Supplementary Material for "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

In this supplementary material, we include more details on our model and several additional results on both datasets. First, we provide detailed derivation of inference and learning algorithms used in our model, which is followed by more experimental results and analysis.

1. Details on inference and learning

For clarity, we introduce the following notations for our scoring functions (i.e., Eqns. 5 and 6 in the main paper), which will be used throughout this section:

$$\begin{aligned}\mu_i^l &= \omega_i^l \mu^l \\ \mu_{ij}^l &= \omega_{ij}^l \mu^l \end{aligned} \tag{1}$$

1.1. Quadratic scoring function for inference

We first provide our derivation of the coordinate-ascent method used in the alternating inference, which searches for the best scoring $\mathbf{x}$ and $\boldsymbol{v}$. The overview of our method is given in Algorithm 1 in the main paper. We now show how to rewrite the scoring function $S(\mathbf{x}, \boldsymbol{v})$ as its quadratic form w.r.t. $\boldsymbol{v}$. Note that $\gamma(\boldsymbol{v}(\mathbf{y})) = (1-\delta)\boldsymbol{v}(\mathbf{y}) + \delta$. So we can write the first term (i.e., the global mask term) in Eqn. 5 as

$$\begin{aligned} S_{c1}(x, \boldsymbol{v}) &= \sum_{l=1}^{4}\sum_{i=1}^{K_l} \omega_i^l \boldsymbol{v}^T \\ &\quad \Bigg[\sum_{\mathbf{y}} \Big((1-\delta)\boldsymbol{v}(\mathbf{y})\boldsymbol{m}_{l,i}^G(\mathbf{x},\mathbf{y})p(C_i^l|\mathbf{y}) \\ &\quad + \delta \boldsymbol{m}_{l,i}^G(\mathbf{x},\mathbf{y})p(C_i^l|\mathbf{y}) - w_b^l\Big)\Bigg] \\ &= \sum_{l=1}^{4}\sum_{i=1}^{K_l} \Bigg[\boldsymbol{v}^T\Big(\omega_i^l \sum_{\mathbf{y}} (1-\delta)\boldsymbol{m}_{l,i}^G(\mathbf{x},\mathbf{y})p(C_i^l|\mathbf{y})\Big)\boldsymbol{v} \\ &\quad + \boldsymbol{v}^T\Big(\omega_i^l \sum_{\mathbf{y}} \delta \boldsymbol{m}_{l,i}^G(\mathbf{x},\mathbf{y})p(C_i^l|\mathbf{y}) - w_b^l\Big)\Bigg] \end{aligned} \tag{2}$$

The other terms in Eqn. 5 and Eqn. 6 can be written in this form similarly. Summing those terms together, we have the following overall scoring function:

$$S(x, \boldsymbol{v}) = \boldsymbol{v}^T A(\mathbf{x})\boldsymbol{v} + \boldsymbol{v}^T B(\mathbf{x}) \tag{3}$$

where

$$A(\mathbf{x}) = \begin{bmatrix} \omega_i^l \sum_{\mathbf{y}} (1-\delta)\boldsymbol{m}_{l,i}^G(\mathbf{x},\mathbf{y})p(C_i^l|\mathbf{y}) \\ \vdots \\ \mu_i^l \sum_{\mathbf{y}} (1-\delta)\boldsymbol{m}_{l,i}^L(\mathbf{x},\mathbf{y})p(C_i^l|\mathbf{y}) \\ \vdots \\ \omega_{ij}^l \sum_{\mathbf{y}} (1-\delta)(\boldsymbol{m}_{1,i}^G \odot \boldsymbol{m}_{l,j}^G)\cdot\varphi \\ \vdots \\ \mu_{ij}^l \sum_{\mathbf{y}} (1-\delta)(\boldsymbol{m}_{1,i}^L \oplus \boldsymbol{m}_{l,j}^L)\cdot\varphi \\ \vdots \end{bmatrix}, \tag{4}$$

$$B(\mathbf{x}) = \begin{bmatrix} \omega_i^l \sum_{\mathbf{y}} \delta \boldsymbol{m}_{l,i}^G(\mathbf{x},\mathbf{y})p(C_i^l|\mathbf{y}) - w_b^l \\ \vdots \\ \mu_i^l \sum_{\mathbf{y}} \delta \boldsymbol{m}_{l,i}^L(\mathbf{x},\mathbf{y})p(C_i^l|\mathbf{y}) - w_b^l \\ \vdots \\ \omega_{ij}^l \sum_{\mathbf{y}} \delta(\boldsymbol{m}_{1,i}^G \odot \boldsymbol{m}_{l,j}^G)\cdot\varphi - w_b^{1,l} \\ \vdots \\ \mu_{ij}^l \sum_{\mathbf{y}} \delta(\boldsymbol{m}_{1,i}^L \oplus \boldsymbol{m}_{l,j}^L)\cdot\varphi - w_b^{1,l} \\ \vdots \end{bmatrix}, \tag{5}$$

where $\odot$ and $\oplus$ are the element-wise product and addition operators, respectively. Please refer to Eqn. 5 for the definition of the variables.

1.2. Learning with a max-margin formulation

We utilize the max-margin Hough transform [2] framework to train our codebook entry and entry pair weight parameters $\mathbf{w} = \{\omega_i^l, \mu_j^l, \omega_{ij}^l, \mu_{ij}^l\}$. During training, our scoring function $S(\mathbf{x}, \boldsymbol{v})$ can be interpreted as a weighted sum

of $\mathbf{w}$ so it can be trained using the objective function of the max-margin formulation as follows

$$\begin{aligned} \min_{\mathbf{w},b,\xi} \ & \frac{1}{2}\mathbf{w}^T\mathbf{w} + C\sum_{i=1}^{T}\xi_i \\ \text{s.t.} \ & z_i(\mathbf{w}^T D_i + b) \geq 1 - \xi_i, \\ & \mathbf{w} \succcurlyeq 0, \xi_i \geq 0, \forall i = 1, 2, \ldots, T \end{aligned} \tag{6}$$

where z_i is the label of the training sample, and D_i^T is the activation matrix for the *i*-th sample defined as

$$D_i^T = \begin{bmatrix} \boldsymbol{v}^T\Big(\sum_{\mathbf{y}} \gamma(\boldsymbol{v}(\mathbf{y}))\boldsymbol{m}_{l,i}^G(\mathbf{x},\mathbf{y})p(C_i^l|\mathbf{y}) - w_b^l\Big) \\ \vdots \\ \boldsymbol{v}^T\sum_{\mathbf{y}} \gamma(\boldsymbol{v}(\mathbf{y}))\boldsymbol{m}_{l,i}^L(\mathbf{x},\mathbf{y})p(C_i^l|\mathbf{y}) \\ \vdots \\ \boldsymbol{v}^T\Big(\sum_{\mathbf{y}} \gamma(\boldsymbol{v}(\mathbf{y}))(\boldsymbol{m}_{1,i}^G \odot \boldsymbol{m}_{l,j}^G)\cdot\varphi - w_b^{1,l}\Big) \\ \vdots \\ \boldsymbol{v}^T\sum_{\mathbf{y}} \gamma(\boldsymbol{v}(\mathbf{y}))(\boldsymbol{m}_{1,i}^L \oplus \boldsymbol{m}_{l,j}^L)\cdot\varphi \\ \vdots \end{bmatrix} \tag{7}$$

2. More experimental results

In this section we present more results of our method on B3DO and NYU datasets, and provide detailed analysis on some of the typical cases we observe in our experiments.

Our detailed results are presented in Fig. 1. Each row from (a) to (f) corresponds to one specific object instance on a test image. From left to right, we present (1) the RGB frame with ground-truth labellings as available in training. Specifically, these are two bounding boxes marked in green and red respectively. The green bounding box indicates visible parts of the instance, while the red one indicates a whole object including both visible and invisible regions. Note that we use a separate pixelwise labelling for evaluating segmentation performance. The pixelwise labelling was manually generated on the Berkeley 3D Object Dataset [1], while on NYU Depth Version 2 [3] it is readily available. Then, we show (2) votes from different layers for object centroid. From the upper-left corner, we show votes from the object layer (red), nearby context layer (green), occluder layer (yellow), and faraway context layer (blue) in clockwise direction. In (3), the next column, the aggregated votes for the object centroid are shown. After that, we show (4) results with our alternating inference algorithm. The whole object hypothesis is shown as a red bounding box, with image cells inferred as visible highlighted in green. Next, we show (5) the corresponding mask prediction. Finally, (6) the segmentation results based on GrabCut is presented.

2.1. Analysis of examples

The examples presented in Fig. 1 include some of the most representative results on both datasets, and reflects various aspects of our model.

Firstly, we can see the multi-layer representation helps build a more discriminative centroid voting codebook by suppressing false alarms in the object layer. This can be easily observed from examples (a), (b) and (e). Our model allows the object layer to generate concentrated peaks while raising or lowering the underlying terrain using the smeared votes from contextual layers. If a local peak from the object layer lacks support from its surrounding context, the vote will be weakened. On the other hand, if all layers have a consensus, the peak will be strengthened.

Secondly, our model captures the appearance of some occluders and use that information to strengthen local centroid peaks, as well as carving out the shape of an object. This is inherently a very challenging task because the appearance of occluders vary greatly, and our model learns their appearances from only coarse-level labels. Successful examples include (d) and (f). In contrast, although the occluder layer gives roughly correct vote positions in (b), the shape voting breaks down on the desktop occluding the chair. In (c), the chair occluding the door is a bit ambiguous and our model fails to fully recover the correct occlusion pattern.

Finally, our model is also capable of localizing truncated objects, as shown in (e) and there are some similar examples in the main paper.

References

[1] 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. 2, 3

[2] S. Maji and J. Malik. Object detection using a max-margin hough transform. In *CVPR*, 2009. 1

[3] P. K. Nathan Silberman, Derek Hoiem and R. Fergus. Indoor segmentation and support inference from rgbd images. In *ECCV*, 2012. 2, 3

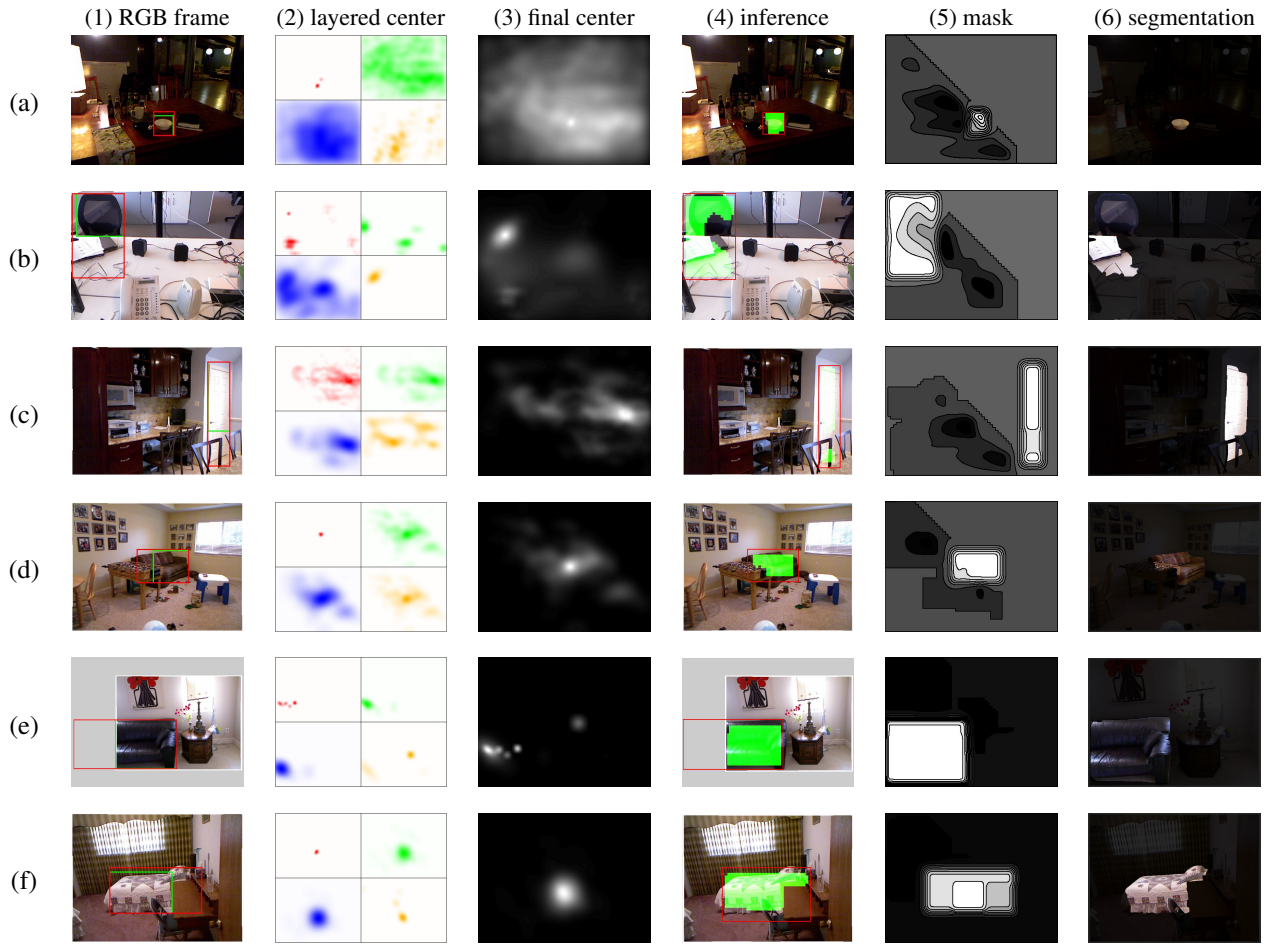

Figure 1. More experimental results of the proposed approach on Berkeley 3D Object Dataset [1] and NYU Depth Dataset [3]. Each row corresponds to a specific instance on a test image. Please refer to Sec. 2 for detailed discussion.