omega^{\mathbb{S}}(x_i + \mathbf{z})$ is calibrated so that it can be interpreted as a probability, we generate the occlusion hypothesis by thresholding this value at 0.5.

Our algorithm takes on average 2 ms per location $\mathbf{z}$ on a 3GHz Core2 Duo CPU. In practice, we run our algorithm only at the hypothesis detections of rLINE2D [24]. The combined computation time is about 1 second per image.

5.4 Distribution of Occluder Sizes

The distribution of object sizes varies in different environments. For a particular scenario, it is natural to only consider objects as occluders if they appear in that environment. The statistics of objects can be obtained from the Internet [41] or, in the household scenario, simply from 100 common household items. Fig. 10 shows the distributions for household objects.

From real world dimensions, we can compute the projected width and height distributions, $p_{\hat{w}}(\hat{w})$ and $p_{\hat{h}}(\hat{h})$, for a given camera viewpoint. The projected width distribution is the same for all viewpoints and is obtained by computing the probability density from (2) for each pair of width and length measurement. These densities are discretized and averaged to give the final distribution of $\hat{w}$.

The projected height distribution, on the other hand, depends on the elevation angle ψ. From (3), $\hat{h}$ is a factor $\cos\psi$ of h. Thus, the projected height distribution, $p_{\hat{h}}(\hat{h})$, is computed by subsampling $p_h(h)$ by $\cos\psi$.

5.5 Single view

We first evaluate the performance for single view object detection. An object is correctly detected if the intersection-over-union (IoU) of the predicted bounding box and the ground truth bounding box is greater than 0.5. Each object is evaluated on all 800 images in this part of the dataset and Fig. 12 shows the precision-recall plot. To summarize the performance, we report the Average Precision in Table 1. A few example detections are shown in Fig. 8.

From the table, the rLINE2D method already significantly outperforms the baseline LINE2D method. One issue with the LINE2D gradient similarity metric (i.e., cosine of the orientation difference) is that it gives high score even to orientations that are very different, resulting in false positives with high scores. The rLINE2D metric of considering only points with the same quantized orientation is more robust to background clutter in the presence of occlusions. However, since rLINE2D only considers information very locally around each edge point, there are still many misclassifications in background clutter. The Gradient Network method performs substantially better than rLINE2D by incorporating edge connectivity directly on low-level gradients. This results in an improvement of 18% in average precision.

When rLINE2D is augmented with occlusion reasoning, there is an absolute improvement of 5% for OPP and 11% for OCLP. For GN, there is a corresponding

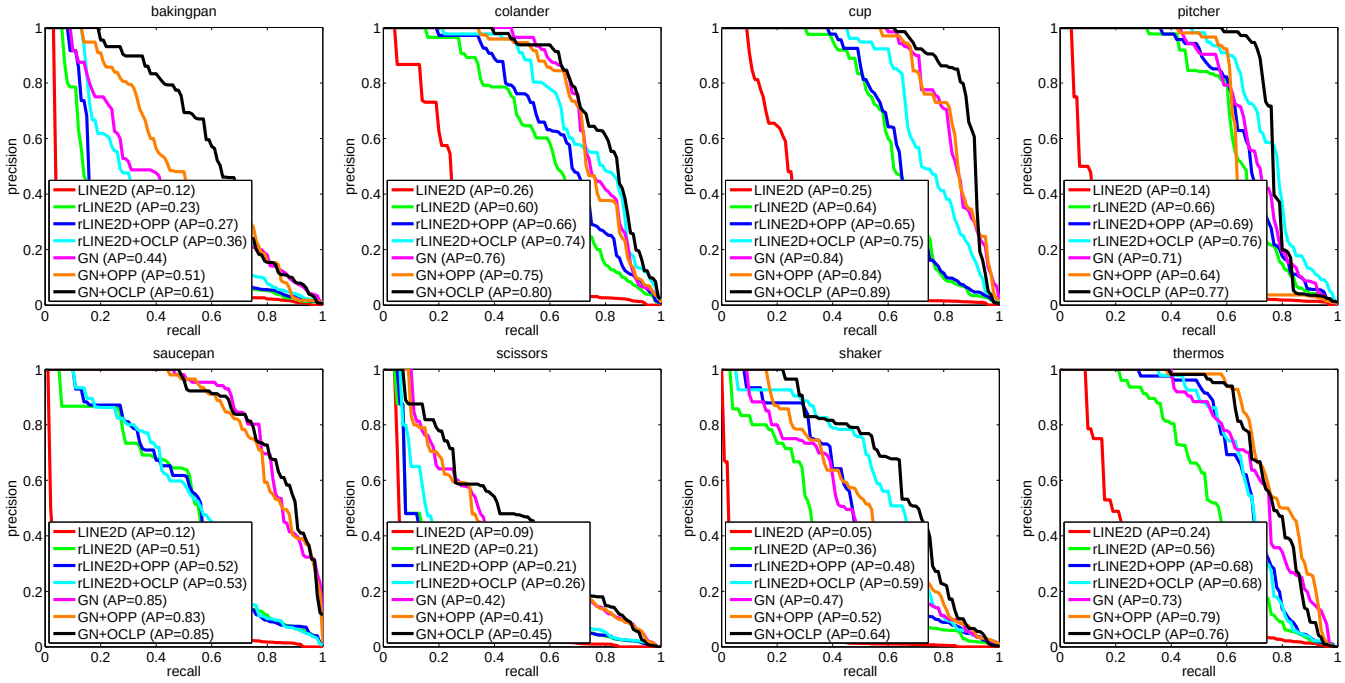


Fig. 12: Precision-recall results for single view. There is significant improvement in performance by using occlusion reasoning.

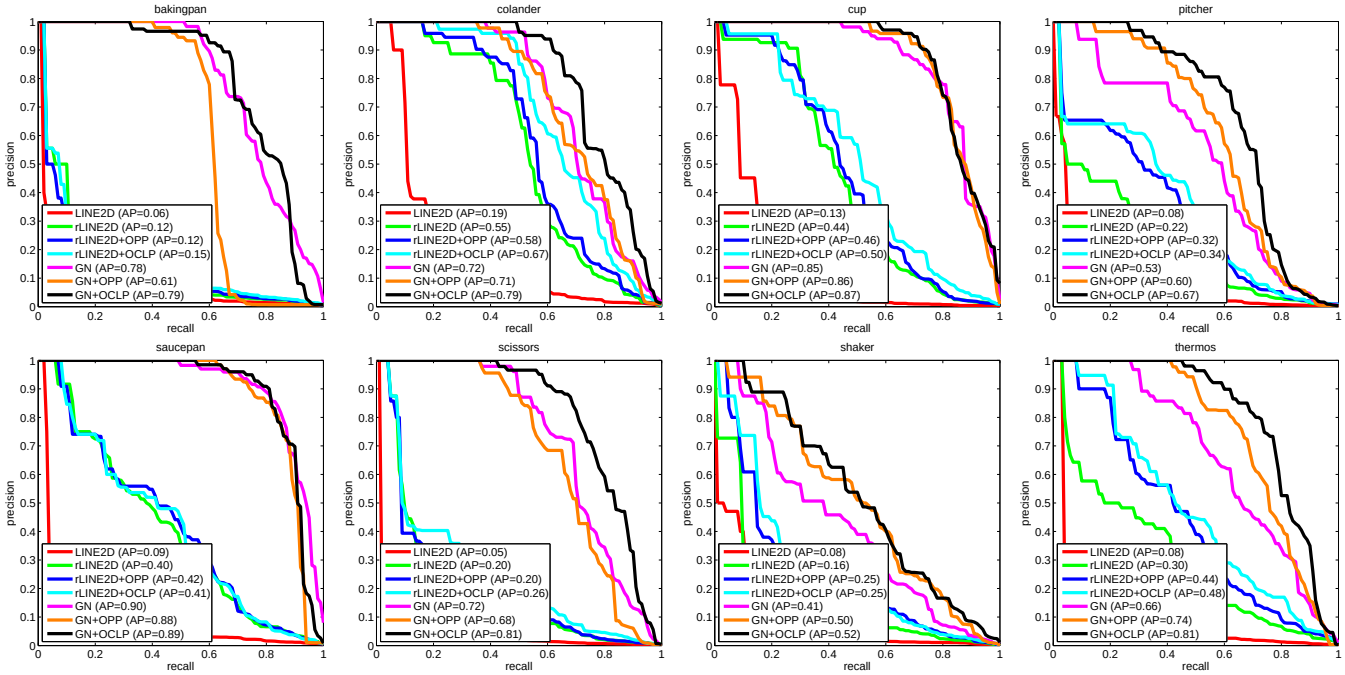


Fig. 13: Precision-recall results for multiple views. The overall performance is lower than the single view experiments due to more false positives, but importantly, we observe similar gains from using our occlusion reasoning

improvement of 1% and 7%. This indicates that both occlusion properties are informative for object detection. The disparity between the gains of OPP and OCLP suggests that accounting for global occlusion layout by OCLP is more informative than considering the *a priori* occlusion probability of each point individually by OPP. In particular, OCLP improves over OPP when one side of the object is completely occluded as shown in Fig. 14. Although the top of the object is validly occluded, OPP assigns a high penalty. By over-penalizing true detections, OPP often makes the performance worse. OCLP, on the other hand, always performs at least on par with the baseline methods and usually performs substantially better.

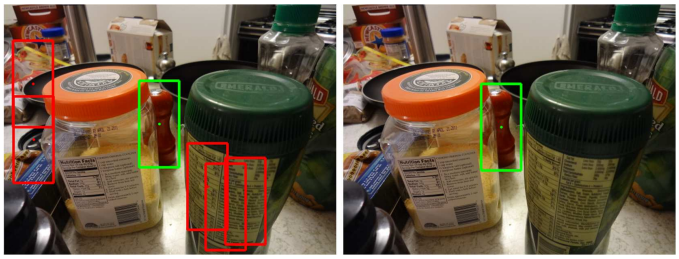


Fig. 14: A typical case where OCLP performs better than OPP. (*left*) For OPP, the false positives in red have higher scores than the true detection in green. The occluded region at the top of the true detection is over-penalized. (*right*) For OCLP, the true detection is the top detection.

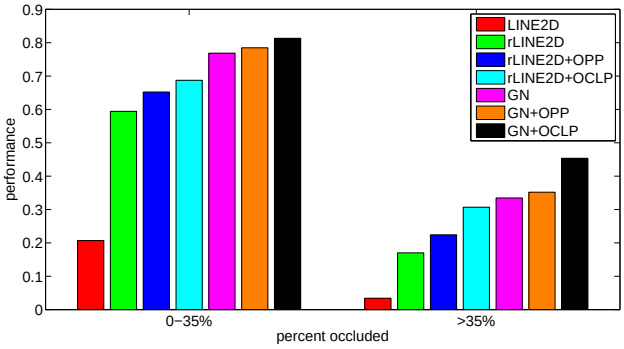


Fig. 15: Performance under different occlusion levels. While our methods improve performance under all levels of occlusions, we see larger gains under heavy occlusions.

Fig. 15 shows the performance under different levels of occlusion. Here, the detection rate is the percentage of top detections which are correct. Our occlusion reasoning improves object detection under both low (0-35%) and high levels (>35%) of occlusions, but provides significantly larger gains for heavy occlusions. For unoccluded objects, we verified that the occlusion model did not degrade performance.

5.6 Multiple views

Next we evaluate the performance for object detection under multiple views. Fig. 13 shows the precision-recall plots and Table 2 reports the Average Precision. Again, we obtain significant improvement gains over the LINE2D system. The performance is lower for rLINE2D due to more false positives from increasing the number of templates, but the relative gains at 5% for OPP and 8% for OCLP are similar to the single view case. GN, on the other hand, is a much more robust template matching technique. With more templates, GN is able to find better aligned viewpoints while keeping a low false positive rate, leading to increased performance over the single view case. We again see similar gains for OCLP at 7%. This demonstrates that our model is effective for representing occlusions under arbitrary view.

Fig. 6 shows a typical false positive that can only be filtered by our occlusion reasoning. Although a majority of the points match well and the missing parts are largely coherent, the detection is not consistent with our occlusion model and is thus penalized and filtered.

Fig. 16: Typical failure cases of OCLP. (*left*) The pitcher is occluded by the handle of the pot which is not accurately modeled by a block. (*right*) The scissor is occluded by a plastic bag resting on top of it. In these cases, OCLP over penalizes the detections.

Fig. 16 shows a couple of failure cases where our assumptions are violated. In the first image, the pot occluding the pitcher is not accurately modeled by its bounding box. In the second image, the occluding object rests on top of the scissor. Even though we do not handle these types of occlusions, our model represents the majority of occlusions and is thus able to increase the overall detection performance.

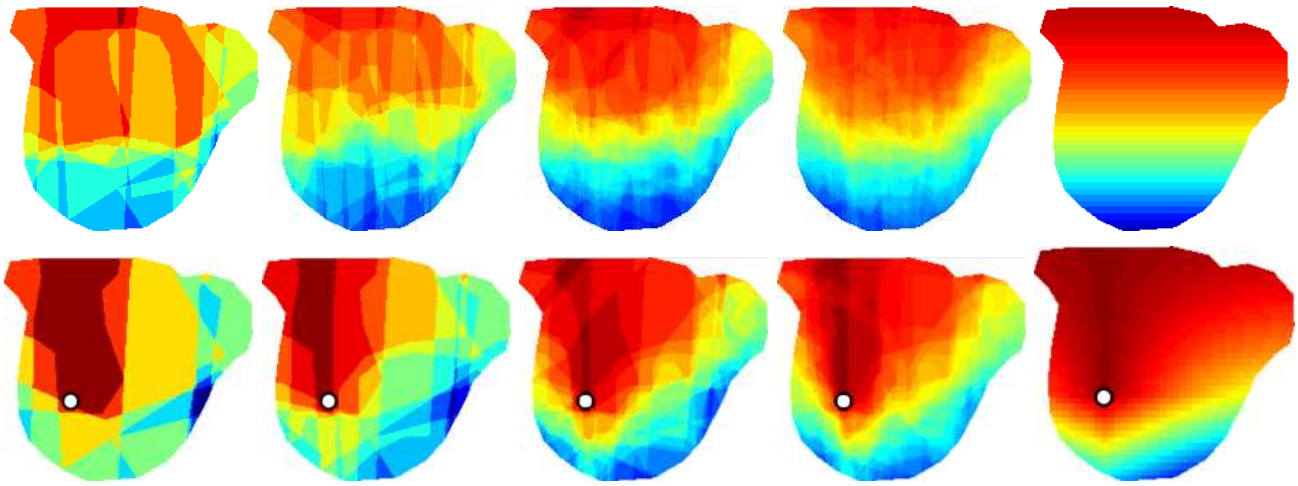
5.7 Learning From Data

To verify that our model accurately represents occlusions in real world scenes, we rerun the above experiments with occlusion priors and conditional likelihoods learned from data. We use the detailed groundtruth occlusion masks in the single view portion of the dataset to obtain the empirical distributions. Fig. 17 compares learning the occlusion prior and occlusion likelihood using 10, 20, 40 and 80 images with our analytical model. The distributions from the empirical and analytical model are very similar.

We use 5-fold cross-validation for quantitative evaluation and Fig. 18 shows the results using different number of images for learning. The learned occlusion properties, lOPP and lOCLP, correspond to their explicit counterparts, OPP and OCLP. The learned occlusion prior, lOPP, performs slightly better than OPP. This is a result of the slightly different distribution seen in Fig. 17 where the sides of the object are more likely to be occluded in the dataset. The learned occlusion conditional likelihood, lOCLP, performs essentially the same as OCLP, but requires 80 images for every view of every object to achieve the same level of performance.

5.8 Parameter Sensitivity

The two parameters of our occlusion reasoning approach are λ_p and λ_c corresponding to the hinge loss parameters for OPP and OCLP. To verify that our approach is not sensitive to the exact choice of these parameters, we



(a) 10 images (b) 20 images (c) 40 images (d) 80 images (e) analytical

Fig. 17: Learning the occlusion distributions from from data. From left to right, columns 1-4 show using 10, 20, 40, and 80 images for learning the occlusion prior (*top*) and the occlusion conditional likelihood (*bottom*). The last column shows the distribution of our analytical model.

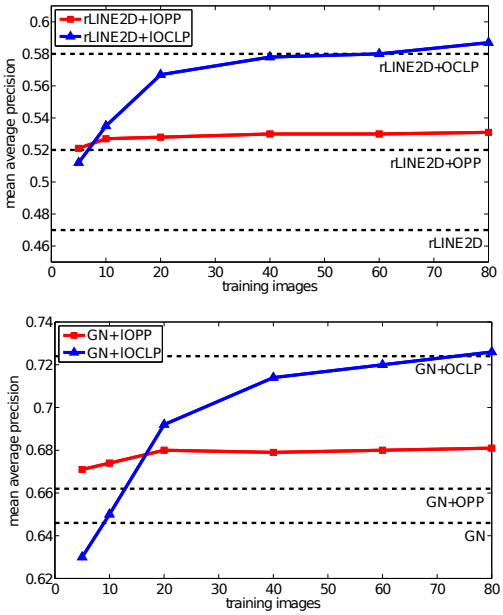


Fig. 18: Learning the occlusion prior and conditional likelihood using groundtruth occlusion segmentations from the single view portion of the dataset. The dotted lines show the performance of our analytic approach, which does not depend on the number of training images. We show the learned occlusion properties, lOPP (red) and lOCLP (blue), corresponding to OPP and OCLP. While lOPP performs slightly better than OPP, it needs about 20 images and is only slightly better. lOCLP needs about 60 to 80 images to achieve the same level of performance as OCLP.

evaluated the performance of the occlusion reasoning for a range of parameter values. Fig. 19 shows the sensitivity of λ_p and λ_c when augmenting both rLINE2D and GN. From the figure, the performance is relatively constant

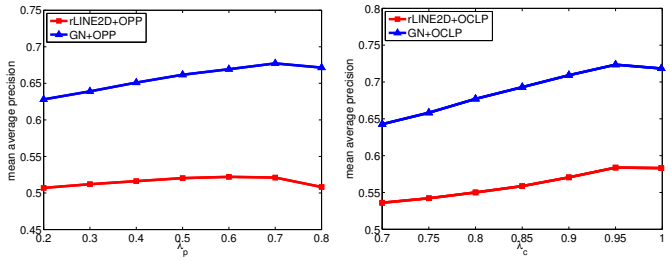


Fig. 19: Parameter sensitivity for (*left*) OPP and (*right*) OCLP. The last data point for OCLP is plotted at $\lambda_c = 0.999$ which penalizes only points that the model believes are definitely occluded (i.e., $P(V_i|\mathcal{V}_{\text{-}i}, O_c) = 1$). The exact choice of the parameter does not affect the performance of the methods significantly.

for a wide range of parameter values and is thus robust to the exact choice.

6 Conclusion

The main contribution of this paper is to demonstrate that a simple model of 3D interaction of objects can be used to represent occlusions effectively for object detection under arbitrary viewpoint without requiring additional training data. We propose a tractable method to capture global visibility relationships and show that it is more informative than the typical *a priori* probability of a point being occluded. Our results on a challenging dataset of texture-less objects under severe occlusions demonstrate that our approach can significantly improve object detection performance.

Acknowledgments

This work was supported in part by the National Science Foundation under ERC Grant No. EEEC-0540865.

REFERENCES

[1] D. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, 2004.

[2] A. Toshev, B. Taskar, and K. Daniilidis, "Object detection via boundary structure segmentation," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2010.

[3] V. Ferrari, L. Fevrier, F. Jurie, and C. Schmid, "Groups of adjacent contour segments for object detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2008.

[4] V. Ferrari, T. Tuytelaars, and L. Van Gool, "Object detection by contour segment networks," in *Proceedings of European Conference on Computer Vision*, 2006.

[5] E. Hsiao and M. Hebert, "Gradient networks: Explicit shape matching without extracting edges," in *Proceedings of AAAI Conference on Artificial Intelligence*, 2013.

[6] S. Hinterstoisser, C. Cagniart, S. Ilic, P. Sturm, N. Navab, P. Fua, and V. Lepetit, "Gradient response maps for real-time detection of texture-less objects," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2011.

[7] E. Hsiao, A. Collet, and M. Hebert, "Making specific features less discriminative to improve point-based 3d object recognition," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2010.

[8] K. Lai, L. Bo, X. Ren, and D. Fox, "A large-scale hierarchical multi-view rgb-d object dataset," in *Proceedings of IEEE International Conference on Robotics and Automation*, 2011.

[9] T. Gao, B. Packer, and D. Koller, "A segmentation-aware object detection model with occlusion handling," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2011.

[10] X. Wang, T. Han, and S. Yan, "An hog-lbp human detector with partial occlusion handling," in *Proceedings of IEEE International Conference on Computer Vision*, 2009.

[11] H. Plantinga and C. Dyer, "Visibility, occlusion, and the aspect graph," *International Journal of Computer Vision*, 1990.

[12] W. Grimson, T. Lozano-Pérez, and D. Huttenlocher, *Object recognition by computer*. MIT Press, 1990.

[13] M. Stevens and J. Beveridge, *Integrating Graphics and Vision for Object Recognition*. Kluwer Academic Publishers, 2000.

[14] R. B. Girshick, P. F. Felzenszwalb, and D. McAllester, "Object detection with grammar models," in *Proceedings of Neural Information Processing Systems*, 2011.

[15] D. Meger, C. Wojek, B. Schiele, and J. J. Little, "Explicit occlusion reasoning for 3d object detection," in *Proceedings of British Machine Vision Conference*, 2011.

[16] R. Fransens, C. Strecha, and L. Van Gool, "A mean field em-algorithm for coherent occlusion handling in map-estimation prob," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2006.

[17] P. Dollár, C. Wojek, B. Schiele, and P. Perona, "Pedestrian detection: An evaluation of the state of the art," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2011.

[18] S. Kwak, W. Nam, B. Han, and J. H. Han, "Learn occlusion with likelihoods for visual tracking," in *Proceedings of IEEE International Conference on Computer Vision*, 2011.

[19] H. Su, M. Sun, L. Fei-Fei, and S. Savarese, "Learning a dense multi-view representation for detection, viewpoint classification and synthesis of object categories," in *Proceedings of IEEE International Conference on Computer Vision*, 2009.

[20] A. Thomas, V. Ferrari, B. Leibe, T. Tuytelaars, B. Schiele, and L. Van Gool, "Towards multi-view object class detection," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2006.

[21] S. Bao, M. Sun, and S. Savarese, "Toward coherent object detection and scene layout understanding," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2010.

[22] D. Hoiem, A. Efros, and M. Hebert, "Putting objects in perspective," *International Journal of Computer Vision*, 2008.

[23] S. Hinterstoisser, V. Lepetit, S. Ilic, P. Fua, and N. Navab, "Dominant orientation templates for real-time detection of texture-less objects," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2010.

[24] E. Hsiao and M. Hebert, "Occlusion reasoning for object detection under arbitrary viewpoint," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2012.

[25] G. Shu, A. Dehghan, O. Oreifej, E. Hand, and M. Shah, "Part-based multiple-person tracking with partial occlusion handling," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2012.

[26] A. Ess, K. Schindler, B. Leibe, and L. Van Gool, "Improved multi-person tracking with active occlusion handling," in *Proceedings of IEEE International Conference on Robotics and Automation, Workshop on People Detection and Tracking*, 2009.

[27] J. Xing, H. Ai, and S. Lao, "Multi-object tracking through occlusions by local tracklets filtering and global tracklets association with detection responses," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2009.

[28] A. Kowdle, A. Gallagher, and T. Chen, "Revisiting depth layers from occlusions," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2013.

[29] S. Tang, M. Andriluka, and B. Schiele, "Detection and tracking of occluded people." in *Proceedings of British Machine Vision Conference*, 2012.

[30] P. Felzenszwalb, D. McAllester, and D. Ramanan, "A discriminatively trained, multiscale, deformable part model," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2008.

[31] A. Vedaldi and A. Zisserman, "Structured output regression for detection with partial truncation," in *Proceedings of Neural Information Processing Systems*, 2009.

[32] L. Sigal and M. J. Black, "Measure locally, reason globally: Occlusion-sensitive articulated pose estimation," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2006.

[33] B. Wu and R. Nevatia, "Detection and segmentation of multiple, partially occluded objects by grouping, merging, assigning part detection responses," *International Journal of Computer Vision*, vol. 82, no. 2, pp. 185–204, 2009.

[34] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake, "Real-time human pose recognition in parts from single depth images," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2011.

[35] C. Wojek, S. Walk, S. Roth, and B. Schiele, "Monocular 3d scene understanding with explicit occlusion reasoning," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2011.

[36] B. Pepik, M. Stark, P. Gehler, and B. Schiele, "Occlusion patterns for object class detection," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2013.

[37] M. Z. Zia, M. Stark, and K. Schindler, "Explicit occlusion modeling for 3D object class representations," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2013.

[38] T. Wang, X. He, and N. Barnes, "Learning structured Hough voting for joint object detection and occlusion reasoning," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2013.

[39] L. Santalo, *Integral geometry and geometric probability*. Addison-Wesley Publishing Co., Reading, MA, 1976.

[40] M. Sun, G. Bradski, B. Xu, and S. Savarese, "Depth-encoded hough voting for joint object detection and shape recovery," in *Proceedings of European Conference on Computer Vision*, 2010.

[41] J. F. Lalonde, D. Hoiem, A. A. Efros, C. Rother, J. Winn, and A. Criminisi, "Photo clip art," in *ACM SIGGRAPH*, 2007.

[42] R. V. Hogg and E. Tanis, *Probability and Statistical Inference*, 8th ed. Pearson, 2009.

Edward Hsiao received the BS with honor in electrical engineering from the California Institute of Technology in 2008. He received the PhD degree in robotics from The Robotics Institute at Carnegie Mellon University in 2013. He was the recipient of the National Science Foundation Graduate Research Fellowship. His current research interests in computer vision include recognizing texture-less objects under arbitrary viewpoint in cluttered scenes and increasing the robustness of object detection under occlusions. He is a student member of the IEEE.

Martial Hebert is a professor at The Robotics Institute, Carnegie Mellon University. His current research interests include object recognition in images, video, and range data, scene understanding using context representations, and model construction from images and 3D data. His group has explored applications in the areas of autonomous mobile robots, both in indoor and in unstructured, outdoor environments, automatic model building for 3D content generation, and video monitoring. He has served on the program committees of the major conferences in the computer vision area. He is a member of the IEEE.

APPENDICES

Many of the results derived for the Occlusion Prior and Occlusion Conditional Likelihood are from the classic field of integral geometry [39]. In the following, we show the detailed derivations.

APPENDIX A
PROBABILITY DENSITY OF $\hat{w}$

We show how to transform a uniform variable over a $\frac{\pi}{2}$ interval by (1) using the *distribution function technique* [42]. First, let's simplify the equation for the projected width:

$$\hat{w}(\theta) = w \cdot \cos\theta + l \cdot \sin\theta \tag{19}$$

$$= \sqrt{w^2+l^2} \cdot \left\{ \frac{w}{\sqrt{w^2+l^2}} \cdot \cos\theta + \frac{l}{\sqrt{w^2+l^2}} \cdot \sin\theta \right\} \tag{20}$$

$$= \sqrt{w^2+l^2} \cdot \left\{ \cos\left[\tan^{-1}\left(\frac{l}{w}\right)\right] \cdot \cos\theta + \sin\left[\tan^{-1}\left(\frac{l}{w}\right)\right] \cdot \sin\theta \right\} \tag{21}$$

$$= \sqrt{w^2+l^2} \cdot \cos\left[\theta - \tan^{-1}\left(\frac{l}{w}\right)\right]. \tag{22}$$

Since the transformation over any $\frac{\pi}{2}$ interval is equivalent, the shift by $\tan^{-1}\left(\frac{l}{w}\right)$ is irrelevant. For simplicity of derivation, consider the interval $[\theta_1, \theta_2]$, where $\theta_1 = \cos^{-1}\left(\frac{l}{\sqrt{w^2+l^2}}\right)$ and $\theta_2 = \cos^{-1}\left(\frac{w}{\sqrt{w^2+l^2}}\right)$. We define the random variable Θ to have a uniform density over this interval:

$$p_\Theta(\theta) = \begin{cases} 2/\pi, & \theta_1 \le \theta \le \theta_2 \\ 0, & else. \end{cases} \tag{23}$$

To compute the probability density of $\hat{w}$, we apply the transformation $g(\theta) = \hat{w}(\theta)/\sqrt{w^2+l^2} = \cos\theta$ to Θ to produce the random variable Y (i.e., $Y = g(\Theta)$). The distribution function technique calculates the density $p_Y(y)$ of Y by first finding the cumulative distribution function $P_Y(y)$ and then taking the derivative. There are two cases.

Case 1: $\cos\theta_2 < y < \cos\theta_1$

$$P_Y(y) = \int_{\cos^{-1} y}^{\theta_2} \frac{2}{\pi} \cdot d\theta \tag{24}$$

$$= -\frac{2}{\pi}\cos^{-1} y + \frac{2}{\pi}\theta_2 \tag{25}$$

$$p_Y(y) = \frac{2}{\pi\sqrt{1-y^2}} \tag{26}$$

Case 2: $\cos\theta_1 < y < 1$

$$P_Y(y) = \int_{-\theta_1}^{-\cos^{-1} y} \frac{2}{\pi} \cdot d\theta + \int_{\cos^{-1} y}^{\theta_2} \frac{2}{\pi} \cdot d\theta \tag{27}$$

$$= -\frac{4}{\pi}\cos^{-1} y + \frac{2}{\pi}(\theta_1+\theta_2) \tag{28}$$

$$p_Y(y) = \frac{4}{\pi\sqrt{1-y^2}} \tag{29}$$

Thus we have that:

$$p_Y(y) = \begin{cases} \frac{2}{\pi\sqrt{1-y^2}}, & \cos\theta_2 < y < \cos\theta_1 \\ \frac{4}{\pi\sqrt{1-y^2}}, & \cos\theta_1 < y < 1 \end{cases} \tag{30}$$

Substituting in θ_1, θ_2 and y, we obtain the probability density function $p_{\hat{w}}(\hat{w})$ in (2).

APPENDIX B
OCCLUSION PRIOR

We show how to compute the area A_{V_i,O_c} covering all the possible positions of the red block in Fig. 3a which occlude the object but keep the point X_i visible. This region is specified in green in Fig. 20. To compute the area A_{V_i,O_c}, we break it up into the area of three parts:

$$A_{V_i,O_c} = \Delta_{1,1} + \Delta_{1,2} + \Delta_2. \tag{31}$$

From the figure, we can see that the purple region has area:

$$\Delta_{1,1} + \Delta_{1,2} = \hat{W}_{obj} \cdot \hat{h}. \tag{32}$$

Note that this area does not depend on the position of X_i. On the other hand, the area of yellow region does depend on the y coordinate of X_i. If the projected height of the block $\hat{h}$ is shorter than y_i, the height of the yellow region is $\hat{h}$. If it is taller, there are less possible positions of the red block and the height of the yellow region is y_i. Thus its area is:

$$\Delta_2 = \begin{cases} \hat{w} \cdot \hat{h}, & \hat{h} \le y_i \\ \hat{w} \cdot y_i, & \hat{h} > y_i \end{cases} \tag{33}$$

Combining (32) and (33), we get the area A_{V_i,O_c} in (7).

APPENDIX C
OCCLUSION CONDITIONAL LIKELIHOOD

We show how to compute the area A_{V_i,V_j,O_c} covering all the possible positions of the red block in Fig. 3a which occlude the object but keep both points X_i and X_j visible. Without loss of generality, assume that X_i is lower than X_j (i.e., $y_i \le y_j$). If this is not the case, we can simply switch the points. The region is specified in green in Fig. 21. To compute the area A_{V_i,V_j,O_c}, we break it up into the area of five parts:

$$A_{V_i,V_j,O_c} = \Lambda_{1,1} + \Lambda_{1,2} + \Lambda_2 + \Lambda_3 + \Lambda_4 \tag{34}$$

From the figure, we can see that the purple region has area:

$$\Lambda_{1,1} + \Lambda_{1,2} = (\hat{W}_{obj} - |x_i - x_j|) \cdot \hat{h} \tag{35}$$

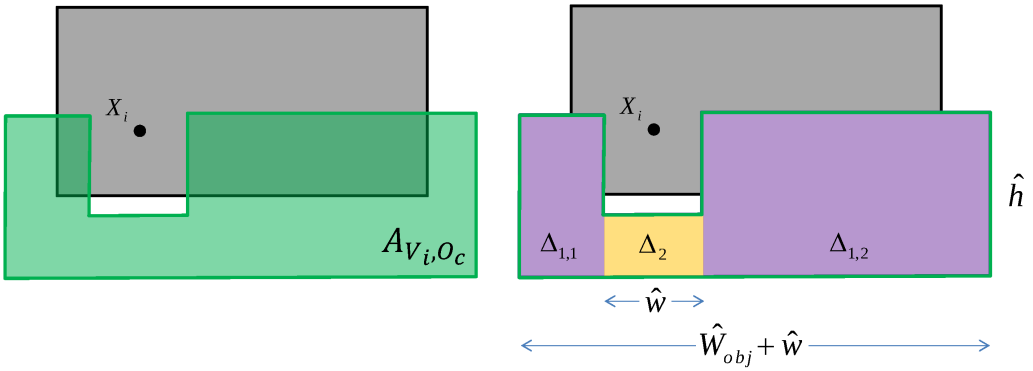


Fig. 20: Detailed illustration of how to compute the area A_{V_i,O_c} covering all the possible positions where the red block in Fig. 3a occludes the object while keeping X_i visible.

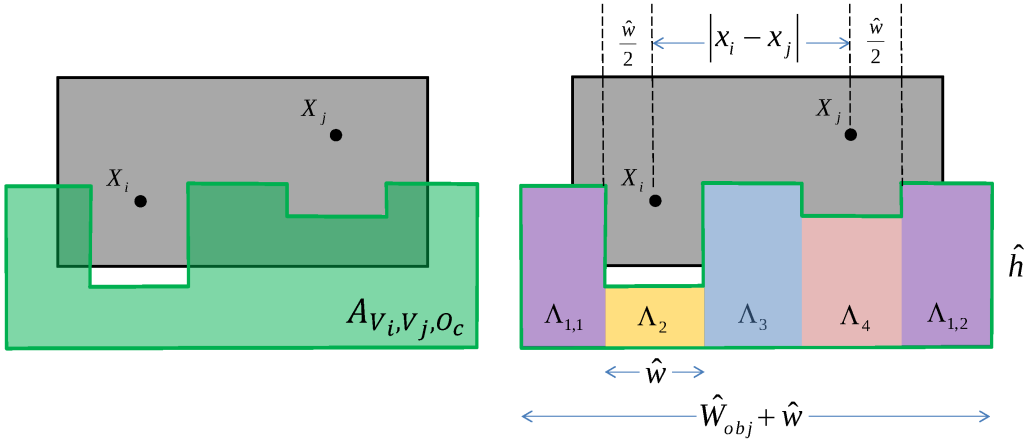


Fig. 21: Detailed illustration of how to compute the area A_{V_i,V_j,O_c} covering all the possible positions where the red block in Fig. 3a occludes the object while keeping both X_i and X_j visible.

The area of the yellow region is the same as for the occlusion prior, and it does not depend on the location of X_j:

$$\Lambda_2 = \begin{cases} \hat{w} \cdot \hat{h}, & \hat{h} \leq y_i \\ \hat{w} \cdot y_i, & \hat{h} > y_i \end{cases} \tag{36}$$

The blue region covers the positions of the block which can fit in between X_i and X_j. If the projected width $\hat{w}$ is greater than $|x_i - x_j|$, it can not fit in this region and the area is 0. However, if it is less than $|x_i - x_j|$, the width of the blue region is $|x_i - x_j| - \hat{w}$. Thus the area is:

$$\Lambda_3 = \begin{cases} (|x_i - x_j| - \hat{w}) \cdot \hat{h}, & \hat{w} \leq |x_i - x_j| \\ 0, & \hat{w} > |x_i - x_j| \end{cases} \tag{37}$$

The orange region covers the positions of the block that are below X_j. When the projected width $\hat{w}$ is less than $|x_i - x_j|$, the computation of the area is similar to Λ_2. However, when it is greater, the possible horizontal positions of the block is restricted by point X_i. In this case, instead of $\hat{w}$ positions, there are only $|x_i - x_j|$ positions. Thus the area is:

$$\Lambda_4 = \begin{cases} \hat{w} \cdot \hat{h}, & \hat{w} \leq |x_i - x_j|, \hat{h} \leq y_j \\ \hat{w} \cdot y_j, & \hat{w} \leq |x_i - x_j|, \hat{h} > y_j \\ |x_i - x_j| \cdot \hat{h}, & \hat{w} > |x_i - x_j|, \hat{h} \leq y_j \\ |x_i - x_j| \cdot y_j, & \hat{w} > |x_i - x_j|, \hat{h} > y_j. \end{cases} \tag{38}$$

Combining (35), (36), (37), (38) and simplifying the equation, we get:

$$\begin{aligned} A_{V_i,V_j,O_c} = & (\hat{W}_{obj} - |x_i - x_j|) \cdot \hat{h} \\ & + \hat{w} \cdot \min(\hat{h}, y_i) \\ & + \delta(\hat{w} \leq |x_i - x_j|) \cdot (|x_i - x_j| - \hat{w}) \cdot \hat{h} \\ & + \min(\hat{w}, |x_i - x_j|) \cdot \min(\hat{h}, y_i) \end{aligned} \tag{39}$$

Integrating over the projected width and height distributions $p_{\hat{w}}$ and $p_{\hat{h}}$, we get the average area in (11).

APPENDIX D
COMPUTING AREA FOR SILHOUETTE

We show how to compute the area of the silhouette A_s used in (14) and (15). Given a mask M, we extract the height of the lowest point $\mathcal{Y}^M(x)$ relative to bottom of the mask for each unique position $x \in \mathcal{X}^M$. Then for an

Algorithm 1 Mask Sliding Min, $\Omega(\mathcal{X}^M, \mathcal{Y}^M, \hat{w}, \hat{h})$

Require: bottom of mask $(\mathcal{X}^M, \mathcal{Y}^M)$, projected width $\hat{w}$, projected height $\hat{h}$

$\mathcal{Y}^M = \min(\mathcal{Y}^M, \hat{h})$
for $x = \min(\mathcal{X}^M) - \frac{\hat{w}}{2} \rightarrow \max(\mathcal{X}^M) + \frac{\hat{w}}{2}$ **do**
    $\mathcal{Z}(x) = \min_{\hat{x} \in [x - \hat{w}/2, x + \hat{w}/2]} \mathcal{Y}^M(\hat{x})$
end for
return $\mathcal{Z}$

occluder with projected width and height $(\hat{w}, \hat{h})$, the area covering all the positions that intersect the bounding box but not the silhouette is given by,

$$A_s = \int \Omega(\mathcal{X}^M, \mathcal{Y}^M, \hat{w}, \hat{h}) \cdot dx, \tag{40}$$

where $\Omega(\mathcal{X}^M, \mathcal{Y}^M, \hat{w}, \hat{h})$ is the *Mask Sliding Min* shown in Algorithm 1. This function considers the highest position to place an occluder at position x while not intersecting the mask. The position is lower than the height of the occluder and lower than the height of all mask points within an interval $[-\hat{w}/2, \hat{w}/2]$ of x.

For a distribution of occluding blocks $p_{\hat{w}}(\hat{w})$ and $p_{\hat{h}}(\hat{h})$ for $\hat{w}$ and $\hat{h}$ respectively, the average areas are then given by:

$$\iiint \Omega(\mathcal{X}^M, \mathcal{Y}^M, \hat{w}, \hat{h}) \cdot p_{\hat{w}}(\hat{w}) \cdot p_{\hat{h}}(\hat{h}) \cdot dx \cdot d\hat{w} \cdot d\hat{h}. \tag{41}$$