Adaptive Interaction Paradigm for Mixed Reality Games

LLOGARI CASAS, 3FINERY LTD, Biggar, United Kingdom of Great Britain and Northern Ireland

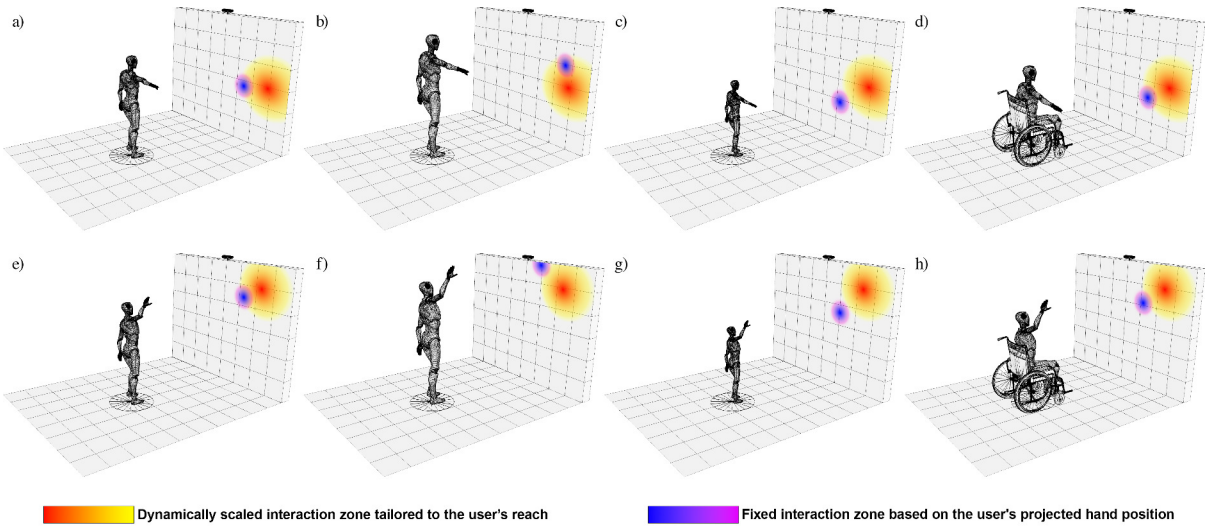

Fig. 1. Illustration of the adaptive interaction paradigm in a MR environment demonstrating interaction zone adjustments based on user-specific characteristics. In sub-figures *(a)-(h)* yellow-red gradients indicate dynamically scaled interaction zones tailored to the user's reach and position and blue-violet gradients represent fixed interaction zones based on the user's projected body position. This visualization highlights the capability to personalize interactions based on individual user profiles, enhancing accessibility, inclusivity, and engagement across diverse physical conditions.

This paper introduces an adaptive interaction paradigm for Mixed Reality (MR) games, designed to enhance accessibility, scalability, and responsiveness in large-scale MR environments. By leveraging depth-sensing technology and real-time 3D skeletal tracking, the paradigm enables virtual elements to dynamically adjust to user movements, creating personalized and inclusive interactions. Unlike traditional fixed interaction models, this approach tailors interaction zones and gesture thresholds to individual user metrics, addressing limitations in current MR designs that fail to accommodate diverse physical abilities. The proposed method employs an egocentric rule-based framework, ensuring low-latency, real-time performance while maintaining transparency and adaptability. Privacy-by-design principles are integral to this approach, with local computation and data anonymization preserving user confidentiality. The effectiveness of this adaptive paradigm is demonstrated through a large-scale MR gameplay use case, with insights from over 5,000 gameplay sessions informing the refinement of interaction

Authors' Contact Information: Llogari Casas, 3FINERY LTD, Biggar, Scotland, United Kingdom of Great Britain and Northern Ireland; e-mail: llogari92@gmail.com.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

© 2025 Copyright held by the owner/author(s). Publication rights licensed to ACM.

ACM 2832-5516/2025/9-ART
https://doi.org/10.1145/3768627

models. Beyond gaming, this paradigm establishes a foundation for broader applications in education, rehabilitation, and accessibility technologies, advancing the state of user-centric MR interaction design.

CCS Concepts: • **Human-centered computing** → **Mixed / augmented reality**; *Accessibility systems and tools*; • **Hardware** → *Sensor devices and platforms*; • **Social and professional topics** → *Privacy*.

Additional Key Words and Phrases: Mixed Reality, Depth-Sensing, Gesture Recognition, Real-Time Interaction, Accessibility, Sensor Data, Privacy, Immersive Gameplay, User-Centric Design, Adaptive Systems

1 Introduction

Mixed Reality (MR) environments are transforming interaction paradigms by seamlessly blending physical and digital worlds, creating immersive and engaging experiences across diverse applications. This paper introduces an adaptive interaction paradigm specifically designed for large-scale MR setups, integrating LED displays and RGBD cameras. By leveraging real-time 3D body tracking, the proposed approach enables virtual elements to dynamically respond to user movements, fostering accessibility, inclusivity and highly responsive interactions.

The core contribution of this work lies in the adaptation of MR environments to enhance user interaction, accessibility, and scalability. Traditional MR interaction paradigms often fail to dynamically adjust to individual user characteristics, resulting in inconsistencies in engagement and usability. To address this, we present an adaptive interaction model that personalizes interaction zones and thresholds based on real-time skeletal tracking, overcoming the limitations of fixed, one-size-fits-all approaches. This rule-based approach ensures real-time, low-latency performance for immersive MR experiences while remaining transparent and adaptable. Interaction parameters such as thresholds and zones dynamically adjust based on user-specific metrics, enabling personalized interactions that accommodate diverse body dimensions and movement styles. Furthermore, the paradigm incorporates stringent privacy safeguards, with data anonymized at the point of capture, computations performed locally and only non-sensitive interaction data retained, adhering to privacy-by-design principles. The effectiveness of this approach is demonstrated through a large-scale MR gameplay use case, with insights from over 5,000 sessions informing the refinement of interaction models and performance metrics. The key contributions of this work include:

- A scalable and adaptive interaction paradigm optimized for large-scale Mixed Reality environments.
- A rule-based method for user-centric interaction modelling, featuring real-time full-body gesture recognition and dynamically adjustable interaction zones.
- Privacy-centric design principles that incorporate ethical strategies for secure and responsible deployment.
- Insights from extensive gameplay data to refine user-centric design and interaction models.

2 Related Work

Mixed Reality environments and depth-sensing technologies have been extensively explored in both academic and industrial research contexts. This section surveys the foundational contributions and recent advancements that underpin the adaptive interaction paradigm presented in this paper.

2.1 Real-Time Depth Data Acquisition

Depth-sensing methods, such as structured light, Time-of-Flight (ToF), and LiDAR, have enabled accurate 3D spatial tracking in MR environments. Kinect's introduction by Microsoft revolutionized gesture-based interactions [Zhang 2012], offering a low-cost solution for capturing full-body movements. More recently, advancements in ToF cameras [Yu et al. 2020] have improved depth accuracy and reduced latency, enabling real-time interaction in dynamic environments. These technologies have been widely adopted in applications ranging from games [Casas et al. 2018] to healthcare [Hargaš and Koniar 2022] and education [Park et al. 2021]. Building on these innovations and following a similar approach to *DanceGraph* [Sinclair et al. 2023], our method utilizes the ZED 2

Stereo camera [Stereolabs 2025], which combines stereo vision with depth sensing to support large-scale MR setups.

2.2 Gesture Recognition and Interaction

Gesture recognition has been a fundamental area of MR research, enabling intuitive interactions between users and virtual environments. Early work by [Wachs et al. 2011] provided a comprehensive survey of vision-based gesture interfaces, emphasizing the importance of skeletal tracking for user-centric design. More recent studies have explored machine learning techniques for gesture classification [Nogales and Benalcazar 2020], enhancing accuracy and robustness in complex environments. For instance, *Mediapipe* [Lugaresi et al. 2019] demonstrated a framework for real-time multi-modal gesture tracking, which inspired our approach to interaction modelling.

While ML-based gesture recognition approaches have demonstrated success in diverse applications [Wu 2024], this paradigm adopts a rule-based method leveraging skeletal key-points. The decision to employ this approach is driven by key considerations of latency, transparency and adaptability, making it particularly well-suited for real-time MR experiences on large-scale displays. Trade-off between both approaches have been extensively evaluated in the literature [Uzuner et al. 2009; van Ginneken 2017]. ML-based models, especially those utilizing deep learning, often require significant computational resources for inference. This can introduce latency that disrupts the immersive nature of MR interaction. In contrast, rule-based approaches operate deterministically and avoid the overhead of neural network computations, achieving real-time performance with average gesture recognition latency below 10 ms on standard hardware. This ensures seamless interaction between the user and virtual elements, critical for maintaining immersion. Rule-based methods offer inherent transparency, as the decision-making process is explicitly defined through mathematical rules [Cippitelli et al. 2016]. This deterministic nature facilitates debugging and optimization, allowing real-time adjustment of thresholds with immediate feedback. Conversely, ML-based approaches often act as a *"black box"*, where gesture classifications are difficult to fine-tune, complicating real-time adaptation processes. Further, a rule-based approach can be capable of dynamically adjusting thresholds and interaction zones based on user-specific metrics. In that context, ML models would require retraining and extensive datasets to generalize across diverse users. For these reasons, our work is built using a rule-based approach that achieves adaptability through simple parameter scaling, eliminating the need for additional data collection.

2.3 Mixed Reality Interfaces

Mixed Reality environmental interaction extends gesture recognition by incorporating spatial relationships between users and virtual objects. Collision detection and bounding box methods have been commonly employed since early days of interactive graphics [Chung and Wang 1996], enabling object manipulation and proximity-based triggers. Recent advancements include incorporating ergonomics into Virtual Reality experiences by adjusting environments to promote inclusivity and enable dynamic personalization [Cabrera-Araya and Rojas-Munoz 2024], aligning closely with our emphasis on adaptive and customized interaction. Additionally, in this space, [Evangelista Belo et al. 2021] introduced *XRgonomics*, a study focused on ergonomic MR design principles to improve user comfort and interaction efficiency. Further to this, recent research on the outstanding grand challenges for Mixed Reality setups [Billinghurst 2021] emphasizes the unique complexities of MR interaction and the potential of full-body input to enable more intuitive, multimodal experiences. Our paradigm addresses these challenges by integrating adaptive, user-centric modelling to enhance interaction and inclusivity.

User studies have also been key in MR environmental interaction by offering insights into user behaviour. Research by [Derby and Chaparro 2022] demonstrated the importance of usability testing for iterative MR development, highlighting the need for frameworks that can adapt to diverse user needs and provide intuitive, accessible experiences. Building on these findings, our approach leverages real-time user feedback and analytics

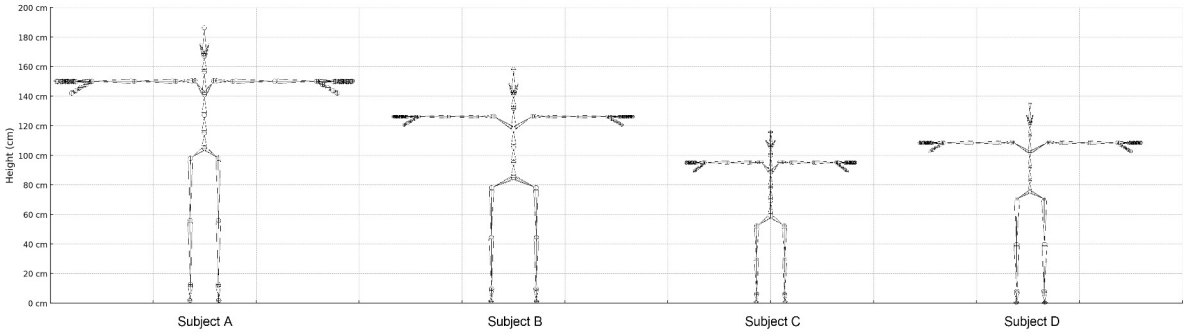

Fig. 2. Skeletal representations of four subjects with varying body dimensions illustrating the method's ability to normalize and adapt interaction zones based on individual height and arm span. This ensures inclusivity and consistency in gesture recognition and interaction across diverse user profiles.

to refine interaction zones and gesture detection thresholds. This ensures seamless and adaptive interactions by accommodating variations in body dimensions and movement mechanics.

3 User-Centric Modelling in MR Interactions

The term User-Centric Modelling refers to the process of designing interactions that adapt to the unique characteristics, movements and abilities of each individual user. In the context of Mixed Reality experiences, this involves tailoring interactions to the user's physical dimensions and gestures to ensure an intuitive and inclusive experience. We employ an egocentric approach, which centres interactions around the user's perspective, where the design and experience are adapted to their specific viewpoint. As depicted in figure 3, our paradigm achieves this by utilizing real-time body tracking and gesture recognition, normalizing skeletal data relative to the user's body size and dynamically adjusting interaction zones and thresholds accordingly.

3.1 MR Environment and Hardware Setup

The MR setup consists of a large-scale interactive environment that integrates a single RGBD camera and an LED display to enable real-time adaptive interactions. The system tracks users' full-body movements using 3D skeletal tracking, allowing virtual elements to dynamically adjust based on individual motion and positioning. The RGBD camera captures depth data and skeletal key points, facilitating interaction mapping without requiring multiple sensor inputs. The LED display serves as the primary visualization interface, rendering interactive content that responds to user movements. The spatial configuration ensures that users can engage naturally with the MR environment, with adaptive interaction zones calibrated based on each individual's reach and mobility.

3.2 Consistent Multi-User Tracking

At the start of the MR experience, the approach identifies the primary user as the individual closest to the center of the depth sensor's field of view. Let $\mathbf{p}_i = (x_i, y_i, z_i)$ represent the position of user i in the camera's coordinate space, and $\mathbf{c} = (x_c, y_c, z_c)$ be the frame's centroid. The distance of user i from the centroid is computed as:

$$d_i = \|\mathbf{p}_i - \mathbf{c}\|_2$$

where $\|\cdot\|_2$ denotes the Euclidean norm. The primary user u_{primary} is selected by minimizing the distance:

$$u_{\text{primary}} = \arg\min_i d_i$$

The method assigns a unique identifier id_i to each tracked user. These identifiers are stored in a priority queue Q:

$$Q = [\mathrm{id}_1, \mathrm{id}_2, \ldots, \mathrm{id}_n]$$

where id_1 has the highest priority. This ensures that the original user retains control, even if other individuals enter the scene. If tracking for the primary user is lost, due to occlusion or temporary exit, the method continuously monitors for the reappearance of their identifier. Re-establishment occurs if:

$$\mathrm{id}_{\mathrm{primary}} \in \mathrm{current\_visible\_ids}$$

where current_visible_ids is the set of identifiers currently detected by the depth camera. Once rediscovered, the user regains their role as the primary user, ensuring persistent interaction.

3.3 User Coordinate Space

To ensure consistency across users with varying body sizes and positions (see figure 2), we define the ***User Coordinate Space*** as a local 3D coordinate space centered on a fixed reference point on the user's body, typically the center of the torso. This coordinate space allows for normalization of skeletal key-points, ensuring that all movements and interactions are relative to the user's body dimensions. By grounding computations in the user coordinate space, interactions become personalized and scalable to different users.

Our implementation utilizes the ZED 2 AI Stereo Camera [Stereolabs 2025], which tracks the user's body using a 38-keypoint format. The camera provides 3D coordinates for key-points representing various body parts. In this context, let $\mathbf{p}_i = (x_i, y_i, z_i)$ denote the 3D position of a key-point i. To normalize these coordinates, they are shifted relative to a reference key-point. Here, $\mathbf{p}_{\mathrm{torso}}$ acts as a fixed origin in the user's local coordinate space ensuring that interactions are proportional to the user's body dimensions.

$$\mathbf{p}_{i,\mathrm{normalized}} = \mathbf{p}_i - \mathbf{p}_{\mathrm{torso}}$$

To account for variability in user size, proportions, and movement capabilities, the method normalizes skeletal data relative to individual body dimensions. A scaling factor s_{scale} is computed as:

$$s_{\mathrm{scale}} = \frac{\mathrm{Arm\ Span}_{\mathrm{user}}}{\mathrm{Arm\ Span}_{\mathrm{default}}},$$

where $\mathrm{Arm\ Span}_{\mathrm{user}}$ is the measured distance between the user's outstretched hands, and $\mathrm{Arm\ Span}_{\mathrm{default}}$ represents a reference value. This factor is applied to normalize interaction thresholds, ensuring consistent gesture recognition across users:

$$\mathbf{p}_i^{\mathrm{normalized}} = s_{\mathrm{scale}} \cdot \mathbf{p}_i.$$

3.4 Interaction Zones

Interaction zones define spatial regions where specific gestures and actions trigger events, acting as the bridge between the user's physical movements and corresponding virtual responses. These zones are dynamically computed in the user coordinate space and are associated with one or more normalized key-points. For instance, a *"wave"* gesture can be detected when the hand key-point $\mathbf{p}_{\mathrm{hand}}$ enters a predefined region:

$$\text{Wave Zone: } x_{\min} < x_{\mathrm{hand}} < x_{\max}, \quad y_{\min} < y_{\mathrm{hand}} < y_{\max}$$

In this case, the boundaries of these zones are not static and are recalculated in real-time based on the user's skeletal dimensions and movement range. This ensures that the zones remain functional and intuitive across varying user positions relative to the depth camera. To this extend, interaction zones can be categorized into static, dynamic, and adaptive types, depending on their purpose and behaviour:

- **Static Zones**. These zones remain fixed relative to the user's normalized body space. For example, a zone directly above the user's head might trigger a "jump" action when the head key-point $\mathbf{p}_{\text{head}}$ enters it.
- **Dynamic Zones**. These zones move in response to the user's actions. For instance, in a catching task, a zone corresponding to an on-screen falling object might move dynamically, requiring the user's hand to align with it.
- **Adaptive Zones**. These zones adjust their size and sensitivity based on the user's movement patterns. For users with limited mobility, larger zones can ensure that smaller gestures still trigger the desired interactions.

Complex interactions often require simultaneous engagement with multiple zones. For example, a *"clap"* gesture can be detected when the left-hand key-point $\mathbf{p}_{\text{left\_hand}}$ and the right-hand key-point $\mathbf{p}_{\text{right\_hand}}$ enter overlapping interaction zones:

$$|x_{\text{left\_hand}} - x_{\text{right\_hand}}| < \epsilon, \quad |y_{\text{left\_hand}} - y_{\text{right\_hand}}| < \epsilon$$

where ϵ is a small threshold value. As users move closer to or farther from the camera, interaction zones are recalibrated to maintain consistent behaviour. For instance, if a user steps back, the interaction zones expand proportionally to compensate for the reduced precision in tracking at greater distances.

3.5 Real-Time Full-Body Gesture Detection

In order for our approach to achieve gesture detection in an immersive real-time MR environment, we leverage 3D skeletal key-points to interpret user movements dynamically. The foundation of this detection lies in the accurate tracking of 3D skeletal key-points. Let $\mathbf{p}_i(t) = (x_i(t), y_i(t), z_i(t))$ represent the position of key-point i at time t. The relative movement of these key-points over time form the basis of a gesture detection. For instance, the relative position vector between two key-points, such as the hands, is given by:

$$\mathbf{v}_{\text{relative}} = \mathbf{p}_{\text{hand1}}(t) - \mathbf{p}_{\text{hand2}}(t)$$

The velocity of a specific key-point, used to identify fast movements like swipes, is calculated as:

$$\mathbf{v}_{\text{velocity}} = \frac{\mathbf{p}_i(t_2) - \mathbf{p}_i(t_1)}{\Delta t}$$

where $\Delta t = t_2 - t_1$ is the time interval between two frames.

Gestures are classified based on predefined movement patterns and velocity thresholds. Our approach supports a variety of gestures commonly used in MR environments, including:

- **Swipe:** This gesture is identified when the hand velocity in the horizontal direction exceeds a threshold:

$$|\mathbf{v}_{\text{hand},x}| > \text{threshold}$$

- **Raise Hand:** This gesture is detected when the vertical position of the hand exceeds the height of the head:

$$y_{\text{hand}} > y_{\text{head}}$$

- **Wave:** A waving motion is identified by detecting periodic lateral hand movements within a specific range:

$$x_{\text{hand}}(t) \text{ oscillates with period } T \text{ and amplitude } A$$

- **Pointing:** This gesture is recognized by analysing the alignment of hand and arm key-points in a straight line.

For further implementation details, see section 3.8 for pseudocode detailing core gesture routines.

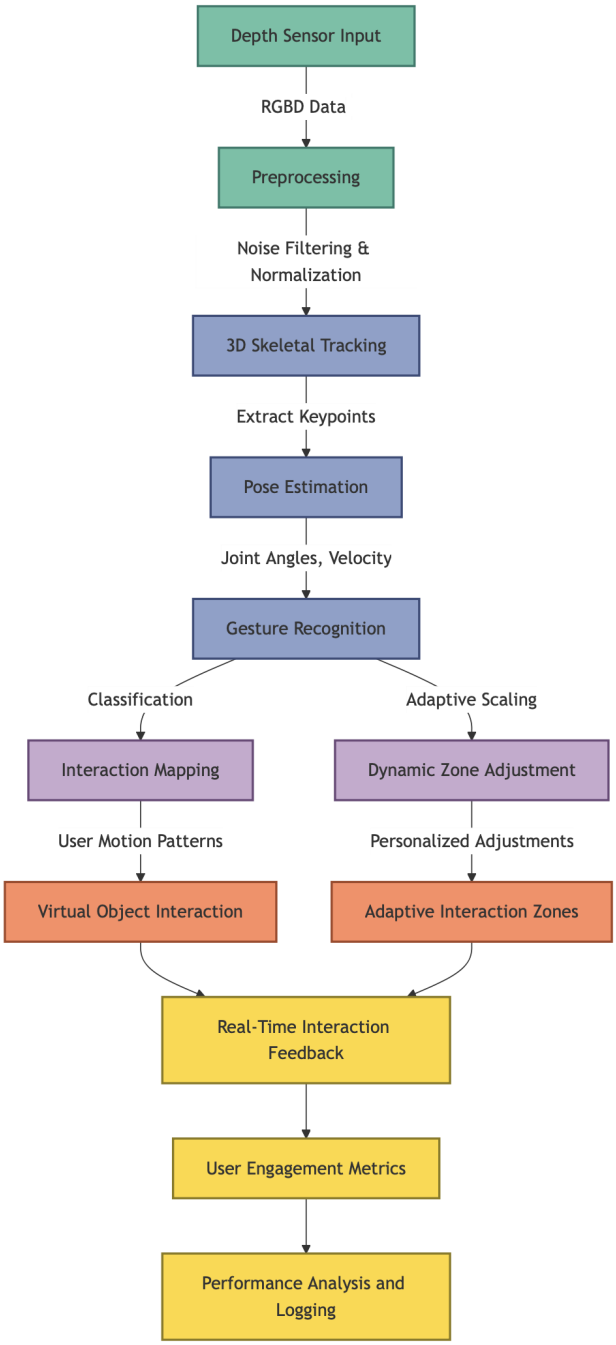

Fig. 3. Color-coded stages of the adaptive interaction pipeline for Mixed Reality environments. Raw depth sensor data is processed to perform skeletal tracking and pose estimation, extracting joint angles and velocities. Gesture recognition informs interaction mapping for motion-based event triggering, while adaptive scaling personalizes interaction zones based on user-specific characteristics. The pipeline provides real-time feedback and logs user engagement metrics for performance evaluation and iterative refinement.

3.6 Environment Interaction Across Realities

Unlike traditional interaction paradigms that rely on input devices such as controllers or keyboards, our method utilizes the user's natural movements to drive the immersive experience. Environmental interaction, in this context, refers to the dynamic and reciprocal engagement between the user and virtual elements within the Mixed Reality environment. Virtual objects displayed on large screens respond dynamically to the user's movements, facilitating intuitive and seamless interactions between both worlds. This approach ensures a consistent and responsive connection between the user and the virtual environment, enhancing both the realism and accessibility of the MR experience. To model these environmental interactions mathematically, we represent the user and virtual objects as bounded regions in a 3D coordinate space. Let the bounding box of the user be defined as:

$$\mathrm{B_u} = \{\mathbf{x} \in \mathbb{R}^3 \mid x_{\min} \le x \le x_{\max},\ y_{\min} \le y \le y_{\max},\ z_{\min} \le z \le z_{\max}\}$$

where the limits are derived dynamically from the user's skeletal key-points. Similarly, virtual objects are defined as:

$$\mathrm{B_o} = \{\mathbf{x} \in \mathbb{R}^3 \mid x'_{\min} \le x \le x'_{\max},\ y'_{\min} \le y \le y'_{\max},\ z'_{\min} \le z \le z'_{\max}\}$$

Interaction events are then triggered when the user's bounding box intersection with a virtual object's bounding box is non-empty:

$$\text{Collision} = (\mathrm{B_u} \cap \mathrm{B_o}) \neq \emptyset$$

Beyond static collision detection, dynamic object interactions are supported through parametrized models. For example, object manipulation such as picking up or throwing relies on continuous tracking of the user's skeletal motion:

- **Picking Up:** A virtual object is "picked up" if the hand key-point enters and remains within the bounding box for a duration Δt_{hold}:
$$\Delta t_{\text{hold}} \ge \tau_{\text{threshold}}$$
- **Throwing:** The velocity vector of a user's hand is computed during the release phase to simulate the object's trajectory:
$$\mathbf{v}_{\text{throw}} = \frac{\mathbf{p}_{\text{hand}}(t_{\text{release}}) - \mathbf{p}_{\text{hand}}(t_{\text{grab}})}{t_{\text{release}} - t_{\text{grab}}}$$
- **Proximity Activation:** Interaction zones are defined as hyper-rectangles in 3D space. Activation occurs when the user's key-points satisfy:
$$\mathbf{p}_{\text{keypoint}} \in \text{BBox}_{\text{zone}}$$

3.7 Adaptive Interaction Dynamics

To enhance accessibility, the paradigm dynamically adjusts sensitivity and thresholds based on user movement patterns. This adaptability is implemented through two mechanisms: an initial calibration phase and real-time adjustments using the last N frames of skeletal data. During the initial calibration phase, users are prompted to perform a small set of predefined gestures (e.g., swipe, raise hand) which the system uses to compute initial thresholds for velocity and range of motion. These parameters initialize the interaction zones and sensitivity levels. Let $\mathbf{p}_i(t) = (x_i, y_i, z_i)$ represent the position of key-point i at time t. The user's range of motion for a given key-point i is computed over a calibration period T_{cal} as:

$$R_i = \max_{t \in [0, T_{\text{cal}}]} \mathbf{p}_i(t) - \min_{t \in [0, T_{\text{cal}}]} \mathbf{p}_i(t)$$

During active use, the system monitors the user's skeletal keypoints across the last $N = 15$ frames and applies low-pass smoothing to adjust zone boundaries in real time. Interaction zones are then scaled proportionally to

R_i, ensuring they match the user's natural motion. The average velocity $\bar{v}_i$ of a key-point is calculated during calibration as:

$$\bar{v}_i = \frac{1}{T_{\text{cal}}} \int_0^{T_{\text{cal}}} \left\| \frac{d\mathbf{p}_i(t)}{dt} \right\| dt$$

A gesture detection threshold $v_{\text{threshold}}$ is then set as:

$$v_{\text{threshold}} = k \cdot \bar{v}_i$$

where $k > 1$ is a proportionality constant to account for natural variability in movement. The user's maximum reach $\mathbf{p}_{\text{max}}$ and minimum reach $\mathbf{p}_{\text{min}}$ are recorded as:

$$\mathbf{p}_{\text{max}} = \max_i \mathbf{p}_i(t), \quad \mathbf{p}_{\text{min}} = \min_i \mathbf{p}_i(t)$$

These values define boundaries for adaptive interaction zones. Once calibration is completed, its parameters are continuously adapted in real-time by analysing the user's recent movement data over the last N frames. Let $\mathbf{p}_i(t_k)$ represent the position of key-point i at frame k. To perform an adjustment to the velocity threshold, the instantaneous velocity of a key-point i at frame k is approximated as:

$$v_i(t_k) = \frac{\|\mathbf{p}_i(t_k) - \mathbf{p}_i(t_{k-1})\|}{\Delta t}$$

where Δt is the time interval between frames. The average velocity over N frames is:

$$\bar{v}_i = \frac{1}{N} \sum_{k=k-N+1}^{k} v_i(t_k)$$

If $\bar{v}_i$ decreases significantly (e.g., due to fatigue), the velocity threshold $v_{\text{threshold}}$ is adjusted:

$$v_{\text{threshold,new}} = \alpha \cdot \bar{v}_i$$

where α is a sensitivity scaling factor. To handle interaction zones expanding or contracting, their boundaries are recalculated based on the user's recent range of motion:

$$R_i^{\text{recent}} = \max_{k=k-N+1}^{k} \mathbf{p}_i(t_k) - \min_{k=k-N+1}^{k} \mathbf{p}_i(t_k)$$

The zone size Z_i is updated as:

$$Z_i = \beta \cdot R_i^{\text{recent}}$$

where β is a proportionality constant to balance sensitivity and usability. Sudden noise and anomalies are identified by analysing deviations from the mean trajectory:

$$\Delta\mathbf{p}_i(t_k) = \mathbf{p}_i(t_k) - \frac{1}{N} \sum_{k=k-N+1}^{k} \mathbf{p}_i(t_k)$$

Significant anomalies ($\|\Delta\mathbf{p}_i(t_k)\| > \epsilon$) are ignored to ensure stability in interactions. This combination of initial calibration and real-time adjustments ensures that a wide range of user abilities are accommodated. Users with limited mobility benefit from expanded interaction zones and reduced velocity thresholds, while more agile users experience interactions that scale dynamically to their capabilities. The reliance on recent frame data allows the method to provide a personalized and responsive experience for all users. The following constant values were used across all tests:

- Gesture velocity threshold scaling constant: $k = 1.25$
- Adaptive sensitivity factor: $\alpha = 0.85$
- Interaction zone scaling constant: $\beta = 1.15$

- Proximity threshold for gestures involving multiple key-points (e.g., clapping): $\epsilon = 0.12\,\text{m}$

3.8 Pseudocode Examples for Rule-Based Gesture Recognition

To support reproducibility and broader adoption of the proposed adaptive interaction paradigm, this section outlines pseudocode examples of the core gesture recognition routines. These rules are implemented on top of a real-time skeletal tracking pipeline using egocentric coordinates (see section 3.3) and all parameters can be adjusted at runtime to accommodate user variation. While the examples below reference the user's right side (e.g., right hand, right elbow), the detection logic is symmetrical and can be equally applied to the left side depending on context or user preference.

3.8.1 Swipe Gesture Detection. The swipe gesture serves as a baseline for interaction due to its simplicity and high recognition rate across user groups. Detection relies on horizontal hand velocity exceeding a threshold, provided the hand is raised above the torso to avoid unintentional triggers. This gesture proved robust in both seated and standing postures, with minimal false positives under natural movement.

ALGORITHM 1: Swipe Gesture Detection

Input: 3D skeletal keypoints at current frame K_t, 3D skeletal keypoints at previous frame K_{t-1}, time delta Δt, threshold $v_{threshold}$
Output: Gesture status (Swipe or None)

$v_x \leftarrow \frac{K_t[\text{right\_hand}].x - K_{t-1}[\text{right\_hand}].x}{\Delta t}$;
$y_{hand} \leftarrow K_t[\text{right\_hand}].y$;
$y_{torso} \leftarrow K_t[\text{torso}].y$;
if $y_{hand} > y_{torso}$ **then**
    if $|v_x| > v_{threshold}$ **then**
        return "Swipe Detected";
return "No Gesture";

3.8.2 Wave Gesture Detection. To detect repetitive waving, the system analyzes horizontal hand oscillations within a short time window. This approach captures gesture periodicity and amplitude, allowing the system to distinguish intentional waves from noise. The wave gesture is well-suited for public displays and ambient interactions, particularly when users are at a distance from the sensing hardware.

ALGORITHM 2: Wave Gesture Detection

Input: Sequence of N horizontal hand positions $\{x_1, x_2, ..., x_N\}$, amplitude threshold A_{min}, period window T
Output: Gesture status (Wave or None)

$oscillations \leftarrow 0$;
for $i \leftarrow 2$ **to** $N - 1$ **do**
    if $(x_{i-1} < x_i)$ ***and*** $(x_i > x_{i+1})$ ***or*** $(x_{i-1} > x_i)$ ***and*** $(x_i < x_{i+1})$ **then**
        $oscillations \leftarrow oscillations + 1$;
$amplitude \leftarrow \max(x_1, ..., x_N) - \min(x_1, ..., x_N)$;
if $oscillations \geq 2$ ***and*** $amplitude \geq A_{min}$ **then**
    return "Wave Detected";
return "No Gesture";

3.8.3 Raise Hand Gesture Detection. A simple yet effective gesture, "raise hand" is detected when the user's hand exceeds the vertical position of the head. This gesture is especially useful in accessibility scenarios, where

limited lateral motion might hinder more complex interactions. Its binary nature makes it a reliable trigger in onboarding or calibration stages.

ALGORITHM 3: Raise Hand Gesture Detection

Input: 3D skeletal keypoints K_t at current frame
Output: Gesture status (Raised or None)

$y_{hand} \leftarrow K_t[\text{right\_hand}].y$;
$y_{head} \leftarrow K_t[\text{head}].y$;
if $y_{hand} > y_{head}$ **then**
    return "Raise Hand Detected";
return "No Gesture";

3.8.4 Pointing Gesture Detection. This gesture identifies when the arm forms a straight line from shoulder to hand, interpreted as a pointing action. It uses the angle between elbow and hand vectors, and is especially relevant for MR applications where users indicate targets, directions or menu selections. Proper detection depends on reliable skeletal joint tracking.

ALGORITHM 4: Pointing Gesture Detection

Input: 3D skeletal keypoints K_t (right_hand, right_elbow, right_shoulder), angle threshold θ_{max}
Output: Gesture status (Pointing or None)

$v_1 \leftarrow K_t[\text{right\_elbow}] - K_t[\text{right\_shoulder}]$;
$v_2 \leftarrow K_t[\text{right\_hand}] - K_t[\text{right\_elbow}]$;
$\theta \leftarrow \arccos\left(\frac{v_1 \cdot v_2}{\|v_1\|\|v_2\|}\right)$;
if $\theta < \theta_{max}$ **then**
    return "Pointing Detected";
return "No Gesture";

This procedure is executed per frame and can be extended to include debounce logic or oscillation detection for gestures like waving. All spatial calculations are performed in the user-centered coordinate space ensuring consistency across different user body types.

4 Real-Time Analytical Driven Interaction

By quantifying user interactions in real time, the analytics component serves as a foundation for refining mechanics, optimizing interaction thresholds and tailoring experiences to the specific needs and abilities of individual users. The module continuously records data from the depth-enabled camera and interaction events, capturing a comprehensive array of metrics to evaluate user behaviour and its performance. Key data points include the 3D coordinates of skeletal key-points, such as the hands, head and torso, tracked over time to monitor and analyse user movements with precision. Additionally, interaction metrics such as session duration, object manipulations, and the frequency of specific actions are tracked. All data is timestamped and stored in a structured format, facilitating both real-time analysis and post-session evaluations. Interaction events are modelled as functions of depth, time, and spatial relationships where $\mathbf{p}_{\text{user}}$ and $\mathbf{p}_{\text{object}}$ are the user and object positions, respectively, and t is the timestamp.

$$e = f(\mathbf{p}_{\text{user}}, \mathbf{p}_{\text{object}}, t),$$

Beyond real-time adjustments, the method also stores data for post-session analysis, enabling the identification of trends in user behaviour. This, for instance, could be preferred gestures, frequently interacted objects and

areas of friction in interaction mechanics, which assist on evaluating the effectiveness of interactive zones and gesture recognition models.

5 Fixed and Adaptive Interaction Comparison

In MR systems, interaction zones are often defined based on fixed screen-space projections of the user's skeletal position. This fixed interaction paradigm, while simple to implement, lacks adaptability and fails to account for variations in user-specific factors such as height, reach and mobility constraints. Figure 1 illustrates a comparative analysis between the conventional fixed interaction approach (blue-violet heat-map) and the proposed adaptive paradigm (yellow-red heat-map). As seen in subfigures *(a)-(h)*, the fixed interaction zone remains constant regardless of the user's pose or physical capabilities. This results in potential accessibility issues, particularly for users with limited mobility, as seen in sub-figure *(d)*, where a seated user is unable to effectively reach the designated target zone. Conversely, when using the adaptive paradigm, interaction zones are dynamically scaled to match the user's reach and pose in real time. The interaction area is adjusted based on key skeletal metrics, including arm span and joint flexibility, to ensure that interaction elements remain within an accessible range for each user. This dynamic approach enables a more intuitive and equitable experience across a diverse range of users, including those with varying physical abilities.

6 Secure and Ethical MR Interaction

To ensure a robust ethical and privacy-focused design for protecting user data while enabling secure and responsible deployment, all collected data is anonymized at the point of capture. Unique identifiers replace personal information, ensuring that no data can be traced back to individual users. Skeletal key-points and interaction events are logged without any association to identifying characteristics, allowing the method to concentrate exclusively on interaction analytics and user modelling. All computations related to depth-sensing data are performed locally on the local unit, eliminating the need to transmit raw unprocessed frames. This minimizes the risk of data breaches and ensures that sensitive visual information remains unexposed. Once processed, only interaction outputs, such as gestures and collisions, are logged, removing potentially sensitive data. The output of computations is securely transmitted to an SQL database, which exclusively stores interaction data and session metrics. The database is designed to avoid retaining raw visual information, adhering to privacy best practices while maintaining the integrity of session analytics. The paradigm explicitly avoids user profiling, ensuring that no behavioural profiles are created based on user interactions. This guarantees that the approach operates transparently, focusing solely on immediate interaction feedback without building persistent user models.

7 Evaluating User Behaviour in MR Gameplay

To assess the effectiveness of our Mixed Reality paradigm, we implemented it as a gaming experience with varying difficulty levels (see figure 4). Over 5,000 gameplay sessions have been analysed. Gameplay durations exhibit significant variability, ranging from as short as 7 seconds to over 110 seconds on average (see figure 5*a*), reflecting the diverse skill levels and engagement patterns of players. Higher difficulty games often result in shorter sessions and lower scores, highlighting the need for carefully balancing mechanics to sustain player interest and ensure accessibility. As depicted in figure 5*b*, easier and medium difficulty games demonstrate a clear preference among players for more approachable challenges, offering a balance between difficulty and sustained engagement.

For predefined and controlled gesture sets, rule-based approaches provide comparable or superior accuracy to ML-based methods. For example, in a controlled test of 1,000 gestures across five users, the rule-based method achieved a gesture recognition accuracy of 97.5%, comparable to ML-based methods trained on similar

Fig. 4. Illustration of the proposed adaptive MR interaction paradigm in action. The images depict users engaging with a large-scale LED display powered by a depth-sensing camera, enabling real-time body tracking and gesture-based interactions with varying interaction and difficulty levels.

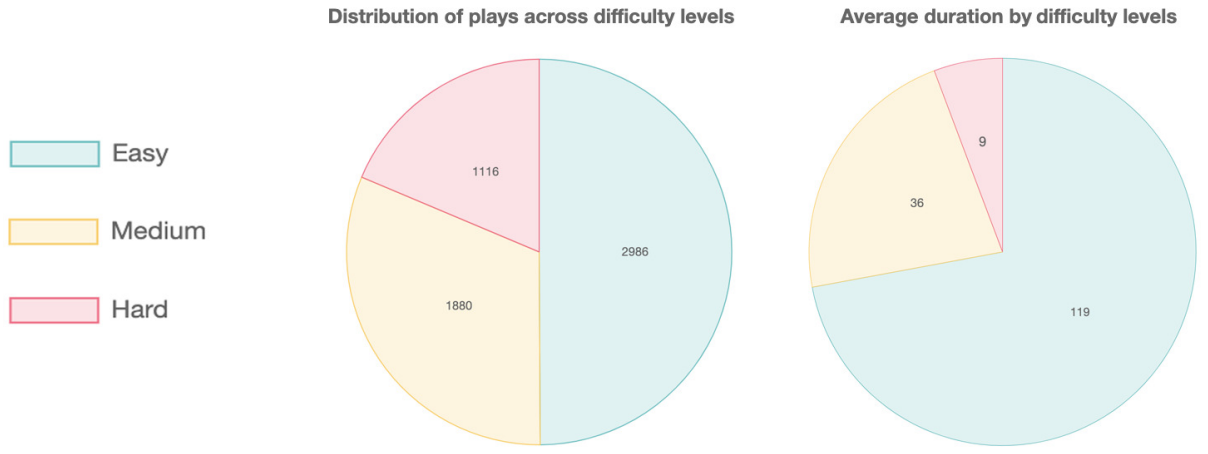

Fig. 5. Distribution of plays and session duration across difficulty levels. The *Easy* level accounts for the highest number of sessions, reflecting its appeal to a broader audience and accessibility for less experienced players. Conversely, *Hard* levels show fewer sessions, likely due to their demanding nature, which may discourage prolonged or repeated attempts by players.

datasets. Furthermore, rule-based methods are less prone to over-fitting or producing bias, as they rely on explicitly defined criteria rather than learned patterns. Further to that, rule-based approaches require minimal computational resources compared to ML-based ones, making them more cost-effective for deployment on large-scale setups. Additionally, their lightweight nature ensures compatibility with a broader range of hardware, from standard CPUs to embedded systems, enabling scalability to diverse MR environments. While the rule-based approach excels in scenarios requiring low latency and transparency, it also serves as a foundation for future hybrid approaches. Lightweight ML models can complement the rule-based approach by handling edge cases

and learning user-specific preferences over time, providing a pathway for integrating the strengths of both approaches.

The results depicted in the graphs on figure 6 demonstrate that the interaction thresholds are dynamically adapted based on user characteristics, ensuring accessibility and usability across diverse profiles. In the first graph on the left-hand side, interaction thresholds are consistently higher than arm span measurements, indicating that the system extends reach allowances to facilitate interaction. Adults show the highest interaction thresholds due to their naturally larger arm span, while mobility-impaired users have slightly reduced arm spans but still benefit from extended interaction thresholds, likely compensating for reach limitations. The graph on the centre highlights the correlation between user height and interaction thresholds. Taller users have the largest interaction thresholds, reflecting their increased reach, while children receive an extended threshold beyond their actual height to enhance usability. This suggests that the system adjusts interaction distances ergonomically, ensuring that users of all sizes can comfortably interact within the MR environment. The graph on the right-hand side focuses on mobility constraints, demonstrating how the system personalizes interaction thresholds to accommodate movement limitations. Users with severe mobility restrictions have the lowest interaction thresholds, minimizing the need for extensive movement. Those with moderate restrictions have an intermediate threshold, striking a balance between accessibility and usability. Unrestricted users exhibit the highest interaction threshold, reflecting their ability to engage in a broader interaction space. Overall, these results confirm that the egocentric interaction model effectively adjusts to physical attributes and mobility constraints, ensuring inclusivity in MR environments. The system dynamically scales interaction thresholds to provide ergonomic comfort and personalized usability, accommodating diverse user needs while maintaining intuitive interaction capabilities.

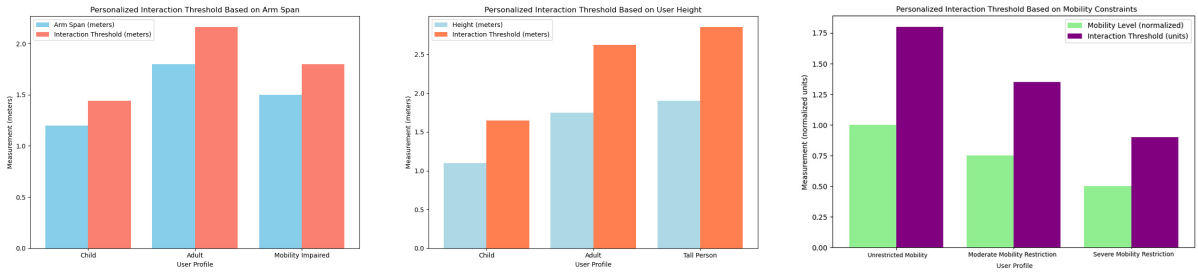

Fig. 6. Personalized Interaction Thresholds Based on User Characteristics. (Left) Interaction threshold variations based on arm span across different user profiles (child, adult, and mobility-impaired users). (Center) Interaction thresholds relative to user height, showing increased thresholds for taller users. (Right) Mobility constraints influencing interaction thresholds, where users with restricted mobility have lower interaction ranges than those with unrestricted mobility.

During our analysis of over 5,000 gameplay sessions, we observed notable differences in gesture performance across user groups, particularly when considering age and physical characteristics. For instance, children often faced challenges with the "pointing" gesture, which requires precise alignment between the hand, elbow, and shoulder. These difficulties were primarily attributed to shorter limb lengths, limited arm extension range, and less stable posture control during extended interactions which are factors consistent with anthropometric findings on human body modelling [Casas 2015]. In contrast, broader gestures such as "swipe" and "raise hand" exhibited high recognition rates across all demographics. Their success can be linked to their larger motion envelopes and the use of velocity-based thresholds, which are more forgiving to anatomical variation and fine motor control. These findings showcase the importance of tailoring gesture sets to account for differences in body scale, motor skills and posture when designing inclusive MR experiences.

8 Limitations and Future Work

Despite its strengths, the current paradigm has limitations that open avenues for further research. One notable limitation is the reliance on pre-defined gesture recognition rules. Implementing hybrid approaches via a combination of rule-based and adaptive ML models could address this by enabling real-time customizable gesture detection. These models could allow the system to continuously learn from user interactions, adapting to unique movement patterns while maintaining the low-latency performance required for MR environments. However, integrating machine learning techniques presents challenges such as increased computational demands for real-time and the need for robust model interpretability to maintain system transparency. Performance in multi-user scenarios is another area requiring further refinement. While the primary user-tracking algorithm works well under controlled conditions, dynamic environments with multiple users introduce challenges such as occlusion, overlapping skeletal data and individuals moving out of the camera's field of view. Addressing these challenges requires the development of optimized tracking algorithms, potentially through synchronized multi-view approaches or sensor fusion techniques that combine data from multiple depth cameras to ensure consistent responsiveness and robust tracking across diverse spatial configurations. Additionally, intelligent user identification strategies using probabilistic tracking models could enhance the accuracy of user-switching mechanisms in shared environments. Expanding applications beyond gaming remains another avenue for future work. In educational environments, MR systems could support collaborative learning experiences by enabling shared interaction zones and real-time performance feedback. Similarly, professional training simulations could benefit from adaptive learning mechanisms that respond to user proficiency, ensuring personalized experiences. In healthcare, the paradigm could be extended to rehabilitation exercises tailored to patients' motor capabilities and progress tracking.

Further to this, the adaptive paradigm described in this paper could be extended beyond large-scale installations to both Augmented Reality and Virtual Reality setups. In AR environments, such as glasses-based systems, the limited field of view and dynamic lighting conditions may require stricter gesture thresholds and more robust occlusion handling. The calibration model could be ported to AR by mapping interaction zones relative to the user's forward gaze rather than a fixed screen plane. In fully immersive VR systems, where users are occluded from the external world, spatial boundaries and visual anchors must be rendered in 3D space to guide interaction. While the current method was implemented with a single RGBD camera, multi-view or headset-embedded depth sensing could substitute this setup. In both contexts, the same rule-based adaptive logic remains valid, provided that skeletal key-points are reliably tracked.

Acknowledgments

This project has been partially supported by European Union's Horizon 2020 research and innovation programme under grant agreement No. 101017779.

References

Mark Billinghurst. 2021. Grand Challenges for Augmented Reality. *Frontiers in Virtual Reality* 2 (2021). doi:10.3389/frvir.2021.578080

Raquel T Cabrera-Araya and Edgar Rojas-Munoz. 2024. INDYvr: Towards an Ergonomics-based Framework for Inclusive and Dynamic Personalizations of Virtual Reality Environments. In *2024 IEEE International Symposium on Mixed and Augmented Reality Adjunct (ISMAR-Adjunct)*. IEEE, 220–222.

Llogari Casas. 2015. Creació de models humans sintètics, home i dona, aptes per a aplicacions tecnològiques i científiques. (2015). https://hdl.handle.net/2445/100602

Llogari Casas, Loïc Ciccone, Gökçen Çimen, Pablo Wiedemann, Matthias Fauconneau, Robert W. Sumner, and Kenny Mitchell. 2018. Multi-reality games: an experience across the entire reality-virtuality continuum. In *Proceedings of the 16th ACM SIGGRAPH International Conference on Virtual-Reality Continuum and Its Applications in Industry* (Tokyo, Japan) *(VRCAI '18)*. Association for Computing Machinery, New York, NY, USA, Article 18, 4 pages. doi:10.1145/3284398.3284411

Kelvin Chung and Wenping Wang. 1996. Quick collision detection of polytopes in virtual environments. In *Proceedings of the ACM Symposium on Virtual Reality Software and Technology* (Hong Kong) *(VRST '96)*. Association for Computing Machinery, New York, NY, USA, 125–132. doi:10.1145/3304181.3304206

Enea Cippitelli, Samuele Gasparrini, E. Gambi, and Susanna Spinsante. 2016. A Human Activity Recognition System Using Skeleton Data from RGBD Sensors. *Computational Intelligence and Neuroscience* 2016 (03 2016), 1–14. doi:10.1155/2016/4351435

Jessyca L. Derby and Barbara S. Chaparro. 2022. The Development and Validation of an Augmented and Mixed Reality Usability Heuristic Checklist. In *Virtual, Augmented and Mixed Reality: Design and Development*, Jessie Y. C. Chen and Gino Fragomeni (Eds.). Springer Publishing, Cham, 165–182.

João Marcelo Evangelista Belo, Anna Maria Feit, Tiare Feuchtner, and Kaj Grønbæk. 2021. XRgonomics: Facilitating the Creation of Ergonomic 3D Interfaces. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems* (Yokohama, Japan) *(CHI '21)*. Association for Computing Machinery, New York, NY, USA, Article 290, 11 pages. doi:10.1145/3411764.3445349

Libor Hargaš and Dušan Koniar. 2022. Usage of RGB-D Multi-Sensor Imaging System for Medical Applications. In *Vision Sensors*, Francisco Javier Gallegos-Funes (Ed.). IntechOpen, Rijeka, Chapter 1. doi:10.5772/intechopen.106567

Camillo Lugaresi, Jiuqiang Tang, Hadon Nash, Chris McClanahan, Esha Uboweja, Michael Hays, Fan Zhang, Chuo-Ling Chang, Ming Yong, Juhyun Lee, Wan-Teh Chang, Wei Hua, Manfred Georg, and Matthias Grundmann. 2019. MediaPipe: A Framework for Perceiving and Processing Reality. In *Third Workshop on Computer Vision for AR/VR at IEEE Computer Vision and Pattern Recognition (CVPR) 2019*.

Ruben Nogales and Marco E. Benalcazar. 2020. A Survey on Hand Gesture Recognition Using Machine Learning and Infrared Information. In *Applied Technologies*, Miguel Botto-Tobar, Marcelo Zambrano Vizuete, Pablo Torres-Carrion, Sergio Montes Leon, Guillermo Pizarro Vasquez, and Benjamin Durakovic (Eds.). Springer International Publishing, Cham, 297–311.

HyeonJung Park, Youngki Lee, and JeongGil Ko. 2021. Enabling Real-time Sign Language Translation on Mobile Platforms with On-board Depth Cameras. 5, 2, Article 77 (June 2021), 30 pages. doi:10.1145/3463498

David Sinclair, Adeyemi Vincent Ademola, Babis Koniaris, and Kenny Mitchell. 2023. DanceGraph: A Complementary Architecture for Synchronous Dancing Online. https://api.semanticscholar.org/CorpusID:259100629

Stereolabs. 2025. *Stereolabs ZED 2 Camera Documentation.* Available online at https://www.stereolabs.com/zed-2/.

Özlem Uzuner, Xiaoran Zhang, and Tawanda Sibanda. 2009. Machine Learning and Rule-based Approaches to Assertion Classification. *Journal of the American Medical Informatics Association* 16, 1 (01 2009), 109–115. doi:10.1197/jamia.M2950

Bram van Ginneken. 2017. Fifty years of computer analysis in chest imaging: rule-based, machine learning, deep learning. *Radiological Physics and Technology* 10, 1 (2017), 23–32. doi:10.1007/s12194-017-0394-5

Juan Pablo Wachs, Mathias Kölsch, Helman Stern, and Yael Edan. 2011. Vision-based hand-gesture applications. *Commun. ACM* 54, 2 (Feb. 2011), 60–71. doi:10.1145/1897816.1897838

Meng Wu. 2024. Gesture Recognition Based on Deep Learning: A Review. *EAI Endorsed Transactions on e-Learning* 10 (Mar. 2024). doi:10.4108/eetel.5191

Pengfei Yu, Shourui Yang, and Shengyong Chen. 2020. Accuracy improvement of time-of-flight depth measurement by combination of a high-resolution color camera. *Appl. Opt.* 59, 35 (Dec 2020), 11104–11111. doi:10.1364/AO.405703

Zhengyou Zhang. 2012. Microsoft Kinect Sensor and Its Effect. *IEEE MultiMedia* 19, 2 (2012), 4–10. doi:10.1109/MMUL.2012.24

Received 10 March 2025; revised 22 June 2025; accepted 11 September 2025