Learning Structured Hough Voting for Joint Object Detection and Occlusion Reasoning

Tao Wang Xuming He Nick Barnes
NICTA & Australian National University, Canberra, ACT, Australia
{tao.wang, xuming.he, nick.barnes}@nicta.com.au

Abstract

We propose a structured Hough voting method for detecting objects with heavy occlusion in indoor environments. First, we extend the Hough hypothesis space to include both object location and its visibility pattern, and design a new score function that accumulates votes for object detection and occlusion prediction. In addition, we explore the correlation between objects and their environment, building a depth-encoded object-context model based on RGB-D data. Particularly, we design a layered context representation and allow image patches from both objects and backgrounds voting for the object hypotheses. We demonstrate that using a data-driven 2.1D representation we can learn visual codebooks with better quality, and more interpretable detection results in terms of spatial relationship between objects and viewer. We test our algorithm on two challenging RGB-D datasets with significant occlusion and intraclass variation, and demonstrate the superior performance of our method.

1. Introduction

Object detection and localization remains a challenging task for cluttered/crowded scenes, such as indoor environments, where objects are frequently occluded by neighboring objects or the viewing window [7, 26]. The partial objects being observed usually provide limited information on the object position and pose, so many previous object detection approaches are prone to failure as they solely rely on image cues from objects themselves.

It is widely acknowledged that contextual information plays an important role in detecting and localizing objects in such adverse conditions. Many context-aware object detection methods have been proposed recently [28, 25, 12, 3]. However, most existing contextual models focus on 2D spatial relationships between objects on the image plane and fewer works have extended the modeling to 3D scenarios [2, 22]. One main difficulty in modeling 3D context was the lack of accessible 3D data. With the recent progress in consumer-level depth sensors (e.g., Kinect), however, it becomes feasible to collect a large amount of high quality depth and registered color images for indoor environments [8, 15].

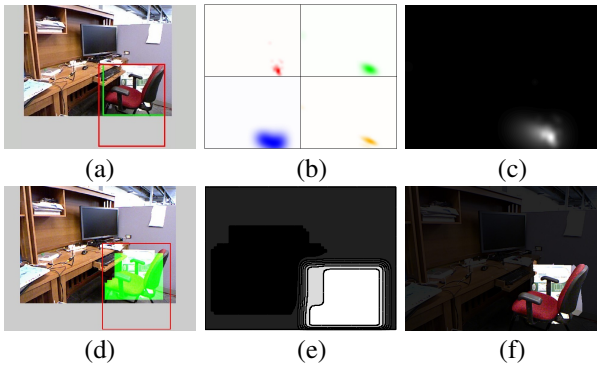

Figure 1. Illustration of the proposed approach. **(a)** RGB frame with object bounding box (red) and visible part bounding box (green). **(b)** Object centroid voting from multiple layers. **(c)** Combined object centroid voting results. **(d)** Detector output (red) with visibility pattern prediction (green). **(e)** Object visibility pattern prediction results. **(f)** Final segmentation results.

Modeling context from a 3D perspective has several advantages over its 2D counterpart conceptually. First, spatial relationships have smaller variations and are easier to interpret semantically; in addition, more spatial relationships in physical world can be captured, instead of being limited to relative positions on image plane. In particular, occlusion can be viewed as a special type of contextual relationship in 3D, which would become an intrinsic component of object and scene models. Finally, joint modeling of an object class and its 3D context may provide effective constraints on the object's scope on image plane and lead to a coarse-level object segmentation. See Fig. 1 for an example.

Our work aims to utilize RGB-D datasets to learn a context-aware object detection model which encodes depth cues and a coarse level of 3D relationships. We focus on training a depth-dependent appearance model for each object class and its context. The learned depth-encoded object

and context model is then applied to 2D images during test so it can be used to facilitate generic object detection [24].

Specifically, we propose a structured Hough voting method that incorporates depth-dependent contexts into a code-book based object detection model. Our model generalizes the traditional Hough voting detection methods in three ways. First, we design a multi-layer representation of *image context* for indoor scenes that captures the layout structure of scenes. An image region contributes to each object hypothesis in a different manner based on its depth layer. Secondly, we define a new object hypothesis space in which both the object's center and its visibility mask will be predicted. Each image patch will generate a weighted vote to a joint score of the object center and its support mask in the image. Finally, we view occlusion as special contextual information, which could provide cues for object localization and help with reasoning about visibility of object parts. The overall output of our approach is a simultaneous object detection and coarse segmentation.

Our detection and segmentation are achieved by maximizing the joint score of object center and visibility mask. We derive an efficient alternating ascent method to search modes of the Hough voting score maps. To learn the model from partially labeled RGB-D data, we adopt an approximate learning procedure based on the max-margin Hough transform [13]. We extensively evaluate our approach on two public RGB-D datasets and demonstrate its efficiency.

The paper is organized as follows. We briefly discuss related work in Section 2. The details of our model structure are introduced in Section 3. Section 4 describes the inference procedure in our structured Hough voting, followed by max-margin learning for model estimation. Experimental evaluation is detailed in Section 5 and Section 6 concludes the paper.

2. Related work

Recently, Hough voting based methods [1] have been widely used in object detection and recognition, and progress has been made in areas including discriminative codebook learning [6, 30], efficient inference methods [10], joint recognition and segmentation [11, 18], and scalable multi-class detection [17]. However, the majority of Hough voting methods focus on improving the target object model and few have studied context and occlusion reasoning. Joint detection and segmentation with Hough voting based methods has been investigated in [11], which only represents the object parts with additional masks and generates segmentation in two separate stages. Previous work also investigated maxima search in high-dimensional Hough spaces [20, 14, 16]. Unlike those methods, our inference iteratively optimizes a well-defined objective function of object center and visibility mask.

Context-aware object detection in 2D scenarios has been well studied [25]. See [28] for a recent review. Many works have incorporated object-level context and rely on semantic contextual information for object segmentation (e.g., [21, 9]). In particular, [29] has shown that reasoning a 2.1D layered object representation in a scene can positively impact object localization. Our work, however, explores depth encoded image context for improving object detection.

Depth information has been incorporated into object feature to improve detection and segmentation performance (e.g., [22, 15]). However, most of existing work relies on the depth cue during test and so could not be applied to 2D images. In terms of depth transfer, the closest related work are [24] and [27] which also use a depth-encoded patch selection process for Hough transform-based detection. However, [24] uses the depth only to prune out patches of incorrect scales, and to create a generative depth model. In [27], we solely focused on object detection with a single layer context model. Recently, [23] has explicitly considered geometric context and 3D scene layout. Our work seeks a unified model that can encode object and context information simultaneously at the object level.

Brox et al. [4] use a part-based poselet detector and align the corresponding part masks to image boundary cues. However, they did not incorporate explicit occlusion and context modeling with depth. Another work which also reasoned about occlusion within bounding boxes for object detectors is [7]. The bounding box representation was augmented with a set of variables to generate a binary occlusion pattern. Again, their method mainly targets the object model itself and relies on object structure.

3. Our approach

3.1. Structured Hough voting

We first briefly review the original Hough voting based object detection method and introduce notation. Hough voting methods (e.g., [11, 6]) generally use object poses as their hypothesis, accumulate scores from each image patch into a confidence map for the hypothesis space, and search for the highest voting scores from the map [1].

Mathematically, suppose we have an image I and an object class of interest o. Let the object hypothesis be $\mathbf{x} \in \mathcal{X}$, where $\mathcal{X}$ is the object pose space. To simplify the notation, we assume each hypothesis is $\mathbf{x} = (a_x, a_y, a_s)$, where a_x and a_y are the image coordinates of the object center and a_s is a scale. Hough voting methods define a scoring function $S(\mathbf{x})$ for each valid location $\mathbf{x}$ on the image plane, which is a summation of weighted votes from every local image patch. To compute the voting weights, an appearance-based codebook is usually learned from the image patches in object class o, denoted by $\mathcal{C} = \{C_i\}_{i=1}^{K}$. Each codebook entry C_i consists of a typical patch descriptor f_{ci} and geometric

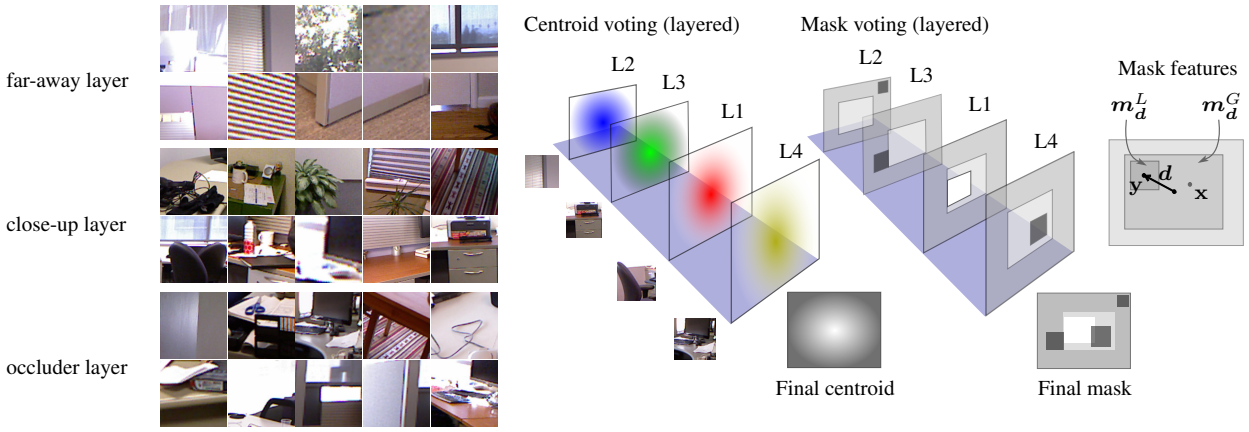

Figure 2. **Left panel**: Top-ranked clusters (presented with the patches closest to the cluster centers) for 3 contextual layers on the Berkeley 3D object dataset. **Right panel**: Illustration of multiple layered object centroid and mask voting. L1 corresponds to the object layer, and L2, L3, L4 correspond to far-away context, close-up context and occluder layers, respectively. For mask voting, brighter regions indicate a higher response, while darker regions indicate a lower response.

features D_i of training patches associated with the i-th entry. A typical geometric feature is the relative positions $\boldsymbol{d}$ of image patches w.r.t. the corresponding object centers.

Given the codebook $\mathcal{C}$, we can write the Hough score function as follows. Denote each image patch $I_{\mathbf{y}}$ by its location $\mathbf{y}$ and feature descriptor $\mathbf{f}_{\mathbf{y}}$,

$$S(\mathbf{x}) \propto \sum_{i=1}^{K} \sum_{\mathbf{y}} \omega_i p(C_i|\mathbf{y}) \sum_{\boldsymbol{d} \in D_i} e^{\left(-\frac{\|(\mathbf{y}-\mathbf{x})-\boldsymbol{d}\|^2}{2\sigma_d^2}\right)} \quad (1)$$

where $\omega_i = p(o|C_i)$ is the entry-to-class probability, $p(C_i|y)$ is the patch-to-entry matching probability, and σ_d is the standard deviation of a Gaussian filter for the object center. Notice that the object hypothesis $\mathbf{x}$ essentially specifies a bounding box. However, the bounding box hypothesis space is limited in its representation power as it is incapable of describing partial objects or its visibility pattern.

We propose to extend the object hypothesis space from a single centroid $\mathbf{x}$ to a joint space $(\mathbf{x}, \boldsymbol{v})$ and define a new score function $S(\mathbf{x}, \boldsymbol{v})$. Here $\mathbf{x}$ specifies the object center (or equivalently its bounding box), and $\boldsymbol{v}$ is a visibility mask indicating which part of object is visible, as shown in Fig. 2. The mask $\boldsymbol{v}$ has the same size as the image I, and $\boldsymbol{v}(\mathbf{y}) = 1$ if the image patch at $\mathbf{y}$ belongs to the object o, and 0 otherwise. For notation simplicity, we reshape $\boldsymbol{v}$ as 1-D vector and denote its element at image location $\mathbf{y}$ as $\boldsymbol{v}_{\mathbf{y}}$.

Our key step is, instead of using Gaussian kernels in Eqn. 1, we introduce a class of voting masks that are capable of representing the relative positions as well as the object visibility pattern. As illustrated in the rightmost figure in Fig. 2, we include a local mask and a global mask for each codebook entry. The local mask predicts if a local patch itself is part of the object, and the global mask casts a vote for the spatial extent of the whole object on the image plane based on the relative geometric feature $\boldsymbol{d}$.

Formally, each codebook entry C_i includes a new set of geometric features $\tilde{D}_i = \{\tilde{\boldsymbol{d}} = (\boldsymbol{d}, \boldsymbol{m}_{\boldsymbol{d}}^L, \boldsymbol{m}_{\boldsymbol{d}}^G)\}$, where $\boldsymbol{m}_{\boldsymbol{d}}^L$ is the local mask feature and $\boldsymbol{m}_{\boldsymbol{d}}^G$ is the global mask feature. The local mask features describe local visibility of object regions, which is similar to the ISM [11]. The global mask features limit the scope of each object in the image plane. A natural choice is an object bounding box-shaped mask. See Fig. 2. Note that by choosing a different family of mask features, our model allows for finer description of the object shape and/or visibility pattern.

For an image patch at $I_{\mathbf{y}}$ and object center hypothesis $\mathbf{x}$, we can compute two average voting masks from the i-th codebook entry as follows:

$$\boldsymbol{m}_i^G(\mathbf{x}, \mathbf{y}) \propto \sum_{\tilde{\boldsymbol{d}} \in \tilde{D}_i} \boldsymbol{m}_{\boldsymbol{d}}^G(\mathbf{x} - \mathbf{y} + \boldsymbol{d}) * G(0, \sigma_d^2) \quad (2)$$

$$\boldsymbol{m}_i^L(\mathbf{x}, \mathbf{y}) \propto \sum_{\tilde{\boldsymbol{d}} \in \tilde{D}_i} \boldsymbol{m}_{\boldsymbol{d}}^L(\mathbf{x} - \mathbf{y}) * G(0, \sigma_d^2) \quad (3)$$

where $\boldsymbol{m}^G$ and $\boldsymbol{m}^L$ are the average global and local voting mask, respectively; $\boldsymbol{m}(\mathbf{x})$ represents the mask with its center shifted to $\mathbf{x}$, $G(\cdot)$ is the Gaussian kernel, and $*$ is the convolution operator. See Fig. 2 for an illustration.

We define the new score function as a matching score between the visibility mask hypothesis $\boldsymbol{v}$ and a weighted sum of the voting mask values,

$$S(\mathbf{x}, \boldsymbol{v}) = \sum_{i=1}^{K} \omega_i \boldsymbol{v}^T \Big[\sum_{\mathbf{y}} \gamma(\boldsymbol{v}(\mathbf{y})) \Big(\boldsymbol{m}_i^G(\mathbf{x}, \mathbf{y}) + \mu \boldsymbol{m}_i^L(\mathbf{x}, \mathbf{y}) \Big) p(C_i|\mathbf{y}) - w_b \Big] \quad (4)$$

where w_b is a global bias to the mask voting score, and μ is the relative weight of the local mask. $\gamma(u)$ is a weighting

function with $\gamma(1) = 1$ and $\gamma(0) = \delta, \delta < 1$. Intuitively, we give a smaller weight to the votes not from the object itself. ω_i gives a relative weight for each codebook entry. It can be shown that when $\boldsymbol{v} = \mathbf{1}$, $\mu = 0$ and the global voting mask has the shape of object bounding box, the new score function is equivalent to the Hough voting score in Eqn. 1.

3.2. Depth-encoded context

The structured Hough voting model can easily incorporate image contextual information by extending the codebook and including votes from both object and context patches. In this work, we design a multi-layer scene representation that captures different types of image cues for detection and integrates them into the model.

Concretely, we group image patches into four layers according to their relationship with the target object: 1) An *object layer* includes all the image patches from the object itself; 2) An *occluder layer* indicates patches occluding the object; 3) A *nearby context layer* consists of context patches within 1 meter of the average object depth; 4) A *far-away context layer* has the rest of the context image patches.

We associate each layer with its own specific parameters as they contribute to object detection and occlusion reasoning in different ways. We first learn a separate codebook-based appearance model for each layer using object labels and depth cues. Denote the i-th codebook entry of layer l as C_i^l, we define a context-aware structured Hough voting model by including the votes from all the layers:

$$S_c(\mathbf{x}, \boldsymbol{v}) = \sum_{l=1}^{4} \sum_{i=1}^{K_l} \omega_i^l \boldsymbol{v}^T \Big[\sum_{\mathbf{y}} \gamma(\boldsymbol{v}(\mathbf{y})) \Big(\boldsymbol{m}_{l,i}^G(\mathbf{x}, \mathbf{y}) + \mu^l \boldsymbol{m}_{l,i}^L(\mathbf{x}, \mathbf{y}) \Big) p(C_i^l | \mathbf{y}) - w_b^l \Big] \quad (5)$$

where K_l is the size of the codebook in layer l. Note that each layer has its own Gaussian kernel width σ_d^l in the voting masks. The details of each layer are as follows.

A. Depth-encoded codebooks. We use HOG features [5] for image patches on the target object and Texton like [21] features for patches from context layers. The initial codebooks are generated by K-means clustering of randomly sampled patches. To capture discriminative patches, we also use an interest point detector to sub-sample the patch pool. The Texton feature, which is a coarser level descriptor, is better for capturing context in a scene. Some examples of image patches in our codebooks are shown in Fig. 2. We can see that different types of scene structure are captured. We further refine the initial codebooks by utilizing depth information available during training. Specifically, we rank each cluster in each layer by its 3D offset variance, and prune out those ranked in the last 25%.

B. Layer-dependent voting masks. We design the global mask feature $\boldsymbol{m}_{\boldsymbol{d}}^G$ and local mask feature $\boldsymbol{m}_{\boldsymbol{d}}^L$ according to the property of each layer. In this work, all the global masks have the same shape as the object bounding box. Thus all active patches contribute to limiting the scope of the object. For the local masks, the object layer has a positive 2D stump with $1/10$th of the object size, while other layers have a negative 2D stump with the same size. Intuitively, the active image patches from context layers help localize the object center but also indicate the local patches that do not belong to the object. In addition, we set the Gaussian blur parameter σ_d^l such that the far away context layer has larger variances in terms of center prediction (3 times).

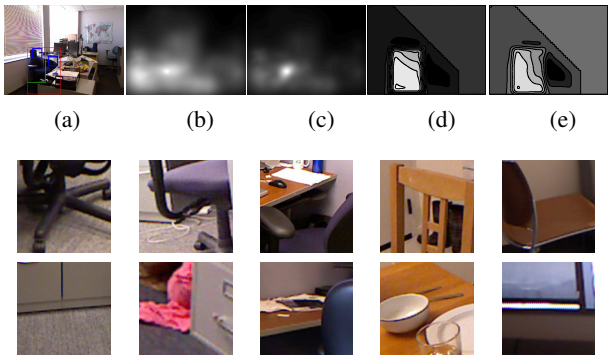

Figure 3. Illustration of the impact of patch pair terms on hypothesis scoring. **Upper panel**: A specific example, with **(a)** RGB frame with an example of a patch pair (in blue rectangles). **(b)** Object centroid voting results without patch pair terms. **(c)** Object centroid voting results with patch pair terms added. **(d)** Shape voting results without patch pair terms. **(e)** Shape voting results with patch pair terms added. **Lower panel**: The highest ranked patch pairs on the Berkeley 3D object dataset. The first row shows on-object patches, and the second row shows off-object patches. Each column corresponds to a patch pair.

3.2.1 Second-order features

In addition to layered codebooks, which are built on single patches, we utilize patch feature pairs to improve the discriminative power of the model [31]. In particular, we focus on co-occurring objects and contextual feature pairs. These pair feature can refine the context relationship and better predict the object boundary.

We incorporate the object-context pair features into our structured Hough voting model by adding a second-order term to the score function: $S(\mathbf{x}, \boldsymbol{v}) = S_c(\mathbf{x}, \boldsymbol{v}) + \alpha S_p(\mathbf{x}, \boldsymbol{v})$, where α is the relative weight, and S_p is the object-context feature pair term. Assume the first layer $l = 1$ is the object layer, S_p can be written as

$$S_p(\mathbf{x}, \boldsymbol{v}) = \sum_{i=1}^{K_1} \sum_{l=2}^{4} \sum_{j=1}^{K_l} \omega_{ij}^l \boldsymbol{v}^T \Big[\sum_{\mathbf{y}, \mathbf{y}'} \gamma(\boldsymbol{v}(\mathbf{y})) \Big(\boldsymbol{m}_{1,i}^G \odot \boldsymbol{m}_{l,j}^G + \mu^l \boldsymbol{m}_{1,i}^L \oplus \boldsymbol{m}_{l,j}^L \Big) \cdot \varphi - w_b^{1,l} \Big] \quad (6)$$

Algorithm 1: Alternating Inference for $S(\mathbf{x}, \boldsymbol{v})$.

Input: Input Image I; Layered Codebooks $\mathcal{C} = \{C_i\}, i = 1 \cdots N_L$; Offsets D_i; Mask templates $\boldsymbol{m_d}(y), \boldsymbol{m'_d}(y), \forall d \in D_i$; Entry weights $\{\omega_i^l, \mu_j^l, \omega_{ij}^l, \mu_{ij}^l\}$; Model parameters $\tau, \alpha, \delta, \kappa$; Local maxima seeds N_{seed}; termination threshold $\varepsilon > 0$; Maximum iterations $T_{\max}$.

Initialization: Let $\boldsymbol{v} = \mathbf{1}$, search for N_{seed} local maxima for $S(\mathbf{x}, \mathbf{1})$: $\mathbf{x}_i, i = 1 \cdots N_{\text{seed}}$.

for each local maxima x_i **do**

  for iteration $= 1 : T_{\max}$ **do**

    1. Obtain a new $\boldsymbol{v_i^*}$ by solving Eqn. 7;
    2. Check for optimal solution: **if** $S(\mathbf{x}_i, \boldsymbol{v_i}) - S(\mathbf{x}_i, \boldsymbol{v_i^*}) < \varepsilon$, **then** break and the problem is solved;
    3. $\boldsymbol{v} \leftarrow \boldsymbol{v_i^*}$, vote again for x_i^* with $\boldsymbol{v_i}$, $\mathbf{x}_i \leftarrow \mathbf{x}_i^*$.

  end

  Mask Recalculation: Obtain a new $\boldsymbol{v_i^*}$ by solving Eqn. 7, $\boldsymbol{v} \leftarrow \boldsymbol{v}^*$.

end

Output: $argmax_{(\mathbf{x}_i, \boldsymbol{v_i})} S(\mathbf{x}_i, \boldsymbol{v_i})$

where $\odot$ and $\oplus$ are the element-wise product and addition operators, respectively. We omit the variable $(\mathbf{x}, \mathbf{y})$ in $\boldsymbol{m}$ for clarity of the notation. ω_{ij}^l is the weight for the object-context codebook entry pairs. The patch pair to entry matching probability $\varphi = p(C_j^l|C_i^1)p(C_i^1|y)p(C_j^l|y')$ and $p(C_j^l|C_i^1)$ is estimated by the feature co-occurence frequency matrix during training. We also use depth information to prune out geometrically unstable or inconsistent codebook pairs as in the previous subsection.

4. Model learning and inference

4.1. Joint inference for object localization

Once the structured Hough voting model is trained with depth-augmented image data, we can apply it to 2D images for object detection and occlusion prediction. Our method infers the object center hypothesis and its visibility mask by maximizing the Hough score function $S(\mathbf{x}, \boldsymbol{v})$. However, due to the large hypothesis space of $(\mathbf{x}, \boldsymbol{v})$, it is difficult to use the original Hough voting approach, or conduct brute-force search. In this section, we propose a coordinate-ascent method which finds the local maxima of the score function.

Specifically, we alternatively maximize the score function with respect to one variable, while keeping the other fixed. When $\boldsymbol{v}$ is fixed, the optimization is the same as the original Hough voting. We only need to carry out a weighted Hough voting step and the local maxima $\mathbf{x}_i^*$ can be retrieved from the Hough map. When the object center is fixed, our Hough score is a quadratic function of the binary vector $\boldsymbol{v}$. To convert $S(\mathbf{x}, \boldsymbol{v})$ into its quadratic form, we notice that $\gamma(\boldsymbol{v}(\mathbf{y})) = (1 - \delta)\boldsymbol{v}(\mathbf{y}) + \delta$. Plugging this into Eqn 5, 6, we can write the score function as

$$S(\mathbf{x}, \boldsymbol{v}) = \boldsymbol{v}^T A(\mathbf{x})\boldsymbol{v} + \boldsymbol{v}^T B(\mathbf{x}) \tag{7}$$

where A and B are matrix functions. We refer readers to the supplementary material for its detailed derivation. We choose to solve a relaxed version of this problem by allowing $\boldsymbol{v}(\mathbf{y}) \in [0, +1]$, which is a constrained quadratic programming problem. We find an approximate binary solution by searching for an optimal threshold to binarize the solution vector. Note that the constraint for the relaxed quadratic programming problem will enforce invisibility for any image location $\mathbf{y}$ outside the bounding box $\mathbf{x}$, i.e., $\boldsymbol{v}(\mathbf{y}) = 0, \forall \mathbf{y} \notin \mathbf{x}$. This greatly reduces the search space.

The inference algorithm is overviewed in Algorithm 1. It initializes the object center hypothesis with the original Hough voting method, and search object hypotheses at multiple scales. Figure 4 shows the iterative inference process.

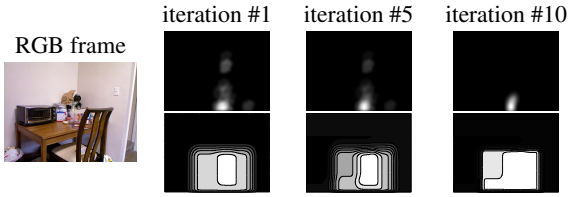

Figure 4. An illustration of how iterative inference updates the object centroid and supporting mask hypotheses. The first row on the right shows object centroid voting, with the corresponding supporting mask estimation in the second row.

4.2. Learning with depth-augmented data

Our model in Eqns. 5 and 6 is linear in terms of its weight vector $\mathbf{w} = \{\omega_i^l, \mu_j^l, \omega_{ij}^l, l = 1, \cdots, 4, i, j = 1, \cdots, K^l\}$. We utilize the max-margin Hough transform [13] framework to train the weight parameters $\mathbf{w}$.

We assume only a coarse labeling of the visibility is available for positive training data. To speed up training, we generate a negative example set that consists of incorrect labeling from applying a simple version of our model with uniform weights, i.e., $\mathbf{w} = \mathbf{1}$. For all the other model parameters, we use cross-validation to find their values using a held-out validation set. We refer readers to the supplementary material for details.

5. Experimental evaluation

5.1. Dataset and setup

We evaluate the proposed structured Hough voting method on two challenging RGB-D object datasets: the

Berkeley 3D Object (B3DO) Dataset (Version 1) [8] and a subset of object classes on the NYU Depth Dataset (Version 2) [15]. B3DO contains 849 images taken in 75 different scenes, and 8 object categories. We follow the experimental settings in [8]. The NYU Depth dataset has a total of 1449 labelled images and we randomly split the images into 3 subsets for training, validation and testing, taking approximately 40%, 20% and 40% of all labelled images. As the dataset was originally designed for pixelwise scene segmentation, it contains many background classes (e.g., wall, ceiling) which are not suitable for our object representation. Therefore, we run experiments with only 5 categories: table, chair, door, bed and sofa.

As labeling of visibility masks is expensive to obtain, we assume only coarse-level labels for our masks. Two bounding boxes are used: one for whole object and the other for visible parts. Some examples of the ground truth labelling is shown in Fig. 7 (and more in supplementary material). For evaluation of segmentation accuracy we also manually label the visibility ground-truth using polygons on the B3DO dataset. We modified some problematic labeling on both datasets but they only take up a small fraction [1].

5.2. Model details

For codebook generation, we randomly sample 200 patches per image from the visible part bounding box and generate 400 clusters for non-object patches using K-means, then rank them according to the patches' offset variance. We then prune these clusters by discarding clusters with 20 or less members, and discard again remaining clusters with ranking in the last 25%. For other layers (i.e., context and occluder), we sample 400 patches per image and generate 800 clusters as the appearance variability is larger with context and occluders. For these layers we follow a similar pruning process after a second round of clustering is performed as discussed in Section 3.

During test, our detector first searches for up to 50 local peaks in the Hough image with $\boldsymbol{v} = \mathbf{1}$, and then runs a full version of inference and computes scoring functions for each of these peaks. Our alternate inference algorithm is likely to converge in a few iterations in most cases so we limit the maximum number of iterations to 20. The inference is efficient and complete detection takes around 5 seconds per image with a quad-core desktop computer, using our parallel MATLAB implementation.

After object location and the corresponding visibility mask is inferred, we run GrabCut [19] in the bounding box specified by $\mathbf{x}$ to generate a final segmentation mask to utilize bottom-up image cues and examine segmentation performance. Based on the shape voting results, we set regions with highest responses as foreground seeds and regions with lowest responses as background seeds, then run GrabCut for 10 iterations to get the final segmentation mask.

[1]The modified labeling can be downloaded from http://users.cecs.anu.edu.au/~taowang.

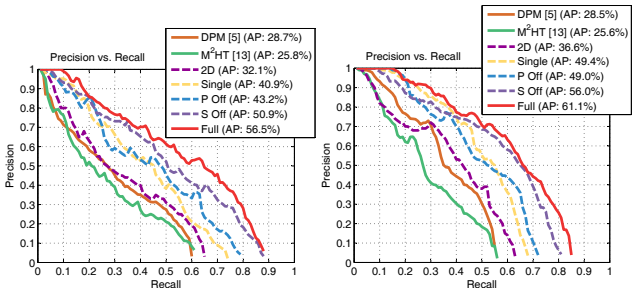

Figure 5. Detection precision-recall curves on the Berkeley 3D Object dataset (left) and the NYU Depth dataset (right). The solid curves correspond to our full model (**Full**) and two baseline methods: Deformable Parts Model (**DPM**) [5] and Max-margin Hough transform (**M^{2}HT**) [13]. The dashed curves correspond to diagnostic results with various components in our full model turned off, i.e., single layer context (**Single**), 2D geometric context (**2D**), patch pair term off (**P Off**), and segmentation off (**S Off**). See details in text.

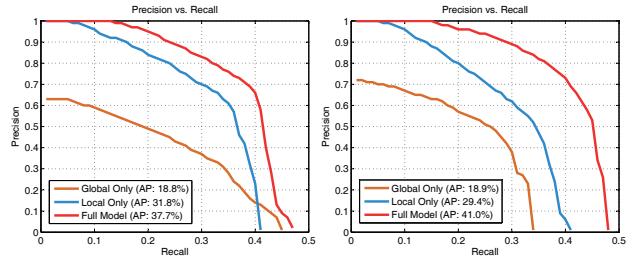

Figure 6. Precision-recall curves on the Berkeley 3D Object dataset (left) and the NYU Depth dataset (right) for segmentation at 50% recall rate in Fig. 5. Simultaneously voting for local feature position and whole object hypothesis for $\boldsymbol{v}$ yields best segmentation results.

5.3. Result comparison and analysis

In this section, we present quantitative evaluation results on the B3DO and NYU Depth datasets as well as some examples for diagnostics. Fig. 5 shows overall precision recall curves using different variants of our method versus state-of-the-art baselines. Per-class performance statistics are shown in Table 1. Specifically, we compare with Deformable Parts Model (DPM) [5], Class-specific Hough Forest (CHF) [6], and max-margin Hough transform (M^{2}HT) [13]. Note that all these methods use 2D image cues only, without encoding contextual cues. [8] also reported results on B3DO with DPM, which is similar to ours. In addition, we include a comparison with Hough voting using additional 2D geometric context, which uses 2D offset only in generating a single-layered contextual codebook. For modelling the object itself with a depth-encoded codebook, we also tried M^{2}HT with a codebook learned with 3D offset, which did not work well due to noisy labels of object

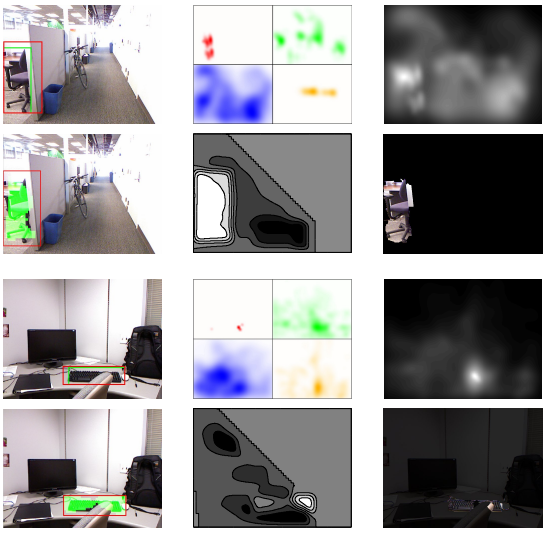

Figure 7. More illustration of the proposed approach. See Fig. 1 for caption details.

centers.

We can see that 2D geometric context contributes to detection performance slightly. With the depth-encoded contextual cues, the performance of our structured Hough voting model is improved significantly. All variants of our model which utilize depth supervision and contextual information achieved a minimum of 10% to 15% average precision increase, with further improvement by refining the contextual model.

We would like to more closely examine the effectiveness and contribution to our object hypothesis of various components in our model. We present the following variants of our model: (a) with single layered context (with patch pair terms and alternate inference); (b) with multiple layers and alternate inference of the visibility pattern, but turning off the patch pair terms in our full model; and (c) with a multiple layered context and patch pair terms, but without running alternate inference (enforcing $\boldsymbol{v} = \mathbf{1}$).

Performance is decreased by around 15% without multiple layers, suggesting that it is essential for the performance improvement. Patch pair terms have a smaller contribution to detection, but are more important for segmentation performance as illustrated in Fig. 3. Finally, alternate inference is important for detection accuracy although the average precision difference is only around 6%.

5.4. Segmentation performance analysis

Finally, we present a segmentation performance analysis with different mask terms enabled. We present the precision-recall of visibility mask at the point of 50% recall in object detection. For each object hypothesis, we obtain a soft segmentation score, which is used to compute the segmentation precision-recall curve in Fig. 6. We can see that both local and global mask features help improve the segmentation performance. It is also clear that simultaneously voting for the local mask position and the whole object mask yields best segmentation performance.

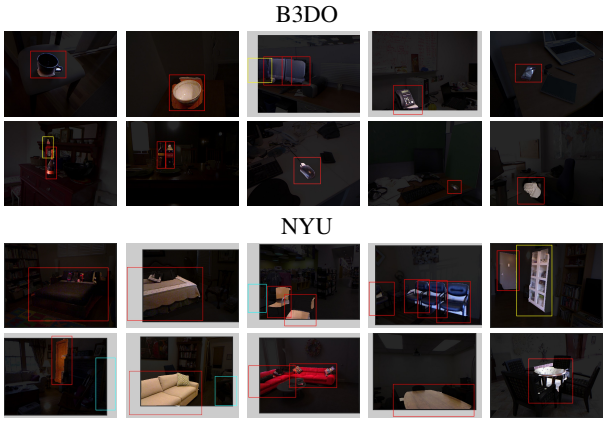

Figure 8. Examples of detection results on the Berkeley 3D object and the NYU Depth datasets. Red boxes indicate correct detections, with segmentation mask overlaid. Yellow boxes indicates false alarms and cyan boxes are missing detections.

6. Conclusion

In this paper, we have presented a novel structured Hough voting model for indoor object detection and occlusion reasoning. We extend the original Hough voting based detection model by introducing a joint Hough space of object location and visibility pattern. The structured Hough model can naturally incorporate both the object and its context information, which is especially important for cluttered indoor scenes. In addition, we utilize depth information at the training stage to build a multilayer contextual model so that a better visual codebook is learned and more detailed object-context relationships can be captured. The efficiency of our approach has been demonstrated on two publicly available RGB-D datasets, and our experiments show we achieve significant improvement over the state-of-the-art 2D object detection approaches.

Acknowledgements. NICTA is funded by the Australian Government as represented by the Department of Broadband, Communications, and the Digital Economy, and the Australian Research Council (ARC) through the ICT Centre of Excellence Program. This research was also supported in part by ARC through its Special Research Initiative (SRI) in Bionic Vision Science and Technology grant to Bionic Vision Australia (BVA).

References

[1] D. Ballard. Generalizing the hough transform to detect arbitrary shapes. *Pattern recognition*, 13(2):111–122, 1981. 2

Subset	B3DO-bottle	B3DO-bowl	B3DO-chair	B3DO-cup	B3DO-keyboard	B3DO-monitor	B3DO-mouse	B3DO-phone	NYU-table	NYU-chair	NYU-door	NYU-bed	NYU-sofa	B3DO-mean AP	NYU-mean AP
Our Approach															
Single Layer	29.8	52.1	57.8	55.7	38.7	65.9	33.9	30.1	32.2	50.2	63.2	19.7	37.2	40.9	49.4
Patch Pair Off	30.2	55.1	58.2	59.2	39.9	70.1	34.2	33.5	30.2	50.7	60.0	20.2	35.8	43.2	49.0
Segmentation Off	40.2	65.1	60.2	64.1	45.5	75.2	39.7	43.5	39.2	59.1	70.8	20.1	45.0	50.9	56.0
Full Model	**45.7**	**79.2**	**65.2**	**66.4**	**50.4**	**80.7**	**42.2**	**44.4**	**45.9**	**66.7**	**72.5**	**23.2**	**49.7**	**56.5**	**61.1**
Baseline Approaches															
DPM [5]	**13.8**	**45.1**	**15.3**	**26.6**	16.1	**76.7**	18.5	17.8	**15.8**	29.4	**63.7**	**13.1**	26.5	**28.7**	**28.5**
CHF [6]	13.1	40.3	9.7	24.5	17.2	68.7	19.3	17.5	13.5	**31.7**	58.2	12.2	21.2	22.7	26.2
M^2HT [13]	13.5	42.3	12.5	24.1	**18.2**	68.9	**20.2**	**20.1**	13.5	29.1	62.1	12.4	**26.7**	25.8	25.6

Table 1. Per-class average precision on the Berkeley 3D Object dataset and the NYU Depth dataset. Mean average precision values are calculated separately for each dataset.

[2] S. Y. Bao, M. Sun, and S. Savarese. Toward coherent object detection and scene layout understanding. In *CVPR*, 2010. 1

[3] M. Blaschko and C. Lampert. Object localization with global and local context kernels. In *BMVC*, 2009. 1

[4] T. Brox, L. Bourdev, S. Maji, and J. Malik. Object segmentation by alignment of poselet activations to image contours. In *CVPR*, 2011. 2

[5] P. Felzenszwalb, R. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. *IEEE Trans. PAMI*, 32(9):1627–1645, 2010. 4, 6, 8

[6] J. Gall and V. Lempitsky. Class-specific hough forests for object detection. In *CVPR*, 2009. 2, 6, 8

[7] T. Gao, B. Packer, and D. Koller. A segmentation-aware object detection model with occlusion handling. In *CVPR*, 2011. 1, 2

[8] A. Janoch, S. Karayev, Y. Jia, J. Barron, M. Fritz, K. Saenko, and T. Darrell. A category-level 3-d object dataset: Putting the kinect to work. In *ICCV Workshops*, 2011. 1, 6

[9] S. Kluckner, T. Mauthner, P. Roth, and H. Bischof. Semantic image classification using consistent regions and individual context. In *BMVC*, 2009. 2

[10] A. Lehmann, B. Leibe, and L. Van Gool. Fast prism: Branch and bound hough transform for object class detection. *IJCV*, 94(2):175–197, 2011. 2

[11] B. Leibe, A. Leonardis, and B. Schiele. Combined object categorization and segmentation with an implicit shape model. In *ECCV Workshops*, 2004. 2, 3

[12] M. Maire, S. Yu, and P. Perona. Object detection and segmentation from joint embedding of parts and pixels. In *ICCV*, 2011. 1

[13] S. Maji and J. Malik. Object detection using a max-margin hough transform. In *CVPR*, 2009. 2, 5, 6, 8

[14] K. Mikolajczyk, B. Leibe, and B. Schiele. Multiple object class detection with a generative model. In *CVPR*, 2006. 2

[15] P. K. Nathan Silberman, Derek Hoiem and R. Fergus. Indoor segmentation and support inference from rgbd images. In *ECCV*, 2012. 1, 2, 6

[16] N. Razavi, J. Gall, P. Kohli, and L. Van Gool. Latent hough transform for object detection. In *ECCV*, 2012. 2

[17] N. Razavi, J. Gall, and L. Van Gool. Scalable multi-class object detection. In *CVPR*, 2011. 2

[18] K. Rematas and B. Leibe. Efficient object detection and segmentation with a cascaded hough forest ism. In *ICCV Workshops*, 2011. 2

[19] C. Rother, V. Kolmogorov, and A. Blake. Grabcut: Interactive foreground extraction using iterated graph cuts. 23(3):309–314, 2004. 6

[20] E. Seemann, B. Leibe, and B. Schiele. Multi-aspect detection of articulated objects. In *CVPR*, 2006. 2

[21] J. Shotton, J. Winn, C. Rother, and A. Criminisi. Textonboost: Joint appearance, shape and context modeling for multi-class object recognition and segmentation. In *ECCV*, 2006. 2, 4

[22] E. Sudderth, A. Torralba, W. Freeman, and A. Willsky. Depth from familiar objects: A hierarchical model for 3d scenes. In *CVPR*, 2006. 1, 2

[23] M. Sun, S. Bao, and S. Savarese. Object detection using geometrical context feedback. *IJCV*, 2012. 2

[24] M. Sun, G. Bradski, B. Xu, and S. Savarese. Depth-encoded hough voting for joint object detection and shape recovery. In *ECCV*, 2010. 2

[25] A. Torralba. Contextual priming for object detection. *IJCV*, 53(2):169–191, 2003. 1, 2

[26] A. Vedaldi and A. Zisserman. Structured output regression for detection with partial truncation. In *NIPS*, 2009. 1

[27] T. Wang, X. He, and N. Barnes. Learning hough forest with depth-encoded context for object detection. In *DICTA*, 2012. 2

[28] L. Wolf and S. Bileschi. A critical view of context. *IJCV*, 69(2):251–261, 2006. 1, 2

[29] Y. Yang, S. Hallman, D. Ramanan, and C. Fowlkes. Layered object detection for multi-class segmentation. In *CVPR*, 2010. 2

[30] P. Yarlagadda, A. Monroy, and B. Ommer. Voting by grouping dependent parts. *ECCV*, 2010. 2

[31] Y. Zhang and T. Chen. Weakly supervised object recognition and localization with invariant high order features. In *BMVC*, 2010. 4