A Unified Hand and Gesture Tracking via Offloading Framework for Object-mediated Interaction in Wearable AR

Woojin Cho*
KAIST KI-ITC PMRC

Taewook Ha†
KAIST UVR Lab

Taejun Son‡
KAIST UVR Lab

Woontack Woo§
KAIST UVR Lab
KAIST KI-ITC ARRC

Figure 1: Concept illustration of object-mediated hand interaction with daily objects

Abstract

We propose a novel object-mediated hand interaction system that enables real-time operation with everyday objects on wearable augmented reality (AR) devices. Despite recent advances, both commercial and academic hand interaction techniques remain constrained, typically requiring external hardware or depending exclusively on bare-hand gestures. Motivated by these constraints, we developed an offloading framework that integrates a high-fidelity transformer-based 3D hand reconstruction model with a dynamic gesture recognition network powered by gated recurrent units (GRU). This architecture ensures stable and accurate gesture recognition even during interaction with physical objects. To evaluate its quantitative performance, we collected a custom dataset based on a predefined gesture set, achieving 93.0% accuracy in 5-fold cross-validation. The complete system implemented on Microsoft HoloLens 2 operates at a real-time framerate, and we further analyze the latency of each step in our framework. Through this interaction paradigm, users can experience immersive and intuitive AR in everyday environments with minimal disruption to natural action behavior. Our projects are available at https://github.com/kaist-uvr-lab/UnifiedHOInteraction.

Index Terms: Augmented Reality, Hand Tracking, Dynamic Gesture Recognition, Real-Time Interaction.

*e-mail: woojin.cho@kaist.ac.kr
†e-mail: hatw95@kaist.ac.kr
‡e-mail: signal725@kaist.ac.kr
§e-mail: wwoo@kaist.ac.kr

1 Introduction

Augmented reality (AR) has rapidly evolved, driven by advances in computing power and resource availability. This progress has enabled the commercialization of lightweight head-mounted displays(HMD), such as Apple Vision Pro [1], Microsoft HoloLens 2 [3], and Meta Quest Pro [4], as well as lightweight glasses like VITURE XR [6], Meta Ray-Ban [5], which offer mobile and immersive experiences. As AR becomes more integrated into daily life, intuitive and natural interaction methods are increasingly essential. Hand-based input stands out for its versatility and leverages well-established motor skills from daily object manipulation, yet vision-based 3D hand pose estimation during object interaction continues to be an active area of research [40]. Consequently, current AR systems rely on either mid-air bare-hand gestures or wearable controllers, both of which limit interaction richness and disrupt natural behavior. To overcome these limitations, we propose a novel hand-based interaction system that leverages real-world objects as mediators, enabling more intuitive and immersive AR experiences.

Incorporating objects as mediators into conventional hand interaction provides several key benefits. First, it allows for a seamless experience by transforming everyday objects into interaction elements within the AR environment, maintaining the natural flow of daily activities. Second, enabling the combination of familiar finger gestures(e.g., sliding, tapping) with a wide range of object types and poses significantly expands the interaction space, without requiring users to learn unfamiliar gesture vocabularies. This capability makes it possible to assign functions to objects that are naturally associated with the intended action or scenario, fostering intuitive and semantically context-aware interactions. Finally, physical objects provide inherent haptic feedback, addressing a key limitation of mid-air gestures, which suffer from the absence of tactile feedback and consequent user fatigue [26, 32, 35]. Within our interaction framework, we define a set of common hand gestures ap-

plicable across everyday objects, while supporting object-specific functionalities determined by their type. We apply the term **object-mediated interaction** to this paradigm, in which real-world objects mediate AR interactions.

There exists a notable gap between two research domains: real-time hand pose estimation in egocentric view, which typically relies on substantial computational resources, and practical hand interaction design for commercial AR devices operating under resource constraints. Although hand pose estimation has been extensively studied across various contexts, most commercial AR headsets support only bare-hand tracking due to limited onboard processing and consistent recognition accuracy. This results in a discrepancy between tracking capabilities and interaction design potential. Consequently, recent research on hand interaction in AR has focused on gesture types that can avoid hand occlusion [29, 47], or on object pose-dependent methods that bypass hand tracking [48, 19].

To bridge this gap, we leverage a novel hand tracker that operates on HMD via an offloading framework, enabling precise hand pose and gesture tracking even in complex hand-object interaction scenarios. Unlike prior works [29, 47, 48, 19] that either simplify gestures to avoid occlusion or omit hand tracking altogether, our approach enables full hand-object interaction while maintaining real-time performance on lightweight AR devices. Our system comprises a device-side module for capturing input images and rendering interaction based on recognized gesture outputs, and a server-side module that performs hand tracking, object, and gesture recognition. On the server, we deploy a robust hand pose estimation model capable of handling occlusions, along with a YOLO-based object tracker. To classify dynamic hand motions as object-mediated gestures, we design a temporal model based on Gated Recurrent Units(GRUs), which offer a simpler structure and superior long-term dependency modeling compared to the conventional Long Short Term Memory(LSTM) model [14]. To further enhance temporal resolution and real-time performance, we integrate a self-attention mechanism, achieving both high recognition accuracy and responsiveness.

In the absence of datasets matching the gesture types introduced in this study, we collected a custom dataset of object-mediated gestures for quantitative evaluation. Our model achieved an average accuracy of 93.0% in per-subject evaluation and an F1-score of 92.9% across individual gesture classes, demonstrating strong generalization and robustness. We also performed an ablation study to assess the contribution of each gesture recognizer component and verified the model on a public human action dataset. To validate the system's real-world applicability, we deployed it on Microsoft's commercial wearable device, HoloLens 2, and analysis end-to-end latency. We further demonstrate several AR application scenarios, including UI control, contextual tool invocation, and gesture-driven content manipulation, which showcase the system's ability to enable intuitive and seamless interaction. By incorporating the proposed mediated interaction, our approach establishes a foundation for scalable and object-adaptive AR interaction that overcomes key limitations of current systems and delivers richer, more seamless user experiences, particularly as AR devices become increasingly lightweight and ubiquitous.

Our main contributions are summarized as follows:

- An offloading-based hand tracking framework that enables advanced pose estimation in occluded scenarios and dynamic gesture recognition on resource-constrained devices.
- Object-mediated hand interaction that extends the interaction space by incorporating everyday objects and enables natural mapping of AR functions in object-centric contexts.
- A complete HoloLens 2 implementation with training code and a custom gesture dataset to facilitate adaptation and extension for various applications.

2 Related Works

In this section, we review recent studies on hand–object interaction in AR environments, along with research on the core components of our proposed framework: 3D hand pose estimation and related interaction studies.

2.1 3D Hand Pose Estimation

Deep learning has become the dominant paradigm in hand pose tracking, enabling robust and scalable 3D hand reconstruction across varied scenarios. Current methods span generative and discriminative frameworks, integrating mesh modeling, graph reasoning, and temporal cues to enhance accuracy and generalization. Generative approaches typically regress hand mesh parameters (e.g., MANO [55]) via autoencoders [38] and intermediate supervision [71, 70, 73], while discriminative methods directly predict joint or mesh coordinates using autoencoders [66, 75], depth maps [49, 50], or heatmaps [72, 12]. Graph-based designs exploit hand topology through spectral [10] and spatial [25] GCNs, with coarse-to-fine reconstruction [24, 13], transformer-based vertex modeling [42, 62], and attention-guided GCNs [60]. For real-time applications, lightweight architectures and efficient losses have been proposed [41, 72, 11], with tracking-history-based regression for Virtual Reality [28]. High-performing transformer-based methods[42, 62, 53] have recently expanded to Vision Transformers [18]. To address both speed and generalization performance in hand–object interaction scenarios, we incorporate two independent modules into our framework, allowing seamless switching between them. Cho et al. [12] proposed a multi-stage GCN-based network that sustains high inference speed even under severe hand–object occlusion. Potamias et al. [53] introduced a large-scale in-the-wild dataset and a high-fidelity transformer-based 3D reconstruction model, achieving real-time performance with strong generalization.

2.2 Hand-Object Interaction

The object-mediated hand interaction examined in this study refers to hand interactions performed while wearing an HMD, in which real-world objects serve as intermediaries for AR functions. Specifically, it involves triggering AR functions through predefined hand gestures in combination with information about the interacting object. While many existing AR devices employ controllers for precise input and tactile feedback, such auxiliary devices have been reported to cause inconvenience and reduce immersion [45, 46]. Accordingly, we review related research on free-hand interactions that operate without additional controllers. Prior works in this area can be broadly divided into two categories: approaches using only the bare hand and those incorporating surrounding objects.

In the first category, the hand itself serves as the sole interface and interaction element in AR environments, exemplified by basic bare-hand interactions implemented in commercial HMDs. This approach offers convenient, controller-free operation and has been extensively studied [37, 74, 52]. To support diverse functions, predefined bare-hand gestures have been proposed; in particular, several studies assign distinct functionalities to specific gestures [58, 15, 57]. Others have leveraged the hand's high degrees of freedom to mimic virtual objects [51, 36] or human/physical motions [7, 33], thereby triggering related effects. However, such gesture-based interactions often require users to perform non-intuitive actions, demanding additional learning. To address this issue, some studies have introduced guidance systems to help users perform gestures more accurately [63], while others have proposed customizable gesture sets to overcome the limitations of predefined, less user-friendly gestures [65]. Nevertheless, assigning individual gestures to each function still imposes a high cognitive load.

The second category expands the range of intuitive hand gestures by integrating physical objects into the interaction process, aligning with the concept of tangible user interfaces [31, 9]. This includes

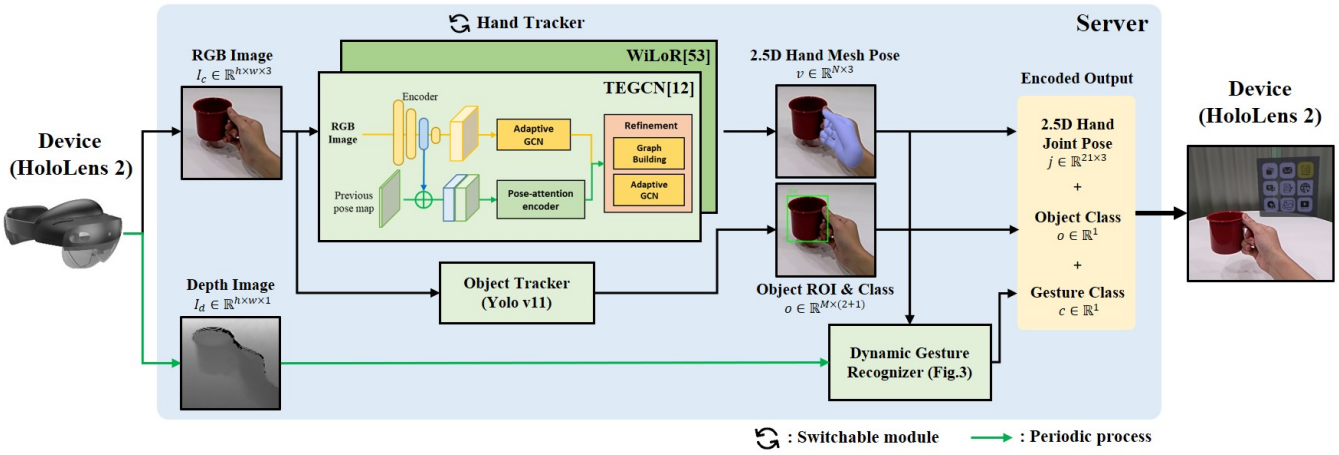

Figure 2: Overview of the proposed system architecture. Image data from HoloLens 2 is processed for hand-object tracking and dynamic gesture recognition. The resulting data are encoded and sent back to the headset for real-time execution.

leveraging environmental features such as object edges, planar surfaces, and 3D object shapes as interactive elements [34, 64, 29]. Other approaches have classified predefined hand-object interactions using additional wrist-worn hardware [39] or have employed physical proxies, such as a spherical device, for AR manipulation tasks [21]. Due to the limited pose-tracking capabilities of current AR HMDs, existing methods have typically focused on either bare-hand conditions or object pose only, excluding the hand. For example, Du et al. [48] explored prototyping tools based on bare-hand gestures and object poses. However, these approaches, constrained by bare-hand pose limitations, have struggled to deliver natural interactions in everyday contexts. In response, we propose an object-mediated hand interaction built on an advanced hand tracker, enabling reliable hand-object interaction on HMD, and present the integrated system along with its application results.

3 Method

We present a fully integrated system within an offloading framework to achieve real-time performance on an HMD, as illustrated in Fig. 2. For each frame, the device captures an RGB image $I_c \in \mathbb{R}^{h\times w\times 3}$ and a depth image $I_d \in \mathbb{R}^{h\times w\times 1}$. Only the RGB images are transmitted on a per-frame basis, while depth images are sent periodically. All encoded images are delivered to the server via TCP-based communication. For each frame, the RGB image is transferred to the selected hand tracker and object tracker. The hand tracker outputs hand-mesh data $v \in \mathbb{R}^{N\times 3}$ representing the 2D positions with relative depth (2.5D) of N vertices. The object tracker generates M regions of interest (RoIs) along with their corresponding class information $o \in \mathbb{R}^{M\times(2+1)}$. The outputs from both trackers, together with the device's depth image I_d, are passed to the gesture recognizer. The module extracts the hand-joint trajectory $j_{\text{traj}} \in \mathbb{R}^{L\times(21\times 3)}$ over a time window L and predicts the gesture class $c \in \mathbb{R}^1$. The recognition results are then encoded and transmitted back to the device. In the following sections, we describe the details of each module and our object-mediated hand-interaction design process.

3.1 Offloading Framework

The proposed offloading framework is implemented with reference to the HoloLens 2 sensor streaming framework [17], utilizing a TCP-based protocol for reliable data transmission between the HoloLens 2 device and the server.

Device The HoloLens 2 captures RGB and depth images in predefined formats and resolutions. RGB frames are acquired in BGRA32 format, while depth frames are obtained in the native short-throw depth format. Depth data is used intermittently within the gesture recognizer to assess interactions with surrounding objects, serving as a trigger for enabling or bypassing the recognition process, and therefore captured every n frames rather than continuously. As analyzed in Section 4.3, varying the resolution and the depth sampling interval n affects communication latency. We selected 360×640 resolution and $n = 10$ for all experiments to balance streaming latency with overall network performance. The streaming pipeline [17] applies hardware-accelerated compression to reduce bandwidth usage: RGB frames are encoded using H.265/HEVC, while depth images are losslessly compressed using PNG encoding. This configuration minimizes transmission delay while preserving the fidelity required for downstream processing.

Server Each stream is decoded with its corresponding method, with RGB frames processed via H.265 and depth images via PNG, before being passed to the internal processing modules. The integrated pipeline produces three outputs: (1) 2.5D joint pose of hand skeleton $j \in \mathbb{R}^{21\times 3}$, (2) gesture class c, and (3) object class o. Although returning the full set of estimated hand mesh vertices does not significantly impact processing speed, such data is not directly usable on the HoloLens 2 in our current setup. Therefore, only the 21×3 joint data is transmitted back to the device for rendering and interaction purposes.

3.2 Hand and Object Tracker

Hand Tracker In organizing the hand tracker, our primary focus was on generalization performance. Specifically, we aimed to ensure robust tracking not only on public benchmark datasets but also in real-world environments where evaluation and application deployment would occur. This requirement was particularly critical in scenarios involving significant hand–object occlusion during interaction. Two candidate modules were considered: TEGCN [12] and WiLoR [53]. Both systems take only RGB image as input and estimate the 2.5D hand mesh vertices. TEGCN [12] is effective in highly dynamic situations with severe occlusion, including fast motion, and operates with relatively high inference speed. WiLoR [53] offers stable performance across a broad range of conditions. Given that our framework allows seamless integration of arbitrary modules on the server side, and considering the complementary strengths of the two trackers, we designed fully integrated system to include both modules. This enables users to switch between trackers according to their specific application requirements. To ensure consistent hand pose tracking across multiple users in quantitative evaluation, we employed the WiLoR [53] tracker exclusively. For application demonstrations requiring faster responsiveness, we primarily utilized TEGCN, exploiting the framework's

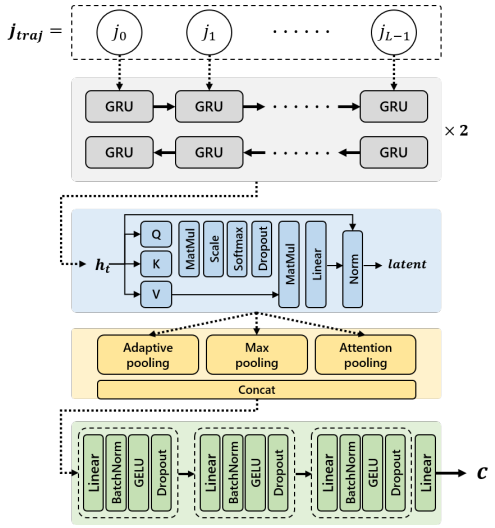

Figure 3: Schematic diagram of the dynamic gesture recognizer architecture. Length of pose history L is set to 16 based on preliminary experiments.

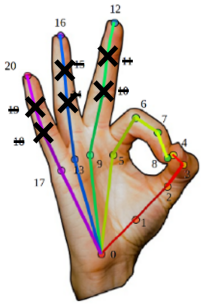

Figure 4: Visualization of the selected joint subset among the 21 hand joints. A total of 15 joints are utilized, corresponding to indices 0–8, 9, 12, 13, 16, 17, and 20.

tracker switching capability.

For both modules, an RGB image $I_c \in \mathbb{R}^{h\times w\times 3}$ is provided as input, from which the mesh data $v \in \mathbb{R}^{N\times 3}$ is extracted, and hand detection is adapted from the *detection-by-tracking* approach proposed in [28]. In the original method, the future hand pose is extrapolated from the previous two tracked poses, and the hand ROI is set around the predicted position. In dynamic motion scenarios, however, we found that this prediction often failed to fully encompass the actual hand location. To mitigate this issue, we center the ROI on the most recent tracked pose and periodically apply full-image hand detection at fixed frame intervals, thereby enhancing re-initialization performance.

Object Tracker In designing the object tracker, we recognized that extensive object information could enable more specific, application-level interactions. However, as the primary goal of this study is to facilitate intuitive hand interactions by combining accurate hand pose data with object information in hand–object interaction contexts, we focused solely on capturing basic object attributes: the ROI center point and class label. To ensure both speed and robustness in typical environments, we adopt YOLO [54], using the release of YOLO-v11. Since the model does not account for hand occlusions, we address expected tracking failures in full-grasp situations by leveraging only the tracked information obtained immediately before direct interaction with the object.

3.3 Dynamic Gesture Recognizer

The objective of the dynamic gesture recognizer module is to classify continuous hand motion sequences using multiple input modalities, including 2.5D hand joints, depth images, object ROIs, and object class data. The recognition process consists of several stages. First, to optimize resource usage, intermittently captured depth images are used to determine whether the hand is interacting with a nearby object. Gesture recognition is then performed only when such interaction is detected. Second, joint data acquired from each frame are accumulated to construct a pose history, which is normalized and reduced to a subset of joints most relevant to the task. Next, the preprocessed partial joint pose history is fed into an enhanced GRU-based model to estimate the gesture class. Finally, a decision filter is applied to mitigate the effect of intermittent prediction noise, and the resulting output is transmitted to the device.

To achieve robust recognition performance across diverse conditions, it is essential to account for the inherent variability in gesture execution duration. Accordingly, we fix the input sequence length L of the recognizer to be shorter than the typical duration of hand actions, thereby training the model on partial gesture sequences rather than complete ones. Since the training dataset encompasses gestures performed at varying speeds, user motions are consistently captured within the recognizer's input window regardless of execution velocity. However, partial inputs may fail to encapsulate all discriminative features of the full gesture, potentially leading to increased error rates. To mitigate this issue, we introduce a decision filtering mechanism, detailed in the Decision Logic section.

3D Proximity Check To reduce false-positive errors in free-hand environments, we implemented a lightweight proximity-check module to trigger the gesture recognizer. Assuming the wrist remains visible in the HMD's egocentric view, a virtual 3D box is defined around the hand based on the wrist's 3D position. If the center of a detected object lies within this box, the system infers a high likelihood of hand–object interaction and identifies the nearest object as the one currently being interacted with, passing this information to subsequent processes.

Preprocessing Recognizing dynamic gestures involving continuous movements requires retaining a history of hand poses across consecutive frames. Instead of using all 21 hand joints, we utilize only a subset, as shown in Fig. 4: the thumb, index finger, and the root–tip joints of the remaining fingers, which are most relevant to hand–object interactions. This selection helps focus the recognizer's attention on the intended gesture. Pose data from the current frame is accumulated together with a history of L previous poses. Furthermore, joint angles are computed from the 2.5D joint positions and included as additional features to enrich the input space. For each pose history, the oldest pose serves as the reference frame, and 2D normalization is applied.

Network Architecture While GRU-based models have shown a strong ability to capture temporal dependencies in sequential hand gesture recognition, we build upon the DeepGRU architecture [44], a widely used GRU-based recognizer known for its efficient modeling of temporal patterns and real-time applicability. The original DeepGRU [44] offers sufficiently fast inference; however, its unidirectional information flow and limited feature aggregation indicate potential for further improvement in generalization performance. To address these issues, we propose enhanced GRU-based gesture recognizer as illustrated in Fig. 3. Given the pose history input $j_{traj} = (j_0, j_1, ..., j_{L-1})$, the proposed network is formulated as follows:

First, we replace its deeper 3-layer unidirectional GRU with a 2-layer bidirectional GRU using 128 units per direction, producing 256-dimensional representations. This design reduces computational cost while enabling the model to exploit both past and future context for a richer understanding of gesture dynamics.

$$\overrightarrow{h_t} = \mathrm{GRU}_{forward}(j_t, \overrightarrow{h_{t-1}}) \tag{1}$$

$$\overleftarrow{h_t} = \mathrm{GRU}_{backward}(j_t, \overleftarrow{h_{t+1}}) \tag{2}$$

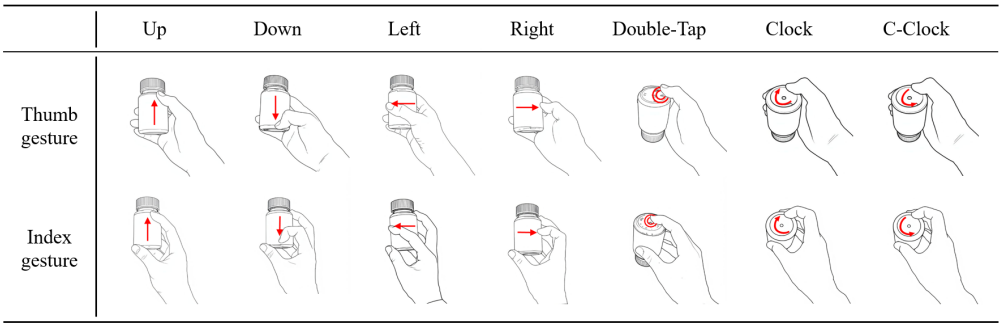

Figure 5: Proposed gesture configuration. Any graspable object can serve as a target and each hand-gesture category can be combined with corresponding object classes to map to a variety of AR functions.

$$h_t = [\overrightarrow{h_t}; \overleftarrow{h_t}] \tag{3}$$

where $\overrightarrow{h_t} \in \mathbb{R}^{128}$ is the forward hidden state, $\overleftarrow{h_t} \in \mathbb{R}^{128}$ is the backward hidden state, $h_t \in \mathbb{R}^{256}$ is the concatenated bidirectional representation, and $[;]$ denotes concatenation operation. Next, given a hidden state h_t, a simple self-attention mechanism is integrated to focus on the most discriminative temporal segments, improving the capture of long-range dependencies.

$$Q = h_t W_Q, \quad K = h_t W_K, \quad V = h_t W_V \tag{4}$$

$$\text{SelfAttention}(h_t) = Norm(\text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V + h_t) \tag{5}$$

where $W_Q, W_K, W_V \in \mathbb{R}^{256\times256}$ are learned projection matrices, $d_k = 256$ is the key dimension, and residual connection with layer normalization $Norm$ is applied. We also employed a mixed pooling strategy that integrates adaptive average pooling, max pooling, and learnable attention-based pooling, enabling the retention of both global and local feature information. Finally, the concatenated features are passed through a multi-layer classifier composed of sequential normalization, dropout, and progressively narrowing fully connected layers with GELU activations and batch normalization at each stage, before the final linear projection to the target gesture classes. This design improves generalization by reducing overfitting and stabilizing feature distributions across layers. The effects of the proposed network modifications were validated in Section 4.3.

Decision Logic To mitigate the persistent issue of false-positive predictions in the recognition model, we classify gestures based on partial hand motion sequences rather than complete sequences. This design allows the system to maintain consistent predictions when a user intentionally performs a gesture, while unintentional false positives typically appear as discontinuous predictions, making them easier to filter. For the final decision, a gesture class is accepted only if it is predicted consecutively for a predefined number of frames. This decision-filtering mechanism effectively suppresses intermittent noise, resulting in more stable and reliable recognition.

3D Pose Lifting on Device Since the predicted 2.5D pose data contain only relative depth information, a 3D pose lifting step is required to reconstruct the full 3D pose for visualization or application use. To improve efficiency, this process is performed directly on the device after the encoded result data are transmitted. Leveraging the fact that the wrist remains visible in most egocentric HMD views, we extract its depth information to enable accurate 3D hand pose lifting. Based on the transmitted 2D wrist position, the corresponding depth value is obtained by referencing the aligned depth image using the HMD's intrinsic camera parameters. When depth holes cause missing data, the wrist depth is replaced with the nearest valid surrounding value to ensure continuity and robustness in the reconstructed 3D pose.

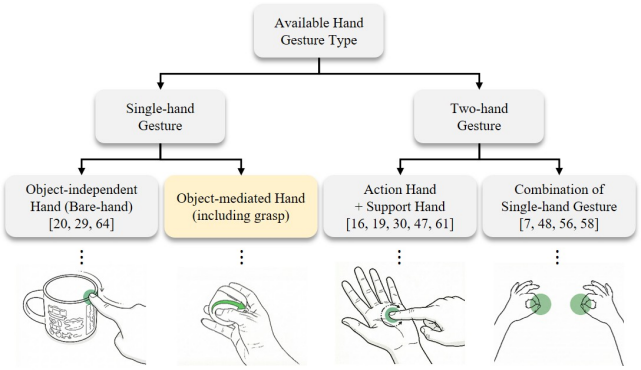

Figure 6: Hand gesture taxonomy with associated references and representative concept illustrations.

3.4 Interaction Design and Data Collection

Design Goal We aim to define straightforward and intuitive hand gestures for interacting with objects, incorporating existing actions while establishing novel object-mediated interactions that can be mapped to specific functions for various object types. To achieve this goal, we developed a taxonomy that encompasses our target gesture groups by synthesizing insights from prior literature. Drawing upon the open challenges in vision-based recognition identified by Zabulis et al. [68], we adopted fundamental classification keywords such as static, continuous, single-hand, and two-hand gestures, reflecting the technical constraints of that context. Furthermore, we refined the findings of Gong et al. [27], who established user-defined gesture sets for spatial tangible interactions. In doing so, we prioritized hand-centric modalities, deliberately distinguishing them from object-oriented gestures where outcomes depend on the object's geometry or pose. Integrating these with the classification metrics (static, dynamic, finger, hand, and bi-hand) employed by Gavgiotaki et al. [23], we present our comprehensive taxonomy in Fig. 6. Instead of relying on the traditional dichotomy of static and dynamic gestures, we focused on the relational dynamics between the entities involved in the interaction. Our taxonomy primarily distinguishes between single-hand and two-hand modalities. Within the single-hand category, we differentiated gestures based on their relationship with surrounding objects: object-independent gestures [64, 29, 20], which are performed autonomously like bare-hand gestures, yet can leverage object context, and object-mediated gestures, which address a critical gap in existing literature, defined as hand gestures occluded by an object that cannot be physically expressed without the object's presence. Regarding two-hand gestures, we adopted a role-based classification to align with existing frameworks. This includes the action-plus-support configuration [19, 47, 30, 61, 16], where one hand serves as an auxiliary reference for the active hand, and the combinatorial configu-

ration [7, 58, 48, 56], where the interaction emerges from the simultaneous execution of two independent single-hand gestures.

Building upon this framework, we focused on defining and implementing object-mediated hand gestures, explicitly targeting a distinct category that prior research has not adequately addressed. As shown in Fig. 5, we identified seven fundamental hand gestures executable while holding objects, centered on the thumb and index finger—the primary digits used for interaction. Our objective is to achieve robust recognition of these proposed gestures regardless of the shape or type of the interacted object, and to assign each gesture to an appropriate AR function based on the target object's properties, thereby enabling seamless object-mediated interaction.

Dataset Acquisition To collect the training and test datasets for gesture recognition, we developed a custom dataset acquisition system on the HoloLens 2 using the proposed offloading framework. To ensure that data were captured under conditions identical to the actual inference environment, dummy data were fed into each tracker and the gesture recognizer during recording, while RGB and depth images for every frame were stored in a buffer. The training dataset comprised short clips, each containing a single gesture performed once, whereas the test dataset consisted of continuous sequences with gestures presented in random order. Participants were encouraged to explore possible variations for each gesture class and record short clips repeatedly. As our objective was to demonstrate the robust operation and recognition performance of our approach, we focused on capturing a wide range of gesture variations performed with everyday objects, taking into account hand shape diversity and the gender distribution of participants. Five participants (three male and two female) with varying hand sizes were recruited, each provided with a private set of easily graspable objects of diverse geometries and sizes. For every object, all combinations of predefined gesture classes and interactable fingers were performed at least four times. Participants were instructed to execute only plausible gestures and to vary camera-relative distance, angle, and range to reflect the frequent viewpoint changes in egocentric HMD scenarios. Additionally, samples for the "Natural" class were collected to enhance classification accuracy. For the test dataset, images were captured from the HoloLens 2 while participants received instructions via a PC interface. Participants were shown the target gesture class and asked to perform it with the right hand, following randomly prompted classes. Each participant used a randomly selected private object, and nine sequences were recorded per gesture. Each sequence lasted 3 seconds to include the preparation phase before gesture execution. This duration was set as the upper bound for recognizable gestures, as finger-level hand gestures taking 2-3 seconds would exceed typical execution speeds. This study was approved by the Institutional Review Board of Korea Advanced Institute of Science and Technology(Approval No.KH2024-180). Informed consent was obtained from all participants.

Dataset Augmentation In addition to commonly used augmentation techniques such as global scaling and 2D rotation, we applied a gradual transition method to the collected dataset. From preliminary experiments, the global scaling factor was set within the range of 0.7–1.3, and the global rotation was applied around the wrist joint within $\pm 15^{\circ}$. The gradual transition was implemented by generating a random 3D vector and applying the same incremental change to the hand pose in each subsequent frame, thereby accumulating the transformation over time. The range of the random vector was set to ± 2. This approach aims to simulate cumulative pose shifting, covering scenarios in which the entire hand moves while performing a gesture. After augmentation, 3D normalization was performed based on the wrist pose in the first frame, and the resulting data were used for training.

4 Experiment

Since no suitable public datasets were available for training the proposed gesture group, we collected a custom dataset as described in Section 3.4. To demonstrate robust recognizer performance at practically applicable levels, we evaluated recognition accuracy and per-step execution time of the proposed framework using this dataset. Furthermore, we conducted an ablation study to validate the architecture of the proposed module and performed a detailed latency analysis of the entire system implemented on the HoloLens 2. To assess the generalizability of the gesture recognition model, we also report accuracy comparisons on the SBU Kinect Interactions dataset [67].

The primary evaluation metrics were the micro F1-score and standard deviation, with 5-fold cross-validation performed separately for each subject. The micro F1-score, which combines precision and recall into a single measure, was computed for our gesture recognition task as follows. A prediction was counted as a true positive (TP) if the predicted gesture class matched the ground truth (GT) class, and as a false positive (FP) if it differed. A false negative (FN) was recorded when the model failed to recognize any gesture class other than the "Natural" class, and a true negative (TN) when the test data contained no gesture. As the recognizer operates continuously over each test sequence, a TP could be counted at most once per sequence, whereas multiple FPs could occur. This strict scoring scheme was adopted to ensure a rigorous assessment of recognition accuracy.

4.1 Implementation Details

HMD–Server Configuration The HMD application running on the HoloLens 2 was hosted on a server with an AMD Ryzen 9 7950X 16-core CPU @ 4.50 GHz, 64 GB of RAM, and an NVIDIA GeForce RTX 4090 GPU. The system was connected via a 5 GHz wireless network, which theoretically supports data transfer speeds of up to 867 Mbps. Network quality tests measured an upload speed of 122.21 Mbps and a download speed of 50.23 Mbps.

Training We employed the Adam optimizer with a learning rate of 0.001, a decay rate of 0.001, and a batch size of 64, together with a cosine annealing learning rate scheduler. Instead of the commonly used categorical cross-entropy loss, we adopted label-smoothing cross-entropy to mitigate overfitting and improve generalization, particularly given the limited size of our dataset. The model was trained for 50 epochs in the *All-case* setting, while for the per-subject folds, early stopping based on validation loss was applied. The *All-case* model was used for all evaluation procedures except the ablation study.

4.2 Quantitative Results

Recognition Accuracy Fig. 7 and Tab. 1 present the aggregated prediction results across all subjects, summarised for each gesture in terms of mean, standard deviation, accuracy, and F1-score. In the confusion matrices, the left panel corresponds to gestures performed with the thumb, and the right panel to those performed with the index finger, both obtained using the WiLoR [53] hand tracker. Note that, numerous gesture predictions were accumulated for each test sequence. While error classes could be counted multiple times within a sequence, the correct class was counted at most once.

Overall, gestures performed with the thumb achieved higher recognition rates. In contrast, certain index-finger gestures, such as *index down* and *index right*, showed lower accuracy. These motions are less familiar and more difficult to perform, and the shape of the object can further restrict the range of index-finger movements among the seven gesture types considered. Consequently, participants were likely unfamiliar with these gestures and produced motion patterns that deviated from the intended forms. The most frequent misclassifications occurred between gestures with opposite directions (e.g., *up* vs. *down*, *left* vs. *right*) or between gestures

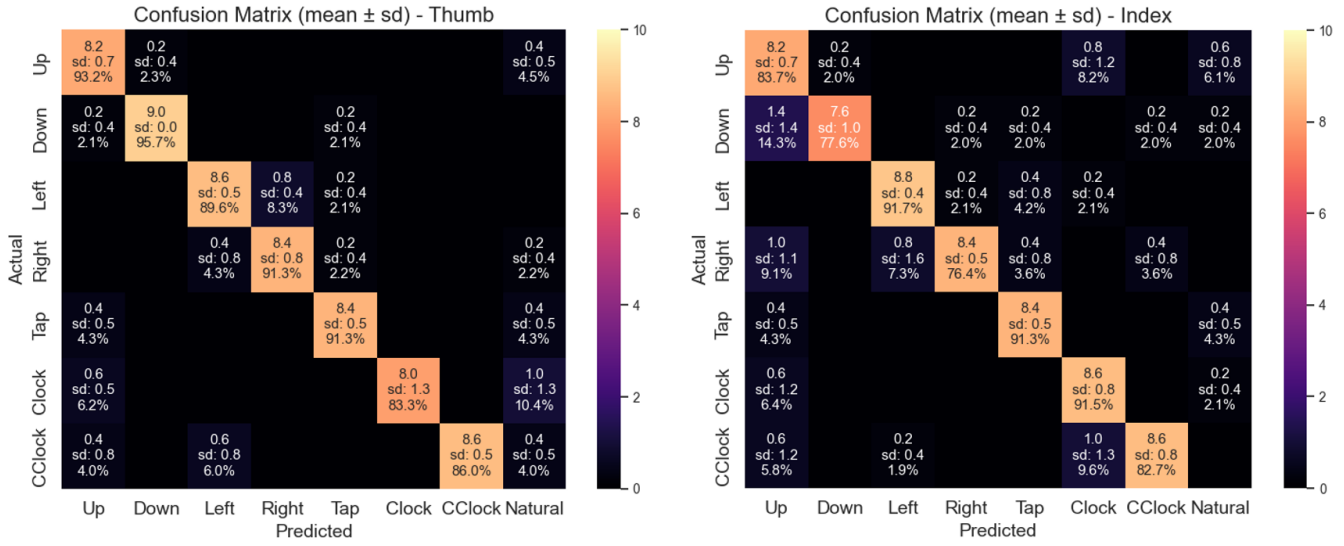

Figure 7: Confusion matrices of all subjects, interaction with thumb(1st row) and with index(2nd row). *Tap* denotes a Double-tap gesture.

	Interacting Finger	Up	Down	Left	Right	Tap	Clock	CClock
F1-score	Thumb	0.965 ± 0.033	0.978 ± 0.029	0.945 ± 0.036	0.955 ± 0.069	0.955 ± 0.047	0.909 ± 0.087	0.925 ± 0.046
	Index	0.911 ± 0.077	0.847 ± 0.099	0.957 ± 0.043	0.866 ± 0.122	0.930 ± 0.075	0.956 ± 0.063	0.905 ± 0.064

Table 1: Mean recognition accuracy for all subject per gesture.

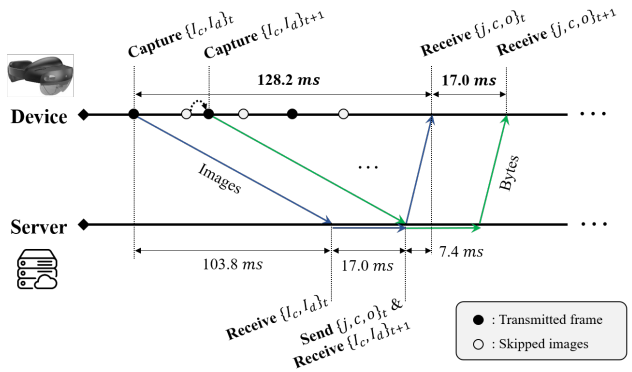

Figure 8: Visualization of device-server latency and framerate with TEGCN hand tracker. Colored line denotes each distinct data flow.

where one motion was partially contained within another (e.g., *tap* vs. *up*). Such errors are likely influenced by the window size of the data sequence provided to the gesture recognizer. Increasing the window size generally improves accuracy by enabling the model to capture longer-range motion patterns, but it also directly increases the latency of recognition results presented to the user, requiring a trade-off.

Tab. 2 summarises the accuracy for the entire gesture set on a per-subject basis. Hand size information was also collected as a potential constraint on the gesture motion range. No significant performance variation was observed in relation to hand morphology, including overall hand size and palm-to-finger ratios. We attribute this to the use of personally familiar objects during data collection, which likely elicited broadly similar motion patterns across participants. The exception was Subject 2, who exhibited comparatively lower accuracy due to motion patterns that deviated from the standard gesture forms demonstrated during data collection.

Integrated System Latency Tab. 3 summarises the latency of each processing stage executed on the server, while Tab. 4 presents the communication latency between the device and server under different RGB resolutions and depth sampling intervals. As shown in Tab. 3, we measured the preprocessing, inference, and postprocessing times for two switchable hand tracker models. In both cases, inference accounted for the largest proportion of the total execution time. WiLoR [53] demonstrated high tracking stability across general environments but required a relatively slower 29.65 ms compared to TEGCN [12]. In contrast, TEGCN [12] achieved accurate tracking in complex environments with a fast inference time of 7.82 ms, although it exhibited reduced stability, with some jittering observed even under static conditions. When aggregating the total processing time from image acquisition to output generation into framerate, WiLoR [53] achieved 25.58 FPS and TEGCN [12] 58.79 FPS, indicating that both models met real-time performance requirements.

Tab. 4 compares communication latency across different data transmission configurations, which are directly influenced by the wireless Wi-Fi environment of the offloading framework. First, we evaluated the impact of image resolution. Although latency did not scale proportionally with the number of transmitted pixels, it increased noticeably as resolution rose. Since the server-side trackers require sufficiently large image inputs for optimal performance, we fixed the captured image resolution at 640×360. Next, we examined the effect of adjusting the sampling interval for periodically captured depth images on communication throughput. Even when depth images were sampled more frequently, throughput did not increase proportionally. This behavior is likely due to fixed processes in the established TCP communication pipeline, such as socket calls and packetization, that are independent of the data volume. Nevertheless, capturing depth images every frame yielded approximately twice the throughput compared to capturing once every 10 frames. As depth images are not required for every frame in our system, we set the sampling interval to $n = 10$ to balance the trade-off between communication latency and recognition interval. For byte format data transmitted from the server to the device, such as hand pose, gesture, and object class, the payload size was sufficiently small, resulting in a latency of only about 7.35 ms.

	Subject 1	Subject 2	Subject 3	Subject 4	Subject 5	Average
Hand size (cm)	19.0 / 8.3 / 13.5 / 6.7	18.3 / 8.4 / 12.5 / 6.5	16.4 / 7.2 / 12.8 / 6.5	17.8 / 7.8 / 12.5 / 5.7	18.0 / 7.9 / 13.2 / 6.6	-
F1-score	0.940 ± 0.083	0.897 ± 0.073	0.933 ± 0.059	0.937 ± 0.055	0.944 ± 0.070	0.930 ± 0.017

Table 2: Mean recognition accuracy per subject. Hand size categories are based on bone lengths between joint pairs (see Fig. 4 for joint indices): wrist-MTIP(0-12), MMCP-MTIP(9-12), wrist-TTIP(0-4), TMCP-TTIP(2-4).

	Execution time (ms)	
	WiLoR	**TEGCN**
Detect hands	2.74	
Preprocess image	1.72	2.56
Inference model	29.65	7.82
Postprocess results	1.22	0.13
3D Proximity check	1.28	
Gesture recognizer	2.48	
Total	**39.09 ms (25.58 FPS)**	**17.01 ms (58.79 FPS)**

Table 3: Execution time per server process stage and selected hand tracker model.

		Latency	
		Device → Server	**Server → Device**
RGB resolution (n =NaN)	1920×1080	757.6	**7.35**
	1080×720	122.0	
	640×360	**92.2**	
	424×240	78.4	
Depth sampling interval	$n = 1$	227.8	
	$n = 5$	124.6	
	$n = 10$	**103.8**	
	n =NaN	92.2	

Table 4: Communication latency analysis on HoloLens 2. Each latency value represents the average over more than 100 frames. n denotes the sampling interval of the depth images.

Fig. 8 visualizes the latency and framerate along the data flow of the proposed framework, using the TEGCN [12] model as an example. For clarity, the segment lengths in the figure do not represent actual time durations. In this configuration, the HoloLens 2 continuously transmits captured data at 88.5 FPS, while the server repeatedly processes the most recent data stored in its buffer (small curved dashed arrow) and returns the results to the device. The pipeline latency, which is the delay from image capture to the availability of the processed result, was measured at 128.2 ms; the output framerate, which represents the number of processed results produced per second, was 17.0 ms(58.8 FPS). Compared with the latency requirement reported in a previous study [8], which specifies that physical interaction tasks such as touch, gesture, and direct manipulation require less than 50 ms, the measured pipeline latency is relatively high. Notably, communication latency accounts for 87.8% of the total pipeline latency. This indicates that, although server-side processing has been optimized to improve the framerate perceived by the user, the overall latency remains dominated by network transmission delays. We expect this to improve with future advancements in wireless communication speed. Qualitative feedback further revealed that gesture recognition results were often produced while the user was still performing the gesture. This is due to the proposed gesture recognizer being trained on partial gesture trajectories, a design choice presumed to have partially mitigated the impact of pipeline latency.

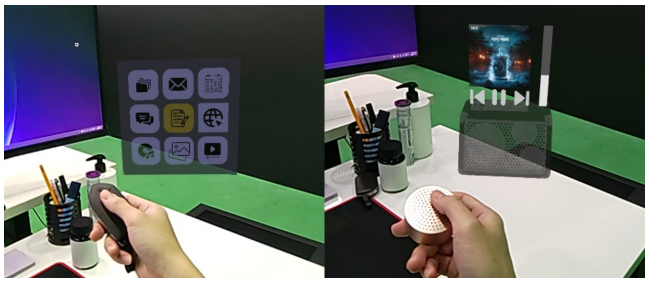

Figure 9: Live-captured views from HoloLens 2 demonstrating the proposed framework: manipulating a sample AR panel (left) and interacting with an AR music application (right).

4.3 Ablation Study

In this section, we evaluate the impact of the proposed improvements to our gesture model and validate its performance by additionally assessing it on a public human activity dataset previously used with the baseline model. Tab. 5 reports the results of a 5-fold cross-validation, where each fold uses one of the five collected subjects as the test dataset and the remaining subjects for training. For each model configuration, we report the number of parameters and the measured latency. The individual improvements introduced in the proposed model, as described in Section 3.3, are denoted as follows: reducing GRU dimensions (**R**), replacing the original attention with a self-attention module (**S**), employing a bidirectional GRU (**B**), applying a mixed pooling strategy (**M**), and using partial keypoints of the hand joints (**P**). The results show a consistent increase in accuracy as each improvement is applied. In the final proposed model, the standard deviation across subjects was the lowest among all configurations, indicating improved generalization, while also achieving the highest accuracy in each individual fold.

To demonstrate that the proposed model does not overfit to our custom hand gesture dataset, we further trained and evaluated it on the SBU Kinect Interactions dataset [67], a complex human activity dataset used for evaluation in the baseline DeepGRU [44] study. The results are presented in Tab. 6. While accuracy remained comparable to other methods, the reduced standard deviation indicates improved generalization. This demonstrates that our model effectively generalizes beyond the custom hand gesture dataset to general human gesture data and, though based on a single comparison, suggests potentially superior cross-subject generalization compared to the baseline.

4.4 Applications and Discussion

Applications To demonstrate the practical utility of the proposed system, we developed example applications: manipulating an AR panel and controlling an AR music application, as illustrated in the object-mediated hand interaction concept in Fig. 1. We assumed a scenario where a user wears a HoloLens 2 while performing routine tasks at a desk, enabling them to execute corresponding interactions by utilizing appropriate surrounding objects as needed. For instance, users utilized cuboid objects resembling a remote controller to move and select icons on the panel as shown in Fig. 9 (left), and employed cylindrical objects resembling a speaker to perform actions such as switching tracks or adjusting volume through predefined gestures as shown in Fig. 9 (right). Both applications demonstrated stable gesture recognition and provided real-time feedback.

Model	# of parameter	Latency (ms)	5-fold Cross-validation (%)					Mean Accuracy (%)	SD
			Sub 1	Sub 2	Sub 3	Sub 4	Sub 5		
base(DeepGRU)	3.8M	1.71	91.95	92.77	76.47	90.05	94.24	89.09	6.46
base+R	0.4M	1.05	90.32	94.46	77.94	91.06	93.20	89.40	5.92
base+R+S	1.0M	1.11	91.39	93.56	75.72	94.76	95.09	90.10	7.31
base+R+S+B	0.8M	0.87	91.74	94.25	80.80	94.68	92.69	90.83	5.12
base+R+S+B+M	1.1M	1.27	92.98	96.03	79.68	94.42	94.56	91.53	6.00
base+R+S+B+M+P (ours)	1.1M	1.20	93.57	96.52	83.15	94.71	94.57	92.50	4.77

Table 5: Ablation study on the gesture recognizer model architecture. The definitions of the model option notations (R, S, B, M, P) are provided in the main text and SD denotes standard deviation.

Modality	Method	Accuracy
Pose	CNN + Kernel Feature Maps [59]	94.3
	GCA-LSTM (stepwise) [43]	94.9
	LSTM + FA + VF [22]	95.0
	VA-LSTM [69]	97.2
	DeepGRU [44]	95.2 ± 2.8
	Ours	95.1 ± 1.6

Table 6: Ablation study of proposed gesture recognizer on SBU Kinect dataset [67].

Beyond these examples, the system can be extended to a variety of real-world scenarios that leverage the natural affordances of physical objects. For instance, a presenter wearing an HMD could manipulate augmented presentation content using a pen while standing in front of a writable board, or a user watching a movie in an AR environment could control playback using a cup in hand. Such applications highlight the potential of associating intuitive, object-specific interactions with augmented content in everyday contexts.

Discussion We outline the findings from employing a real-time hand tracker to facilitate daily object-mediated interactions and identify factors contributing to robust performance. First, we observed that occluded hand joints only need to be sufficiently accurate, whereas visible joints should be prioritized for precision. This suggests that, during training, weighting the loss function by joint visibility might improve interaction performance. Second, reducing false positives emerged as a key challenge, particularly when prioritizing natural motion dynamics. We observed the effectiveness of strategies such as decision filters (e.g., 3D proximity checking), incorporating contextual depth, and collecting noisy motion data. Third, understanding the typical motion patterns exhibited by users was important. For instance, participants often aligned their motions with the axes of objects. Considering these tendencies, incorporating object pose information may offer a more robust approach, effectively accounting for significant motion differences within the same gesture class. Finally, optimizing the temporal window size was crucial to balance the trade-off between recognition accuracy and latency. Overall, our approach improved recognition reliability and responsiveness by capturing partial gestures and applying continuous filtering.

4.5 Limitations

As our primary focus was developing a generalizable hand and gesture tracker for commercial HMDs, we validated tracking stability and recognition accuracy using our custom dataset and cross-validation, but did not conduct formal user studies to assess the usability of the proposed interaction. Furthermore, the distinctive features of our approach—including object-held hand and arbitrary graspable objects—precluded direct comparison with existing methods, thus limiting access to public gesture datasets that consist of larger human populations. We expect this work to inspire future efforts in collecting large-scale datasets for object-mediated interaction research.

During implementation with multiple users and real-world applications, we observed instances of tracking and gesture recognition failures due to the limited field of view(FoV) of HoloLens 2. A common cause was user motions extending beyond the FoV of the HMD camera or being only partially captured. As a result, users were compelled to raise their arms unnaturally or lower their heads to keep their hands within the HMD's FoV. To minimize such forced motions, we constrained both dataset collection and application scenarios to seated, desk-based conditions; this reduced but did not fully eliminate the problem. However, this is a hardware-dependent limitation related to the natural range of user activity, and it can be mitigated by leveraging downward-facing cameras available on more recent devices such as Apple's Vision Pro [1] and Samsung's Galaxy XR [2].

5 Conclusion

In this work, we presented a unified offloading-based framework for advanced hand and gesture tracking, enabling robust object-mediated interaction on resource-constrained wearable AR devices. By combining an advanced 3D hand tracker for hand–object interaction with a GRU-based dynamic gesture recognizer, the system maintained stable performance even under challenging occlusion scenarios, which has not been addressed in prior studies. The proposed approach was validated through a custom object-mediated gesture dataset, extensive ablation studies, and real-world deployment on the HoloLens 2, achieving high recognition accuracy, improved generalization, and real-time responsiveness. Furthermore, the modular offloading design allows flexible integration of alternative tracking modules, ensuring adaptability to diverse application domains. With the growing adoption of lightweight wearable glasses with limited onboard resources, this approach offers strong potential for delivering high-performance tracking without overburdening the device. Overall, our work establishes a foundation for scalable, object-aware, and intuitive AR interactions that seamlessly incorporate physical objects as mediators of the AR experience, paving the way for richer and more immersive everyday applications. Our future scope includes porting the proposed framework to heterogeneous HMDs and assessing its practical usability through in-depth user evaluations.

Acknowledgments

This work was supported by Institute of Information communications Technology Planning Evaluation (IITP) grant funded by the Korea government(MSIT) (RS-2019-II191270, and RS-2024-00397663).

REFERENCES

[1] Apple Vision Pro. https://www.apple.com/apple-vision-pro/. 1, 9

[2] Galaxy XR. https://www.samsung.com/sec/xr/galaxy-xr/galaxy-xr/. 9

[3] Hololens 2. https://www.microsoft.com/hololens. 1

[4] Meta Quest Pro. https://www.meta.com/kr/en/quest/quest-pro/. 1

[5] Meta Ray-Ban. https://www.meta.com/kr/ai-glasses/. 1

[6] VIRTUE XR. https://luma.viture.com/. 1

[7] R. Arora, R. H. Kazi, D. M. Kaufman, W. Li, and K. Singh. Magicalhands: Mid-air hand gestures for animating in vr. In *Proceedings of the 32nd annual ACM symposium on user interface software and technology*, pp. 463–477, 2019. 2, 6

[8] C. Attig, N. Rauh, T. Franke, and J. F. Krems. System latency guidelines then and now–is zero latency really considered necessary? In *International Conference on Engineering Psychology and Cognitive Ergonomics*, pp. 3–14. Springer, 2017. 8

[9] M. Billinghurst, H. Kato, I. Poupyrev, et al. Tangible augmented reality. *Acm siggraph asia*, 7(2):1–10, 2008. 2

[10] J. Bruna, W. Zaremba, A. Szlam, and Y. LeCun. Spectral networks and locally connected networks on graphs. *arXiv preprint arXiv:1312.6203*, 2013. 2

[11] X. Chen, Y. Liu, Y. Dong, X. Zhang, C. Ma, Y. Xiong, Y. Zhang, and X. Guo. Mobrecon: Mobile-friendly hand mesh reconstruction from monocular image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 20544–20554, 2022. 2

[12] W. Cho, T. Ha, I. Jeon, J. Jeon, T.-K. Kim, and W. Woo. Temporally enhanced graph convolutional network for hand tracking from an egocentric camera. *Virtual Reality*, 28(3):143, 2024. 2, 3, 7, 8

[13] H. Choi, G. Moon, and K. M. Lee. Pose2mesh: Graph convolutional network for 3d human pose and mesh recovery from a 2d human pose. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VII 16*, pp. 769–787. Springer, 2020. 2

[14] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*, 2014. 2

[15] S. Das, A. Nasser, and K. Hasan. Fingerbutton: Enabling controller-free transitions between real and virtual environments. In *2023 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pp. 533–542. IEEE, 2023. 2

[16] P. Dhaka, K. Katsuragawa, K. Hasan, et al. Exploring augmented reality user interface transitions across mid-air, on-body and physical surfaces. In *2024 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pp. 1177–1186. IEEE, 2024. 5

[17] J. C. Dibene and E. Dunn. Hololens 2 sensor streaming. *arXiv preprint arXiv:2211.02648*, 2022. 3

[18] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 2

[19] R. Du, A. Olwal, M. Le Goc, S. Wu, D. Tang, Y. Zhang, J. Zhang, D. J. Tan, F. Tombari, and D. Kim. Opportunistic interfaces for augmented reality: Transforming everyday objects into tangible 6dof interfaces using ad hoc ui. In *CHI Conference on Human Factors in Computing Systems Extended Abstracts*, pp. 1–4, 2022. 2, 5

[20] C. Dupré, C. Appert, S. Rey, H. Saidi, and E. Pietriga. Tripad: Touch input in ar on ordinary surfaces with hand tracking only. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems*, pp. 1–18, 2024. 5

[21] D. Englmeier, J. Dörner, A. Butz, and T. Höllerer. A tangible spherical proxy for object manipulation in augmented reality. In *2020 IEEE conference on virtual reality and 3d user interfaces (VR)*, pp. 221–229. IEEE, 2020. 3

[22] Z. Fan, X. Zhao, T. Lin, and H. Su. Attention-based multiview re-observation fusion network for skeletal action recognition. *IEEE Transactions on Multimedia*, 21(2):363–374, 2018. 9

[23] D. Gavgiotaki, S. Ntoa, G. Margetis, K. C. Apostolakis, and C. Stephanidis. Gesture-based interaction for ar systems: a short review. In *Proceedings of the 16th International Conference on PErvasive Technologies Related to Assistive Environments*, pp. 284–292, 2023. 5

[24] L. Ge, Z. Ren, Y. Li, Z. Xue, Y. Wang, J. Cai, and J. Yuan. 3d hand shape and pose estimation from a single rgb image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10833–10842, 2019. 2

[25] J. Gilmer, S. S. Schoenholz, P. F. Riley, O. Vinyals, and G. E. Dahl. Neural message passing for quantum chemistry. In *International conference on machine learning*, pp. 1263–1272. PMLR, 2017. 2

[26] M. Giordano, O. Georgiou, B. Dzidek, L. Corenthy, J. R. Kim, S. Subramanian, and S. A. Brewster. Mid-air haptics for control interfaces. In *Extended Abstracts of the 2018 CHI Conference on Human Factors in Computing Systems*, pp. 1–8, 2018. 1

[27] W. Gong, S. Santosa, T. Grossman, M. Glueck, D. Clarke, and F. Lai. Affordance-based and user-defined gestures for spatial tangible interaction. In *Proceedings of the 2023 ACM Designing Interactive Systems Conference*, pp. 1500–1514, 2023. 5

[28] S. Han, B. Liu, R. Cabezas, C. D. Twigg, P. Zhang, J. Petkau, T.-H. Yu, C.-J. Tai, M. Akbay, Z. Wang, et al. Megatrack: monochrome egocentric articulated hand-tracking for virtual reality. *ACM Transactions on Graphics (ToG)*, 39(4):87–1, 2020. 2, 4

[29] F. He, X. Hu, J. Shi, X. Qian, T. Wang, and K. Ramani. Ubi edge: Authoring edge-based opportunistic tangible user interfaces in augmented reality. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*, pp. 1–14, 2023. 2, 3, 5

[30] Z. He, X. Wang, Y. Shi, C. Hsia, C. Liang, and C. Yu. Palmpad: Enabling real-time index-to-palm touch interaction with a single rgb camera. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*, pp. 1–16, 2025. 5

[31] S. J. Henderson and S. Feiner. Opportunistic controls: leveraging natural affordances as tangible user interfaces for augmented reality. In *Proceedings of the 2008 ACM symposium on Virtual reality software and technology*, pp. 211–218, 2008. 2

[32] J. D. Hincapié-Ramos, X. Guo, P. Moghadasian, and P. Irani. Consumed endurance: a metric to quantify arm fatigue of mid-air interactions. In *Proceedings of the SIGCHI conference on human factors in computing systems*, pp. 1063–1072, 2014. 1

[33] C.-W. Hung, R.-C. Chang, H.-S. Chen, C. H. Liang, L. Chan, and B.-Y. Chen. Puppeteer: Exploring intuitive hand gestures and upper-body postures for manipulating human avatar actions. In *Proceedings of the 28th ACM Symposium on Virtual Reality Software and Technology*, pp. 1–11, 2022. 2

[34] R. Jain, J. Shi, R. Duan, Z. Zhu, X. Qian, and K. Ramani. Ubi-touch: Ubiquitous tangible object utilization through consistent hand-object interaction in augmented reality. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology*, pp. 1–18, 2023. 3

[35] S. Jang, W. Stuerzlinger, S. Ambike, and K. Ramani. Modeling cumulative arm fatigue in mid-air interaction based on perceived exertion and kinetics of arm motion. In *Proceedings of the 2017 CHI conference on human factors in computing systems*, pp. 3328–3339, 2017. 1

[36] Y. Jang, I. Jeon, T.-K. Kim, and W. Woo. Metaphoric hand gestures for orientation-aware vr object manipulation with an egocentric viewpoint. *IEEE Transactions on Human-Machine Systems*, 47(1):113–127, 2016. 2

[37] K. Kin, C. Wan, K. Koh, A. Marin, N. C. Camgöz, Y. Zhang, Y. Cai, F. Kovalev, M. Ben-Zacharia, S. Hoople, et al. Stmg: A machine learning microgesture recognition system for supporting thumb-based vr/ar input. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*, pp. 1–15, 2024. 2

[38] D. P. Kingma and M. Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013. 2

[39] C.-J. Lee, R. Zhang, D. Agarwal, T. C. Yu, V. Gunda, O. Lopez, J. Kim, S. Yin, B. Dong, K. Li, et al. Echowrist: Continuous hand pose tracking and hand-object interaction recognition using low-power active acoustic sensing on a wristband. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*, pp. 1–21, 2024. 3

[40] V. Lepetit. Recent advances in 3d object and hand pose estimation. *arXiv preprint arXiv:2006.05927*, 2020. 1

[41] G. M. Lim, P. Jatesiktat, and W. T. Ang. Mobilehand: Real-time 3d hand shape and pose estimation from color image. In *Neural Information Processing: 27th International Conference, ICONIP 2020, Bangkok, Thailand, November 18–22, 2020, Proceedings, Part IV*, pp. 450–459. Springer, 2020. 2

[42] K. Lin, L. Wang, and Z. Liu. End-to-end human pose and mesh reconstruction with transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 1954–1963, 2021. 2

[43] J. Liu, G. Wang, L.-Y. Duan, K. Abdiyeva, and A. C. Kot. Skeleton-based human action recognition with global context-aware attention lstm networks. *IEEE Transactions on Image Processing*, 27(4):1586–1599, 2017. 9

[44] M. Maghoumi and J. J. LaViola Jr. Deepgru: Deep gesture recognition utility. In *International symposium on visual computing*, pp. 16–31. Springer, 2019. 4, 8, 9

[45] A. Masurovsky, P. Chojecki, D. Runde, M. Lafci, D. Przewozny, and M. Gaebler. Controller-free hand tracking for grab-and-place tasks in immersive virtual reality: Design elements and their empirical study. *Multimodal Technologies and Interaction*, 4(4):91, 2020. 2

[46] R. McGloin, K. Farrar, and M. Krcmar. Video games, immersion, and cognitive aggression: does the controller matter? *Media psychology*, 16(1):65–87, 2013. 2

[47] V. Mollyn and C. Harrison. Egotouch: On-body touch input using ar/vr headset cameras. In *Proceedings of the 37th Annual ACM Symposium on User Interface Software and Technology*, pp. 1–11, 2024. 2, 5

[48] K. Monteiro, R. Vatsal, N. Chulpongsatorn, A. Parnami, and R. Suzuki. Teachable reality: Prototyping tangible augmented reality with everyday objects by leveraging interactive machine teaching. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*, pp. 1–15, 2023. 2, 3, 6

[49] M. Oberweger and V. Lepetit. Deepprior++: Improving fast and accurate 3D hand pose estimation. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pp. 585–594, 2017. 2

[50] M. Oberweger, P. Wohlhart, and V. Lepetit. Hands deep in deep learning for hand pose estimation. *arXiv preprint arXiv:1502.06807*, 2015. 2

[51] S. Pei, A. Chen, J. Lee, and Y. Zhang. Hand interfaces: Using hands to imitate objects in ar/vr for expressive interactions. In *Proceedings of the 2022 CHI conference on human factors in computing systems*, pp. 1–16, 2022. 2

[52] K. Pfeuffer, J. Obernolte, F. Dietz, V. Mäkelä, L. Sidenmark, P. Manakhov, M. Pakanen, and F. Alt. Palmgazer: Unimanual eye-hand menus in augmented reality. In *Proceedings of the 2023 ACM Symposium on Spatial User Interaction*, pp. 1–12, 2023. 2

[53] R. A. Potamias, J. Zhang, J. Deng, and S. Zafeiriou. Wilor: End-to-end 3d hand localization and reconstruction in-the-wild. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 12242–12254, 2025. 2, 3, 6, 7

[54] J. Redmon. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016. 4

[55] J. Romero, D. Tzionas, and M. J. Black. Embodied hands: Modeling and capturing hands and bodies together. *ACM Transactions on Graphics, (Proc. SIGGRAPH Asia)*, Nov. 2017. 2

[56] Z. P. Sin, Y. Jia, R. C. Li, H. V. Leong, Q. Li, and P. H. Ng. Illumotion: an optical-illusion-based vr locomotion technique for long-distance 3d movement. In *2024 IEEE Conference Virtual Reality and 3D User Interfaces (VR)*, pp. 924–934. IEEE, 2024. 6

[57] Z. Song, J. J. Dudley, and P. O. Kristensson. Efficient special character entry on a virtual keyboard by hand gesture-based mode switching. In *2022 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pp. 864–871. IEEE, 2022. 2

[58] Z. Song, J. J. Dudley, and P. O. Kristensson. Hotgestures: Complementing command selection and use with delimiter-free gesture-based shortcuts in virtual reality. *IEEE Transactions on Visualization and Computer Graphics*, 2023. 2, 6

[59] Y. Tas and P. Koniusz. Cnn-based action recognition and supervised domain adaptation on 3d body skeletons via kernel feature maps. *arXiv preprint arXiv:1806.09078*, 2018. 9

[60] T. H. E. Tse, K. I. Kim, A. Leonardis, and H. J. Chang. Collaborative learning for hand and object reconstruction with attention-guided graph convolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1664–1674, 2022. 2

[61] U. Tsimbalistaia, C. Berger, H. Gellersen, and P. Manakhov. On-body icons: Designing a 3d interface for launching apps in augmented reality. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*, pp. 1–15, 2025. 5

[62] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 2

[63] X. Wang, B. Lafreniere, and J. Zhao. Exploring visualizations for precisely guiding bare hand gestures in virtual reality. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*, pp. 1–19, 2024. 2

[64] R. Xiao, J. Schwarz, N. Throm, A. D. Wilson, and H. Benko. Mrtouch: Adding touch input to head-mounted mixed reality. *IEEE transactions on visualization and computer graphics*, 24(4):1653–1660, 2018. 3, 5

[65] X. Xu, J. Gong, C. Brum, L. Liang, B. Suh, S. K. Gupta, Y. Agarwal, L. Lindsey, R. Kang, B. Shahsavari, et al. Enabling hand gesture customization on wrist-worn devices. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*, pp. 1–19, 2022. 2

[66] L. Yang, S. Li, D. Lee, and A. Yao. Aligning latent spaces for 3d hand pose estimation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 2335–2343, 2019. 2

[67] K. Yun, J. Honorio, D. Chattopadhyay, T. L. Berg, and D. Samaras. Two-person interaction detection using body-pose features and multiple instance learning. In *2012 IEEE computer society conference on computer vision and pattern recognition workshops*, pp. 28–35. IEEE, 2012. 6, 8, 9

[68] X. Zabulis, H. Baltzakis, and A. A. Argyros. Vision-based hand gesture recognition for human-computer interaction. *The universal access handbook*, 34:30, 2009. 5

[69] P. Zhang, C. Lan, J. Xing, W. Zeng, J. Xue, and N. Zheng. View adaptive recurrent neural networks for high performance human action recognition from skeleton data. In *Proceedings of the IEEE international conference on computer vision*, pp. 2117–2126, 2017. 9

[70] X. Zhang, H. Huang, J. Tan, H. Xu, C. Yang, G. Peng, L. Wang, and J. Liu. Hand image understanding via deep multi-task learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 11281–11292, 2021. 2

[71] X. Zhang, Q. Li, H. Mo, W. Zhang, and W. Zheng. End-to-end hand mesh recovery from a monocular rgb image. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 2354–2364, 2019. 2

[72] X. Zheng, P. Ren, H. Sun, J. Wang, Q. Qi, and J. Liao. Sar: Spatial-aware regression for 3d hand pose and mesh reconstruction from a monocular rgb image. In *2021 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pp. 99–108. IEEE, 2021. 2

[73] Y. Zhou, M. Habermann, W. Xu, I. Habibie, C. Theobalt, and F. Xu. Monocular real-time hand shape and motion capture using multi-modal data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5346–5355, 2020. 2

[74] F. Zhu, L. Sidenmark, M. Sousa, and T. Grossman. Pinchlens: Applying spatial magnification and adaptive control-display gain for precise selection in virtual reality. In *2023 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pp. 1221–1230. IEEE, 2023. 2

[75] C. Zimmermann and T. Brox. Learning to estimate 3D hand pose from single RGB images. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 4903–4911, 2017. 2