Hand-Guided Object Tracking Using Hand-Object Consistency*

Jiwon Yang[1], Taewook Ha[1], Woontack Woo[1,2,*]

[1]KAIST UVR Lab, 291 Daehak-ro, Yuseong-gu, 34141, Daejeon, Republic of Korea
[2]KAIST KI-ITC Augmented Reality Research Center, 291 Daehak-ro, Yuseong-gu, 34141, Daejeon, Republic of Korea

Abstract

We propose a robust method for estimating the pose of occluded objects by hand during user interaction in a Head-Mounted Display (HMD) environment. Existing approaches to the occlusion problem often predict the hand and object jointly to improve efficiency, but their applicability in HMD environments is limited by high computational cost and poor generalization to occluded objects. Our approach applies hand pose changes to object pose changes based on the confidence levels of both the hand and the object. Evaluation conducted on 20 distinct grasping pose types demonstrated a lower Mean Per-Vertex Position Error (MPVPE) compared to conventional interpolation methods. Consequently, the proposed method enables effective estimation of occluded objects using fewer computational resources.

Keywords

Object poses estimation, Hand poses estimation, AR/VR, HMD

1. Introduction

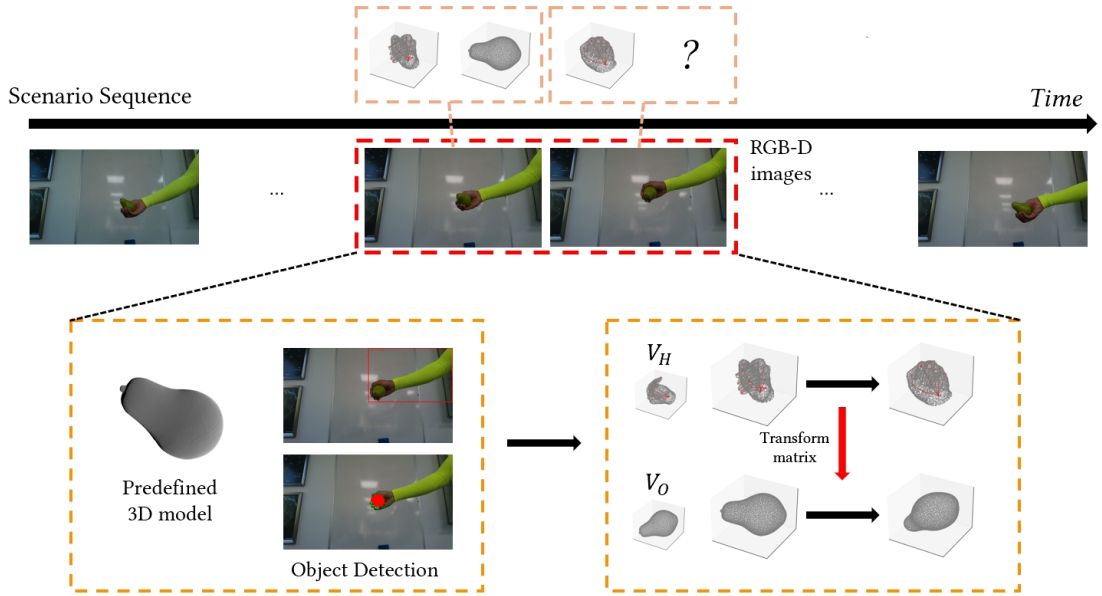

Figure 1: The structure of the hand-guided object tracking system

Object recognition plays a critical role in scenarios where users interact with objects via Head-Mounted Display (HMD) devices. When real-world objects held by users are not accurately tracked and such information fails to be transmitted to the device, natural interaction becomes impaired [1]. Although existing object detection models perform well under conditions where objects are clearly

* *APMAR'25: The 17th Asia-Pacific Workshop on Mixed and Augmented Reality, Sep. 26-27, 2025, Busan, South Korea*
* Corresponding author.
yjw514@kaist.ac.kr (J. Yang); hatw95@kaist.ac.kr (T. Ha); wwoo@kaist.ac.kr (W. Woo)
0009-0004-7946-8577(T. Ha); 0000-0002-5501-4421 (W. Woo)

visible, recognition accuracy tends to degrade significantly when occluded by the user's hand. Hence, addressing occlusion problems in computer vision is essential for providing realistic immersion in HMD environments [2].

Current approaches to occlusion mitigation predominantly employ deep learning and generative models to simultaneously predict hand and object states, often improving temporal efficiency compared to sequential prediction methods [3][4][5]. However, these methods exhibit three main limitations. First, as demonstrated in [3] and [5], while robust occlusion-resistant prediction is feasible, the computational load is high, making these methods resource-inefficient for HMD devices where limited processing power is available, often consuming excessive computation for accurate hand-object pose estimation. Second, as shown in [4], limitations arise with previously unseen objects, leading to poor generalization and inability to cover diverse hand-object interaction patterns typical in real-world HMD usage. Third, these methods typically do not consider fail-safe strategies for pose estimation or object tracking failures, which are crucial for practical deployment in HMD scenarios.

We proposes a method that estimates the current pose and position of objects occluded beyond a certain threshold by the user's hand by leveraging hand motion information. Our approach enables robust inference of occluded object movement with low computational overhead by applying hand pose changes to object pose estimation. The method integrates object recognition results from both current and previous frames, assessing the sufficiency of available information. When data confidence is adequate, the method heavily relies on current frame information for pose estimation; otherwise, it references prior frames' data to compensate for missing or unreliable inputs. This temporal data utilization ensures applicability to time-series data and supports real-time object pose estimation on resource-constrained HMD platforms.

Quantitative evaluation was performed using multiple hand-object interaction scenarios [6], representing various object grasping and rotation patterns. Frames were segmented based on object rotation direction, and the Mean Per-Vertex Position Error (MPVPE) between predicted and ground truth poses was computed. Experimental results show that applying hand pose variations to object pose estimation significantly outperforms conventional interpolation techniques in tracking occluded objects during hand-object interactions in HMD settings.

The contributions of this study are threefold. First, it presents an efficient pose estimation method tailored for real-time HMD interaction environments under limited computational resources, specifically addressing occlusion caused by the hand. Second, it enhances robustness by leveraging both current and past frame recognition data, enabling compensatory estimation when immediate information is insufficient or unreliable. Third, it validates improved tracking performance and generalization through comprehensive quantitative experiments involving realistic hand-object grasping and pose scenarios. Consequently, this work demonstrates the feasibility of reliable, real-time occluded object pose estimation for collaborative and interactive applications utilizing HMD devices.

2. Related Works

2.1. 3D Hand-Object Poses Estimation

Research on pose estimation in hand-object interaction scenarios from images or videos continues to advance. The H+O framework [7] proposed a method that simultaneously performs 3D hand-object pose estimation, object recognition, and action classification using a single RGB image, rather than separately estimating 3D poses of the person or objects. However, since it relies solely on a single RGB input, the lack of depth-related information poses inherent limitations when the hand and object occlude each other, adversely affecting prediction accuracy. More recently, HOISDF [3] employed Global Signed Distance Fields (SDF) to jointly estimate 3D hand-object poses even under occlusion. While this approach benefits from modeling contact and proximity between the hand and object simultaneously, it suffers from the high computational and memory demands of SDF processing. Additionally, severe occlusions necessitate further refinement to accurately capture fine details at contact regions. Similarly, Lin et al. [4] proposed a method that selectively shares or separates features

at the backbone network level to improve simultaneous pose estimation from a single RGB image under occlusion conditions. Despite its effectiveness, this approach lacks generalization to unseen objects and does not sufficiently address the challenges posed by invisible hand-object contact areas.

Other research efforts have sought to apply alternative AI models to hand-object pose estimation. Semi-supervised frameworks have been proposed to enhance pose estimation performance under occlusion and limited 3D labeled data from single images [8]. However, the absence of 3D object models resulted in no pseudo-labeling for objects, and the quality of such labels critically influences performance, limiting comprehensive resolution in complex multi-object and hand interaction scenarios. Additionally, some studies employ deep learning-based feedback loop frameworks to simultaneously estimate 3D hand and object poses purely via deep neural networks [9]. Nonetheless, these deep learning-based object detection approaches generally entail substantial computational overhead, rendering them inefficient for deployment on resource-constrained HMD devices [5]. Therefore, our approach aims to provide a less computationally demanding alternative compared to existing methods that require high computational costs.

2.2. Human-Object Interaction Detection (HOI detection)

Research on detecting how human-object interactions (HOI) occur continues to advance. The UnionDet framework [10] proposed a single-stage prediction approach that directly infers the interaction regions of human-object pairs, aiming to overcome the speed limitations of conventional multi-stage HOI detection methods. However, it exhibited limitations in handling overlapping instances and multiple simultaneous interactions. Another approach utilized a transformer-based model to predict sets of humans, objects, and interactions without requiring explicit human-object matching [11]. This method benefits from eliminating computationally expensive post-processing, resulting in significantly faster inference speeds. Nevertheless, it faces increased computational costs when dealing with complex images containing numerous human object interaction instances. More recently, attempts have been made to combine Convolutional Neural Networks (CNNs) with multi-resolution wavelet analysis to address the trade-off between computational speed and detection accuracy [12]. However, this approach infers interactions solely from 2D images without incorporating full 3D information, limiting its applicability in scenarios where depth information is essential. Therefore, instead of jointly predicting both the hand and the object within the interaction space, our approach applies the motion of the hand to the target object, thereby reducing computational overhead and improving inference speed.

3. Method

3.1. System Structure

In real-world interactions, the hand and object typically move independently until contact occurs. Therefore, temporal information is utilized to set the initial pose of the stationary object, and subsequent hand pose variations are applied to update the object pose when it becomes occluded. Because the occluded object's pose is predicted using the remaining recognition results, this approach operates efficiently without requiring additional computational resources.

The system is broadly divided into two stages. The first stage recognizes the hand and object separately based on 2D images from the camera viewpoint. We assume that 3D information of the target object is provided beforehand, and an RGB-D image captured either immediately at contact onset or while in contact is supplied as input. This enables pose estimation of the object in its initial static state, as well as the detection of the hand pose at the instant of contact. From the moment hand-object interaction begins, an object detection model is employed to evaluate the recognition confidence of both hand and object. To improve reliability, cropped regions based on the estimated hand and object

locations are fed into the object detection model. These detection results subsequently inform the application of hand pose changes to update the object pose.

The second stage estimates the pose based on the detection confidence scores from the first stage. When the object detection confidence is sufficiently high, indicating accurate recognition, the object pose is updated using a dedicated pose estimation model. Conversely, if the object confidence is low, the most recently recognized object pose is updated by applying hand pose variations. This update process also considers the confidence of the hand pose estimation as well as the temporal interval since the last object pose update.

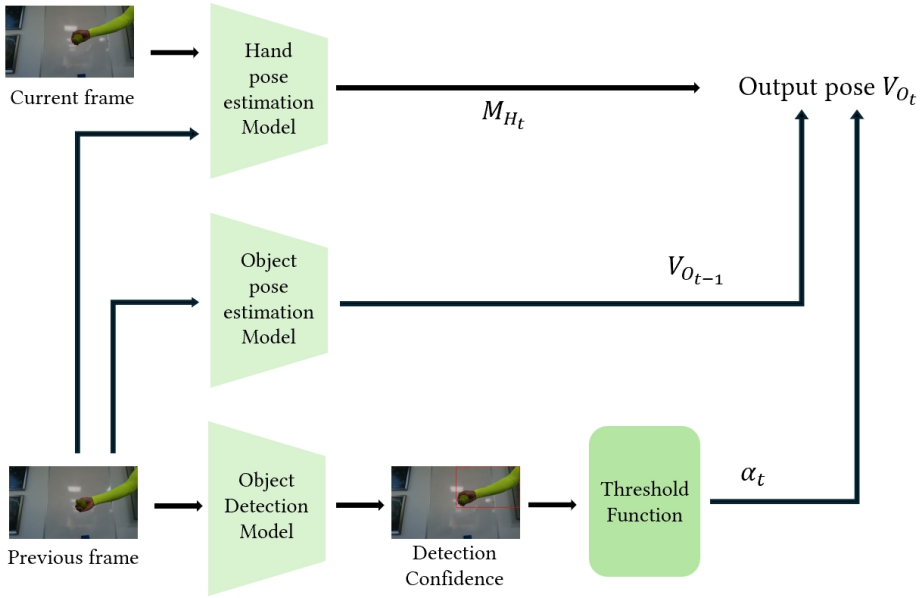

Figure 2: The Data processing pipeline of the hand-guided object tracking system

3.2. Object pose estimation based on hand pose

Equation (1) describes the object pose update method in the second stage introduced in Section 3.1. Let V_{O_t} denote the object poses at time t. The predicted object poses are obtained by applying a rotational transform matrix M_{H_t}, representing the hand's rotation at time t, to the object poses from the previous time step $V_{O_{t-1}}$, scaled by a weighting coefficient α_t. Formally:

$$V_{O_t} = \alpha_t \cdot M_{H_t} \cdot V_{O_{t-1}} \quad . \tag{1}$$

Here, α_t quantifies the degree of trust in the previous object pose estimate when computing the current pose update.

$$\alpha_t = \begin{cases} 1, & \text{if } S_t^{obj} \geq \tau_{obj} \\ No\ \ update, & \text{if } S_t^{obj} < \tau_{obj} \text{ and } S_t^{hand} < \tau_{hand} \\ (\lambda^{\Delta t} \cdot S_t^{hand}), & otherwise \end{cases} \tag{2}$$

Equation (2) defines the calculation of the weighting coefficient α_t. Let S denote the recognition confidence score for the object or the hand at time t, and let τ be a predefined confidence threshold. If the recognition confidence for either the object or the hand falls below τ, it is considered that the respective entity is insufficiently visible, and the system either fully references or disregards the previous frame's information accordingly. In other cases, the hand confidence is used with a decay factor proportional to the number of frames elapsed since the last reliable object pose update to calculate α_t. When both the object confidence and the hand confidence drop below respective minimum thresholds, the current frame's pose estimation becomes unreliable. Therefore, the system preserves the pose from the most recent reliable frame to minimize estimation error.

4. Experiment

4.1. Dataset

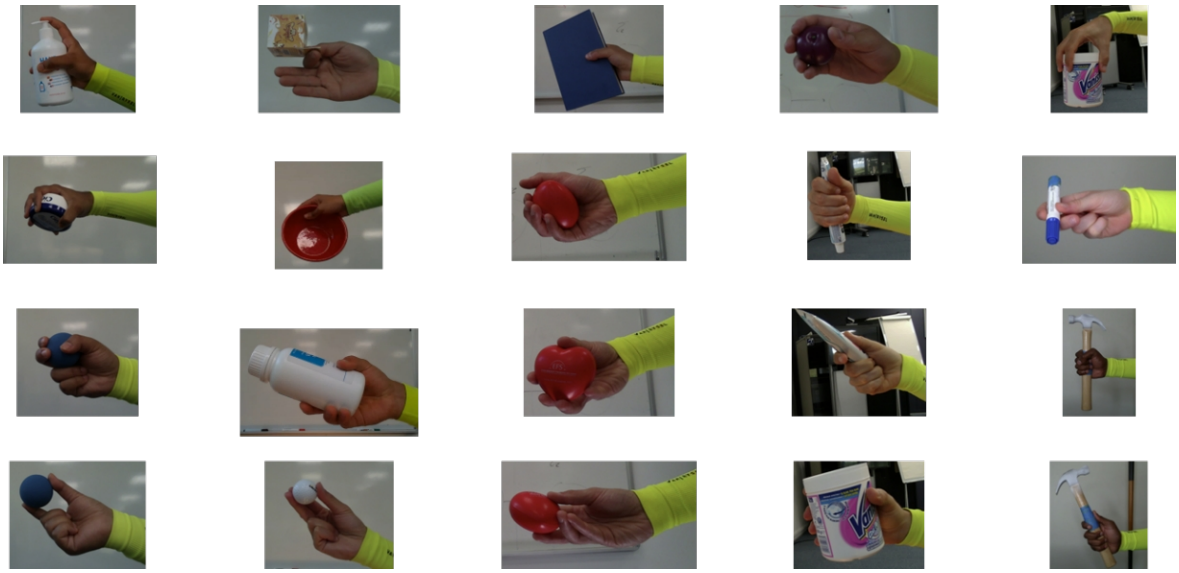

Figure 3: Representative scenarios of 20 grasp methods in the SHOWME dataset [3]

To consider scenarios in which objects are partially occluded by the hand, the SHOWME dataset [6] was utilized. This dataset defines 20 grasp types derived from the comprehensive grasp taxonomy of 33 types presented in [13], and it comprises a total of 96 scenarios based on variations in object categories and hand movements. In this study, experiments were conducted using a subset of 20 scenarios, each corresponding to one of the 20 selected grasp types. The selection criteria focused on scenarios where the modeling information of rendered results aligned well when projected onto the RGB images. To account for fast-moving objects, not all data recorded at 30 frames per second (fps) was used; instead, one frame was sampled every 10 frames, effectively yielding a 3 fps frame rate for the experiments. Camera parameters, including the distance between the camera and the object, were directly utilized as provided in the dataset.

4.2. Performance Metric

The evaluation metric employed in this study is the Mean Per-Vertex Position Error (MPVPE). MPVPE quantifies the average positional discrepancy between the vertices of the ground truth (GT) mesh and those of the estimated mesh. For both the proposed method and the interpolation baseline, predicted object meshes are separately saved as obj files to compute this metric. Specifically, the MPVPE is calculated by comparing the obj files of the predicted object mesh against the corresponding ground truth object mesh provided in the SHOWME dataset. Lower MPVPE values indicate smaller deviations between the predicted and actual vertex positions, thus representing higher estimation

accuracy. The results are analyzed by plotting graphs for each grasp type and rotation direction to provide detailed performance insights.

4.3. Experimental Method

Prior to deployment on HMD devices, the original dataset values are treated as ground truth and used to evaluate the prediction accuracy. The method follows the previously described system workflow. The model is applied to cropped images focusing exclusively on the hand and object regions within each frame of the dataset. This cropping aims to isolate the hand-object interaction, preventing interference from other objects in the scene and ensuring that confidence scores reflect only the scenario-specific hand and object. The crops were generated using the rendered results provided by the SHOWME dataset.

As a baseline, an interpolation method was considered. When the confidence scores for the hand and object in the current frame fall below the threshold used in the proposed method, the object pose is estimated as the midpoint between the previous and subsequent frames. This interpolation approach is analogous to the proposed idea of predicting object pose based on the hand pose change between consecutive frames. The interpolation is applied specifically at the point where the weighting coefficient is calculated during object pose estimation at time t in the system. Otherwise, all other processing steps remain identical. This setup enables direct comparison of evaluation metrics between the proposed method and the interpolation baseline.

Significant variability exists in object detection rates across scenarios, heavily influenced by the training quality of the detection model. Experiments were conducted not only using the raw detection results but also by artificially adjusting detection rates per scenario. This allowed assessment of which method performs better relative to the detection model's effectiveness. When using unmodified detection results, object pose prediction is triggered only if the object fails to be detected by the detection model, which in this study is Mediapipe. When detection rates are artificially manipulated, undetected frames are randomly prioritized for pose prediction using the respective methods. Hand pose confidence for prediction always relies on Mediapipe model outputs.

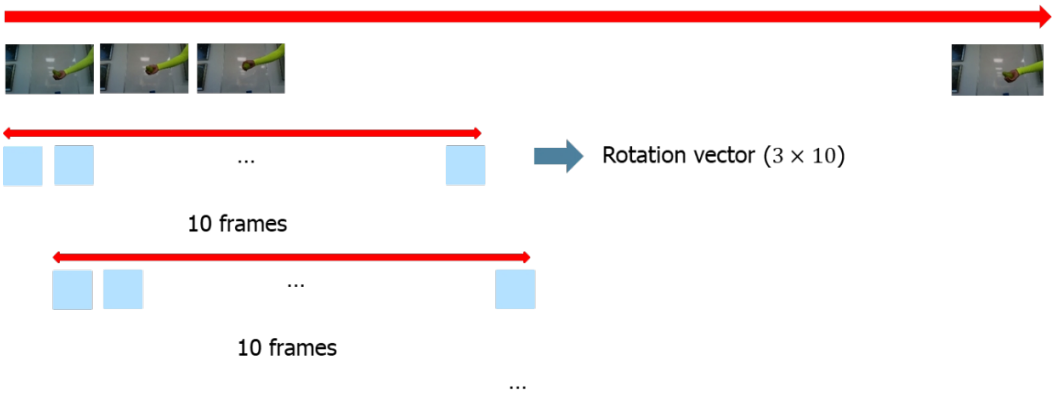

Figure 4: How to group minimum rotation units for scenario rotation order and direction classification

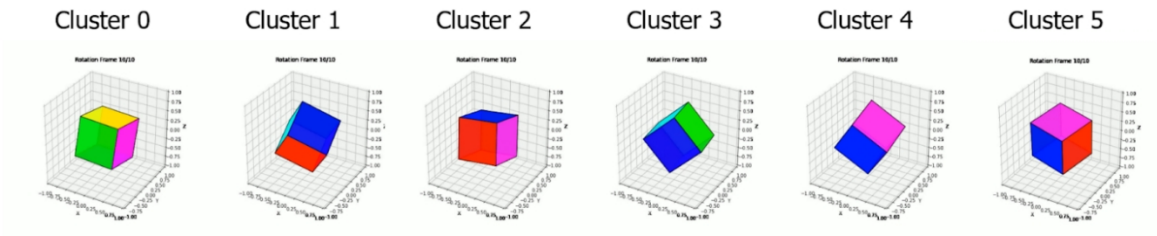

Figure 5: Types of rotation classification using the k-means algorithm (6 types)

When utilizing the dataset, the grasping method, object type, and rotation sequence/direction are not consistent. Therefore, we additionally grouped the data into rotation units of 10 frames, which served as the minimum rotation segment for classification. In other words, one unit corresponds to a 100-frame video (10 data points). For all 20 scenarios, these units were grouped, and the corresponding rotation vectors were classified into rotation types using the k-means clustering algorithm. Based on these rotation types, we examined which rotation directions each scenario is more specialized in, thereby enabling more accurate estimation. The number of clusters was experimentally adjusted by varying k until clusters with identical directions and motion tendencies no longer appeared. Consequently, the six clusters obtained represent distinct directions or tendencies (i.e., consistency of rotation).

5. Discussion

5.1. Comparison of MPVPE with Interpolation Methods

First, when specifying different detection rate ratios, we compared the Mean Per-Vertex Position Error (MPVPE) results for each scenario and detection rate using both the proposed method and the interpolation baseline. The results demonstrated that the proposed method consistently achieved lower MPVPE values across all scenarios. Scenario 14 (Medium Wrap) exhibited a substantial performance gap favoring the proposed method regardless of the object detection rate. In contrast, Scenarios 4(Inferior Pincer), 8(Tripod), and 13(Quadpod) showed relatively minor differences between methods, irrespective of detection rates. These scenarios involve grasps on smaller objects, which may contribute to smaller absolute errors in both methods. Additionally, although these scenarios feature longer durations with diverse rotations, the smaller radius of object rotation results in relatively low errors even when using conventional interpolation.

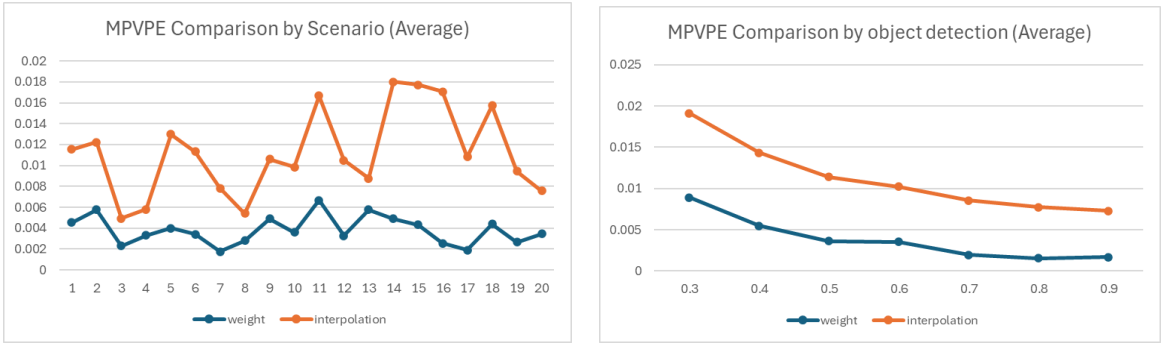

Figure 6: MPVPE results by scenario (left) and detection ratio (right) when using the method of this study and the interpolation method when the detection result ratio is specified.

Across all detection rate variations, the proposed method consistently outperformed the interpolation approach. Furthermore, as detection rates decreased, the performance gap widened, indicating that the proposed method is particularly effective when recognition confidence is low. Conversely, performance stabilized when detection rates surpassed a certain threshold. This plateau is

likely due to the decay factor in the weighting coefficient α_t, which diminishes proportionally with the number of frames elapsed since the last reliable object pose update. Higher detection rates reduce the number of frames over which decay applies, leading to more stable pose estimations.

To further analyze results by rotation type, scenarios were grouped into six rotation clusters. Clusters 0 and 2 have opposite rotation directions, so they revealed significant differences in MPVPE despite the symmetrical rotation axes. For example, Scenarios 9 (Parallel Extension), 10 (Power Sphere), and 11 (Precision Sphere) frequently exceeded an MPVPE of 0.002 in Cluster 0, whereas in Cluster 2, most scenarios remained below this threshold. This discrepancy is hypothesized to result from longer occlusion durations caused by the rotation direction. Longer occlusions increase the number of frames over which decay in pose confidence is applied, thereby reducing prediction reliability and increasing error. Thus, the proposed method demonstrates better performance when occlusion occurs in shorter, repeated intervals rather than in prolonged continuous segments.

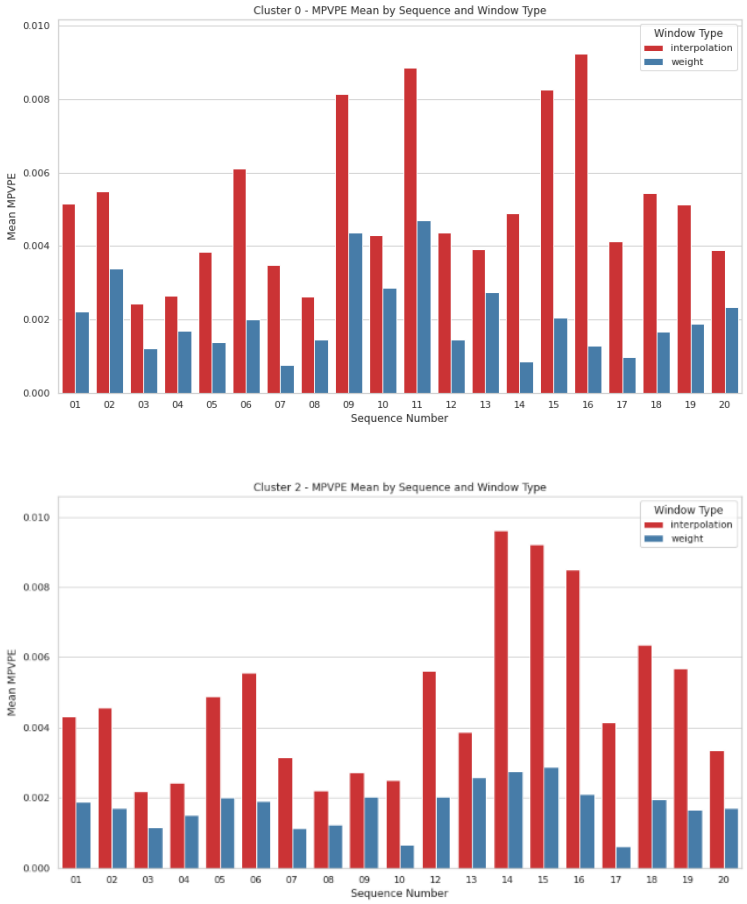

Figure 7: Classification by rotation direction – the upper and lower figures represent opposite rotations. (If the rotation does not exist in the scenario, it is removed from the figure)

Next, the comparison between Clusters 4 and 5 focused on whether rotation direction remained consistent or changed midway. Cluster 4 generally exhibited higher MPVPE values, with Scenario 13 (Quadpod) showing a twofold increase compared to Cluster 5. This result suggests that predicting object pose from hand pose changes is more straightforward when rotation direction remains constant. When rotation direction changes, the hand's rotational velocity typically decreases, resulting in longer occlusion intervals and greater difficulty in accurate prediction.

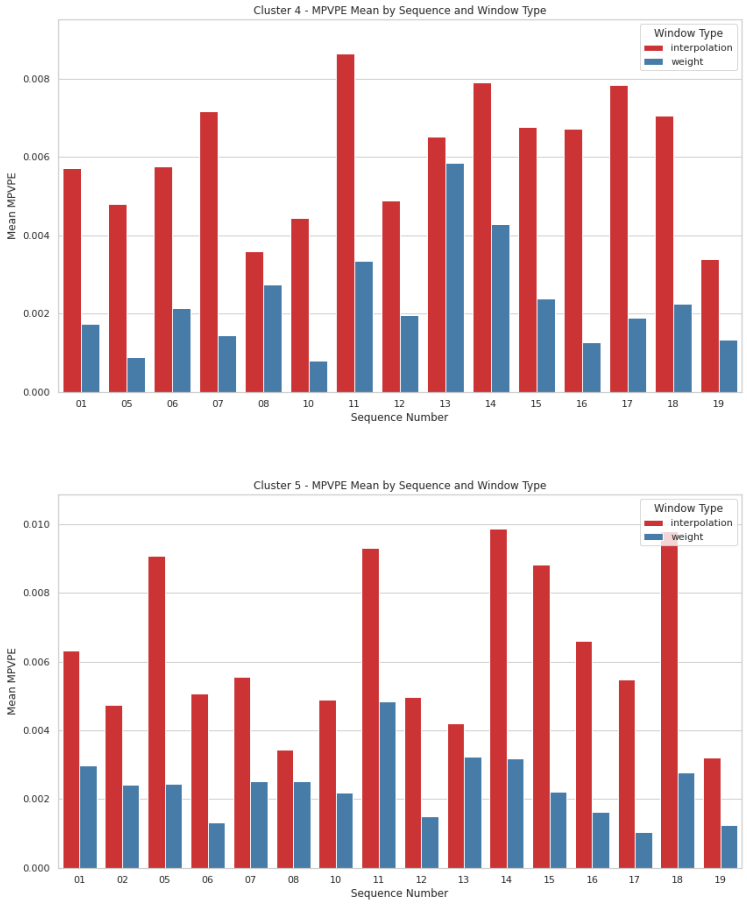

Figure 8: Classification by rotational consistency – Cases where the rotation direction changes midway(upper), Cases where the rotation direction is constant (lower). (If the rotation does not exist in the scenario, it is removed from the figure)

5.2. Resource Usage/Computation Time

Table 1. presents the resource consumption and computation time of the proposed system. For the first stage, object detection, we employed the Mediapipe object detection model. The reported results include the entire process, from detecting the target regions of the hand and object in the given RGB images, cropping these regions, and storing the outputs. For the second stage, pose estimation, the original pipeline should include estimating past and current hand poses using a hand pose estimation model, followed by object pose estimation based on those hand poses. However, considering that the outcomes can vary significantly depending on the specific hand pose estimation model employed, we excluded that part and calculated values only for the remaining structural components of the system. The tested scenario involved the tripod sequence, conducted under the same conditions as the MPVPE comparison experiment described in Section 5.1, using a total of 126 frames.

Table 1
Step-by-step operation time/resource usage

Step	Object detection	Pose estimation
Average FPS ↑	11.88	281.669
Average time per frame [ms] ↓	83.11	3.55

Total FLOPs [TFLOPs] ↓	0.063	0.252
Max Memory Usage [MB] ↓	688.03	2261.82

According to a recent study analyzing the impact of frame rate on user experience in virtual reality environments [14], most users perceive a sense of real-time interaction at frame rates above 30 FPS, while a frame rate of 60 FPS or higher is recommended to ensure full immersion and to mitigate simulator sickness. Since the FPS value achieved by the system proposed in this study exceeds 60 FPS, it can be considered sufficient for users to perceive real-time responsiveness. Furthermore, because the proposed method requires only a minimal computational time, it is expected that parallel utilization of multiple models would not introduce significant performance issues. Given the computational speed of the example object detection model employed, it can be inferred that the overall system's FPS is ultimately determined by the specific object detection and hand pose estimation models utilized. Therefore, if real-time capable object detection and hand pose estimation models are employed, the system architecture demonstrated in this study can be effectively applied to HMD devices in real-time scenarios.

In addition, the FLOPs value of the proposed method is relatively small when compared to the computational capabilities of current HMD devices and smartphones, thereby confirming its feasibility for deployment on such platforms. Finally, the maximum memory consumption of approximately 2 GB further indicates that the system is well within the RAM capacity of modern HMD devices, ensuring its practical applicability [15].

6. Conclusion

We proposed a method to estimate the pose of objects occluded by the hand through the utilization of hand pose changes. To evaluate the performance advantage of our method compared to the conventional interpolation approach, tests were conducted on 20 grasp types from the SHOWME dataset[6]. Our method consistently outperformed the baseline regardless of the performance of the object detection model. Furthermore, even when experiments were stratified by object rotation directions, the proposed method demonstrated superior performance with a substantial margin.

In addition, we calculated the step-by-step processing time and computational resource usage to examine whether the proposed method could be utilized on real-time HMD devices. Since our method requires very little time per frame, we demonstrated that it can be applied in terms of processing time, provided that the object detection and hand pose estimation models to be used together are appropriately selected for real-time operation. Furthermore, in terms of FLOPs, we confirmed that the method is applicable when considering the performance levels of HMD devices and general smartphones.

Although the SHOWME dataset used in this study contains certain instances of directional changes, frames exhibit predominantly linear tendencies. Hence, it remains necessary to evaluate whether the proposed approach can be generalized effectively to datasets characterized by more complex motion patterns. Furthermore, since the SHOWME dataset is limited to single-hand manipulation of an object, additional validation is required in scenarios that align more closely with the research objective—namely, multi-user interaction with objects in immersive HMD environments, where multiple users may manipulate a single object simultaneously. In addition, to comprehensively evaluate different grasping methods, we utilized a dataset that can be classified into 20 grasp types and performed experiments under the assumption that the hand and object in the images are observed from the user's

perspective. To further validate the user's direct manipulation of objects, we plan to conduct additional user studies.

In this regard, future work may consider incorporating a weighting term that accounts for complex movements. For example, there is determining which hand's confidence level should be applied when estimating object pose changes in multi-hand scenarios. Such an extension would enhance the robustness of the proposed estimation method for interaction with virtual avatars, which constitutes the final objective of this research. Additionally, as this study has prioritized performance validation of the proposed method, relatively less attention has been devoted to the selection of the hand pose estimation model. Further investigations into model selection could thus provide a more comprehensive guideline for the effective application of the proposed framework.

Acknowledgement

This paper was supported by Korea Institute for Advancement of Technology(KIAT) grant funded by the Korea Government(MOTIE) (RS-2025-02304167, HRD Program for Industrial Innovation)

Declaration on Generative AI

During the preparation of this work, the author used GPT-5 in order to: Grammar and spelling check. After using these tools, the author reviewed and edited the content as needed and takes full responsibility for the publication's content.

References

[1] M. M. Shah, H. Arshad, and R. Sulaiman, "Occlusion in augmented reality," in Proc. 2012 8th Int. Conf. Information Science and Digital Content Technology (ICIDT), Jeju, Korea, 2012.

[2] Q. Feng, H. P. H. Shum, and S. Morishima, "Resolving hand-object occlusion for mixed reality with joint deep learning and model optimization," Comput. Animat. Virtual Worlds, vol. 31, no. 4–5, e1956, 2020. doi: 10.1002/cav.1956.

[3] H. Qi, C. Zhao, M. Salzmann, and A. Mathis, "HOISDF: Constraining 3D hand-object pose estimation with global signed distance fields," in Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition (CVPR), 2024. doi: 10.48550/arXiv.2402.17062.

[4] Z. Lin, C. Ding, H. Yao, Z. Kuang, and S. Huang, "Harmonious feature learning for interactive hand-object pose estimation," in Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition (CVPR), 2023, pp. 12989–12998.

[5] B. Mounika, P. Udayaraju, Ch. V. Varma, T. V. Narayana, P. Jyothi, and Ch. Devi, "Exploring spiking neural networks and deep learning techniques for occlusion detection in AR and VR images," in Proc. 2024 Int. Conf. Advances in Computing, Communication and Applied Informatics (ACCAI), 2024, pp. 1–8. doi: 10.1109/ACCAI61061.2024.10601809.

[6] A. Swamy, V. Leroy, P. Weinzaepfel, F. Baradel, S. Galaaoui, R. Bregier, M. Armando, J.-S. Franco, and G. Rogez, "SHOWMe: Benchmarking object-agnostic hand-object 3D reconstruction," in ACVR Workshop at ICCV, 2023. doi: 10.48550/arXiv.2309.10748.

[7] B. Tekin, F. Bogo, and M. Pollefeys, "H+O: Unified egocentric recognition of 3D hand-object poses and interactions," in Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition (CVPR), 2019, pp. 4511–4520.

[8] S. Liu, H. Jiang, J. Xu, S. Liu, and X. Wang, “Semi-supervised 3D hand-object poses estimation with interactions in time,” in Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition (CVPR), 2021, pp. 14687–14697.

[9] M. Oberweger, P. Wohlhart, and V. Lepetit, “Generalized feedback loop for joint hand-object pose estimation,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 42, no. 8, pp. 1898–1912, Aug. 2020. doi: 10.1109/TPAMI.2019.2907951.

[10] B. Kim, T. Choi, J. Kang, and H. J. Kim, “UnionDet: Union-level detector towards real-time human-object interaction detection,” in Proc. Eur. Conf. Computer Vision (ECCV), 2020. doi: 10.48550/arXiv.2312.12664..

[11] B. Kim, J. Lee, J. Kang, E.-S. Kim, and H. J. Kim, “HOTR: End-to-end human-object interaction detection with transformers,” in Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition (CVPR), 2021, pp. 74–83.

[12] Q. B. Pay, V. M. Baskaran, J. Y. Loo, K. Wong, and S. See, “Conceptualizing multi-scale wavelet attention and ray-based encoding for human-object interaction detection,” in Proc. Int. Joint Conf. Neural Networks (IJCNN), 2025. doi: 10.48550/arXiv.2507.10977.

[13] T. Feix, R. Pawlik, H.-B. Schmiedmayer, J. Romero, and D. Kragić, “A comprehensive grasp taxonomy,” in Proc. IEEE-RAS Int. Conf. Humanoid Robots (Humanoids), 2010, pp. 327–333.

[14] J. Wang, R. Shi, W. Zheng, W. Xie, D. Kao, and H.-N. Liang, “Effect of frame rate on user experience, performance, and simulator sickness in virtual reality,” IEEE Trans. Vis. Comput. Graphics, vol. 29, no. 5, pp. 2478–2488, May 2023. doi: 10.1109/TVCG.2023.3247057.

[15] D. Heaney, “Quest 3 full specs – Compared with Quest 2, Quest Pro, Pico 4 & Apple Vision Pro,” UploadVR. Available: https://www.uploadvr.com/quest-3-specs/. [Accessed: Aug. 18, 2025].