ViHOPE: Visuotactile In-Hand Object 6D Pose Estimation with Shape Completion

Hongyu Li *
Brown University
Providence, RI
hongyu@brown.edu

Snehal Dikhale, Soshi Iba, and Nawid Jamali
Honda Research Institute USA
San Jose, CA
{snehalsubhash_dikhale, siba, njamali}@honda-ri.com

Abstract

In this paper, we present ViHOPE, a framework for estimating the 6D pose of an in-hand object using visuotactile perception. In our framework, we employ a conditional Generative Adversarial Network to complete the shape of an in-hand object based on volumetric representation. This completed shape is then utilized to estimate the 6D pose, demonstrating that our approach outperforms prior methods. We assess the effectiveness of our model by training and testing on a synthetic dataset. In both the visuotactile shape completion task and the visuotactile pose estimation task, our approach outperforms the state-of-the-art by a significant margin. We present our pivotal lesson learned: the value of explicitly completing object shapes. Furthermore, we ablate our framework to confirm gains from explicit shape completion and demonstrate that our framework produces models that are robust to sim-to-real transfer on a real-world robot platform.

1 Introduction

An accurate 6D pose can benefit many applications, such as robotic manipulation (Qin et al., 2022; Chen et al., 2022a), autonomous driving (Geiger et al., 2012), and social navigation (Chen et al.). Recent advances in deep learning techniques promising results (Wang et al., 2019a; Chen et al., 2022b; Hu et al., 2019). These methods, when combined with iterative refinement (Besl and McKay, 1992; Wang et al., 2019a; Li et al., 2018), leverage the object's 3D model to obtain a more accurate estimate. However, many methods find the presence of intermediate to extreme occlusions challenging, particularly in dexterous manipulation where the object is being held, grasped, or sometimes completely obscured by the robot hand.

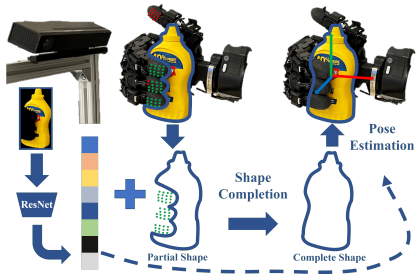


Figure 1: **A high-level overview of the proposed framework**. The green dots represent taxels of the tactile sensors that are in contact. RGB image and depth map are retrieved from an RGB-D sensor for object segmentation and visual feature extraction.

In an effort to improve the quality of reception, researchers have explored the use of tactile sensors(Bimbo et al., 2016; Dikhale et al., 2022; Villalonga et al., 2021). Villalonga et al. (2021) leverage a template-based approach to match the visuotactile observation with the rendered shapes. Dikhale et al. (2022) use visuotactile data and utilize an end-to-end deep neural network and address the 6D pose estimation as a regression problem. However, they do not explicitly leverage the 3D geometry of the object.

*This work was completed when Hongyu Li was an intern at Honda Research Institute USA, Inc.

NeurIPS 2023 Workshop on Touch Processing: a new Sensing Modality for AI .

To this end, we introduce ViHOPE—Visuotactile In-Hand Object Pose Estimator (Fig. 1). ViHOPE takes visuotactile observation as input and explicitly optimizes the object shape while estimating the pose. We hypothesize that jointly optimizing the shape and pose of the object will provide more accurate estimates of the 6D pose of the object. Additionally, during deployment, by providing an estimate of the complete shape of the in-hand object, it increases explainability and potentially expands the range of applications, such as grasping (Varley et al., 2017). Specifically, we first train an autoencoder to capture the geometry prior of the object and encode the object shape into a latent space. We then leverage a GAN to transfer the latent code from the partial shape space to that of the complete shape. We then use the estimated complete latent code, and the visual feature to estimate the 6D pose.

We conduct experiments on a synthetic dataset by Dikhale et al. (2022) and a physical robot platform. We evaluate the performance of ViHOPE on two tasks: shape completion and pose estimation. In the shape completion task, we show improved performance by a large margin compared with the prior work (Watkins-Valls et al., 2019), where our model faithfully reconstructs the complete shape even under heavy occlusion. In the pose estimation task, we demonstrate our model outperforms the state-of-the-art visuotactile pose estimator (Dikhale et al., 2022). We also present results of ablation studies, in which we remove the shape completion module to confirm the effectiveness and robustness of our approach, which explicitly optimizes shape.

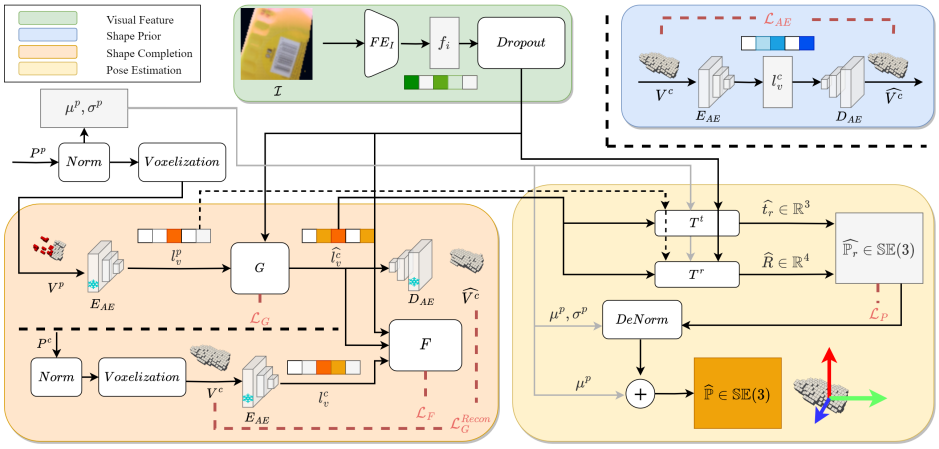


Figure 2: The proposed framework consists of three phases. Phase one trains the autoencoder (the blue box) in isolation to learn a shape prior. Phase two trains the shape completion module (the orange box) with frozen weights for E_{AE} and D_{AE} from phase one. Phase three uses the completed shape (in latent space) to estimate the object's pose in an end-to-end manner.

2 Methodology

Our objective is to determine the 6D pose of an object based on an image, its corresponding depth map, and tactile feedback from the robot hand. We assume the 3D model of the object is available during training.

The proposed visuotactile pose estimation framework consists of: a visuotactile shape completion module and a pose estimation module. We provide a high-level overview in Fig. 2. The shape completion module explicitly optimizes the object's shape from a partial observation of the object from the vision and tactile sensors (Section 6.1). The pose estimation module uses the output of the shape completion module, in the form of latent code, along with the visual features and point cloud normalization scalars to estimate the 6D pose of the object (Section 6.2).

3 Experiments

Our experiments are designed to assess the efficacy of our proposed framework: 1) in the level of accuracy achieved by our shape completion module in reconstructing object shapes; 2) the impact of explicit shape completion on the quality of 6D pose estimation; 3) the contribution of each component

of the framework to pose estimation accuracy; and 4) the sensitivity of the framework to variations in occlusion levels and tactile contact points. This section outlines our experimental setup including model training and performance evaluation. The model was trained and tested on a synthetic dataset, and then transferred to a real physical robot to study the framework's robustness in sim-to-real transfers. We refer readers to our supplementary Sec. 6.3 for more experiment design details.

4 Results

4.1 Shape Completion

We compare our shape completion module with the seminal work from Watkins-Valls et al. (Watkins-Valls et al., 2019) using two metrics: Intersection over Union (IoU) and Chamfer Distance (CD). We implement their proposed model (Varley et al., 2017; Watkins-Valls et al., 2019) using PyTorch with a minor modification. To ensure a fair evaluation, a consistent voxel occupancy grid resolution was used. Our implementation utilizes a 32^3 voxel grid for input and output, instead of the 40^3 voxel grid utilized in the original work. This modification allowed seamless integration of the shape completion module into our established pipeline. Results in Table 1 show a 265.5% increase in IoU and an 88% decrease in CD, demonstrating the robustness of our model under challenging conditions.

Table 1: Quantitative shape completion result.

Method	IoU ↑	CD ↓
Watkins-Valls et al. (2019)	0.142	0.125
Vision Only	0.341	0.042
ViHOPE (Ours)	**0.519**	**0.015**

We also evaluate the performance of the shape completion model by removing the tactile modality (Vision Only). The result suggests the significance of tactile modality for completing the shape of an in-hand object. We provide additional qualitative results in Fig. 5.

4.2 6D Pose Estimation

Table 2: A comparison of our approach with the state-of-the-art is presented in the first half of the table, followed by the results of our ablation studies in the second half. The modalities used by the methods are highlighted in the second column.

Method	Modalities			Position Error (cm) ↓	Angular Error (deg) ↓	ADD (cm) ↓	ADD-S (cm) ↓
	RGB	point cloud	tactile				
PoseCNN (Xiang et al., 2018)	✓			6.146 ± 0.023	10.897 ± 0.082	-	-
DenseFusion (Wang et al., 2019a)	✓	✓		0.640 ± 0.004	9.969 ± 0.117	1.037 ± 0.008	0.571 ± 0.003
ViTa (Dikhale et al., 2022)	✓	✓	✓	0.299 ± 0.002	8.074 ± 0.105	0.825 ± 0.007	0.474 ± 0.002
No-Shape-Completion	✓	✓	✓	0.258 ± 0.018	4.104 ± 0.049	0.400 ± 0.019	0.282 ± 0.018
No-Vis-GAN	✓	✓	✓	1.613 ± 0.333	4.132 ± 0.058	1.745 ± 0.333	1.623 ± 0.332
No-Vis-MLP	✓	✓	✓	$\mathbf{0.156 \pm 0.001}$	5.677 ± 0.083	0.403 ± 0.004	$\mathbf{0.214 \pm 0.001}$
No-Tactile	✓	✓		1.614 ± 0.015	17.228 ± 0.165	2.023 ± 0.017	0.774 ± 0.009
No-Point-Cloud	✓		✓	0.861 ± 0.010	14.245 ± 0.096	1.478 ± 0.011	0.655 ± 0.009
ViHOPE (Ours)	✓	✓	✓	0.194 ± 0.009	$\mathbf{2.873 \pm 0.036}$	$\mathbf{0.298 \pm 0.009}$	$\mathbf{0.214 \pm 0.008}$

4.2.1 Comparsion with state-of-the-art

We compare the performance of our pose estimation network with two seminal works: i) the visuotactile-based estimator (ViTa) (Dikhale et al., 2022), and ii) the RGB-D-based estimator, DenseFusion (Wang et al., 2019a). In Fig. 3, we provide a per-instance numerical analysis on 11 YCB objects. Our approach outperforms ViTa and DenseFusion by a large margin on each object with statistical significance, suggesting explicit shape optimization is more effective compared to implicit methods.

4.2.2 Performance under different occlusion level

We evaluate the performance of our model under different levels of occlusion. The results of our evaluation are presented in Fig. 4, where it can be observed that the model demonstrates a robust performance in the presence of increasing levels of occlusion. It is worth noting that our method maintains its performance as compared to ViTa, which suggests that our model is able to handle occlusion effectively and still produce competitive performance. We further evaluate the performance by removing the tactile modality (Vision Only). The result confirms the significant contribution of tactile modality under severe occlusions.

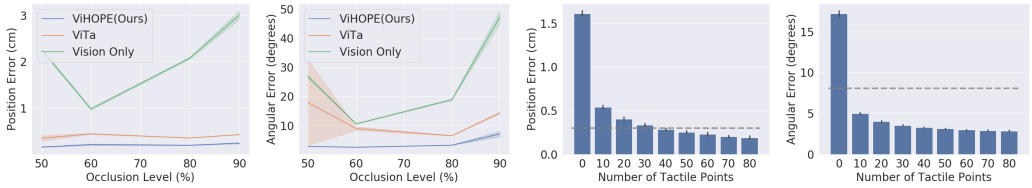


Figure 4: **Left**: Performance under different levels of occlusion. **Right**: The performance of our approach under different levels of tactile contact points. The dashed gray line represents the performance of ViTa using 1000 tactile contact points.

4.2.3 Performance under different tactile points

The spatial resolution of real-world tactile sensors can vary significantly. Therefore, we analyze the performance of our model under different tactile contact points (Fig. 4). It is noteworthy that our model, which was trained with 80 tactile points, demonstrates robust performance when presented with a reduced number of points. As expected, the performance of the model deteriorates as the number of tactile points is reduced and drops significantly when the tactile modality is removed entirely. It is worth noting that compared to ViTa, which requires tactile input, our model can still provide pose estimation even without tactile feedback, although with degraded performance. Upon analyzing the position error, we observe that, up to a reduction of 40 tactile points, our model outperforms ViTa, which uses 1000 points. The angular error results show that our model consistently outperforms ViTa.

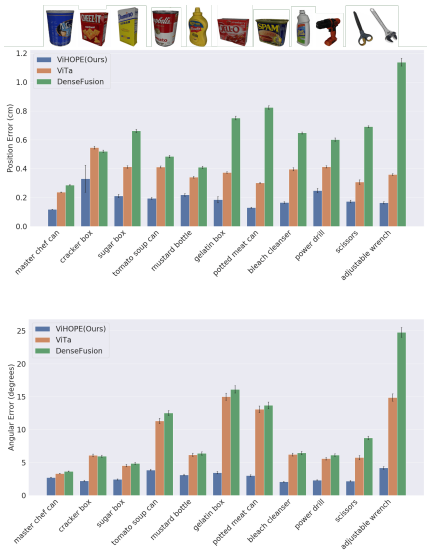


Figure 3: Performance comparison with the state-of-the-art.

4.2.4 Ablation Studies

We perform ablation studies to examine the effectiveness of our design choices (Table 2). A more detailed analysis of our ablation study results is in supplementary Sec. 6.4.

4.2.5 Real-world experiment

We validate the sim-to-real robustness of our framework using our robot platform, with a subset of YCB objects that could be grasped by the Allegro Hand. The hand moves along a trajectory covering different poses. We apply a novel hand-grasping pose that doesn't exist in our training dataset, which shows our model's ability to generalize. Our model efficiently operates at 11.2ms / 89Hz using an NVIDIA RTX 6000, thus capable of real-time deployment. We provide a visualization of the result in supplementary Sec. 6.6.

5 Conclusion

We presented ViHOPE, a novel framework for estimating the 6D pose of an in-hand object using visuotactile perception. We focus on instance-level pose estimation in this paper. In the future, we are interested in extending our pose estimator to work on more challenging scenarios, for example, lack of annotated data Zhang et al. (2022), and category-level pose estimation Wang et al. (2019b). Another interesting future direction is incorporating other sensory information, such as pressure.

6 Supplementary Material

6.1 Shape Completion

Our framework starts from a volumetric shape completion module, which consists of two steps: 1) learning a shape prior from the object model, i.e., full observation, 2) recovering the complete shape by learning a mapping from partial observation to a complete one, in the latent space.

6.1.1 Learning the shape prior

We first train an autoencoder (the blue box in Fig. 2) to capture the shape prior of the object. To capture the prior under different orientations, we apply random $SO(3)$ rotations to the object $\mathcal{O}$, and voxelize it to form an augmented dataset. The autoencoder consists of two components: an encoder E_{AE} and a decoder D_{AE}. The encoder encodes the input voxel occupancy grid $V \in \mathbb{R}^{n_x \times n_y \times n_z}$ into a latent code $l_v \in \mathbb{R}^{n_l}$ using a set of 3D convolutional layers. The decoder recovers the original voxel occupancy grid from the latent code as $\widehat{V} \in \mathbb{R}^{n_x \times n_y \times n_z}$ using symmetrical 3D deconvolutional layers. We apply a batch normalization layer after each layer, followed by a ReLU activation function. We set $n_x = n_y = n_z = 32$ and $n_l = 128$, empirically, and optimize the autoencoder model using the Jaccard index loss

$$\mathcal{L}_{AE} = 1 - \frac{|V^c \cap D_{AE}(E_{AE}(V^c))|}{|V^c \cup D_{AE}(E_{AE}(V^c))|}. \tag{1}$$

Note, higher voxel grid resolution can improve accuracy but comes with increased computation and memory costs. Some works have explored methods to mitigate these costs (Tatarchenko et al., 2017; Li et al.); however, such optimization techniques are beyond the scope of this work.

6.1.2 Recovering the complete shape

After training the autoencoder to convergence, we optimize the entire shape completion model (the orange box in Fig. 2). The encoder E_{AE} and the decoder D_{AE} of the shape completion module are initialized with the weights from the previous step and are frozen in this step (Wu et al., 2020).

The inputs to the shape completion model consist of occluded observational data, that is, a partial visuotactile point cloud $P^p \in \mathbb{R}^{3 \times N}$ and an RGB image $\mathcal{I} \in \mathbb{R}^{H \times W \times 3}$. We first obtain a semantic segmentation mask $\mathcal{S}$ of the object $\mathcal{O}$ using the RGB image $\mathcal{I}$. We then segment the depth map $\mathcal{D}$ using the mask $\mathcal{S}$ as $\mathcal{D}^p$ and transform $\mathcal{D}^p$ to its respective point cloud form $P^{\mathcal{D}}$. Combining the point cloud $P^{\mathcal{D}}$ with the tactile point cloud $P^{\mathcal{T}}$, we obtain the observation $P^p = P^{\mathcal{D}} \cup P^{\mathcal{T}}$ of the object $\mathcal{O}$. To improve the training efficiency, we first normalize the input partial point cloud P^p using its centroid $\mu^p \in \mathbb{R}^3$ and the farthest distance from the centroid $\sigma^p \in \mathbb{R}$. The normalized point cloud is voxelized as V^p and encoded using the frozen encoder E_{AE} into a partial latent vector l_v^p in the partial latent space $\mathcal{M}^p$. During training, we use the ground-truth complete voxel grid V^c to calculate the losses.

To recover the complete shape from the partial observation, we seek a mapping for shape latent code from the partial latent space to the complete latent space $\mathcal{M}^p \mapsto \mathcal{M}^c$. Inspired by Wu et al. (2020), we find the mapping using a cGAN (Mirza and Osindero, 2014). We condition the cGAN on the partial latent vector l_v^p and the visual feature f_i, which is extracted from the input image $\mathcal{I}$ using a pretrained ResNet (He et al., 2016) feature extractor FE_I. FE_I is finetuned during the training period and regularized by a dropout layer. The dropout layer is only activated during training and deactivated during testing. Therefore our generator is trained as $G : (\mathcal{M}^p, p(f_i)) \mapsto \mathcal{M}^c$. We pass the estimated complete latent vector $\widehat{l_v^c} \in \mathcal{M}^c$ from the generator to the discriminator F along with the ground-truth complete latent vector l_v^c, obtained by applying the same procedure on the ground-truth complete point cloud P^c. Having the same conditioning as G, the discriminator F is trained as a binary classifier to distinguish the real complete latent vector l_v^c and the fake complete latent vector $\widehat{l_v^c}$. At the end, we feed the estimated complete latent vector $\widehat{l_v^c}$ into the frozen decoder D_{AE} to reconstruct the complete shape $\widehat{V^c}$.

The shape completion model is optimized using three losses: the discriminator loss $\mathcal{L}_F$, the generator loss $\mathcal{L}_G$, and the reconstruction loss $\mathcal{L}_G^{Recon}$. The discriminator loss penalizes the discriminator if it can't distinguish the real and fake latent vectors. On the other side of this min-max game, the

generator loss encourages the generator to fool the discriminator. We leverage the loss functions from LSGAN (Mao et al., 2017) to stabilize the training

$$\begin{aligned}\mathcal{L}_F = &\mathbb{E}_{V^c,\mathcal{I}}[F_{cGAN}(E_{AE}(V^c), FE_I(\mathcal{I})) - 1]^2 \\ &+ \mathbb{E}_{V^p,\mathcal{I}}[F_{cGAN}(G_{cGAN}(E_{AE}(V^p), FE_I(\mathcal{I})))]^2\end{aligned} \tag{2}$$

$$\mathcal{L}_G = \mathbb{E}_{V^p,\mathcal{I}}[F_{cGAN}(G_{cGAN}(E_{AE}(V^p), FE_I(\mathcal{I}))) - 1]^2. \tag{3}$$

To further stabilize the GAN training and guide the model, we include the reconstruction loss $\mathcal{L}_G^{Recon}$ that directly measures the differences between the ground-truth complete shape V^c and the estimated complete shape $\widehat{V^c} \triangleq D_{AE}(G_{cGAN}(E_{AE}(V^p), FE_I(\mathcal{I})))$ using Jaccard index loss

$$\mathcal{L}_G^{Recon} = 1 - \frac{|V^c \cap \widehat{V^c}|}{|V^c \cup \widehat{V^c}|}. \tag{4}$$

Therefore, the overall training objective for our shape completion model is

$$\underset{G,FE_I}{\operatorname{argmin}} \underset{F}{\operatorname{argmax}} \mathcal{L}_F + \mathcal{L}_G + \alpha \mathcal{L}_G^{Recon}, \tag{5}$$

where α is the weight for the reconstruction loss.

6.2 Pose Estimator

We use two simple four-layer MLP models to estimate the pose of the object. Our pose estimators T^t and T^r take the estimated complete shape latent code $\widehat{l_v^c}$, visual features f_i, normalization factors μ^p and σ^p, and a skip connection from partial latent l_v^p as input and estimate the 3D translation residual $\widehat{t_r} \in \mathbb{R}^3$ and 3D rotation in quaternion form $\widehat{R} \in \mathbb{R}^4$, respectively. Note, similar to the prior works (Wang et al., 2019a; Dikhale et al., 2022), instead of estimating the absolute translation t, we estimate the residual of the translation $t_r = t - \mu^p$. We use the residual pose $\widehat{\mathbb{P}_r} = [\widehat{R}|\widehat{t_r}]$ to calculate the point cloud loss $\mathcal{L}_P$ (Wang et al., 2019a):

$$\mathcal{L}_P = \frac{1}{k} \sum_{x \in \mathcal{K}} ||(Rx + t_r) - (\widehat{R}x + \widehat{t_r})||, \tag{6}$$

where $\mathcal{K}$ denotes a set of points randomly sampled from the object's 3D model, and k represents the cardinality $|\mathcal{K}|$. The point cloud loss minimizes the distance between the points on the ground-truth pose and their respective points on the models transformed using the estimated pose. The overall loss function is shown in Equation 7,

$$\underset{G,FE_I,T^t,T^r}{\operatorname{argmin}} \underset{F}{\operatorname{argmax}} \mathcal{L}_F + \mathcal{L}_G + \alpha \mathcal{L}_G^{Recon} + \beta \mathcal{L}_P, \tag{7}$$

where β is the weight for the point cloud loss.

To speed up convergence, our method is trained in four steps: i) The autoencoder ($E_{AE} + D_{AE}$) is trained in isolation, and weights are frozen; ii) The shape completion module ($G + F$) is trained; iii) The pose estimator ($T^t + T^r$) is trained while freezing the shape completion module; iv) the shape completion module in unfrozen and trained end-to-end ($G + F + T^t + T^r$), similar to (Liang et al.; Gervet et al., 2023).

6.3 Experiment Details

Synthetic Dataset We use VisuoTactile synthetic dataset from Dikhale et al. (2022) to train our framework. In this dataset, a subset of 11 YCB objects (Calli et al., 2015) are selected based on their graspability. A total number of 20K distinct in-hand poses are simulated per object. In particular, Unreal Engine 4.0 has been used to render photo-realistic observational data of a 6 DoF robot arm

with a 4-fingered gripper equipped with 12 32x32 tactile sensors (3 per finger). A main RGB-D camera captures images of the robot holding an object. Each tactile sensor captures object surface contact points in a point cloud format. Each data sample is generated by randomizing the in-hand object pose, the robot fingers configuration, and the robot arm orientation and position. Domain randomization is also applied for the color and pattern of the background and workspace desk.

Real Robot We test our model on the Allegro Hand (Wonik Robotics) and the Sawyer robot arm (Rethink Robotics). Our Allegro Hand is instrumented with the uSkin 4x4 tactile sensors and uSkin Curved tactile sensors from XELA Robotics. Each finger has three 4x4 tactile sensors (16 taxels) and one Curved tactile sensor (30 taxels). We use a Microsoft Kinect V2 camera to collect RGB-D data.

Implementation Details We utilize an ImageNet pre-trained ResNet-34 model as our visual feature extractor FE_I. We apply center crop, Gaussian blur, and color jitter augmentations to the input image. We implement our pose estimators T^r and T^t using four-layer MLPs with layers of $[512, 256, 32, 4]$ and $[512, 256, 32, 3]$, respectively. We trained our autoencoder for 500 epochs, shape completion module for 1000 epochs, pose estimator for 1000 epochs, and the entire end-to-end model for 2000 epochs using the Adam optimizer with a learning rate of 1×10^{-3}, 5×10^{-4}, 1×10^{-3}, and 5×10^{-5}, respectively. We set the reconstruction loss α and the point cloud loss β (Equation 7) to 30 and 5000, respectively.

6.4 Ablation Studies Analysis

In this section, we provide a more detailed analysis of the ablation studies results shown in Tab. 2.

No-Shape-Completion To evaluate the contribution of explicit shape completion, we remove the shape completion module from our proposed framework, which is achieved by feeding the partial latent vector l_v^p to the pose estimator instead of the complete vector $\widehat{l_v^c}$. Our ablation result shows that by removing the shape completion module, the position error drops by 24.8%, and the angular error drops by 30.0%, suggesting, jointly and explicitly optimizing the shape and the pose during visuotactile pose estimation is effective.

No-Vis-Gan To analyze the gain from visual cues in the shape completion module, we remove the visual feature conditioning f_i. We observe that removing the visual conditioning from the shape completion module resulted in a significant deterioration of performance, highlighting the importance of incorporating visual cues. The study shows that, without visual cues, the partial geometry feature is ambiguous for inferring the complete shape under heavy hand occlusion.

No-Vis-MLP To examine the contribution of visual features in pose estimators, we remove the visual feature input from the pose estimators. We notice that removing the visual features degrades the angular error performance. This makes sense because our dataset contains symmetrical objects. The object geometry alone is insufficient for accurately determining the pose of symmetrical objects, such as a mustard bottle, which requires the utilization of visual features to distinguish between its front and back.

No-Tactile & No-Point-Cloud Two separate studies were conducted to evaluate the contribution of the tactile points $P^{\mathcal{T}}$ (No-Tactile) and the point cloud from the vision sensor $P^{\mathcal{D}}$ (No-Point-Cloud). Our results suggest a significant drop in performance when either the tactile points or the point cloud input from the visual sensors is removed, emphasizing the significant contribution of both modalities to pose estimation.

6.5 Qualitative Results on Shape Completion

Fig. 5 represents a visualization of the input and output of our shape completion module, which uses visuotactile observations to faithfully complete the shape of the in-hand object.

6.6 Real-world Experiment

In Fig. 6, we show three consecutive frames of the Allegro Hand holding a scissor. Our model could accurately estimate the 6D pose of the in-hand object, given a noisy and partial segmentation mask. In

(a) Camera View (b) Partial Shape (c) Completed Shape (d) Ground Truth

Figure 5: Visuotactile shape completion results. The gray and red voxel represents RGB-D and tactile observations, $P^{\mathcal{D}}$ and $P^{\mathcal{T}}$, respectively. From left to right, we show the cropped RGB image from the main camera, the observed partial shape of the in-hand object, the completed shape by our approach, and the ground-truth shape.

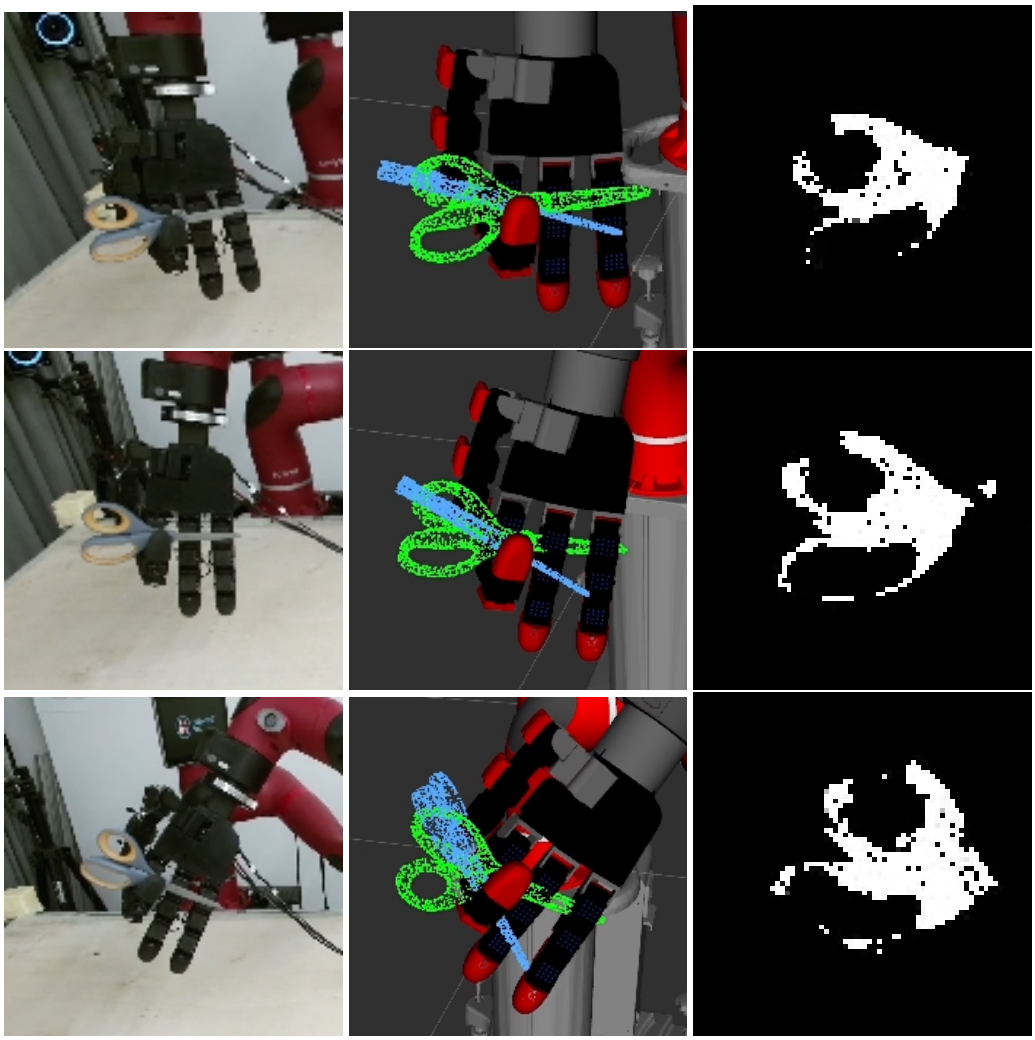
the bottom row, we demonstrate a failure case of our method where the estimated pose of the scissors is flipped 180 degrees due to the near-symmetrical geometry and visual feature of the scissors. We refer readers to our supplementary videos for our experiment videos that include more objects and real-time quantitative comparisons.

References

Y. Qin, H. Su, and X. Wang, "From One Hand to Multiple Hands: Imitation Learning for Dexterous Manipulation from Single-Camera Teleoperation," Apr. 2022, arXiv:2204.12490 [cs]. [Online]. Available: http://arxiv.org/abs/2204.12490

T. Chen, J. Xu, and P. Agrawal, "A System for General In-Hand Object Re-Orientation," in *Proceedings of the 5th Conference on Robot Learning*. PMLR, Jan. 2022. [Online]. Available: https://proceedings.mlr.press/v164/chen22a.html

A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? The KITTI vision benchmark suite," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*, Jun. 2012.



(a) Observation (b) Estimation (c) Segmentation

Figure 6: We compare our method (in green) against ViTa (Dikhale et al., 2022) (in blue) in real-world. From left to right are the observation from the RGB-D sensor, the pose estimations, and the noisy partial segmentation task we obtain.

Y. F. Chen, M. Everett, M. Liu, and J. P. How, "Socially aware motion planning with deep reinforcement learning," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*.

C. Wang, D. Xu, Y. Zhu, R. Martin-Martin, C. Lu, L. Fei-Fei, and S. Savarese, "DenseFusion: 6D Object Pose Estimation by Iterative Dense Fusion," 2019. [Online]. Available: https://openaccess.thecvf.com/content_CVPR_2019/html/Wang_DenseFusion_6D_Object_Pose_Estimation_by_Iterative_Dense_Fusion_CVPR_2019_paper.html

H. Chen, P. Wang, F. Wang, W. Tian, L. Xiong, and H. Li, "EPro-PnP: Generalized End-to-End Probabilistic Perspective-N-Points for Monocular Object Pose Estimation," 2022. [Online]. Available: https://openaccess.thecvf.com/content/CVPR2022/html/Chen_EPro-PnP_Generalized_End-to-End_Probabilistic_Perspective-N-Points_for_Monocular_Object_Pose_Estimation_CVPR_2022_paper.html

Y. Hu, J. Hugonot, P. Fua, and M. Salzmann, "Segmentation-Driven 6D Object Pose Estimation," 2019. [Online]. Available: https://openaccess.thecvf.com/content_CVPR_2019/html/Hu_Segmentation-Driven_6D_Object_Pose_Estimation_CVPR_2019_paper.html

P. J. Besl and N. D. McKay, "Method for registration of 3-D shapes," in *Sensor Fusion IV: Control Paradigms and Data Structures*, vol. 1611. SPIE, Apr. 1992. [Online]. Available: https://www.spiedigitallibrary.org/conference-proceedings-of-spie/1611/0000/Method-for-registration-of-3-D-shapes/10.1117/12.57955.full

Y. Li, G. Wang, X. Ji, Y. Xiang, and D. Fox, "DeepIM: Deep Iterative Matching for 6D Pose Estimation," 2018. [Online]. Available: https://openaccess.thecvf.com/content_ECCV_2018/html/Yi_Li_DeepIM_Deep_Iterative_ECCV_2018_paper.html

J. Bimbo, S. Luo, K. Althoefer, and H. Liu, "In-Hand Object Pose Estimation Using Covariance-Based Tactile To Geometry Matching," *IEEE Robotics and Automation Letters*, vol. 1, no. 1, Jan. 2016.

S. Dikhale, K. Patel, D. Dhingra, I. Naramura, A. Hayashi, S. Iba, and N. Jamali, "VisuoTactile 6D Pose Estimation of an In-Hand Object Using Vision and Tactile Sensor Data," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, Apr. 2022.

M. B. Villalonga, A. Rodriguez, B. Lim, E. Valls, and T. Sechopoulos, "Tactile Object Pose Estimation from the First Touch with Geometric Contact Rendering," in *Proceedings of the 2020 Conference on Robot Learning*. PMLR, Oct. 2021. [Online]. Available: https://proceedings.mlr.press/v155/villalonga21a.html

J. Varley, C. DeChant, A. Richardson, J. Ruales, and P. Allen, "Shape completion enabled robotic grasping," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Sep. 2017.

D. Watkins-Valls, J. Varley, and P. Allen, "Multi-Modal Geometric Learning for Grasping and Manipulation," in *2019 International Conference on Robotics and Automation (ICRA)*, May 2019.

Y. Xiang, T. Schmidt, V. Narayanan, and D. Fox, "PoseCNN: A Convolutional Neural Network for 6D Object Pose Estimation in Cluttered Scenes," May 2018, arXiv:1711.00199 [cs]. [Online]. Available: http://arxiv.org/abs/1711.00199

K. Zhang, Y. Fu, S. Borse, H. Cai, F. Porikli, and X. Wang, "Self-Supervised Geometric Correspondence for Category-Level 6D Object Pose Estimation in the Wild," Sep. 2022. [Online]. Available: https://openreview.net/forum?id=ZKDUlVMqG_O

H. Wang, S. Sridhar, J. Huang, J. Valentin, S. Song, and L. J. Guibas, "Normalized Object Coordinate Space for Category-Level 6D Object Pose and Size Estimation," 2019. [Online]. Available: https://openaccess.thecvf.com/content_CVPR_2019/html/Wang_Normalized_Object_Coordinate_Space_for_Category-Level_6D_Object_Pose_and_CVPR_2019_paper.html

M. Tatarchenko, A. Dosovitskiy, and T. Brox, "Octree Generating Networks: Efficient Convolutional Architectures for High-Resolution 3D Outputs," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct. 2017.

H. Li, Z. Li, N. U. Akmandor, H. Jiang, Y. Wang, and T. Padir, "Stereovoxelnet: Real-time obstacle detection based on occupancy voxels from a stereo camera using deep neural networks," in *2023 International Conference on Robotics and Automation (ICRA)*.

R. Wu, X. Chen, Y. Zhuang, and B. Chen, "Multimodal Shape Completion via Conditional Generative Adversarial Networks," in *Computer Vision – ECCV 2020*, ser. Lecture Notes in Computer Science, A. Vedaldi, H. Bischof, T. Brox, and J.-M. Frahm, Eds. Cham: Springer International Publishing, 2020.

M. Mirza and S. Osindero, "Conditional Generative Adversarial Nets," Nov. 2014, arXiv:1411.1784 [cs, stat]. [Online]. Available: http://arxiv.org/abs/1411.1784

K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," 2016. [Online]. Available: https://openaccess.thecvf.com/content_cvpr_2016/html/He_Deep_Residual_Learning_CVPR_2016_paper.html

X. Mao, Q. Li, H. Xie, R. Y. K. Lau, Z. Wang, and S. P. Smolley, "Least Squares Generative Adversarial Networks," Apr. 2017, arXiv:1611.04076 [cs]. [Online]. Available: http://arxiv.org/abs/1611.04076

Y. Liang, B. Chen, and S. Song, "SSCNav: Confidence-Aware Semantic Scene Completion for Visual Semantic Navigation," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*.

T. Gervet, S. Chintala, D. Batra, J. Malik, and D. S. Chaplot, "Navigating to objects in the real world," *Science Robotics*, vol. 8, no. 79, p. eadf6991, Jun. 2023, publisher: American Association for the Advancement of Science. [Online]. Available: https://www.science.org/doi/abs/10.1126/scirobotics.adf6991

B. Calli, A. Singh, A. Walsman, S. Srinivasa, P. Abbeel, and A. M. Dollar, "The YCB object and Model set: Towards common benchmarks for manipulation research," in *2015 International Conference on Advanced Robotics (ICAR)*, Jul. 2015.