Estimation of Hand-interacting Object Poses with Boundary Guidance

Sin-Yu Fu and I-Chen Lin[0000−0001−9924−4723]

College of Computer Science, National Yang Ming Chiao Tung University, Taiwan

Abstract. Estimating poses of objects that interact with hands is a key task for tangible user interface. It is highly challenging due to its inherence of self- and mutual occlusion. Previous approaches often predict 2D object keypoints from features to establish 2D-3D correspondence during object pose estimation. However, the features for the object and hand are usually intermixed and lead to unreliable output keypoints and inaccurate object pose estimation. To address this issue, we propose a novel Boundary-guided Network (BG-Net). This network takes two cooperative branches for the object and hand. It can effectively capture the object region and utilizes the region as guidance to narrow down the area for keypoint searching. Additionally, we introduce an efficient and effective loss function, min-max boundary distance (MMBD) loss, which restricts the range of estimated keypoint locations. This further benefits the 2D-3D mapping. Experiments demonstrate that the proposed model outperforms related state of the arts for object pose estimation in multiple interactive hand-object benchmarks.

Keywords: Object 6D pose estimation · Hand posture · Region-aware framework.

1 Introduction

The interplay between hands and objects is one of the most frequent actions conducted by human beings, wherein the interactions are affected not only by the hand postures but also by those of target objects. Hence, estimating hand-object poses can help understanding human actions. For the emerging tangible interface and augmented reality, accurately estimating poses of objects that interact with hands is the key issue since such systems generate visual feedback according to estimated 6D object poses (three dimensional rotations and translations, respectively) [19, 21, 8].

To estimate 6D object posture from a single image, several approaches adopt fusing features from a RGB-D image [2, 3]. Since it is easier to access RGB images, recent work pays attention on object pose estimation from a RGB image, and the 3D models of target objects are usually available. One strategy [20, 14] is to directly regress object poses, for instance, to learn a prediction model that can map an input image to the corresponding 6D poses. Although such methods are quite effective, they do not fully leverage the projective geometry of known 3D object models.

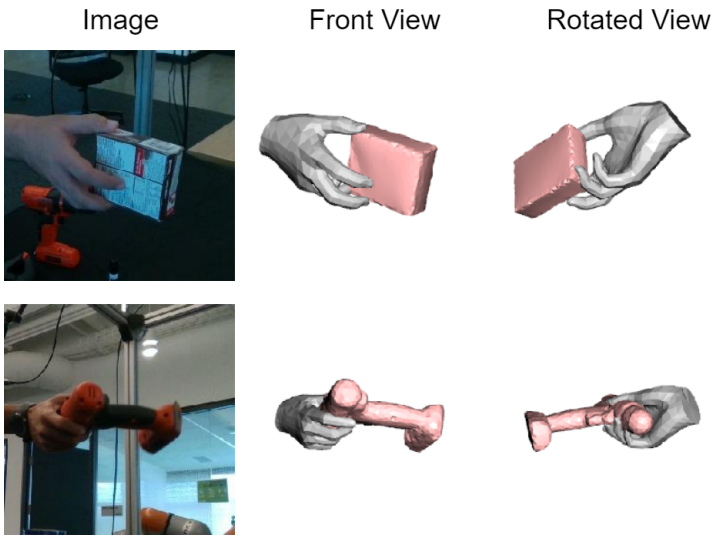

Fig. 1. Object and hand poses estimated by the proposed method from monocular RGB images, respectively.

Another stragety [10, 17, 9, 15, 18] makes use of keypoints of objects. During the inference time, these methods predict the object keypoint locations within the input image. After the 2D keypoint locations are associated with 3D ones, a Perspective-n-Point (PnP) algorithm can be employed to estimate the 6D object pose from these 2D-to-3D correspondences. While recent keypoint-based methods moderately tolerate partial occlusion, their performance usually becomes unstable when the target object is grabbed. When a user holds an object with her (or his) hand, features in the occluded regions often deviate significantly from the object characteristics. Under such circumstances, it is challenging to determine the object boundary and geometric shape, and thereby the accuracy of the output poses degrades. Several recent methods [13, 12] notice the challenge of estimating hand-interacting object poses and take hand-object correlations into account. We found that there is competition between hand and object features when they utilize a single backbone to extract features and keep them in the same space.

To address the aforementioned issues, we propose a novel *Boundary-Guided Network* (BG-Net) to estimate 6D poses of an object that is interacting with a human hand. BG-Net capitalizes on features of the object and hand and learns their correlations. Our network is designed to comprise two branches: one focusing on the object and the other dedicated to the hand. This dual-stream design can avoid feature competition and extract distinctive object and hand features. To mitigate the uncertainties in prediction, we predict the object mask as guidance and enable the network to regress object keypoints from promising regions.

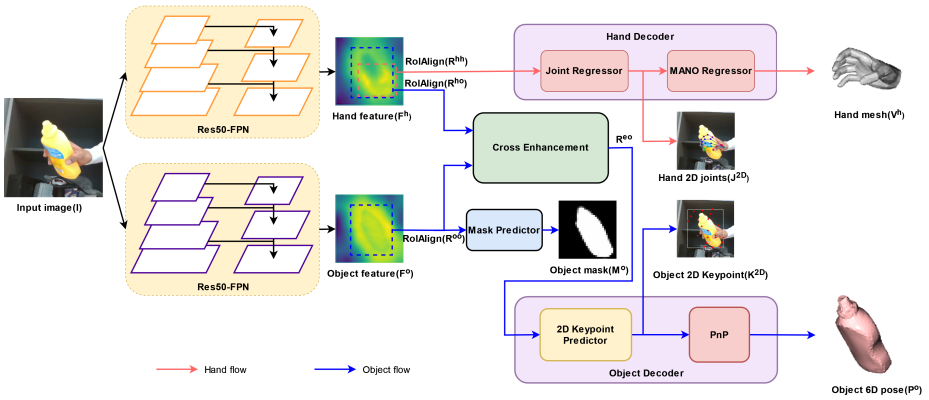

Fig. 2. Overview of the proposed BG-Net.

By adopting attention mechanism [16] with the guided object region, our model learns the correlation of interactions between each pixel of the hand and object with less ambiguity. Even when object areas are partially occluded by the hand during interaction, our framework can still gain additional cues from joint features. Consequently, the posture of the hand can assist in inferring the distribution of object keypoints during occlusion.

Furthermore, given the potential interference from the image background, we observed that when the object features lack clarity, object keypoints tended to gather within the interior of the object and make the following PnP algorithm difficult to estimate adequate poses. Thus, we introduce a novel loss function, *min-max boundary distance* (MMBD) loss. This loss function compels the outmost 2D keypoints to align with the object bounding box, and therefore enhances the reliability of 2D keypoints, even in scenarios where the object is seriously occluded. To verify the effectiveness of our proposed method, we conducted multiple experiments on two popular hand-object interaction datasets: HO3D [4] and Dex-YCB [1]. Experiments demonstrate that our proposed framework reach state-of-the-art performance for pose estimation of hand-interacting objects.

In summary, our contributions include:

- A new framework BG-Net for hand-interacting object pose estimation is proposed. It utilizes object amodal masks as guidance and directs the network attention toward crucial regions. This approach enables better delineation of the geometric shape of the object and leads to precise keypoint prediction.
- Our proposed MMBD loss, aligning the outermost keypoint 2D coordinates to the projected object keypoint bounding box, can effectively reduce the prediction errors caused by occlusions and enhance the accuracy of poses.

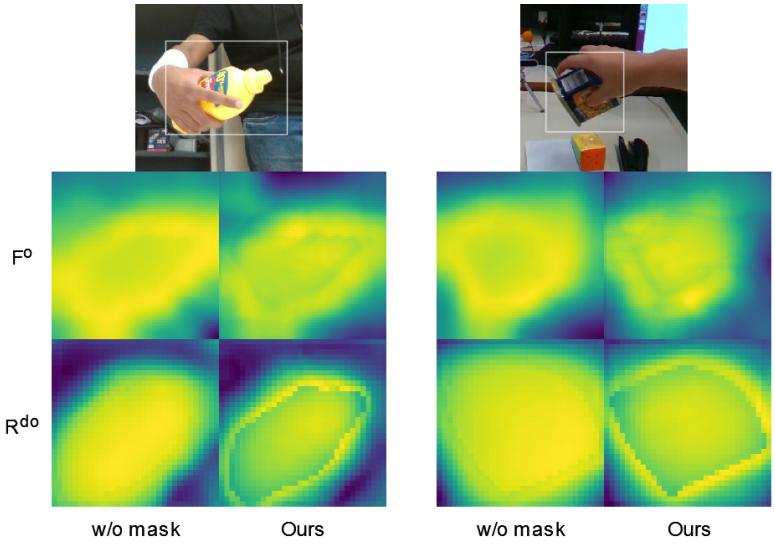

Fig. 3. Visualization of the feature maps F^o, the object feature maps from our backbone, and R^{do}, the intermediate feature maps within the object decoder. R^{do} is the last feature map before the 2D object keypoint regression. With amodal mask prediction (right columns), our network improves its capability of extracting the boundary of the object.

2 Methodology

As illustrated in Fig. 2, our BG-Net consists of two branches to predict hand and object pose respectively. Each branch utilizes its own backbone to extract features, and the object branch includes an additional mask predictor to learn the object contour. Subsequently, the cross enhancement module leverages hand information to provide more cues for occluded object regions. Afterward, the object 2D keypoints K^{2D} is estimated by 2D keypoint predictor, in which the proposed MMBD loss and other loss functions benefit the 2D keypoint alignment during training. In parallel, the 2D joint locations J^{2D} are estimated by the joint regressor. Finally, the hand and object decoders output the 3D hand mesh V^h and the 6D object pose P^o according to 2D joints and keypoints, respectively. The following sections explain each component and the loss functions we used. To ease the explanation, we use F to denote feature maps that contain the same region as the input image, and the superscript h and o represents hand and object, respectively. R denotes feature maps that are cropped and resized after RoIAlign [6]. The first and second superscripts of R denote the source and cropped region, respectively. For example, R^{ho} denotes the feature map cropped from hand feature map F^h and its cropped region is aligned with the predicted object region.

2.1 Backbone and Mask Predictor

As mentioned in the introduction, previous methods [13, 12] used a single-stream backbone to extract both hand and object features. We found that they might compete with each other during feature learning, and it lessens the distinction of these features. As a result, given an RGB image $\mathrm{I} \in \mathbb{R}^{256\times256\times3}$, we employ two separate ResNet-50 models [7] to extract hand and object features. Two distinct Feature Pyramid Networks (FPN) [11] are utilized to fuse the output features from multiple levels within each branch individually. The extracted features for hand and object are denoted as $\mathrm{F}^h \in \mathbb{R}^{64\times64\times256}$ and $\mathrm{F}^o \in \mathbb{R}^{64\times64\times256}$. With our dual-branch design, when the hand and object are partially occluded by each other, the respective network can still correctly acquire information from the regions relevant to their estimation target.

As shown in the middle of Fig. 2, after FPN, we obtain $\mathrm{R}^{hh} \in \mathbb{R}^{32\times32\times256}$ from F^h by RoIAlign according to the hand bounding box. We apply a similar operation to obtain R^{oo} and R^{ho} from F^o and F^h according to the object bounding box. R^{ho}, hand-to-object feature, is used as auxiliary information for object pose estimation in our cross enhancement module in Section 2.2.

Although we have obtained a rough region of the object by the object bounding box, there is still a portion of area, between the object and the boundary of the given bounding box, belongs to the background. This background area can still disturb the pose prediction. Hence, we introduce a mask predictor that not only outputs the visible part of the object but also predicts occluded areas caused by interactions between hand and object.

Specifically, we utilize R^{oo} as input for the mask predictor, which includes four convolutional layers and a sigmoid function to output the object amodal mask M^o. This mask serves a dual purpose: it aids the backbone in focusing on the object and guides subsequent modules to prioritize the object. In other words, such additional prediction compels the feature extractor to acquire adequate features that benefit the visible and occluded area estimation, and that helps our system predict more accurate keypoints around the object boundary. We illustrate how the amodal mask affects the learned features of the object branch in Fig. 3.

2.2 Cross Enhancement

The interaction between hands and objects is highly correlated, allowing visible parts within the image to contribute information to analysis of occluded regions. Previous research [13, 12], has yielded promising results by applying attention mechanisms to enhance object features. However, they employed features extracted from the object bounding box(blue box in Fig. 5.a) as query and intersecting areas between the hand and object(green box in Fig. 5.a) as key and value for the attention module.

Such a design has two limitations. Firstly, when hands and objects do not overlap, this module fails to produce meaningful learning outcomes. Secondly,

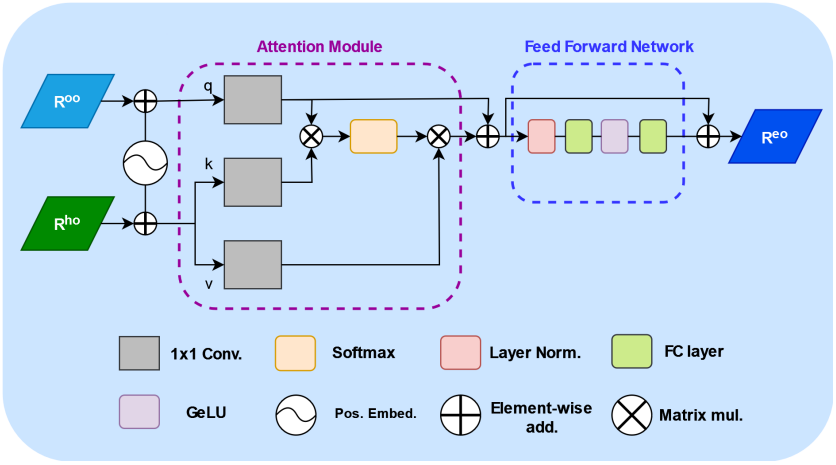

Fig. 4. The structure of our cross enhancement module. This module utilizes the object features R^{oo} (query) to identify its correlation with hand features over the object region R^{ho} (key and value) and outputs the enhanced features R^{eo} accordingly.

due to the permutation-invariance property of Transformer [16], typical approaches often incorporate positional embeddings to retain spatial information. Nonetheless, the aforementioned methods [13, 12] treated the overlapped regions as key and value(Fig. 5.f), which mostly do not align with the query(Fig. 5.d) size generated by the object bounding box. Such spatial misalignment hindered the use of positional embeddings.

In our paper, according to an identical object bounding box, we extract and align features regarding objects, R^{ho}(Fig. 5.e) and R^{oo}(Fig. 5.d) from hand F^{h}(Fig. 5.b) and object F^{o}(Fig. 5.c) features, respectively. This enables our module to persistently serve as a self-attention module even when there are no interactions between hands and objects. Additionally, our query, key, and value are situated in the same spatial domain by this strategy, and it allows us to add positional embeddings and ensures that the process of computing attention scores maintains spatial relationships. The illustration of cross enhancement is shown in Fig. 4.

Specifically, we add learnable positional embeddings to R^{ho} and R^{oo} and employ three separate 1×1 convolutions to derive query q, key k, and value v from R^{oo} and R^{ho}. They are then fed into a multi-head attention module following a feed-forward network, and finally we can output the enhanced object features $\mathrm{R}^{eo} \in \mathbb{R}^{32\times32\times256}$.

2.3 Min-Max Boundary Distance Loss

Based on the cross-enhanced features, our object decoder then predicts projected 2D keypoints of an object. Afterward, the 6D object pose can be estimated from

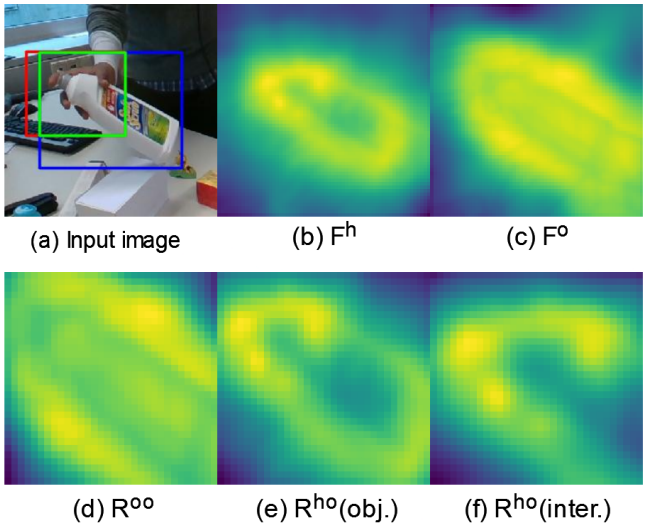

Fig. 5. Illustration of features. (a) Bounding boxes for the hand, hand-object overlap and object are in red, green, and blue , respectively. (b) and (c) Feature maps for hand and object branches after FPN. (d) Features from F^o after RoIAlign according to the blue box in (a). (e) and (f) Features from F^h after RoIAlign according to the blue and green boxes in (a).

2D keypoints by a PnP algorithm. Fig. 6 show the defined keypoints on the 3D bounding box of an object (object keypoint amount, $N^o = 21$ in our case).

During early trials, we observed that the estimated locations of prominent, especially outermost, keypoints tend to shrink toward the object center, as shown in Fig. 7(c). Even with L2 keypoint distance loss, the model took an conservative way to fit in with various cases, including occlusion. Based on these gathered keypoints, the following PnP method then predicts a farther 3D location for the object. If we directly take object depth as a depth loss, we have to incorporate PnP computation into the network and lose the flexiblity of our framework.

Hence, we propose a Min-Max Boundary Distance (MMBD) loss based on projected keypoints to effectively correct the shrunk keypoint problem. This novel loss compares the bounding boxes of projected keypoints. The objective of this loss function is to encourage the outermost predicted 2D keypoints to align with the ground-truth bounding box of projected keypoints. The MMBD loss $\mathcal{L}_{\mathcal{MMBD}}$ is formulated as:

$$\mathcal{L}_{\mathcal{MMBD}} = \sum_{s \in S} (\min_{k \in K} \|k_x - s_x\|_1 + \min_{k \in K} \|k_y - s_y\|_1), \tag{1}$$

where S includes the coordinate of the top-left corner and bottom-right corner of the 2D object bounding box, K indicated the N^o keypoints. The subscript x and y denote the x coordinate and y coordinate respectively. The loss sums the

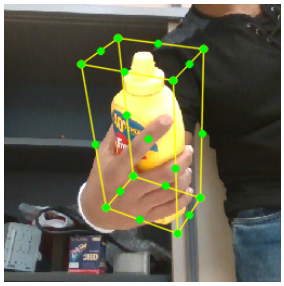

Fig. 6. Visualization of keypoints of an object, including eight corners, twelve midpoint on edges, and one central point of the 3D object bounding box.

distances between the four edges of the ground-truth bounding box projection and their closest predicted 2D keypoints.

As shown in Fig. 7, in the images without using MMBD loss, the outermost keypoints are not aligned with the bounding box and the error of estimated depth is large. By contrast, with the MMBD loss, it can be observed that the outermost keypoints are pulled toward the bounding box. This, in turn, enhances our object pose estimation and reduces the error of the output object depth.

2.4 Decoder and Overall Loss functions

Our hand and object decoder share the same architecture as previous works [13, 12], except that we employ three residual blocks instead of six convolutional blocks in the object decoder to better retain features learned from preceding boundary-guided processes and preserve the contours of the object as shown in Fig. 3.

Besides the MMBD loss mentioned above, multiple loss functions are applied in our framework during training. We apply the binary cross entropy loss $\mathcal{L}_{BCE}$ for our object mask M^o:

$$\mathcal{L}_{mask} = \mathcal{L}_{BCE}(M^o, \hat{M}^o), \tag{2}$$

where $\hat{M}^o$ is the corresponding ground-truth amodal mask. We briefly describe the remaining loss used for hand and object supervision since they are the same as [13, 12]. The overall hand and object loss are as below:

$$\begin{aligned}\mathcal{L}_{hand} =& \alpha_{mano}\mathcal{L}_{mano} + \alpha_{J^{2D}}\mathcal{L}_{J^{2D}} + \\ & \alpha_{J^{3D}}\mathcal{L}_{J^{3D}} + \alpha_{V^h}\mathcal{L}_{V^h},\end{aligned} \tag{3}$$

$$\begin{aligned}\mathcal{L}_{obj} =& \alpha_{MMBD}\mathcal{L}_{MMBD} + \alpha_{p2d}\mathcal{L}_{p2d} + \\ & \alpha_{conf}\mathcal{L}_{conf} + \alpha_{mask}\mathcal{L}_{mask},\end{aligned} \tag{4}$$

where $\mathcal{L}_{mano}$ denotes the L2 loss for MANO parameters θ and β. $\mathcal{L}_{J^{2D}}$ is the L2 loss for 2D joint predictions. $\mathcal{L}_{J^{3D}}$ and $\mathcal{L}_{V^h}$ are the L2 loss for 3D joints and

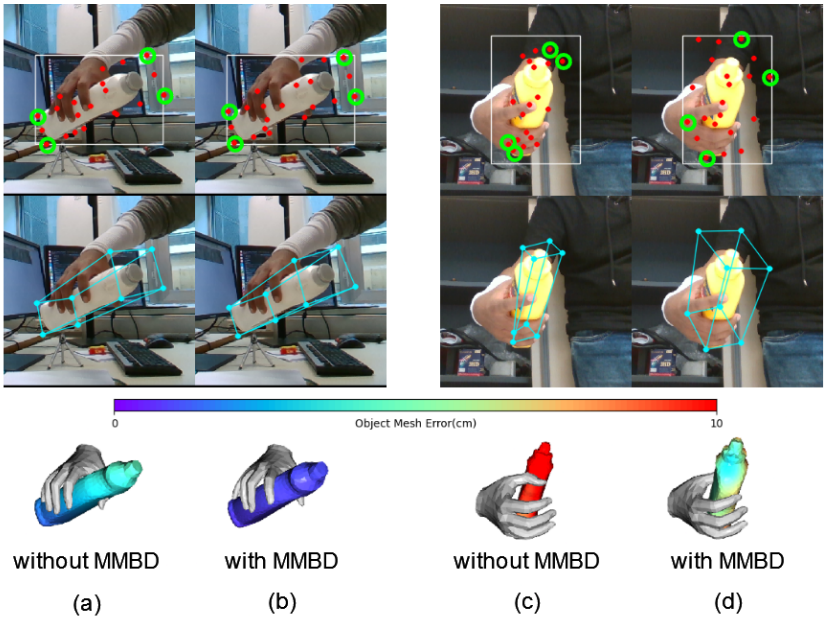

Fig. 7. Visualization of the effect of MMBD loss. (a)(c) are outputs without MMBD; (b)(d) are the corresponding outputs with MMBD loss. Red dots and green circles indicate the predicted object keypoints and the outermost ones. Our MMBD loss significantly assists in aligning the outermost keypoints along the boundaries.

3D hand mesh. $\mathcal{L}_{p2d}$ and $\mathcal{L}_{conf}$ are the L1 loss for 2D object keypoints and their confidence scores.

$\alpha_{mano}, \alpha_{J^{2D}}, \alpha_{J^{3D}}, \alpha_{V^h}, \alpha_{MMBD}, \alpha_{p2d}, \alpha_{conf}$ and α_{mask} are hyper-parameters for balancing each loss. (In our case, two terms of weights for MANO pose and shape are 10 and 10^{-1}. The others are 10^2, 10^4, 10^4, 20, 500, 10^2, 10^2, respectively.) Finally, our total loss function is defined as :

$$\mathcal{L}_{total} = \mathcal{L}_{hand} + \mathcal{L}_{obj}. \tag{5}$$

3 Experiments

3.1 Datasets and Evaluation Metrics

We adopted two popularly used hand-object datasets, HO3D [4] and DexYCB [1] for our experiments. HO3D consists of 66,000 training images and 11,000 testing images, covering 10 different objects. DexYCB is a more challenging dataset, encompassing 582,000 images and featuring interactions with 21 distinct objects. This dataset presents a greater diversity of interactions between hands and objects. We employed the official **s0** split to partition the dataset into training and

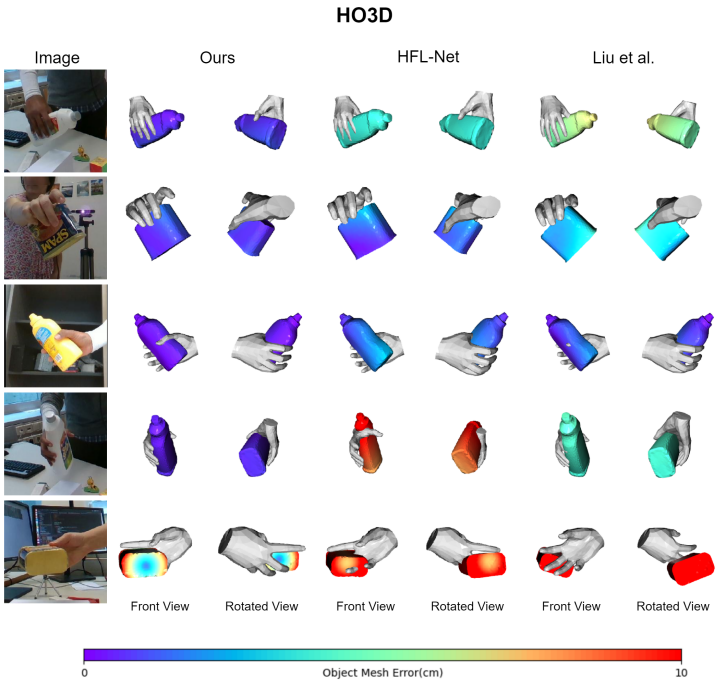

Fig. 8. Qualitative comparison of the proposed BG-Net and state-of-the-art hand-object pose estimation methods [13, 12] on HO3D [4] dataset.

testing sets. We followed the evaluation metrics applied in HFL-Net [12] for fair comparisons. For our primary task, 6D object pose estimation, we apply the popular ADD-0.1D. It evaluates the percentage of object 3D vertices error within 10% of the object diameter of the dataset. For the hand pose estimation, besides evaluating the average joint error, joint error with procrustes alignment (PA) is another popular metric. It first aligns the centroids, scales and orientations of two shapes and evaluates the differences.

3.2 Implementation Details

We cropped and resized the input images from the dataset to 256×256 pixels, centered around the midpoint of the hand and object. During training on the HO3D dataset, we employed the Adam optimizer with an initial learning rate of 1e-4 and a weight decay rate of 0.7 every 10 epochs. We set the batch size as 32 and trained the model with 60 epochs on a single NVIDIA RTX4090 GPU. To augment the data, we utilized techniques such as color jittering, random rotation, translation, and scaling. Please refer to the supplementary document for other details. The codes will be available from the project page of the authors.

Methods	ADD-0.1D↑ cleanser	bottle	can	avg
Liu et al. [13]	88.1	61.9	53.0	67.7
HFL-Net [12]	81.4	**87.5**	52.2	73.3
Ours	**94.7**	80.2	**65.8**	**80.2**

Table 1. Comparison with state-of-the-art methods on object pose estimation on HO3D [4] dataset. "avg" denotes the average among all object categories. Our method achieves the best performance on average.

Methods	Error(PA)↓ Joint	Mesh	F-score↑ F@5	F@15
Liu et al. [13]	10.1	9.7	53.2	95.2
ArtiBoost [22]	11.4	10.9	48.8	94.4
Keypoint Trans. [5]	10.8	-	-	-
HFL-Net [12]	8.9	8.7	57.5	96.5
Ours	9.7	9.7	53.1	95.3

Table 2. Comparison with state-of-the-art methods on hand pose estimation on HO3D [4] dataset. Even though our goal is object pose estimation, our estimated hand poses are comparable to those of related methods.

3.3 Comparisons with State-of-the-art Methods

HO3D Our work emphasizes 6D object pose estimation in an interactive scenario, and the comparison with state of the arts is shown in Table 1. Our results achieved 80.2% accuracy on ADD-0.1D, surpassing the second-best method by 6.9%. It demonstrates the effectiveness of object pose estimation through our boundary-guided network. Qualitative comparisons are shown in Figure 8. Even in cases where a large portion of hands or objects are occluded, or when object features are ambiguous, our model generates a more precise object pose compared to that of [13, 12].

Even though the proposed work focuses on hand-interacting object pose estimation, our BG-Net can still estimate accurate hand poses comparable to recent methods as shown in Table 2. Although our approach did not achieve the best performance on hand posture, our method still outperforms Liu et al. [13], which has a similar hand pose estimation structure to ours.

Methods	ADD-0.1D(s)↑	Joint↓	Joint(PA)↓
Liu et al. [13]	29.8	15.27	6.58
HFL-Net [12]	30.2	**12.56**	**5.47**
Ours	**46.2**	12.7	5.53

Table 3. Comparison with state-of-the-art methods on Dex-YCB [1] dataset. Our method achieves competitive results with the best approach [12] on hand pose estimation and outperforms the others on object pose estimation by a large margin.

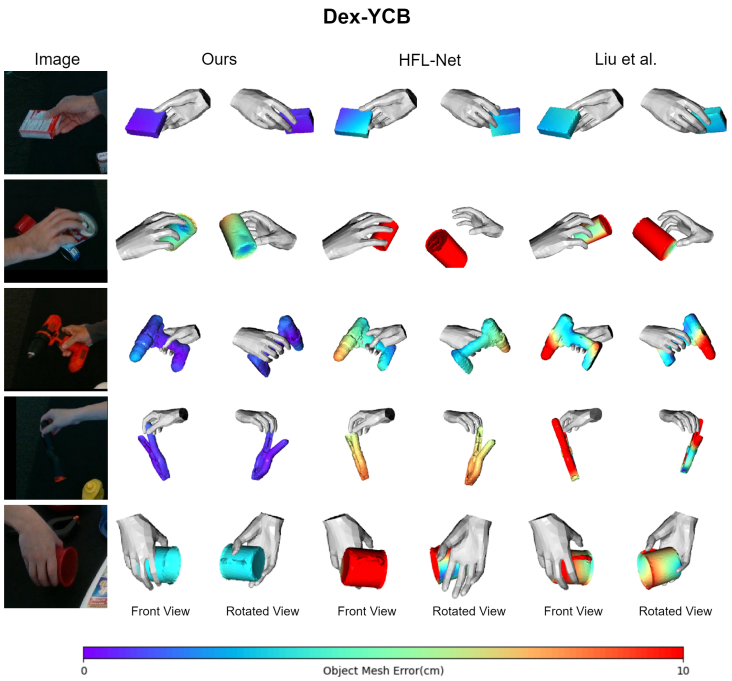

Fig. 9. Qualitative comparison of the proposed BG-Net and state-of-the-art hand-object pose estimation methods [13, 12] on DexYCB [1] dataset.

Dex-YCB Table 3 summarize results of object and hand pose estimation on Dex-YCB dataset. The errors of joint estimation by our method with and without Procrustes Alignment are 12.7mm and 5.53mm, respectively. They are on a par with state-of-the-art approaches. For object pose estimation, our results reach 46.2% on ADD-0.1D(s), substantially outperforming HFL-Net [12] by 16%. We attribute this advance to our double-stream architecture and amodal mask in tackling the challenges of learning from such a diverse object dataset, where twenty one objects are included. Our approach allows the object backbone to concentrate solely on extracting object-specific features, while the mask aids in learning object boundaries, and our MMBD loss helps correct improperly estimated depth of an object. Qualitative comparisons are shown in Fig. 9.

Our framework has shown its advantage of estimating hand-interacting object poses and it employs 59,073,760 trainable parameters, while there are 46,080,659 and 34,480,019 trainable parameters in HFL-Net [12] and Liu et al. [13], respectively.

3.4 Ablation Study

To verify the effectiveness of our proposed methods, we conducted ablation study on the HO3D [4] dataset.

Methods	ADD-0.1D↑ cleanser	bottle	can	avg
w/o mask	93.3	**80.7**	59.7	77.6
w/o cross enhance.	93.6	77.0	57.9	76.3
w/o residual blocks	93.2	73.3	60.4	75.7
w/o MMBD loss	92.6	72.9	60.2	75.2
Ours	**94.7**	80.2	**65.8**	**80.2**

Table 4. Ablation study on the major components and MMBD loss.

Effectiveness of the Major Components and MMBD Loss As our designed approach mainly focuses on enhancing object pose estimation, we report the ADD-0.1D in Table 4. In the first experiment, we removed the mask predictor. The result indicates that the absence of the mask decreases the accuracy in pose estimation. The visualization in Fig. 3 shows that prediction with the amodal mask accentuates the object boundaries in feature maps. In the second experiment, we removed the cross enhancement module, and no additional information from hand features is provided. It results in a 2.6% performance drop. It manifests that the hand poses can provide useful features for hand-interacting object pose estimation.

For the third experiment, we replaced the three residual blocks in the object decoder with six convolutional layers, similar to [13, 12]. The result reveals that residual blocks play a significant role in preserving previously learned features. They prevent losing the cues provided by the contours of the object mask and clues from hand features. The fourth experiment and Fig. 7 validate the proposed MMBD loss. They show that without MMBD loss, the performance substantially degrades. These experiments demonstrate that the employed components and MMBD loss indeed benefit the pose estimation performance for objects that are partially occluded by a hand.

Effectiveness of Double-Stream Backbone Wile related methods [13, 12] took a single-stream backbone, we adapted a double-stream backbone. To verify the effectiveness of our double-stream backbone, we replaced the architecture of our model with a shared ResNet-50 with FPN for both hand branch and object branch while keeping other components unchanged. Table 5 shows that applying our double-stream backbone, combined with the proposed modules and loss functions, provides a 7.8% improvement in object pose estimation compared to a framework adopting the single-stream backbone.

Additionally, there is a 0.3mm enhancement in average hand joint and mesh errors. This outcome demonstrates that using two separate backbones to learn hand and object features enables an easier learning process for respective targets without interference. The mask predictor also better guides the object backbone in learning object boundaries.

Methods	ADD-0.1D↑	Joint↓	Mesh↓
Single-stream	72.4	10.0	10.0
Ours	**80.2**	**9.7**	**9.7**

Table 5. Ablation study on single-stream and double-stream architectures.

	ADD-0.1D↑			
Methods	cleanser	bottle	can	avg
intersect.	93.8	74.3	56.5	74.9
intersect. + pos.	93.3	71.0	62.3	75.5
object bbox.	91.8	75.2	62.3	76.4
Ours	**94.7**	**80.2**	**65.8**	**80.2**

Table 6. Ablation study on different settings for R^{ho} in cross enhancement. "Intersect." and "object bbox." denote that we use the hand-object overlapped region(green box in Figure 5.a) or the object bounding box(blue box in Figure 5.a) to produce R^{ho}. "Pos." indicates that positional embeddings are appended.

Different Settings for R^{ho} in Cross Enhancement Table 6 compares the results of using different bounding boxes to produce R^{ho}(Fig. 5.e & Fig. 5.f) in cross enhancement, along with the incorporation of positional embeddings. In the first and second settings, R^{ho} is extracted from the overlapping region of the hand and the object(green box in Fig. 5.a), while the second setting additionally integrates positional embeddings. It can be observed that the incorporation of positional embeddings in such settings merely gains 0.6% improvement on ADD-0.1D. It is worth noting that the performance of the first and second settings is not as good as when we do not employ cross enhancement in our model (the second row in Table 4). This suggests that when there is a spatial inconsistency among the key, value, and query in transformer, attention mechanism does not successfully benefit the model.

By contrast, in the third and fourth settings, R^{ho} is extracted based on the object bounding box(blue box in Fig. 5.a). Compared to the third setting, the fourth setting includes positional embeddings and exhibits a 3.8% enhancement on ADD-0.1D. This underscores the significance of positional embeddings for preserving spatial information, when the key, value, and query share identical space on feature maps.

4 Conclusion

This paper presents the Boundary-Guided Network (BG-Net) for 6D post estimation of objects interacting with a hand. This framework adapts a double-stream framework to enhance the object and hand feature distinction respectively. In this framework, we estimate and utilize the object amodal mask to guide the object branch in learning object-specific features and identifying object boundaries for accurate prediction of 2D object keypoints. Moreover, we propose a novel min-max boundary distance (MMBD) loss. It tackles the gathering is-

sue of predicted keypoints and therefore reduces the depth error of the output object pose. Experiments demonstrate that our method surpasses state-of-the-art methods on hand-interacting object pose estimation, and it also achieves comparable performance in hand pose estimation.

References

1. Chao, Y.W., Yang, W., Xiang, Y., Molchanov, P., Handa, A., Tremblay, J., Narang, Y.S., Van Wyk, K., Iqbal, U., Birchfield, S., et al.: Dexycb: A benchmark for capturing hand grasping of objects. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9044–9053 (2021)
2. Cheng, Y., Zhu, H., Sun, Y., Acar, C., Jing, W., Wu, Y., Li, L., Tan, C., Lim, J.: 6d pose estimation with correlation fusion. In: International Conference on Pattern Recognition (ICPR). pp. 2988–2994 (2021)
3. Feng, H., Zhang, L., Yang, X., Liu, Z.: Mixedfusion: 6d object pose estimation from decoupled rgb-depth features. In: International Conference on Pattern Recognition (ICPR). pp. 685–691 (2021)
4. Hampali, S., Rad, M., Oberweger, M., Lepetit, V.: Honnotate: A method for 3d annotation of hand and object poses. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 3196–3206 (2020)
5. Hampali, S., Sarkar, S.D., Rad, M., Lepetit, V.: Keypoint transformer: Solving joint identification in challenging hands and object interactions for accurate 3d pose estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 11090–11100 (2022)
6. He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask r-cnn. In: Proceedings of the IEEE international conference on computer vision. pp. 2961–2969 (2017)
7. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)
8. Hsu, W.T., Lin, I.C.: Associating real objects with virtual models for vr interaction. In: SIGGRAPH Asia 2021 Posters. p. 24:1–2 (2021)
9. Hu, Y., Hugonot, J., Fua, P., Salzmann, M.: Segmentation-driven 6d object pose estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3385–3394 (2019)
10. Huang, W.L., Hung, C.Y., Lin, I.C.: Confidence-based 6d object pose estimation. IEEE Transactions on Multimedia **24**, 3025–3035 (2022)
11. Lin, T.Y., Dollár, P., Girshick, R., He, K., Hariharan, B., Belongie, S.: Feature pyramid networks for object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2117–2125 (2017)
12. Lin, Z., Ding, C., Yao, H., Kuang, Z., Huang, S.: Harmonious feature learning for interactive hand-object pose estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12989–12998 (2023)
13. Liu, S., Jiang, H., Xu, J., Liu, S., Wang, X.: Semi-supervised 3d hand-object poses estimation with interactions in time. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 14687–14697 (2021)
14. Mo, N., Gan, W., Yokoya, N., Chen, S.: Es6d: A computation efficient and symmetry-aware 6d pose regression framework. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6718–6727 (2022)

15. Peng, S., Liu, Y., Huang, Q., Zhou, X., Bao, H.: Pvnet: Pixel-wise voting network for 6dof pose estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4561–4570 (2019)
16. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. Advances in neural information processing systems **30** (2017)
17. Wang, D., Zhou, G., Yan, Y., Chen, H., Chen, Q.: Geopose: Dense reconstruction guided 6d object pose estimation with geometric consistency. IEEE Transactions on Multimedia **24**, 4394–4408 (2021)
18. Wang, G., Manhardt, F., Tombari, F., Ji, X.: Gdr-net: Geometry-guided direct regression network for monocular 6d object pose estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 16611–16621 (2021)
19. Wu, L.C., Lin, I.C., Tsai, M.H.: Augmented reality instruction for object assembly based on markerless tracking. In: Proceedings of the 20th ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games. p. 95–102. I3D '16, Association for Computing Machinery (2016)
20. Xiang, Y., Schmidt, T., Narayanan, V., Fox, D.: Posecnn: A convolutional neural network for 6d object pose estimation in cluttered scenes. arXiv preprint arXiv:1711.00199 (2017)
21. Yamaguchi, M., Mori, S., Mohr, P., Tatzgern, M., Stanescu, A., Saito, H., Kalkofen, D.: Video-annotated augmented reality assembly tutorials. In: Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology. p. 1010–1022. UIST '20, Association for Computing Machinery (2020)
22. Yang, L., Li, K., Zhan, X., Lv, J., Xu, W., Li, J., Lu, C.: Artiboost: Boosting articulated 3d hand-object pose estimation via online exploration and synthesis. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2750–2760 (2022)