Gaze-Guided 3D Hand Motion Prediction for Detecting Intent in Egocentric Grasping Tasks

Yufei He

Delft University of Technology

TU Delft

Gaze-Guided 3D Hand Motion Prediction for Detecting Intent in Egocentric Grasping Tasks

by

Yufei He

Student Name	Student Number
Yufei He	5694248

Supervisors: Prof. Arno H. A. Stienen, Prof. Xucong Zhang
Faculty: Faculty of Mechanical Engineering, Delft

TU Delft

Contents

Gaze-Guided 3D Hand Motion Prediction for Detecting Intent in Egocentric Grasping Tasks

Yufei He, Xucong Zhang, and Arno H. A. Stienen

***Abstract*— Human intention detection with hand motion prediction is critical to drive the upper-extremity assistive robots. However, the traditional methods relying on physiological signal measurement are restrictive and often lack environmental context. We propose a novel approach that integrates gaze information, historical hand motion sequences, and environmental object data to predict future sequences of intended hand poses, adapting dynamically to the assistive needs of the patient without prior knowledge of the intended object for grasping. Specifically, we propose to use a vector-quantized variational autoencoder for robust hand pose encoding with an autoregressive generative transformer for effective hand motion sequence prediction. We demonstrate the usability of these novel techniques in a pilot study with healthy subjects. To train and evaluate the proposed method, we collect a dataset consisting of various types of grasp actions on different objects from multiple subjects. Through extensive experiments, we demonstrate that the proposed method can successfully predict sequential hand movement. Especially, the gaze information shows significant enhancements in prediction capabilities, particularly with fewer input frames, highlighting the potential of the proposed method for real-world applications.**

***Index Terms*— Intention Detection, Hand Motion Generation**

I. INTRODUCTION

Upper extremity movement disorders due to conditions like stroke, traumatic brain injury, and nerve damage can severely restrict the ability of individuals to perform daily tasks [1], [2]. Upper extremity assistive robots are designed to support arm and hand functions, enhancing the reacquisition of motor function through structured and adaptive exercises [3]. These robots provide targeted, intensive, and repetitive [1] training tasks that effectively mimic activities of daily living [4], thereby facilitating effective assistance. Intention detection is crucial for operating robots because it enables the robot to understand the user's desired actions, allowing it to provide customized assistance [5]. It becomes more important when transitioning from clinic-based rehabilitation robots, which use visual targets for user interaction, to home assistive robots that support everyday tasks like washing dishes or dressing. These domestic settings challenge robots to quickly understand and assist with the user's intentions. Conventional methods such as surface electromyography (sEMG) and electroencephalography (EEG) have been used for intention detection, which can directly measure physiological signals [6]. However, these methods usually have restricted movement and require frequent recalibrations. Moreover, these methods lack the sense of environmental context around the user, which is vital for analyzing interactions. Additionally, these signals in post-stroke conditions are also disturbed [7], [8] and thus become hard to correlate with movement. Recent developments in computer vision have made vision signals useful as direct or supplementary sources for intention interpretation [9]. Vision-based techniques can enable robots to learn from natural human behaviors and observe how people interact with the environment, significantly enhancing the robot's ability to assist in a manner that aligns closely with human needs.

Gaze information is important for revealing user intent by identifying areas of interest before physical actions and is effectively used in assistive robots to predict human motion [10]–[13]. However, these applications often focus only on classification tasks and may suffer from inaccuracies due to false positives [9]. For instance, external distractions can divert attention away from primary objectives. Integrating egocentric-view visual signals can provide a more robust environmental context, enhancing the interpretation of user intentions through cues like human poses and object shapes. Although research has utilized egocentric signals [14]–[16], or combined them with gaze data [17], [18], the focus has generally been on semantic predictions rather than predicting explicit future hand motions. Developing this capability is essential for assistive robots to offer effective assistance throughout training. Studies for hand motion prediction often only include the interactive object [19]–[24], or have limited hand movement with fixed start and end positions [19]–[24]. Consequently, these methods have not been studied in driving assistive robot applications.

In response to this challenge, we propose a novel task for intention detection: given a set of potential grasping objects and initial hand movements, we want to predict intended future hand motions. This task focuses on two fundamental aspects: 1) utilizing only implicit environmental context, and 2) producing explicit hand motion outputs represented by 21 hand key points. To tackle this task, we have developed a method that employs gaze and egocentric-view visual signals to predict future hand motions. This setting is practical for assistive robot applications because the user can operate the robot with the head-mounted device. This approach integrates three critical types of information: gaze data from an eye-tracker, historical hand motion records, and object details captured through egocentric video. We demonstrate the practical usability of our novel method in a pilot study with healthy subjects. To train and verify our approach, we collect a dataset from these subjects, containing various grasping types and objects. Before each grasping process, the positions of objects are randomized to ensure the robustness of the model against variations in object placement, thereby enhancing its ability to generalize across different real-world scenarios.

We developed a method consisting of a Vector-Quantized Variational AutoEncoder (VQ-VAE) and an auto-regressive generative transformer. The VQ-VAE is used for encoding hand pose, allowing for capturing in-distribution features from history for accurate motion prediction. The transformer is used for future hand motion sequence generation based on any given input frame. We also have a feature fusion architecture, which comprises several linear layers and could transform the dimensions of gaze and object features and fuse them with encoded hand motion embeddings. We validated the generalizability of our model in our self-collected dataset across different subjects and motions, and we explored the impact of various gaze fusion methods on model performance. Our findings indicate robust generalization across diverse validation settings, particularly in distance accuracy. Notably, the model with gaze integration significantly outperforms the no-gaze model, especially as the number of input frames decreases, highlighting the value of gaze information when historical data is limited. Furthermore, our model demonstrates

Yufei He and Arno H. A. Stienen are with Department of Biomechanical Engineering, Faculty of Mechanical Engineering, Delft University of Technology.

Xucong Zhang is with Department of Intelligent Systems, Delft University of Technology.

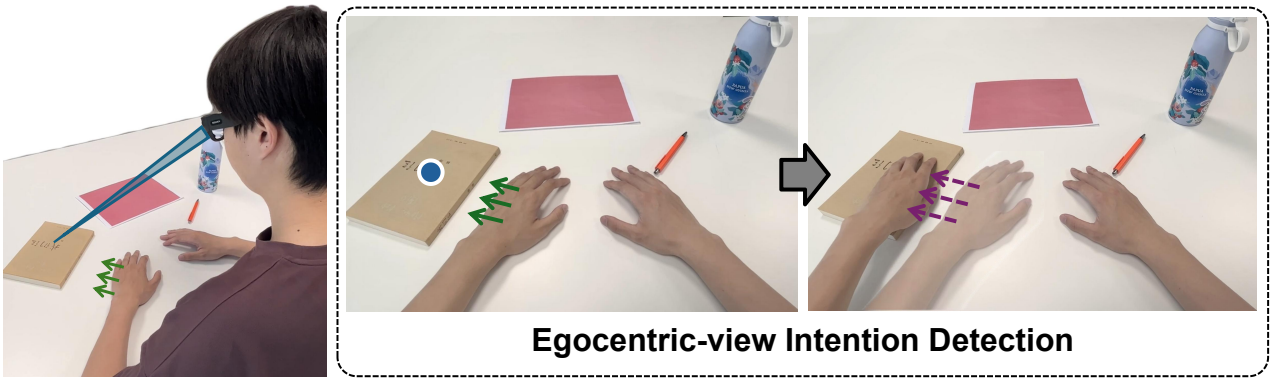


Fig. 1: **Overview of gaze-guided human intention detection.** Left: We equip the user with wearable eye-tracking glasses to obtain the gaze fixation point (blue ray), the initial hand motion indicated (green arrow), and object locations as the input. The positions of objects are randomized before each grasping process. Right: We aim to use the egocentric-view data to predict a sequence of hand motions leading up to the final grasping action on the object indicated as the purple arrow. The developed system can be used to drive the upper-extremity assistive robot.

enhanced noise resistance compared to the no-gaze model. The results show that, compared with the no-gaze model, our model has the potential to provide accurate and timely predictions in real-time situations.

In summary, our paper introduces a novel approach to hand motion prediction that enhances hand movements in interactive tasks. The main contributions are:

1) We propose a new task of explicit hand motion sequence prediction given implicit environmental context towards the goal of driving upper extremity assistive robots.
2) We introduce a novel method that combines gaze data with egocentric visual signals for hand motion prediction.
3) We validate that our model generalizes effectively to the grasping behaviors of both new subjects and objects, illustrating its broad applicability.

II. RELATED WORKS

A. Human Motion Generation

Motion generation is the process of creating natural, human-like motion from multimodal inputs such as text [25], [26], speech [27], [28], and motion history [29]–[35]. 3D human motion prediction, which uses the motion sequence history as a condition, is one of the most important motion generation tasks.

Recurrent Neural Networks (RNNs) and Graph Convolution Networks (GCNs) have been two popular methods used to capture temporal and spatial relationships in human movement. Fragkiadaki et al. [29] developed the Encoder-Recurrent-Decoder (ERD) network. It incorporates nonlinear encoder and decoder networks and a Long Short-Term Memory (LSTM) network to predict future human motion. Jain et al. [30] introduced a Structural-RNN that segments the skeletal hierarchy into clusters to encode semantic similarities among different body parts, taking spatio-temporal information into consideration. Extending beyond action-specific models, Martinez et al. [31] developed a method for multi-action contexts. They integrated residual connections in the decoder to model velocities, thereby enhancing the smoothness and accuracy of motion prediction. Additionally, GCNs have been employed to grasp the spatial relationships among joints. Mao et al. [32] proposed the DCT-GCN model, which utilizes the Discrete Cosine Transform (DCT) to encode temporal pose information in trajectory space, while graph convolutional layers with residual connections learn spatial relationships.

With the advancement of transformers, studies have leveraged attention mechanisms to enhance motion prediction tasks. To improve previous works, Mao et al. [33] integrate an attention mechanism that assesses the similarity between current motion contexts and historical motion sub-sequences. Aksan et al. [34] introduced a dual attention concept, incorporating spatial and temporal attention modules that operate in parallel. This design enables the model to simultaneously access current and past information, enriching its contextual understanding. Cai et al. [35] developed a transformer-based approach with a progressive decoding strategy to predict DCT coefficients, focusing on central to peripheral extensions based on structural connectivity. This method also uses a memory-based dictionary to preserve and utilize global motion patterns from the training data, enhancing prediction accuracy.

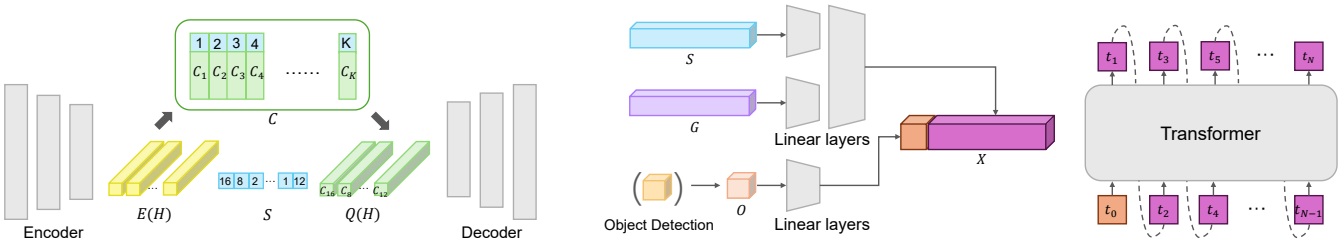
B. Hand-Object Motion Generation

Recent advancements in hand motion generation have attracted significant research interest, with hand-object interaction being a particularly popular area. A variety of datasets capturing both hand and object interactions have been developed [36]–[42], enabling studies on grasp generation [43]–[45]. For example, Jiang et al. [43] refined grasping gestures using contact maps on objects. However, these studies have been limited to generating static gestures. While some efforts have extended to creating motion sequences, these often depend on explicit conditions such as the geometry or position of the object [19]–[24], initial or final hand positions [19]–[24], trajectories [46], or textual descriptions [19]. Christen et al. [23] introduced a method for synthesizing diverse hand motions based on the start and end poses of an object.

Although effective in generating accurate hand motions, the previous works do not fully capture the underlying human intentions. In this work, we explore the integration of gaze data, a strong indicator of human intention, with hand motion to predict future hand movements. We also aim to predict the grasping process under a more naturalistic condition by presenting multiple potential grasping objects instead of a single predetermined one, thereby enhancing the adaptability and realism of the motion prediction.

III. METHOD

Our method performs intention detection via 3D hand motion prediction from the input of user eye gaze, historical hand motions, and object information. To perform the task, our method consists of



(a) Hand Motion VQ-VAE (b) Hand Motion Generator

Fig. 2: **Overview of our framework for hand motion prediction.** The framework consists of two main components: (a) Hand-Motion VQ-VAE, which encodes hand motion into codebook indices, C represents codebook, S represents the encoded hand motion indices; and (b) Hand Motion Generator, which contains feature Fusion layers and a transformer. In feature fusion layers, the encoded hand motion S is integrated with eye-gaze and object features G and O, together forming fused feature X. The transformer predicts future hand motion indices in an auto-regressive manner using a transformer architecture. These indices are subsequently decoded using the VQ-VAE decoder to obtain the predicted hand motions.

two modules, including the hand motion VQ-VAE for discrete hand pose codebook learning and the hand motion generator to predict the sequence of hand motions. In this section, we first describe the problem in formulation and then introduce each module individually.

A. Problem Formulation

The intention detection can be formulated as a model M that predicts a sequence of future 3D hand motion $\hat{H}$ based on an initial sequence of 3D hand motion H, a corresponding sequence of eye gazes G, and the representation of the possible interactive objects O in the first frame. The task is formally defined as:

$$\hat{H} = M(H, G, O). \tag{1}$$

$H = \{h_t\}_{t=1}^{\tau}$ is a sequence of input hand motion, where τ is the input frame number. The 3D hand pose at frame t, denoted by $h_t \in \mathbb{R}^{126}$, is defined by the positions of 21 3D hand joint locations (x,y,z) for both hands. This configuration includes 20 finger joints and one wrist position per hand, according to the Mediapipe framework [47], resulting in a total dimension of $126 = 21 \times 2 \times 3$. Similarly, $G = \{g_t\}_{t=1}^{\tau}$ is a sequence of eye gaze fixation points, represented as $g_t \in \mathbb{R}^3$, is characterized by the 3D eye fixation point (x,y,z) in the world coordinate system. Objects in the scene are represented by maximum four of 3D points $O = \{o^k\}$, where $o_k \in \mathbb{R}^{12}$. For instance, a sheet of paper is described using the positions of its four corners, while a pen is represented by the positions of its tip and bottom, reflecting their distinct shapes. The predicted sequence 3D hand motion $\hat{H} = \{\hat{h}_t\}_{t=\tau+1}^{T}$ consists of 3D hand pose at frame t, $\hat{h}_t \in \mathbb{R}^{126}$, maintains the same dimension as the input hand pose h_t while are in the future sequence, starting from frame $\tau+1$ until the end frame T.

B. Hand Motion VQ-VAE

Hand poses have a large space of movements that is difficult to model. A similar problem exists in the human body pose modeling, where the VQ-VAE [48] has been proposed to encode the continuous body movements into discrete classes within a latent space [49]. We utilize the VQ-VAE to learn multiple hand poses, which can be represented as discrete classes in the motion generation phase. Specifically, given the input sequence of hand motion H, our goal is to encode this sequence into discrete embeddings using an encoder E coupled with a learnable codebook C, and then reconstruct the sequence via a decoder D. An overview of the Hand Motion VQ-VAE model is presented in Fig. 2 (a). The encoded features serve as inputs for the hand motion generation network. The codebook is defined as $C = \{c_i\}_{i=1}^{K}$, where each c_i belongs to $\mathbb{R}^{D_c}$, K represents the size of the discrete latent space, and D_c is the dimensionality of each embedding vector. The sequence is encoded as $E(H) = \{e_t\}_{t=1}^{\lfloor T/l \rfloor}$, with each embedding $e \in \mathbb{R}^{D_c}$ and l denoting the downsampling scale. The discrete embeddings $Q = \{q_t\}_{t=1}^{\lfloor T/l \rfloor}$ and indices $S = \{s_t\}_{t=1}^{\lfloor T/l \rfloor}$ for each frame are computed as:

$$q_t = \arg\min_{c_i \in C} \|e_t - c_i\|_2 \tag{2}$$

$$s_t = \arg\min_{i} \|e_t - c_i\|_2 \tag{3}$$

1) Network Architecture: Inspired by previous work [25], we integrated a convolutional architecture featuring a combination of convolution layers, residual blocks, and ReLU activation functions in developing the encoder and decoder in our model. Specifically, the encoder utilizes two convolutional layers with a stride of two for temporal downsampling, reducing the temporal length by a factor of four. This approach not only minimizes computational demands but also reduces noise within the input data. In contrast, the decoder employs nearest-neighbor interpolation for upsampling, facilitating the reconstruction of the complete hand motion sequence.

2) Optimization Strategy: To optimize the VQ-VAE model, the loss function L_{vq} consists of reconstruction loss, embedding loss, and commitment loss, detailed as follows:

$$L_{recon} = \begin{cases} 0.5(h_t - \hat{h}_t)^2/\beta, & \text{if } |h_t - \hat{h}_t| < \beta \\ |h_t - \hat{h}_t| - 0.5\beta, & \text{otherwise} \end{cases} \tag{4}$$

$$L_{embed} = \|\text{sg}[e_t] - q_t\|_2^2 \tag{5}$$

$$L_{commit} = \gamma \|e_t - \text{sg}[q_t]\|_2^2 \tag{6}$$

where β and γ are hyper-parameters that influence reconstruction loss and commitment loss, respectively. "sg" represents the stop-gradient operator, which prevents the back-propagation of gradient, treating the variable as a constant during the optimization process. The total loss is written as $L_{vq} = L_{recon} + L_{embed} + L_{commit}$.

C. Hand Motion Generator

With a trained hand-motion VQ-VAE model, the input hand-motion sequence $H = \{h_t\}_{t=1}^{T_g}$ is encoded into a sequence of quantized indices $S = \{s_t\}_{t=1}^{T_d}$, where $T_d = \lfloor T_g/l \rfloor$. As demonstrated in Fig. 2 (b), these indices are fused with gaze features G and conditioned on object features O, serving as inputs X to the hand-motion generator. This generator operates in an autoregressive manner, producing predicted hand motion indices $\hat{S} = \{\hat{s}_t\}_{t=1}^{T_d+1}$. Given the combined features

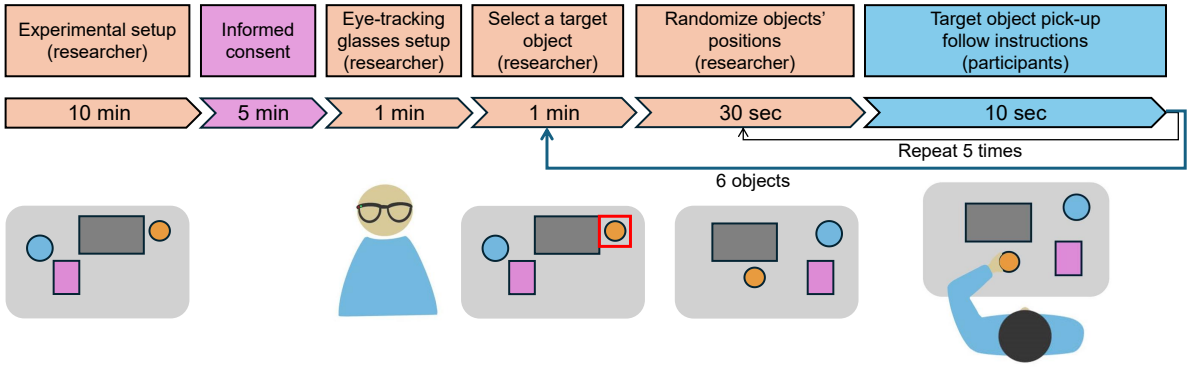


Fig. 3: **Data Collection Procedure.** This flowchart outlines the sequence of activities involved in the experiment, with activities differentiated by color: orange for researcher tasks, blue for participant tasks, and purple for joint tasks. The procedure begins with the researcher setting up the experiment, followed by obtaining informed consent. The researcher then sets up the eye-tracking glasses for the participant, selects a target object and randomizes the positions of all objects. Participants perform the target object pick-up task five times, each in a newly randomized position, across a total of six objects.

up to the previous frame X_{t-1}, the probability of each code book index being selected for the next frame hand-motion is calculated as $p_i(s_t|X_{t-1})$. The next-frame hand-motion index is determined by:

$$\hat{s}_t = \arg\max_i p_i(s_t|X_{t-1}) \tag{7}$$

The sequence of indices is mapped to the learned codebook embeddings, forming $\hat{Q} = \{\hat{q}_t\}_{t=1}^{T_d+1}$, where each $\hat{q}_t$ corresponds to $c_{\hat{s}_t}$ from the codebook. This encoded sequence $\hat{Q}$ is then processed by the decoder D, which reconstructs the predicted hand motion sequence $\hat{H} = \{\hat{h}_t\}_{t=1}^{T_p}$.

1) Feature Combination: One of the key components of the proposed model is a feature combination of hand and eye gaze. The hand-motion indices are represented using a lookup table as token embeddings, resulting in a hand embedding sequence $S' = \{s'_t\}_{t=1}^{T_d}$, with each $s'_t \in D_h$. To align the dimensional differences between gaze features and hand-motion token embeddings, the gaze features are expanded using a linear layer, producing $G' = \{g'_t\}_{t=1}^{T_d}$ where each $g'_t \in D_g$. These embeddings are then concatenated and passed through a linear feature fusion layer followed by a ReLU function, resulting in the combined hand-eye embeddings $F(S,G) = \{f_t\}_{t=1}^{T_d}$, where each $f_t \in \mathbb{R}^{D_x}$. Notably, we do not apply out-of-shelf object detection here, due to the existence of well-established methods for accurate real-time object detection, allowing us to focus on other aspects of our study. Instead, we manually extract object positions from the first frame and transform via a linear layer to match D_x, forming $O' \in \mathbb{R}^{D_x}$, which acts as a conditioning input. The object embeddings are concatenated at the start of the sequence to create $X = \text{Concat}(O', F) = \{x_t\}_{t=0}^{T_d}$, with each $x_t \in \mathbb{R}^{D_x}$.

2) Decoder-only Transformer Architecture: In the hand-motion generator, we employ decoder-only transformers with masked self-attention layers similar to [25] for human pose generation, enabling the model to sequentially learn the input tokens. The masked self-attention is calculated as follows:

$$Q = XW^Q; K = XW^K; V = XW^V \tag{8}$$

$$Att(Q,K,V) = \text{softmax}\left(\frac{QK^T - M}{\sqrt{D_x}}\right)V \tag{9}$$

$$M_{i,j} = \begin{cases} 0 & \text{if } i \geq j, \\ -\infty & \text{if } i < j. \end{cases} \tag{10}$$

W^Q, W^K, and $W^V \in \mathbb{R}^{D_x \times D_x}$ represent the linear projection weights for queries, keys, and values, respectively. Att is the soft attention operation. M is the mask ensuring predictions for a position do not depend on the following positions.

TABLE I: Summary of Motions and Interactions

Motion	Grasping Type [51]	Involved Object	Num. Hands
Pick up a bottle	Type A	Bottle	1
Move a piece of paper	Type B	Paper	1
Pick up a book	Type C	Book	1
Pick up a phone	Type C	Phone	1
Pick up a pen	Type D	Pen	1
Pick (an) earphone(s)	Type D	Earphone(s)	1 or 2
Write on paper	Type B, D	Paper, Pen	2

3) Optimization Strategy: The loss for the transformer model is computed as a classification task that the predicted probabilities of each index are compared against the actual hand-motion indices. To emphasize the significance of the predictions for future positions, particularly the last output embedding, we assign a higher weight to the last index in the sequence. The loss is calculated as follows:

$$L_{transformer} = -\sum_{t=1}^{N} w_t \cdot \log(p(\hat{s}_t|X_{t-1})), \tag{11}$$

where w_t is the weight assigned to each index, and N is the length of the learned sequence. Specifically, w_N, the weight for the last index, is greater than the weights assigned to other indices.

D. Dataset Collection

We utilize the Project Aria Glasses from Meta [50] to capture eye-tracking data and an egocentric view of grasping procedures. For this study, 15 volunteers were recruited to participate in the data collection process. Prior to the experiments, participants received comprehensive instructions detailing the tasks and procedures, and each object was associated with a specific grasping type [51], as shown in Table I. The dataset collection was approved by our ethical committee.

1) Experiment Procedure: The entire experimental procedure is depicted in Fig.3. Participants, already equipped with eye-tracking glasses, are seated at a table with their hands placed palms down. An instructor randomly positions a target object on the table for grasping alongside other objects to simulate a real-life scenario. The table measures 1.15 meters in length and one meter in width, all objects being within easy reach of participants who may lean

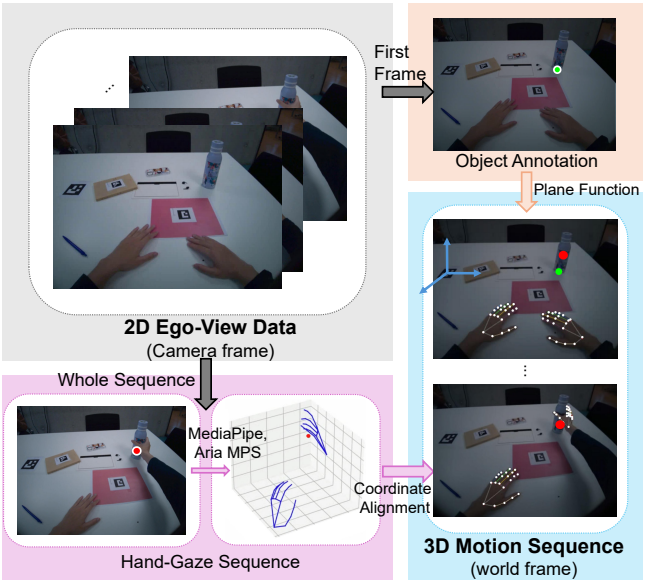


Fig. 4: **Data processing pipeline.** This figure illustrates the sequence of steps applied to process egocentric video data for analysis: (a) Raw 2D images are captured from an egocentric-view video. (b) Throughout the entire sequence, the Mediapipe framework and Aria MPS are utilized to extract 3D hand motion, while Aria MPS extracts 3D gaze points. (c) The object representation is manually annotated on the first frame of the video. (d) A world coordinate is employed to integrate the hand-gaze sequence with the object representation into a unified 3D world frame.

forward slightly to grasp them. Participants are instructed to pick up the object with their preferred hand using a predetermined type of grasp. Each object is grasped five times by each participant from randomly determined positions, totaling thirty grasping attempts. With the exception of earphones—which may be placed singly or in pairs, requiring bimanual coordination if paired—all objects can be grasped with one hand. Additionally, a bimanual task—writing on paper—is included and performed once per participant.

This study primarily focuses on the reaching phase of the grasping process. We hypothesize that participants enter the experiment with preconceived notions regarding the position, size, and shape of the object, which influence their initial gaze direction. It is therefore hypothesized that the gaze of the participant is directed specifically toward the intended object. Upon the start of the recording, the gaze of the participant is expected to shift toward this object, resulting in prolonged fixation. For data augmentation during the training process, the positions of non-grasping objects are randomized based on our hypothesis that the participant gaze is only related to the grasping object.

2) 3D Eye-Hand-Object Data Acquisition: The data acquisition procedure, illustrated in Fig. 4, utilizes only 2D information directly available from the video footage due to device limitations. To convert this into 3D data, we initiate a 2D-to-3D mapping process. Initially, we extract relative 3D hand joints from the 2D video using MediaPipe [47]. Subsequently, using the Project Aria Machine Perception Service (MPS), we obtain 3D wrist positions and gaze data from an egocentric viewpoint. By assuming uniform joint-to-wrist lengths across participants, we project these relative 3D joints into a global 3D space based on the known 3D wrist positions. For object positioning, we manually identify their exact locations in the video and map these into 3D space using the plane function defined by the table surface. The 3D eye gaze point is provided by the Aria glasses. Lastly, we synchronize the hand motion, gaze, and object data within a unified coordinate system origin at 1.15 meters to the right and 0.5 meters from the bottom of the table plane.

IV. EXPERIMENTS AND RESULTS

A. Evaluation Metric and Baseline

We calculate the Euclidean distance of the palm positions as position errors between the predicted motion and the ground truth. We also calculate the pose errors after correcting the position errors.

Position Error: The position error quantifies the deviation in palm position between the prediction and the ground truth as follows:

$$e_{\text{position}} = \frac{1}{T}\sum_{t=1}^{T} \|p_t - \hat{p}_t\|_2, \tag{12}$$

where p_t and $\hat{p}_t$ are the ground truth and predicted palm positions at each frame t, and T is the total number of frames.

Pose Error: The pose error is computed by first correcting the predicted palm positions for the distance error and then calculating the average across all joints J:

$$\hat{h}'_{t,j} = \hat{h}_{t,j} - (\hat{p}_t - p_t), \tag{13}$$

$$e_{\text{pose}} = \frac{1}{T \times J}\sum_{t=1}^{T}\sum_{j=1}^{J} \|h_{t,j} - \hat{h}'_{t,j}\|_2, \tag{14}$$

where $h_{t,j}$ and $\hat{h}'_{t,j}$ represent the ground truth and adjusted predicted joint positions at each frame t for each joint j, and J denote the number of joints.

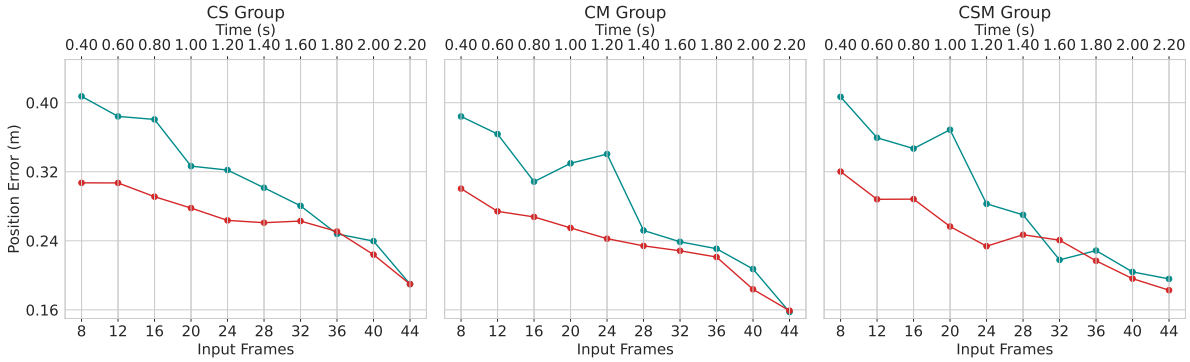
For the upper extremity assistive robot applications, we are particularly concerned about the prediction of the final grabbing pose given the current input. It has practical usage in giving signals to assistive robots early. To reflect the ability of early prediction, we computed the position and pose error only on the final grabbing pose as *End-Pose error*, which we consider only the last frame ($t = T$), where $p_t = p_T$, $\hat{p}_t = \hat{p}_T$; $h_{t,j} = h_{T,j}$, $\hat{h}'_{t,j} = \hat{h}'_{T,j}$.

We establish a baseline model that only takes hand motion sequences and object embeddings as input, i.e. without the eye-gaze feature. The model architecture remains the same as our proposed model. With this baseline, we want to investigate the effectiveness of the eye gaze feature for the human intention detection task.

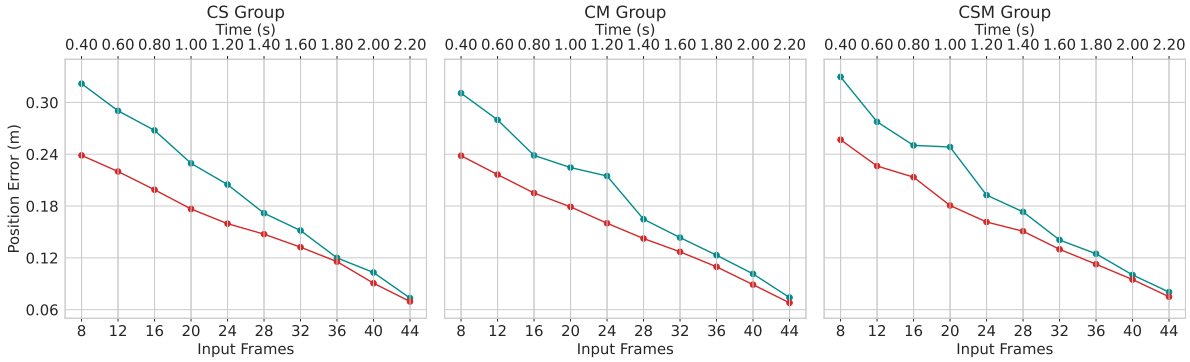
B. Cross-Subject and Motion Generalization

To evaluate the generalization capability of the proposed method in terms of subjects and motions, we designed three evaluation settings. Two specific actions, "pick up a book" and "write on a piece of paper", were selected for motion validation that had always been removed from any training procedure. The first evaluation setting is cross-subject (CS), where we performed the five-fold cross-subject evaluation on the 15 subjects from our self-collected dataset. Note that the subjects are different in the training and test sets, while hand actions are the same in this setting. The second evaluation setting is cross-motion (CM), where we train and test on the same groups of subjects while testing only on the "pick up a book" and "write on a piece of paper" actions that were not presented in the training set. Note that the subjects are the same in the training and test sets, while the hand actions are different in this setting. The third evaluation setting is across both subjects and motions, where we performed the five-fold cross-subject evaluation while only testing on the "pick up a book" and "write on a piece of paper" actions. Note that both subjects and hand actions are different in this setting.

All results reported were derived from this comprehensive cross-validation strategy. Position and pose errors were evaluated across a



(a) End-Pose Position Error

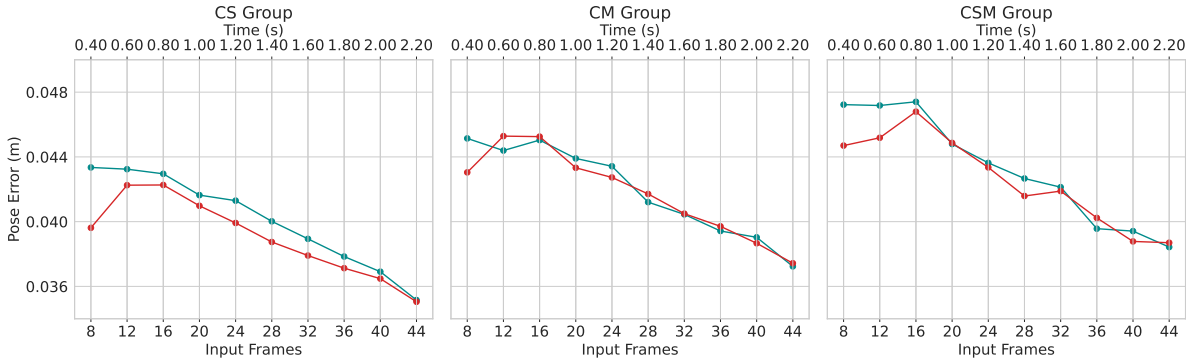


(b) Average Position Error

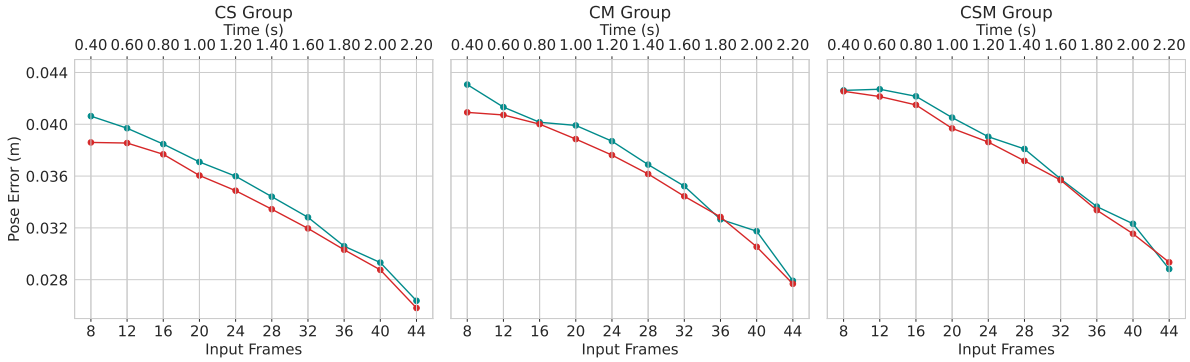
Fig. 5: **Position Errors (in *m*) across Various Input Frames and Time (in *s*).** This figure displays the end-pose (first row) and average (second row) *position* errors within the CS, CM, and CSM groups across different numbers of input frames and time. Red lines represent results with gaze, and green lines represent results without gaze.

TABLE II: **Position error Comparison of End-Pose by Fusion Type.** This table displays the position error (in *m*) for end-pose across different input frames, comparing various fusion types within the CS, CM, and CSM groups.

Validation Type	Fusion Type	Input Frames									
		8	12	16	20	24	28	32	36	40	44
CS	w/o Gaze	0.4074	0.3841	0.3806	0.3265	0.3220	0.3014	0.2805	0.2480	0.2396	0.1899
	Linear	**0.3071**	**0.3070**	**0.2911**	**0.2779**	**0.2636**	**0.2610**	0.2629	0.2508	**0.2240**	**0.1898**
	Convolution	0.3320	0.3195	0.3417	0.3044	0.2765	0.2819	0.2705	0.2635	0.2344	0.1954
	Summation	0.4149	0.3265	0.3171	0.3199	0.2829	0.2715	**0.2611**	**0.2425**	0.2260	0.1967
CM	w/o Gaze	0.3840	0.3637	0.3085	0.3297	0.3405	0.2520	0.2389	0.2308	0.2073	**0.1575**
	Linear	**0.3004**	**0.2741**	**0.2677**	**0.2549**	**0.2425**	**0.2341**	**0.2284**	0.2212	**0.1838**	0.1590
	Convolution	0.3427	0.2988	0.3041	0.2934	0.2810	0.2719	0.2629	0.2475	0.2222	0.1822
	Summation	0.3494	0.3223	0.3146	0.2988	0.2656	0.2561	0.2295	**0.2007**	0.1882	0.1624
CSM	w/o Gaze	0.4068	0.3593	0.3468	0.3687	0.2828	0.2701	**0.2179**	0.2287	0.2039	0.1959
	Linear	**0.3202**	**0.2880**	**0.2883**	**0.2566**	**0.2337**	**0.2470**	0.2407	0.2168	**0.1962**	0.1828
	Convolution	0.3479	0.3233	0.3309	0.3108	0.2681	0.2665	0.2458	0.2349	0.2210	0.1863
	Summation	0.3315	0.3067	0.2980	0.2805	0.2783	0.2706	0.2245	**0.2146**	0.2139	**0.1713**



(a) End-Pose Pose Error



(b) Average Pose Error

Fig. 6: **Pose Errors (in *m*) across Various Input Frames and Time (in *s*).** This figure displays the end-pose (first row) and average (second row) *pose* errors within the CS, CM, and CSM groups across different numbers of input frames and time. Red lines represent results with gaze, and green lines represent results without gaze.

range of input frames from 8 to 44 (a range of time from 0.4 seconds to 2.2 seconds), increasing in increments of four to show the early prediction of the proposed method.

We show the results of end-pose error and average position error in Fig. 5. Across all evaluation settings, the position errors demonstrated a decreasing trend as the number of input frames increased for both end-pose and average position errors. Models integrating gaze information generally exhibited lower errors across CS, CM, and CSM settings, although there were exceptions. All three settings exhibited similar position error patterns, indicating that the model generalizes well across different settings. Notably, the disparity in position errors between models with and without gaze became more pronounced with fewer input frames. Gaze-enhanced models showed smaller errors, suggesting the potential of gaze-enhanced models to provide more accurate and immediate corrections in real-time applications where rapid response is crucial.

The average error across the entire grasping process was evaluated similarly to the end-pose error, as depicted in Fig. 5 (b). All groups—CS, CM, and CSM—demonstrated consistent trends where models with gaze information outperformed those without. This indicates that gaze information plays a critical role in guiding the movement process. The performance gap between the gaze and no-gaze models became more pronounced as the number of input frames decreased, suggesting that gaze information is particularly beneficial in the early stages of input where less historical data is available to aid prediction.

As shown in Fig. 6, for pose error, the CS group shows a similar but less pronounced improvement with the integration of gaze, suggesting that gaze contributes positively but more modestly to pose accuracy. However, in the CM and CSM groups, pose errors showed no significant differences with respect to gaze usage. The CS group exhibited the lowest pose errors compared to the CM and CSM groups, suggesting that gaze integration is more effective in motion settings where the validation conditions closely match the training conditions.

C. Ablation Study on Gaze Fusion Techniques

The gaze information is shown to be critical for hand motion prediction in our previous experiments. To investigate the optimal way of integrating gaze information into the model, we compared our linear feature integration with two simple yet effective gaze-fusion methods convolutional fusion and direction summation. For the convolutional fusion method, we incorporated convolution layers equipped with 1×1 kernels. In the direct summation method, we first expanded the gaze feature to match the dimensionality of the hand motion features before adding them together. These methods were tested following the procedures described in the previous experiment settings. We analyzed the results based on the end-pose error. The results for position and pose errors are validated across the CS, CM, and CSM groups, as depicted in Tab. II.

The linear combination method in our model generally surpasses other methods in reducing position error across the CS, CM, and

CSM groups, although there are a few exceptions. Specifically, for shorter input sequences ranging from 4 to 28 frames, the linear combination consistently delivers superior performance compared to other methods. As the length of the input sequences increases, providing a longer historical context, the performance of the other methods becomes comparable. This means, under conditions of limited historical data, our linear combination method demonstrates enhanced effectiveness.

D. Effect of Noise on Hand Joint

In our previous experiments, data was corrected and smoothed to ensure a noise-free environment. However, noise is inherently present in the dataset collection process. The hand motion data was captured using Aria Glasses RGB camera, which produces a fish-eye output. This, combined with the camera's narrow field of view and instances where the subject moves their head, often results in parts of the hands being occasionally missing or appearing at the very edges of the frame, which can lead to extensive distortion. Additionally, mapping the hand motion into 3D involves significant coordinate transformations, which can influence the reliability and accuracy of hand motion detection. These factors collectively pose challenges to the accuracy of applying such methods in real-time prediction scenarios.

To simulate high noise levels in hand motion estimation, we introduced per-joint Gaussian noise to the system. This noise level and its distribution were chosen based on previous research [52]. We applied Gaussian noise with mean errors of 0.1 m, 0.15 m, 0.2 m, 0.25 m, and 0.3 m, calculating the associated Gaussian standard deviations using the mean of the Chi distribution as follows:

$$\sigma = e\sqrt{\frac{\pi}{8}} \tag{15}$$

This noise was independently applied to each joint for every frame. To ensure consistency, the same noise was applied for both models with and without gaze integration at the same timestep. We specifically analyzed the impact of this noise on the shortest input sequence of 8 frames. The results of this analysis are presented in Fig. 7.

As the noise level rises from 0.1 m to 0.3 m, a consistent increase in both position and pose errors is observed. The integration of gaze information consistently reduces these position errors across all groups, demonstrating its effectiveness in noisy conditions. Notably, when the input noise level is below 2.0 m, the position error closely approximates the error observed without noise, illustrating the robustness of the gaze-enhanced model. This indicates the resilience of the gaze-applied model. The benefit of gaze integration becomes more significant with larger noise levels in all groups, contributing significantly to the stability and accuracy of the system in noisy environments. For pose errors, a similar but less pronounced trend is evident. Gaze-enhanced models outperform non-gaze models in the CS group, yet demonstrate a reduced effect in mitigating pose errors within the CM and CSM groups. This suggests that in scenarios involving cross-motion validation, the gaze information is less beneficial for unknown pose accuracy.

V. DISCUSSION

In this study, we developed a gaze-guided method for human intention detection, utilizing a hand pose VQ-VAE for encoding motion and a decoder-only transformer for hand motion sequence prediction. This article aims to prove the concept that the gaze-enhanced model is effective for real-time intention detection. We have shown that gaze guidance not only significantly improves distance accuracy, particularly in the initial stages of prediction These characteristics underscore the suitability of our method for real-time applications. While we currently use manual object annotation, this method could be replaced with real-time object detection techniques to further boost the real-time functionality of the system.

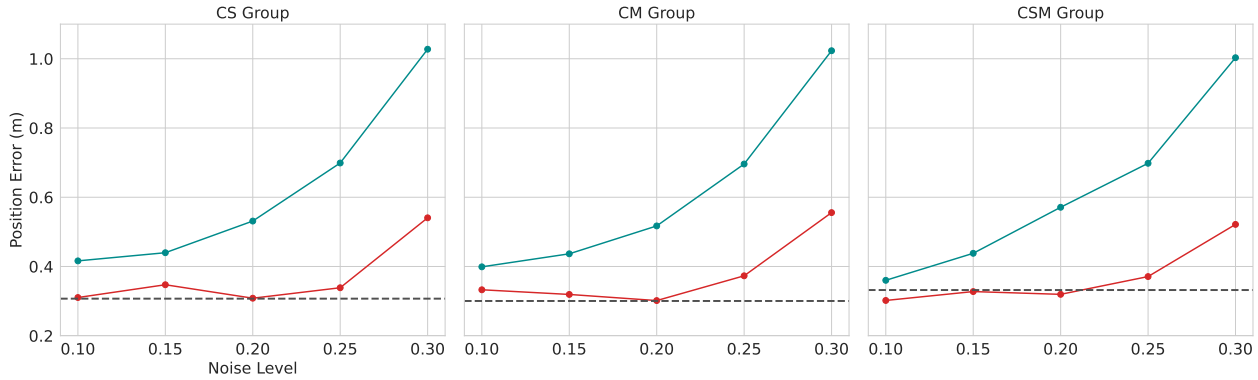
We employed the VQ-VAE in our method, which utilizes a learned codebook to select features. This characteristic ensures robustness, as the model primarily generates learned, in-distribution motion patterns. Consequently, the decoded hand motions are likely to reflect patterns observed during the training process, thus avoiding the generation of unrealistic motions. The "pick-from-codebook" nature of VQ-VAE inherently reduces noise by aligning the hand features with the nearest embeddings in the codebook, serving as a natural filter for anomalies. Moreover, the model downsampling design in its encoder and decoder effectively smooths raw motion data. Despite these advantages, it is important to acknowledge that the VQ-VAE can sometimes introduce inaccuracies into the generated motions. This issue typically arises when the model encounters data points that deviate significantly from the training distribution, challenging the model's ability to accurately reproduce those motions.

In the results, the cross-motion validation group trained with the gaze model did not show an improvement in pose error compared to the non-gaze model. Such shortcomings could be attributed to inadequate object representation, where only a few points were used to annotate each object. This could result in the model's limited capability in generalizing motion and object interactions. Additionally, the limited variety of objects involved in the training is also a reason. Consequently, the model fails to effectively learn the relationship between specific grasping types and object geometry. To address this issue, a more detailed and accurate object representation, such as a 3D mesh or point cloud, may be required. A potential generalization method could involve using "Segment Anything" [53] to segment the objects, then transfer points in the segmented sections into 3D point clouds. This would allow the creation of 3D affordance maps, which indicate which sections of an object are suitable for specific types of grasps.

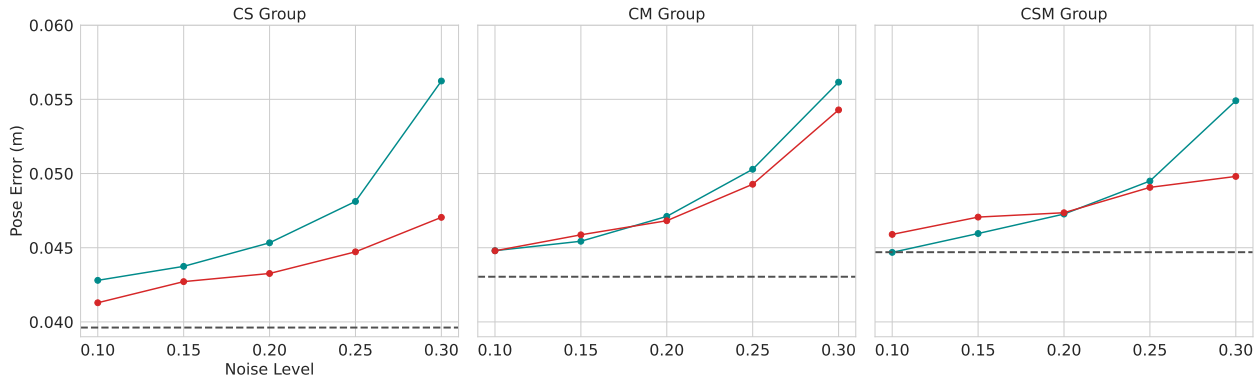
In the context of implementing assistive robots, the occurrence of larger errors in the initial frames may initially seem problematic. However, this is mitigated by the dynamic capabilities of these robots, which are designed to continuously adjust based on real-time feedback. This feature allows for the correction of any initial inaccuracies in assistance as more precise motion data becomes available in subsequent frames. The ability to recalibrate and refine assistance forces ensures that the robot remains functional and effective, even when starting with less accurate predictions. Regarding the acceptability of an error margin of approximately 16 cm, as detailed in Fig. 5 of our experiments, this level of precision is sufficient for assistive robots to approximate the direction toward the intended target. This margin allows the robot to guide the user's hand movements within a close range of the object, after which the user can make finer adjustments manually if necessary. This degree of accuracy supports effective interaction with various objects without requiring the robot to perform with absolute precision, thereby simplifying the technological requirements and enhancing the system's practicality for everyday use.

VI. CONCLUSION

This study presents an intention detection method that effectively integrates gaze data and egocentric visual cues to predict hand motion sequences, particularly in grasping tasks. By incorporating a VQ-VAE and an auto-regressive generative transformer, our approach not only predicts future hand poses with a high degree of accuracy but also demonstrates robustness against significant noise levels



(a) End-Pose Position Error



(b) End-Pose Pose Error

Fig. 7: **End-Pose Position and Pose Error (in *m*) across Different Noise Levels.** This figure displays end-pose position error (a) and pose error (b) in CS, CM, and CSM groups with 8 input frames (0.4 seconds). Red lines represent results with gaze, and green lines represent results without gaze. Gray dashed lines indicate, with 8 input frames, the end-pose position and pose errors when using a gaze-inclusive model without noise, applied for the corresponding group.

and adaptability to different subjects and objects. These findings underscore the efficacy of our gaze-enhanced model, facilitate its application in real-time interactive environments where rapid and reliable intention detection is critical.

REFERENCES

[1] A. C. Lo, P. D. Guarino, L. G. Richards, J. K. Haselkorn, G. F. Wittenberg, D. G. Federman, R. J. Ringer, T. H. Wagner, H. I. Krebs, B. T. Volpe, C. T. Bever, D. M. Bravata, P. W. Duncan, B. H. Corn, A. D. Maffucci, S. E. Nadeau, S. S. Conroy, J. M. Powell, G. D. Huang, and P. Peduzzi, "Robot-assisted therapy for long-term upper-limb impairment after stroke," *New England Journal of Medicine*, vol. 362, no. 19, pp. 1772–1783, 2010.

[2] S. Luo, Q. Meng, S. Li, and H. Yu, "Research of intent recognition in rehabilitation robots: a systematic review," *Disability and Rehabilitation: Assistive Technology*, vol. 19, no. 4, pp. 1307–1318, 2024.

[3] A. Basteris, S. M. Nijenhuis, A. H. Stienen, J. H. Buurke, G. B. Prange, and F. Amirabdollahian, "Training modalities in robot-mediated upper limb rehabilitation in stroke: a framework for classification based on a systematic review," *Journal of neuroengineering and rehabilitation*, vol. 11, pp. 1–15, 2014.

[4] M. Guidali, A. Duschau-Wicke, S. Broggi, V. Klamroth-Marganska, T. Nef, and R. Riener, "A robotic system to train activities of daily living in a virtual environment," *Medical & biological engineering & computing*, vol. 49, pp. 1213–1223, 2011.

[5] D. Wang, X. Gu, and H. Yu, "Sensors and algorithms for locomotion intention detection of lower limb exoskeletons," *Medical Engineering & Physics*, p. 103960, 2023.

[6] Y. Gu, Y. Xu, Y. Shen, H. Huang, T. Liu, L. Jin, H. Ren, and J. Wang, "A review of hand function rehabilitation systems based on hand motion recognition devices and artificial intelligence," *Brain Sciences*, vol. 12, no. 8, p. 1079, 2022.

[7] X. Zhang, Z. Wei, X. Ren, X. Gao, X. Chen, and P. Zhou, "Complex neuromuscular changes post-stroke revealed by clustering index analysis of surface electromyogram," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 25, no. 11, pp. 2105–2112, 2017.

[8] M. S. N. Ag Lamat, M. S. H. Abd Rahman, W. A. Wan Zaidi, W. N. N. W. Yahya, C. S. Khoo, R. Hod, and H. J. Tan, "Qualitative electroencephalogram and its predictors in the diagnosis of stroke," *Frontiers in Neurology*, vol. 14, p. 1118903, 2023.

[9] Y. Guo, X. Gu, and G.-Z. Yang, *Human–robot interaction for rehabilitation robotics*. Springer, 2021.

[10] A. Frisoli, C. Loconsole, D. D. Leonardis, F. Banno, M. Barsotti, C. Chisari, and M. Bergamasco, "A new gaze-bci-driven control of an upper limb exoskeleton for rehabilitation in real-world tasks," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 42, pp. 1169–1179, 2012.

[11] D. Novak and R. Riener, "Enhancing patient freedom in rehabilitation robotics using gaze-based intention detection," in *2013 IEEE 13th international conference on rehabilitation robotics (ICORR)*, pp. 1–6, IEEE, 2013.

[12] C. Lin, C. Zhang, J. Xu, R. Liu, Y. Leng, and C. Fu, "Neural correlation of eeg and eye movement in natural grasping intention estimation," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 31, pp. 4329–4337, 2023.

[13] V. Crocher, R. Singh, J. Newn, and D. Oetomo, "Towards a gaze-informed movement intention model for robot-assisted upper-limb reha-

bilitation," in *2021 43rd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*, pp. 6155–6158, IEEE, 2021.

[14] R. Girdhar and K. Grauman, "Anticipative video transformer," in *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 13505–13515, 2021.

[15] B. Soran, A. Farhadi, and L. Shapiro, "Generating notifications for missing actions: Don't forget to turn the lights off!," in *2015 IEEE International Conference on Computer Vision (ICCV)*, pp. 4669–4677, 2015.

[16] A. Furnari and G. M. Farinella, "Rolling-unrolling lstms for action anticipation from first-person video," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 11, pp. 4021–4036, 2021.

[17] Y. Li, M. Liu, and J. M. Rehg, "In the eye of the beholder: Gaze and actions in first person video," *IEEE transactions on pattern analysis and machine intelligence*, vol. 45, no. 6, pp. 6731–6747, 2021.

[18] S. Adebayo, S. McLoone, and J. C. Dessing, "Hand-eye-object tracking for human intention inference," *IFAC-PapersOnLine*, vol. 55, no. 15, pp. 174–179, 2022.

[19] A. Ghosh, R. Dabral, V. Golyanik, C. Theobalt, and P. Slusallek, "Imos: Intent-driven full-body motion synthesis for human-object interactions," in *Computer Graphics Forum*, vol. 42, pp. 1–12, Wiley Online Library, 2023.

[20] Q. Li, J. Wang, C. C. Loy, and B. Dai, "Task-oriented human-object interactions generation with implicit neural representations," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 3035–3044, 2024.

[21] O. Taheri, V. Choutas, M. J. Black, and D. Tzionas, "Goal: Generating 4d whole-body motion for hand-object grasping," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13263–13273, 2022.

[22] Y. Wu, J. Wang, Y. Zhang, S. Zhang, O. Hilliges, F. Yu, and S. Tang, "Saga: Stochastic whole-body grasping with contact," in *European Conference on Computer Vision*, pp. 257–274, Springer, 2022.

[23] S. Christen, M. Kocabas, E. Aksan, J. Hwangbo, J. Song, and O. Hilliges, "D-grasp: Physically plausible dynamic grasp synthesis for hand-object interactions," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 20577–20586, 2022.

[24] J. Zheng, Q. Zheng, L. Fang, Y. Liu, and L. Yi, "Cams: Canonicalized manipulation spaces for category-level functional hand-object manipulation synthesis," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 585–594, 2023.

[25] J. Zhang, Y. Zhang, X. Cun, S. Huang, Y. Zhang, H. Zhao, H. Lu, and X. Shen, "T2m-gpt: Generating human motion from textual descriptions with discrete representations," *arXiv preprint arXiv:2301.06052*, 2023.

[26] B. Jiang, X. Chen, W. Liu, J. Yu, G. Yu, and T. Chen, "Motiongpt: Human motion as a foreign language," *Advances in Neural Information Processing Systems*, vol. 36, 2024.

[27] S. Ginosar, A. Bar, G. Kohavi, C. Chan, A. Owens, and J. Malik, "Learning individual styles of conversational gesture," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3497–3506, 2019.

[28] U. Bhattacharya, E. Childs, N. Rewkowski, and D. Manocha, "Speech2affectivegestures: Synthesizing co-speech gestures with generative adversarial affective expression learning," in *Proceedings of the 29th ACM International Conference on Multimedia*, pp. 2027–2036, 2021.

[29] K. Fragkiadaki, S. Levine, P. Felsen, and J. Malik, "Recurrent network models for human dynamics," in *Proceedings of the IEEE international conference on computer vision*, pp. 4346–4354, 2015.

[30] A. Jain, A. R. Zamir, S. Savarese, and A. Saxena, "Structural-rnn: Deep learning on spatio-temporal graphs," in *Proceedings of the ieee conference on computer vision and pattern recognition*, pp. 5308–5317, 2016.

[31] J. Martinez, M. J. Black, and J. Romero, "On human motion prediction using recurrent neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2891–2900, 2017.

[32] W. Mao, M. Liu, M. Salzmann, and H. Li, "Learning trajectory dependencies for human motion prediction," in *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 9489–9497, 2019.

[33] W. Mao, M. Liu, and M. Salzmann, "History repeats itself: Human motion prediction via motion attention," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIV 16*, pp. 474–489, Springer, 2020.

[34] E. Aksan, M. Kaufmann, P. Cao, and O. Hilliges, "A spatio-temporal transformer for 3d human motion prediction," in *2021 International Conference on 3D Vision (3DV)*, pp. 565–574, IEEE, 2021.

[35] Y. Cai, L. Huang, Y. Wang, T.-J. Cham, J. Cai, J. Yuan, J. Liu, X. Yang, Y. Zhu, X. Shen, *et al.*, "Learning progressive joint propagation for human motion prediction," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VII 16*, pp. 226–242, Springer, 2020.

[36] O. Taheri, N. Ghorbani, M. J. Black, and D. Tzionas, "Grab: A dataset of whole-body human grasping of objects," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part IV 16*, pp. 581–600, Springer, 2020.

[37] S. Hampali, M. Rad, M. Oberweger, and V. Lepetit, "Honnotate: A method for 3d annotation of hand and object poses," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 3196–3206, 2020.

[38] Y.-W. Chao, W. Yang, Y. Xiang, P. Molchanov, A. Handa, J. Tremblay, Y. S. Narang, K. Van Wyk, U. Iqbal, S. Birchfield, *et al.*, "Dexycb: A benchmark for capturing hand grasping of objects," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9044–9053, 2021.

[39] T. Kwon, B. Tekin, J. Stühmer, F. Bogo, and M. Pollefeys, "H2o: Two hands manipulating objects for first person interaction recognition," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 10138–10148, 2021.

[40] L. Yang, K. Li, X. Zhan, F. Wu, A. Xu, L. Liu, and C. Lu, "Oakink: A large-scale knowledge repository for understanding hand-object interaction," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 20953–20962, 2022.

[41] Y. Liu, Y. Liu, C. Jiang, K. Lyu, W. Wan, H. Shen, B. Liang, Z. Fu, H. Wang, and L. Yi, "Hoi4d: A 4d egocentric dataset for category-level human-object interaction," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 21013–21022, 2022.

[42] Z. Fan, O. Taheri, D. Tzionas, M. Kocabas, M. Kaufmann, M. J. Black, and O. Hilliges, "Arctic: A dataset for dexterous bimanual hand-object manipulation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12943–12954, 2023.

[43] H. Jiang, S. Liu, J. Wang, and X. Wang, "Hand-object contact consistency reasoning for human grasps generation," in *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 11107–11116, 2021.

[44] K. Karunratanakul, J. Yang, Y. Zhang, M. J. Black, K. Muandet, and S. Tang, "Grasping field: Learning implicit representations for human grasps," in *2020 International Conference on 3D Vision (3DV)*, pp. 333–344, IEEE, 2020.

[45] K. Karunratanakul, A. Spurr, Z. Fan, O. Hilliges, and S. Tang, "A skeleton-driven neural occupancy representation for articulated hands," in *2021 International Conference on 3D Vision (3DV)*, pp. 11–21, IEEE, 2021.

[46] M. Zhang, Y. Ye, T. Shiratori, and T. Komura, "Manipnet: neural manipulation synthesis with a hand-object spatial representation," 2021.

[47] C. Lugaresi, J. Tang, H. Nash, C. McClanahan, E. Uboweja, M. Hays, F. Zhang, C.-L. Chang, M. G. Yong, J. Lee, *et al.*, "Mediapipe: A framework for building perception pipelines," *arXiv preprint arXiv:1906.08172*, 2019.

[48] A. Van Den Oord, O. Vinyals, *et al.*, "Neural discrete representation learning," *Advances in neural information processing systems*, vol. 30, 2017.

[49] M. S. Yasar and T. Iqbal, "Vader: Vector-quantized generative adversarial network for motion prediction," in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 3827–3834, IEEE, 2023.

[50] K. Somasundaram, J. Dong, H. Tang, J. Straub, M. Yan, M. Goesele, J. J. Engel, R. De Nardi, and R. Newcombe, "Project aria: A new tool for egocentric multi-modal ai research," *arXiv preprint arXiv:2308.13561*, 2023.

[51] G. Prange, L. Smulders, J. van Wijngaarden, G. Lijbers, S. Nijenhuis, P. Veltink, J. Buurke, and A. Stienen, "User requirements for assistance of the supporting hand in bimanual daily activities via a robotic glove for severely affected stroke patients," in *2015 IEEE International Conference on Rehabilitation Robotics (ICORR)*, pp. 357–361, 2015.

[52] M. Salvato, N. Heravi, A. M. Okamura, and J. Bohg, "Predicting hand-object interaction for improved haptic feedback in mixed reality," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 3851–3857, 2022.

[53] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo, *et al.*, "Segment anything," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4015–4026, 2023.

A Review of Vision-Based Techniques for Upper Extremity Gesture Recognition and Intention Detection Using Artificial Intelligence

Yufei He 5694248

***Abstract*—Upper-extremity gesture recognition is an important research area in assistive robotics, allowing for the interpretation of human nonverbal communication. Intention detection and early motion prediction emerge as important extensions of learning human gesture features. This aspect of assistive robotics goes beyond simply recognizing and responding to gestures; it also includes anticipating human behaviors before they occur, allowing for more proactive and timely robot responses. Using advances in computer vision and artificial intelligence, robots can observe and interpret a wide range of human movements. This review systematically explores the application of artificial intelligence in upper extremity gesture recognition and intention detection. It examines various AI technologies, for example, Convolutional Neural Networks and Long Short-Term Memory networks, to understand their efficacy in different scenarios like hand or object detection, pose estimation, gesture recognition, and intention detection. The review aims to identify the most suitable AI approaches for each context, highlighting the advancements and challenges in this rapidly evolving field.**

***Index Terms*—gesture recognition, intention detection, artificial intelligence, computer vision, upper extremity**

I. Introduction

Human-robot interaction (HRI) is becoming increasingly important in the evolving dynamic between humans and machines, with applications in a wide range of fields, including healthcare. In this sector, HRI finds profound application in assistive technologies, especially in the context of upper limb rehabilitation. Robots equipped with HRI capabilities are transforming the way rehabilitation is conducted, aiding patients in regaining mobility and strength in their shoulders, arms, hands, and fingers [1]. This focus on upper limb movement and usage is vital, as these parts of the human body are central to both expressive and functional human gestures and intentions [2]. The ability of robots to accurately interpret and respond to these gestures and intentions is critical in the field of rehabilitation, allowing for a more effective and customized collaboration between patients and therapeutic machines [3].

Advancements in artificial intelligence (AI) result in great enhancements in HRI. AI incorporates robots with cognitive abilities that enable a comprehensive understanding and interpretation of human behaviors and conditions. By analyzing data from computer vision, AI facilitates precise movement recognition, analysis, and prediction, tailoring a more personalized form of interaction. Deep learning models like Convolutional Neural Networks (CNN) and Long Short Term Memory (LSTM) models are particularly effective in these applications. The integration of AI into HRI is revolutionary, opening up new possibilities in various functionalities while improving the accuracy, effectiveness, and wisdom of the device [3], [4].

Upper-extremity gesture recognition is an important research area in assistive robotics and human-computer interactions (HCI), enabling the interpretation of human non-verbal communication [5]. Advances in computer vision and AI have enabled robots to observe and understand a broad range of human movements [6]. Hand gestures are extensively studied due to their universality in conveying diverse expressions and commands and their natural use in daily human activities, making them an intuitive method for robot interaction with touchless control [7]. By analyzing hand movements, patterns, and interactions with objects, assistive robots gain a thorough understanding of current poses and can make insightful observations about human intentions. This comprehension improves HRI's ability to perform in a natural and convenient manner.

Intention detection and early motion prediction emerge as crucial extensions of learning human gesture features [8], [9]. This aspect of assistive robotics goes beyond simply recognizing and responding to gestures; it involves predicting human behaviors before they occur, allowing for more proactive and timely robot responses [10]. For example, in a scenario where a person intends to grab an object from a table cluttered with multiple items, the robot can determine which object the person aims to reach before the action is performed and give targeted help. As shown in Figure 1, classical methods of intention detection, such as surface electromyography (sEMG) and electroencephalography (EEG), focus on measuring electrical muscle activities [11]. While robust, these wearable-device-based approaches can impede movement, cause fatigue, and require frequent calibrations [3]. Most importantly, these methods could not provide information about the environment. Conversely, vision-based techniques avoid these drawbacks and enable robots to observe not only human motion but also the interaction with objects. This predictive capability is powered by advanced AI models that analyze patterns of movement, environmental context, and large historical interaction data to infer probable future actions. Such foresight in robotics opens up possibilities for offering assistance even before a request is explicitly made, thus aligning closely with human behavior and expectations.

This literature review aims to examine the advancements in upper-extremity gesture recognition and intention detection, with a specific focus on diverse AI techniques. It explores and compares various methodologies for how advances in AI

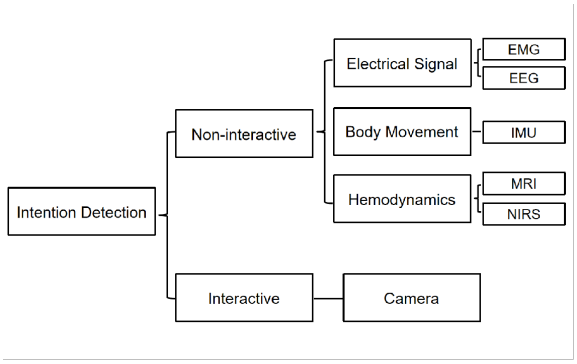


Fig. 1. Classical motion intention detection methods [12].

and computer vision have improved robots' ability to interpret and anticipate human action. The review will highlight current achievements and identify challenges, ultimately contributing to assistive robotic technologies for improved patient care.

II. Method

A. Search Strategy

For a systematic and organized methodology in conducting this literature review, the guidelines of Preferred Reporting Items for Systematic Reviews and Meta-Analyses (PRISMA) were followed. PRISMA offers a structured framework for reporting on systematic reviews and meta-analyses, outlining essential elements that ought to be incorporated in such reviews [13].

In order to comprehensively gather relevant literature in the field of upper-extremity gesture recognition and intention detection, this review primarily utilized two databases: Scopus and Web of Science. These databases were selected for their extensive coverage of scientific and technical research articles.

The search queries were carefully designed to cover the breadth of research on the topic: upper-extremity gesture recognition and intention detection using computer vision in the context of AI and assistive robotics. To ensure comprehensive coverage of this topic, the queries were structured to include several key areas: upper extremity, intention detection, artificial intelligence, computer vision, and applications in assistive robotics. In each category, relevant keywords were carefully selected and linked using the "OR" operator to capture any of the terms within a single category. The "AND" operator was then employed between these categories to ensure that the retrieved articles addressed all aspects of the research area. The following keywords and boolean operators were used to construct the search query:

Utilization of keywords should ensure the inclusion of research covering all aspects of upper body parts.

- Upper Extremity: "upper limb" OR "upper extremity" OR "hand" OR "finger" OR "arm" OR "upper body" OR "shoulder".

To cover a broad spectrum of non-verbal communication studies, a variety of keywords were applied. These keywords are essential to capturing research that detects human gestures and their implications.

- Gesture/Intention: "intent" OR "intention" OR "gesture" OR "motion" OR "pose".
- Detection: "classification" OR "detection" OR "estimation" OR "prediction" OR "recognition".

To capture studies focusing on artificial intelligence and computer vision techniques, relevant keywords were applied.

- AI: "AI" OR "artificial intelligence" OR "deep learning" OR "neural network".
- Computer Vision: "computer vision".

To refine the focus on assistive robotics within the research context, specific terms were chosen. These terms ensure that the research retrieved is directly relevant to the development and use of robots in aiding human activities, particularly in the rehabilitation field.

- Assistive Application: "rehabilitation" OR "assist*" OR "exoskeleton" OR "help*" OR "robot*".

The final inclusion search queries are shown in Table I.

To refine the search and guarantee its relevance to the specific research topic, a set of exclusion keywords was strategically defined. This method aimed to filter out search results that did not align with the study's primary focus. Articles falling outside the scope of the study, specifically those related to the lower extremity and electromyography (EMG)-related approaches, were systematically excluded. In each category, the keywords were linked via the "OR" operator. The 'NOT' operator was used between categories, effectively separating these unrelated terms from the primary search queries. The following keywords and boolean operators were used to construct the exclusion query:

- Lower extremity: "lower limbs" OR "leg" OR "lower AND body" OR "feet" OR "lower AND extremity" OR "hip" OR "gait" OR "walk".
- EMG-related: "EMG" OR "electromyography" OR "sEMG".

The search results were automatically restricted to publications from the past five years and those written in English. Furthermore, a manual search was conducted. A reason is that some papers use "vision" as a keyword in the title and abstract instead of "computer vision". The other reason is that intention detection has received less attention in the literature than gesture recognition. These studies were derived from the citations of papers identified through Web of Science and Scopus that are on the topic of intention detection. Zotero, a reference management software, was employed for the organization and categorization of the final search results. Additionally, Zotero's functionality for automatic duplicate removal was utilized to facilitate the efficient aggregation of papers.

B. Selection Criteria

The initial phase of article screening involved a thorough review of only the titles and abstracts of the papers. This preliminary evaluation was conducted to quickly assess the relevance of each study to the research topic. The selection of studies at this stage was based on the following criteria:

- Paper Type: Eligible studies must be original scientific articles rather than reviews, surveys, or datasets.

Upper Extremity		Gesture/Intention		Detection		AI		Computer Vision		Assistive Application
upper extremity		intention		detection		AI		computer vision		rehabilitation
OR		OR		OR		OR				OR
upper limb		intent		classification		artificial intelligence				assist*
OR		OR		OR		OR				OR
upper body		gesture		estimation		deep learning				exoskeleton
OR		OR		OR		OR				OR
hand	AND	motion	AND	prediction	AND	neural network	AND		AND	help*
OR		OR		OR						OR
finger		pose		recognition						robot*
OR										
arm										
OR										
shoulder										

TABLE I
SEARCH QUERIES DIVIDED BY CATEGORY

- Content Relevance: The content of the articles must be directly related to gesture recognition and intention detection using an AI approach, aligning closely with the core focus of this review.
- Body Part Specificity: The studies should specifically involve the upper limb (shoulders, arms, hands, and fingers) for gesture recognition and intention detection.
- Focus on Algorithm Design: The primary emphasis of the articles should be on the development and innovation of algorithms, as opposed to hardware implementation.
- Computer Vision as a Primary Source: The selected studies should mainly focus on computer vision information.
- Purpose of the Article: The article should aim to develop new methods in the field rather than evaluating or comparing existing methods.

In the secondary phase of screening, a more detailed examination of the paper content was undertaken. During this phase, each paper was carefully assessed for its relevance and alignment with the review's objectives. Papers were rejected based on specific criteria listed as follows:

- Scope of Study: Excludes research focused on body parts other than the upper extremity.
- AI Approach Specificity: Excludes papers that do not provide a detailed description of AI methodologies.
- Sensor Information Relevance: Papers that predominantly involved sensor information other than cameras were excluded to maintain a focus on computer vision-based approaches.
- Content Relevance: excludes studies centered on engineering application. Those who focus on how to apply gesture recognition to interactive robots rather than designing gesture recognition algorithms, for example, should be excluded. The articles that have limited relevance to gesture recognition and intention detection should also be excluded.

C. Data Categorization

The finalized articles included in this review were systematically organized based on their specific purposes and gesture types. This organization involves two primary categorization criteria:

- Nature of Detected Gestures: Articles were classified based on whether the gestures are 'static' or 'dynamic'.
- Research Purpose: The articles were further categorized according to their objectives, including 'hand or object detection', 'pose estimation', 'gesture recognition', and 'intention prediction', to distinguish between the various applications and focuses of each study.

This structured classification aids in providing a clear overview of the research landscape. In the following section, the results will be discussed in these categories.

III. RESULT

A. Study Selection

The literature review's study selection process is outlined in Figure 2. A total of 195 articles were initially identified from various sources, with 180 from the Web of Science and 15 identified manually. The ineligible papers were removed afterwards, leaving 161 articles remaining. A screening of titles and abstracts further narrowed the selection to 79 articles for full-text screening. Of these, 4 were not retrievable, and 42 were excluded based on specific criteria, resulting in 33 papers being included in the review. Table II shows an overview of all included papers and their key approaches.

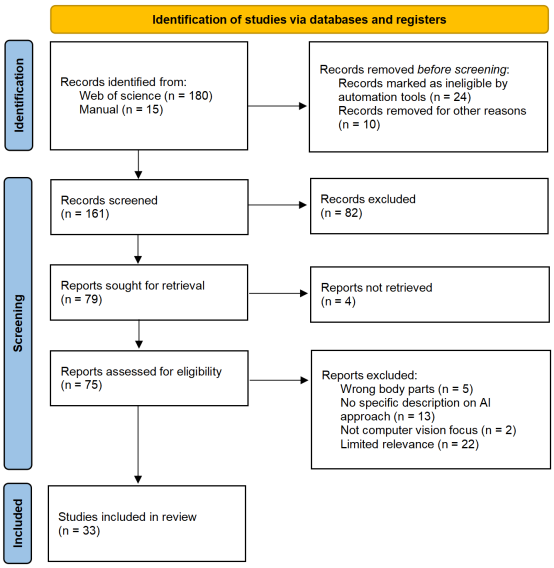


Fig. 2. Flowchart of study selection using PRISMA method.

Author	Object and body part	Purpose	Detection type (static/dynamic)	Model
Yin et al. [14]	hand	pose estimation; gesture recognition	dynamic	Resnet: feature extraction; Transformer: hand-object interaction; Stacked hourglass model: hand keypoints pinpointing; CNN: mesh regression; Temporal convolution: gesture recognition
Mazhar et al. [15]	upper body	hand detection; gesture recognition	static and dynamic	Spatio attention: hand region cropping; CNN: subtle hand movement distinguishing; LSTM: dynamic gesture recognition
Wang et al. [16]	hand	hand detection; pose estimation	dynamic	YOLO-SH: real-time hand detection; CNN: keypoint extractor; Differential adaptive kalman filter: hand position tracking
Liu et al. [17]	hand	gesture recognition; intention recognition	dynamic	HRnet: feature extraction as a heatmap GCN: multi-scale action recognition; Bidirectional CNN and bidirectional LSTM: intention recognition
Iglesias et al. [18]	hand	gesture recognition	dynamic	Small CNN (Darknet-based): fast detection;
Serj et al. [19]	hand	gesture recognition	dynamic	CNN and LSTM-based model; Time-distributed layer
Dutta et al. [20]	hand	gesture recognition	static	CenterNet architecture with attention: Encoder-Decoder network with DA-Net module; CNN for detection branches
Santavas et al. [21]	hand	pose estimation	static	DenseNet-based: Inverted residual blocks: information maintenance; Attention: focusing on relevant parts
Chanda B. et al. [22]	hand	hand detection; gesture recognition	static	U-Net: semantic segmentation; CNN: classification
Liu et al. [23]	hand	intention prediction	dynamic	CNN- and LSTM-based model
Zhang et al. [24]	hand	pose estimation		Hourglass module: feature extractor; Plane regression module (CNN-based): heatmap generation; Depth regression module (CNN-based): local offset maps generation
Gao et al. [25]	hand	hand detection; pose estimation; gesture recognition	dynamic	Faster RCNN with attention: hand detection; OpenPose-based: 3D pose estimation; 3DCNN and ConvLSTM: gesture classification
Li et al. [26]	hand	gesture recognition	static	CNN-SVM: gesture classification
Güler et al. [27]	hand	gesture recognition	static	CCNN: Capsule networks: different angular values recognition; Dynamic routing: updating weights
Zhang et al. [28]	hand	pose estimation; gesture recognition	static	Bidirectional pyramid structure: pose estimation; Asymmetric convolution structure: high-resolution heatmap generation; Deconvolution structure: keypoints and larger heatmap generation

Adebayo et al. [29]	hand, eye fixation, object	hand tracking; intention prediction	dynamic	Mediapipe: hand traker; YOLO v5: object detection; Bidirectional LSTM: intention detection
Tan et al. [30]	hand	gesture recognition	static	ViT (transformer-based): Linear projection; Transformer encoder: global dependencies capture
Korkmaz et al. [31]	hand	hand detection; gesture recognition	dynamic	SSD: hand detection; CNN: gesture classification
Wang et al. [32]	hand, object	intention prediction	dynamic	RNN-based generative model: action plot prediction; GMM: object location distribution; Faster RCNN: object-bounding box finding; FCN: hand detection; LSTM: action segmentation (generating action label for each frame)
Tang et al. [33]	hand	gesture recognition	static and dynamic	SSD: hand position locating; CNN: classification
Amit et al. [34]	hand	gesture recognition	dynamic	LSTM
Mohammed et al. [35]	hand	gesture recognition	static	CNN (SqueezeNet and MobileNet-based)
Baumgartl et al [36]	hand	gesture recognition	static	CNN (MobileNet-based)
Bodla et al. [8]	upper body included, object	intention prediction	dynamic	RNN: human-object sequence prediction; MLP: human-object relation reasoning
Koppula et al. [9]	upper body included, object	intention prediction	dynamic	Structural SVM
Hu et al. [37]	upper body included, object	intention prediction	dynamic	Fast RCNN: object region detection GNN: spatial messages being conducted; Temporal-gated Conv: temporal dynamic of actions obtaining; CNN: spatio-temporal feature processing
Khan et al. [38]	hand	hand detection	static	Mask-RCNN: hand segmentation
Bo et al. [39]	hand	hand detection	static	Deeplabv3 with Resnet-50: encoder; Dense Attention Mechanism; SqueezeNet: decoder;
Sahoo et al. [40]	hand	gesture recognition	static	DRCAM
Ting et al. [41]	hand	pose estimation	static	3DCNN
Jafari et al. [42]	hand	gesture recognition	static	HOG: feature extraction; CNN: classification
Zhu et al. [43]	hand	pose estimation	static	SE-Hourglass: feature map generation; CNN: spatial relationship predicting
Hou et al. [44]	hand	gesture recognition	dynamic	TCN; Attention branch: mask generation

TABLE II: Results of article properties and key findings

Purpose	static	dynamic
Hand detection	4	4
Pose estimation	5	4
Gesture recognition	12	9
Intention prediction	-	6

TABLE III
ARTICLE CATEGORIZATION RESULTS

The included papers in the literature review were categorized based on their research purposes. The distribution was as follows: 8 papers focused on hand (object) detection, 9 on pose estimation, 21 on gesture recognition, and 6 on intention prediction. Furthermore, within each of these purpose categories, the detected gestures were classified as either static or dynamic. Dynamic gestures are known for their variety and expressive nature, while static gestures, in contrast, are characterized by their simplicity and specific positions [5]. Detailed numbers for each category and classification are provided in Table III. These topics will be further discussed in the subsequent sections.

B. Object and Hand Detection

In the realm of hand and object detection, two predominant methods are frequently employed: bounding box detection and segmentation, as shown in Figure 3. The choice of method, particularly for segmentation, is largely influenced by the quality of the images and the gesture types [5]. The segmentation method is better suited for static hand gestures. Bounding box detection, on the other hand, shows versatility as it can be effectively applied to both dynamic and static gestures.

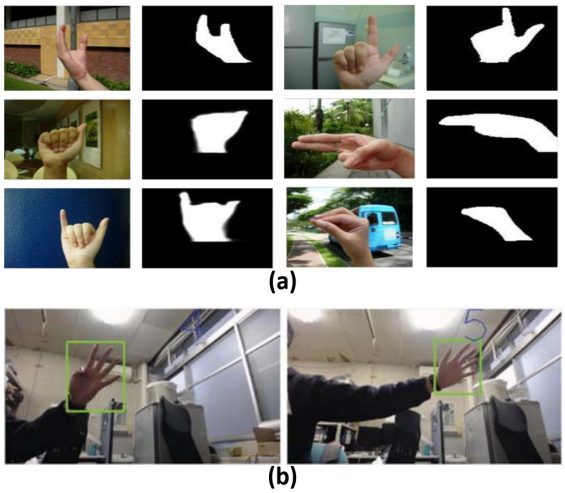


Fig. 3. Different hand detection methods. (a) Hand detection using bounding boxes [22]; (b) Hand segmentation [33].

Despite advancements, object and hand detection methods encounter significant challenges. One major difficulty is the variability of image backgrounds and skin tones, which can greatly affect the accuracy of detection algorithms [45]. Other factors, such as lighting conditions, occlusions, and the angle of hand positioning, also contribute to the complexity of accurately detecting hand gestures [45]. These challenges highlight the need for sophisticated and versatile AI models to effectively handle the diverse scenarios encountered in hand gesture recognition.

1) Static Gesture: Chanda B. et al. [22], Bo et al. [39] and Khan et al. [38] have implemented distinct neural network architectures for hand segmentation. Chanda B. et al. used a U-Net [46] architecture for semantic segmentation, focusing on binary classification to differentiate between image foreground and background. This involves processing input images into RGB and grayscale segments using an encoder-decoder structure with convolutional blocks. Bo et al. introduced DenseAttentionSeg, an attention-based network using Deeplabv3 [47] with Resnet-50 [48] for encoding and SqueezeNet [49] for decoding, enhanced by a Dense Attention Mechanism for feature adjustment. Mask-RCNN, which uses a ResNet-50 as its backbone, was used by Khan et al. for hand segmentation. It also contains a region proposal network for bounding box generation and a RoI align layer for mask prediction.

Mazhar O. et al. [15] used a spatial attention module for the cropping of the hand region. The technique begins with the use of OpenPose to extract key points and the skeleton of the hands. These coordinates, derived from the hand skeleton, are then utilized to crop the hand images accurately, with the assistance of hand depth estimators. This approach proves to be versatile, as it is effective for both static and dynamic hand gesture analysis, demonstrating its utility in various gesture recognition contexts.

2) Dynamic Gesture: In dynamic gesture recognition, bounding box detection is widely employed. You Only Look Once (YOLO) [50] models are particularly effective. Wang et al. [16] utilized YOLO v3 with ShuffleNet [51] (YOLO-SH) as the backbone for real-time hand detection, where ShuffleNet's grouped convolution aids in parameter reduction and ensures effective information fusion. Adebayo Samuel et al. [29] employed YOLO v5 for object detection through transfer learning.

Faster regional convolutional neural network (Faster-RCNN) [52] is also prevalent. Gao et al.'s [25] approach involved Faster RCNN with a bi-stream attention module, outperforming the traditional Faster RCNN with VGG-16 in hand image feature extraction. Hu et al. adopted faster RCNN with a ResNet-50 backbone for object region detection.

Additionally, Tang et al. [33] and Korkmaz et al. [31] used Single Shot MultiBox Detector (SSD) [53], based on VGG-16 [54] but with convolutional layers replacing fully connected layers, striking a balance between speed and accuracy for dynamic gesture detection.

C. Pose Estimation

Pose estimation plays a vital role in recognizing gestural behavior by extracting information on hand and body posture from images or video streams. The two main methods in this field are regression-based and heatmap-based estimation, both relying on CNNs for effective visual data analysis [55]. Regression-based techniques focus on pinpointing the coordinates of key body points directly, while heatmap-based

methods generate pixel scores to indicate the likelihood of being key points on the human body. The heatmap-based method is currently prominent in the field of pose estimation, an example of which is shown in Figure 4.

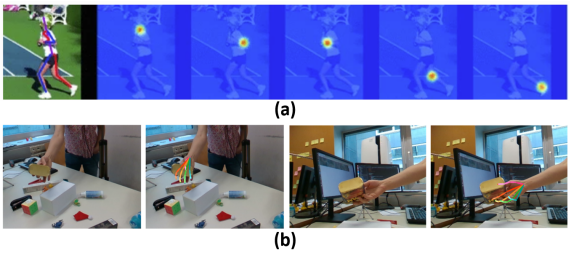


Fig. 4. Hourglass model: an example of a heatmap-based method for pose estimation. (a) Example output produced by hourglass model [56]; (b) The skeleton of hand pose estimation resulting from Hourglass model [14].

Pose estimation encounters several difficulties due to the intricate nature of human gestures and the environment. The difficulties of varying foregrounds and backgrounds that are encountered in hand detection can also cause problems in the pose estimation process. Despite these, factors such as the complexity of hand morphology and occlusions created by interactions with objects also significantly increase the challenge [57]. Additionally, compared to lower body poses, upper body poses usually possess finer structure, subtler differences, and a lower range of motion, requiring more accurate keypoint localization. For example, the complexity of hand poses, due to the numerous joints in each hand, requires the precise identification of over 20 keypoints per hand to accurately interpret these subtle variations.

1) Static Gesture: In the field of static pose estimation, various approaches related to CNNs have been employed. Ting et al. [41] employed a 3DCNN model for gesture recognition. Santavas N. et al. [21] developed a DenseNet- [58] based model, integrating unique architectural features like inverted residual blocks with concatenated skip connections for better information preservation. Additionally, it includes attention-augmented inverted bottleneck blocks, focusing the network more effectively on pertinent elements of the input. Zhang et al. [28] utilized a bidirectional pyramid structure with convolutional layers, focusing on reducing feature loss and enhancing the extraction of small target features. Their use of an asymmetric convolution structure further facilitated the generation of high-resolution heatmaps.

The hourglass model is popular among various CNN-based models. Zhang et al. [24] implemented a dual-module strategy, employing an Hourglass module [56] for feature extraction, followed by a CNN-based plane regression module for heatmap generation, and another for depth regression to create local offset maps. Zhu et al. [43] also adapted the Hourglass model in their approach but replaced the standard residual block with the SE-residual block. This block incorporates a squeeze-and-excitation (SE) module into the residual block, enhancing the network's ability to utilize feature information by focusing on the interdependencies between channels. Following this, a CNN-based network is employed to predict a dense representation of spatial relationships between pixels and hand keypoints using the features extracted by the modified Hourglass module.

2) Dynamic Gesture: CNN-based models are also commonly used in dynamic gestures. Wang et al.'s approach [16] uses CNN-based methods for feature extraction and keypoint regression, focusing on the relative positions within feature structures. Liu et al. [17] implemented a CNN-based HRNet [59] to generate heatmaps for feature extraction, merging multi-channel data for final keypoint visualization. Gao et al. [25] adopted an OpenPose-based hand pose estimation, utilizing Part Conference Map (PCM) and Part Affinity Fields (PAFs) models to derive hand pose points and subsequently construct the hand skeleton.

The hourglass model can also be employed for dynamic gestures. Yin et al. [14] developed a spatio-pose estimation method using CNN, comprising two models: a 2D keypoint localization network and a mesh regression network. The former network employs a stacked hourglass network for pinpointing hand keypoints, while the latter combines four CNN layers with three fully connected layers.

D. Gesture Recognition

Gesture recognition technology facilitates the interpretation and replication of human hand and body movements, an example of which is shown in Figure 5. Utilizing advanced technologies like CNN, LSTM, and GCN, this field has significantly evolved, focusing on accurately identifying and classifying gestures from various inputs. Researchers have developed sophisticated algorithms and network architectures, such as combining CNN with LSTM or using graph convolution networks, to enhance recognition capabilities.

Fig. 5. An example of gesture recognition result [20].

1) Static Gesture: In the domain of static gesture recognition, Convolutional Neural Networks (CNNs) are predominantly used. Chanda et al. [22] employed a fully connected CNN. Jafari et al. [42] used the Histogram of Oriented Gradients (HOG) feature extraction method and CNN for classification. More sophisticated architectures are utilized by other researchers: Mohammed et al. [35] implemented SqueezeNet- and MobileNet- [60] based CNN architectures, while Baumgartl et al. [36] used a MobileNet-based CNN architecture, achieving end-to-end gesture classification.

Researchers have integrated additional modules to enhance CNN's capabilities. Li et al. [26] integrated CNN with a Support Vector Machine (SVM) model, forming a CNN-SVM framework. In this setup, CNN effectively extracts features, which are then classified in a high-dimensional space by SVM.

The SVM component constructs an optimal classification surface in this space, enabling the effective segregation of samples into multiple classes, thus providing a more nuanced and precise gesture recognition capability.

Sahoo et al. [40] developed a CNN architecture with attention named the Densely Connected Residual Channel Attention Module (DRCAM). The Residual Channel Attention Module (RCAM) combines residual units with channel attention modules for multiscale representation. The deeper CNN architecture is realized using a cascading structure of RCAM, which focuses on learning the distinct aspects of hand gestures. This architecture benefits from dense connectivity in its cascading structure, ensuring information flow and feature reuse.

Güler et al. [27] have utilized a convolutional capsule neural network (CCNN) model, which combines the strengths of CNNs and capsule networks [61], particularly excelling in handling images from various angles. The model initially uses a CNN to generate a detailed feature map from the image sequence, employing multiple convolution kernels. These scalar outputs from the CNN are then fed into the capsule network. In the capsule network, scalar outputs are transformed into vector output capsules, enhancing the model's ability to recognize and interpret different angular values of the images.

Dutta et al. [20] implemented the CenterNet architecture [62], integrated with an Attention module, specifically the Dual Attention Network (DA-Net). This system features an encoder-decoder network with a unique two-way attention mechanism. The encoder transforms the input image into a low-resolution feature map, which is then processed by DA-Net to establish a contextual relationship between local and global features. The output from the attention network is then upsampled and directed into three convolutional detection branches, which are responsible for predicting the center point, width, height, and offsets of the center point in the images.

Beyond CNN, the transformer model can also be used for gesture recognition. Tan et al. [30] employed the Vision Transformer (ViT) [63], a model adapted from the traditional Transformer architecture. Distinct from conventional CNNs, the ViT treats an image as a sequence of tokens, similar to the way Natural Language Processing (NLP) handles text. This method enables the ViT model to effectively process larger image sizes and exhibit enhanced generalization capabilities across various tasks, all without the need for task-specific architectural modifications.

2) Dynamic Gesture: In dynamic gesture recognition research, various CNN architectures are utilized. Tellaeche Iglesias et al. [18] modified a small CNN architecture from DarkNet, while Korkmaz et al. [31] and Tang et al. [33] used simple, fully connected CNNs for classification. Yin et al. [14] innovatively designed a gesture recognition network employing temporal convolution for dynamic feature extraction in gesture sequences. To enhance this, they incorporated dilated convolution into the time convolution process, effectively maintaining time resolution and expanding the receptive field. This approach allows for more effective capture of key information across different time scales.

Hou et al. proposed Spatial-Temporal Attention Res-TCN (STA-Res-TCN), which consists of a main branch for feature processing and an attention mask branch. The Temporal Convolution Network (TCN) is built from stacked units of 1-dimensional convolution across the temporal domain, acting as the main branch. The attention branch generates the same-size masks at each layer, which softly weight the feature maps extracted by the main branch. It helps the model to adaptively focus more on the informative frames and features.

LSTMs [64], known for their proficiency in processing temporal information, widely used for temporal feature extraction. Amit et al. [34] combined LSTM with a multi-layer perception (MLP) structure for gesture recognition. LSTM, a type of recurrent neural network, could selectively remember patterns over extended periods, model sequential data, and understand complex human behavior dynamics. Meanwhile, the MLP functions as the recognizer, interpreting the data processed by the LSTM.

A combination of CNN and LSTM has also been explored. Mazhar et al. [15] employed this approach for dynamic gesture recognition, fine-tuning the Inception V3 model on a dataset of background-substituted hand gestures. This fine-tuned model, serving as the CNN block, learns to focus on pixels exclusively occupied by hands, effectively acting as a feature extractor. The features extracted by this CNN block are then processed over time using LSTM networks, enabling the detection of dynamic gestures in video sequences.

Serj et al. [19] developed a novel architecture, TD-CNN-LSTM, for hand gesture recognition. This deep architecture integrates four TD (time-distributed) blocks as distinct layers. The first three blocks each comprise a convolutional layer followed by a max-pooling layer, which reduces the width and height of the feature maps. The final TD block includes a flattening layer and two LSTM layers, designed to decrease the output sizes.

Gao et al. [25] created a network framework for gesture recognition that combines 3DCNN and ConvLSTM. ConvLSTM is adept at learning long-term spatio-temporal features, whereas the 3DCNN module captures short-term spatio-temporal features efficiently. The framework also incorporates 2DCNNs after ConvLSTM to learn high-level spatio-temporal features.

The Graph Convolution Network (GCN) is another effective approach for real-time human action recognition. Liu et al. [17] developed a GCN-based network capable of integrating both local and global information from different graph structures, enhancing the accuracy of human action recognition. This network employs graph convolution operations to produce outputs for specific time instances, effectively capturing the complex dynamics of human movements.

E. Intention Detection

In intention detection, unlike hand detection, pose estimation, or gesture recognition, gestures can only be dynamic, relying on a sequence of movements for inference. Despite its importance, the use of computer vision in upper extremity intention detection is still underexplored. Consequently, some of the approaches mentioned below are not focused on the

upper extremities only, but they contain upper body gestures. This area presents a valuable opportunity for further research and development, as computer vision could offer insights into human intentions by analyzing visual interactions between humans and environments.

Liu et al. (2019) developed the Motion Recognition and Prediction (MRP) network for predicting human intentions in tasks like computer disassembly. The network comprises two parts: a foundational CNN and an LSTM network. The CNN, based on the first 13 layers of VGG-16, extracts spatial motion features. Following this, the LSTM network identifies temporal motion patterns, enabling the system to predict actions before they occur. The LSTM's temporal memory features are used for offline training, while online predictions are made using test data.

Bodla et al. [8] developed a method that also involves similar temporal-sequence prediction and spatio-relation reasoning, but with a different arrangement. The method first learns each object's sequences separately using RNN. Following this, the Human-Object Relational Network (HORN) takes these pose and object features from the sequence prediction stage and further enhances them by learning the complex relationships between humans and objects.

Eye fixation can also be added as a visual clue for intention detection. Adebayo et al. [29] designed a method for predicting human intentions using hand movement, eye fixation, and object interaction. A bidirectional LSTM was chosen for motion inference. It learns sequential features by alternating between forward and backward passes, resulting in faster and more accurate networks. Each layer is made up of two LSTMs, one for each direction. Their neural network has seven layers, three of which are bidirectional LSTM layers and four of which are fully connected layers with a softmax activation function for classification.

Wang et al. [32] developed a series of models to predict human-object interactions. The action plot used in this method for a single time step contains the action label, the action duration, the set of active objects participating, the object states, and the end position. The approach begins with an action plot RNN generative model that forecasts action plots for the next time step. Then, Gaussian mixture models are used to learn about object arrangements and their distribution representations. Action segmentation is required to identify the start and end times of each action in order to generate action labels for each frame in our videos. A VGG-16-based CNN model is used for video segmentation. It extracts meaningful image features for each frame and then uses an LSTM model to classify action labels based on the extracted features. Finally, the actions that are likely to occur in the near future can be predicted using the predicted labels and the Action Plot RNN. Figure 6 illustrates an example of the process. The hand-object interactions can be captured and interpreted by an action plot (Fig. 6 a). It also shows the prediction of likely future actions based on these interactions (Fig. 6 b).

Hu et al. [37] developed a scene-aware spatio-temporal graph neural network (SA-STGNN) to model spatio-temporal interactions, along with a few-shot early action predictor for future action labeling. This model uses a temporal-gated CNN to capture action dynamics over time and a graph neural network (GNN) for spatial message processing, effectively combining these elements for advanced action recognition.

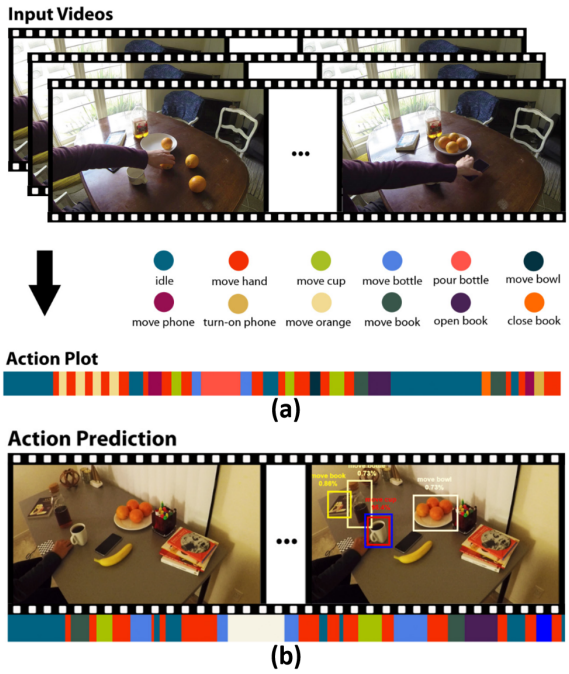


Fig. 6. intention prediction approaches developed by Wang et al. [32]. (a) To encode actions, the action plot is used to describe the participating objects and the object states. (b) The approach could predict the actions that are likely to happen in the future.

IV. Discussion

A. Artificial Intelligence

Using AI in gesture recognition offers significant advantages, such as improved accuracy, the ability to process complex or occluded gestures [14], and real-time analysis capabilities [17], [18], [31], [34]. AI models can learn from vast datasets, leading to a more comprehensive understanding and responsiveness to various human movements in different environmental contexts. Compared to traditional image processing steps, deep learning models often provide end-to-end solutions [20], [23], offering simpler and more direct processing steps for users. This streamlined approach facilitates ease of use, making these models particularly appealing for various applications. However, these systems also face challenges like requiring large amounts of training data and high computational demands. Hence, selecting and creating datasets and tailoring AI models to specific applications is crucial. Modern methods employ CNNs for spatial feature extraction and RNNs, especially LSTM networks, for analyzing temporal aspects. This evolution has expanded applications into more efficient and accurate HCI.

B. Convolutional Neural Network

CNNs are widely used in various applications due to their exceptional ability to capture features in images. They are

particularly effective in target detection, semantic segmentation, and feature extraction. The design and structure of CNNs, however, need to be carefully tailored according to the specific application at hand. This customization ensures optimal performance and accuracy in extracting and processing the relevant features from images for different tasks.

1) Semantic Segmentation: In semantic segmentation, classical models typically consist of an encoder and a decoder structure. Unlike fully convolutional networks (FCNs) with a single decoder layer, more advanced methods place greater emphasis on decoder innovation. The decoder, crucial for segmentation, uses up-sampling and deconvolution layers to extract more effective and abstract features. It also complements information and maintains the resolution of images, playing a key role in the overall performance of the segmentation process.

2) Object detection: In object detection, three architectures are usually used, including YOLO, Faster R-CNN and SSD. YOLO uses anchor boxes for detection, which are predetermined bounding boxes of specified height and width placed at each input image location [20]. This unique approach to processing the entire image in a single stage offers a substantial speed advantage. Due to the lack of regional sampling, YOLO has relatively good performance in extracting global information but shows limitations for detecting local and small objects. SSD is a kind of model that balances both accuracy and speed. Similar to the architecture and concept of YOLO, it is also a one-stage approach [20]. It employs several convolutional layers to produce progressively smaller feature maps, enabling it to detect objects of various sizes. Hence, similar to Faster R-CNN, it can focus on learned region proposals [31], [33].

Faster R-CNN is renowned for its high accuracy in object detection, attributed to its region proposal networks (RPN) and the refinement of these proposals in subsequent stages [18]. It excels in complex or small object detection scenarios [25], compared to YOLO and SSD. The primary drawback of faster R-CNN is its slower inference speed, resulting from the two-step process involving region proposal and object detection [20]. This, coupled with its higher computational resource requirement, renders it less suitable for real-time applications when compared to YOLO and SSD.

In the context of gesture recognition, YOLO could provide sufficient accuracy, particularly due to its efficiency in extracting larger features. This capability is especially relevant when the primary goal is to detect the rough position of human hands. In many hand gesture datasets, the hands are typically positioned centrally and occupy a large portion of the image, making YOLO an ideal choice for such scenarios where detailed precision is less critical. However, when it comes to intention detection involving object interactions, the requirements can be more demanding. For these applications, a method that focuses on the detailed status of smaller objects is necessary. For instance, when the task involves discerning detailed aspects of small objects, such as determining whether a cup is filled, a more precise and intricate analysis is necessary. In these scenarios, methods capable of detecting fine details in small objects are essential. Thus, employing architectures like faster R-CNN or SSD becomes advantageous.

3) Pose Estimation: Pose estimation focuses on identifying the position and orientation of body parts, requiring neural networks to either directly regress keypoints or generate heatmaps for specific body parts. The main methods discussed in this article, including the Hourglass model, HRNet and OpenPose are all heatmap-based methods. CNN emerges as the predominant tool in the reviewed studies for pose estimation, owing to its efficacy in spatial feature analysis. The addition of attention mechanisms in some approaches underscores the focus on relevant parts of the input, enhancing accuracy. Notably, the methods applied do not significantly differ between static and dynamic gestures, as both primarily concentrate on spatial features instead of temporal relations.

The Hourglass Model and OpenPose represent state-of-the-art deep learning models in pose estimation. OpenPose excels in real-time, multi-person 2D pose estimation, particularly for large-scale poses like full-body or upper-body. However, it shows limitations in the accuracy of hand gestures and requires adaptations for 3D pose estimation [25]. The stacked Hourglass model, using a combination of top-down and bottom-up processing within each unit and relying on residual modules, captures spatial features at multiple scales [24], [43]. It can effectively identify both local features, such as those of the hands, and global features of the entire body, making it more suitable for local human pose estimation compared to OpenPose.

4) Gesture Recognition: Gesture recognition, which involves interpreting human body language, requires neural networks to accurately classify a set of features. In this field, popular lightweight models like Inception, SqueezeNet, and MobileNet aim to reduce the number of parameters without compromising efficiency. Inception employs multiple filter sizes in the same layer, allowing for diverse feature observation. It also employs strategies to decrease computational complexity, making it suitable for dynamic gesture recognition. SqueezeNet's fire module combines different filter sizes to reduce parameters significantly [35]. MobileNet uses a different methodology to reduce computational complexity by using depthwise separable convolutions [35], [36]. It substantially reduces parameters while increasing operational speed, making it efficient for mobile and embedded applications. Although SqueezeNet and MobileNet have been primarily used for static gesture recognition in the examined article, their capabilities extend to classifying dynamic gestures as well. Both models have comparable accuracy and speed, with SqueezeNet being faster and mobilenet being more accurate [35].

In conclusion, the versatility of the CNN model makes it suitable for a wide range of applications, with different architectural designs catering to specific use cases. For instance, pose estimation often involves a combination of downsampling and upsampling to capture spatial hierarchies effectively. In contrast, gesture recognition typically employs small kernels in deep convolutional networks for detailed feature extraction. It is important to carefully design the model and select the appropriate backbone. Even within the same application, varying focus points necessitate architectural adjustments. Take pose estimation as an example: if the goal is to produce a

high-resolution heatmap, a standard downsampling process might not be ideal, as it could reduce the detail level required for accurate representation. This underscores the need for a tailored approach in CNN architecture to meet the specific demands of each application effectively.

C. Recurrent Neural Network

RNNs are extremely effective in dynamic gesture recognition due to their ability to capture time-series features. This makes them particularly well-suited to recognizing dynamic gestures, where the temporal sequence of movements is critical. This aspect becomes even more important in the context of intention detection, as dynamic gestures are a fundamental component of all intention detection scenarios. Nonetheless, RNNs face challenges, most notably the vanishing gradient problem. This issue arises during backpropagation in deep RNNs, where gradients can shrink exponentially, making it difficult to learn and retain information from earlier inputs. LSTM, a special kind of RNN, addresses this by introducing gates that regulate the flow of information, allowing them to preserve long-term dependencies. Hence, it is very popular and extensively used in gesture recognition and intention detection.

The integration of RNNs with other models, such as CNNs, has led to significant advancements in gesture recognition. This hybrid approach combines CNNs' proficiency in extracting spatial features from images with RNNs' ability to analyze temporal sequences. In many advanced models, LSTM layers are strategically placed after CNNs. CNNs can first process image frames to extract spatial features, which are then fed into an RNN to understand the progression of gestures over time and provide a comprehensive understanding of motion and intention. This combination harnesses the strengths of both architectures, resulting in more robust and accurate gesture recognition systems.

A novel application of RNN is in the creation of action plots, as detailed in Wang et al. [32]. An action plot is essentially a sequence of actions performed by a human hand that leads to a state change in the scene. Utilizing RNN, transition probabilities can be learned and future actions can be predicted. As a result, RNN can be used to not only recognize given sequences but also predict the unknown future from them, giving the model a strong capability in the field of intention detection.

D. Application Areas

Vision-based techniques for upper extremity gesture recognition and intention detection are transforming many fields, with healthcare being one of the most prominent. In the healthcare domain, these technologies play a crucial role in rehabilitation and physical therapy. These technologies enable the creation of systems that assist patients in their recovery by predicting their next movements and providing targeted assistance. For instance, a rehabilitation system or exoskeleton could apply specific forces to aid patients in achieving certain movements based on accurate predictions. Additionally, these systems offer valuable feedback to both patients and therapists, ensuring the effectiveness and correctness of recovery exercises. This not only enhances the quality of care but also accelerates the recovery process.

Beyond healthcare, vision-based gesture recognition finds applications in virtual reality (VR) and augmented reality (AR), enhancing user experiences in gaming and training simulations. In the automotive industry, gesture recognition can be used in vehicle control systems, allowing drivers to interact with infotainment systems or other in-car technology through simple hand movements, reducing distractions. In smart homes, gesture recognition offers a convenient and intuitive way of interacting with various home appliances. These diverse applications underscore the versatility of vision-based gesture recognition and intention detection, opening up new possibilities for interaction and accessibility across multiple areas.

V. Conclusion

In conclusion, this review has highlighted the significant role of AI in advancing upper-extremity gesture recognition and intention detection within the field of assistive robotics. It shows the effectiveness of technologies such as CNNs, LSTMs, and GCNs in a variety of scenarios, ranging from hand and object detection to pose estimation, gesture recognition, and intention detection. The findings show that CNNs excel at extracting spatial features, which are critical for accurate gesture recognition, whereas LSTMs excel at processing temporal features, making them highly effective for dynamic gestures. The findings also highlight the importance of selecting appropriate AI approaches tailored to specific applications. The review also identifies a gap in research regarding intention detection using computer vision, indicating the need for additional research. This exploration paves the way for further innovations in AI, enhancing human-robot interaction, and expanding the capabilities of assistive robotics.

References

[1] A. Basteris, S. M. Nijenhuis, A. H. Stienen, J. H. Buurke, G. B. Prange, and F. Amirabdollahian, "Training modalities in robot-mediated upper limb rehabilitation in stroke: a framework for classification based on a systematic review," *Journal of neuroengineering and rehabilitation*, vol. 11, pp. 1–15, 2014.

[2] D. Sarma and M. K. Bhuyan, "Methods, databases and recent advancement of vision-based hand gesture recognition for hci systems: A review," *SN Computer Science*, vol. 2, no. 6, p. 436, 2021.

[3] Y. Gu, Y. Xu, Y. Shen, H. Huang, T. Liu, L. Jin, H. Ren, and J. Wang, "A review of hand function rehabilitation systems based on hand motion recognition devices and artificial intelligence," *Brain Sciences*, vol. 12, no. 8, p. 1079, 2022.

[4] M. A. Vélez-Guerrero, M. Callejas-Cuervo, and S. Mazzoleni, "Artificial intelligence-based wearable robotic exoskeletons for upper limb rehabilitation: A review," *Sensors*, vol. 21, no. 6, p. 2146, 2021.

[5] N. Fadel and E. I. A. Kareem, "Computer vision techniques for hand gesture recognition: Survey," in *International Conference on New Trends in Information and Communications Technology Applications*, pp. 50–76, Springer, 2022.

[6] Y. Li and J. Xu, "Gesture recognition related technology and development challenges," in *2022 IEEE 2nd International Conference on Electronic Technology, Communication and Information (ICETCI)*, pp. 568–571, IEEE, 2022.

[7] A. D. Harale and K. J. Karande, "Literature review on dynamic hand gesture recognition," in *AIP Conference Proceedings*, vol. 2494, AIP Publishing, 2022.

[8] N. Bodla, G. Shrivastava, R. Chellappa, and A. Shrivastava, "Hierarchical video prediction using relational layouts for human-object interactions," p. 12141–12150, 2021.

[9] H. S. Koppula, R. Gupta, and A. Saxena, "Learning human activities and object affordances from rgb-d videos," *The International Journal of Robotics Research*, vol. 32, p. 951–970, July 2013.

[10] D. Wang, X. Gu, and H. Yu, "Sensors and algorithms for locomotion intention detection of lower limb exoskeletons," *Medical Engineering & Physics*, p. 103960, 2023.

[11] T. Du Plessis, K. Djouani, and C. Oosthuizen, "A review of active hand exoskeletons for rehabilitation and assistance," *Robotics*, vol. 10, no. 1, p. 40, 2021.

[12] J. Lobo-Prat, P. N. Kooren, A. H. Stienen, J. L. Herder, B. F. Koopman, and P. H. Veltink, "Non-invasive control interfaces for intention detection in active movement-assistive devices," *Journal of neuroengineering and rehabilitation*, vol. 11, no. 1, pp. 1–22, 2014.

[13] M. J. Page, J. E. McKenzie, P. M. Bossuyt, I. Boutron, T. C. Hoffmann, C. D. Mulrow, L. Shamseer, J. M. Tetzlaff, E. A. Akl, S. E. Brennan, *et al.*, "The prisma 2020 statement: an updated guideline for reporting systematic reviews," *International journal of surgery*, vol. 88, p. 105906, 2021.

[14] Q. Yin, Q. Zhao, H. Jia, Y. Gao, L. Feng, C. Li, Y. Cheng, and B. Lin, "3d hand pose estimation and gesture recognition based on hand-object interaction information," 2023.

[15] O. Mazhar, S. Ramdani, and A. Cherubini, "A deep learning framework for recognizing both static and dynamic gestures," *Sensors*, vol. 21, no. 6, 2021.

[16] S. Wang, C. Guo, R. Yang, Q. Zhang, and H. Ren, "A lightweight vision-based measurement for hand gesture information acquisition," *IEEE Sensors Journal*, vol. 23, p. 4964–4973, Mar. 2023.

[17] C. Liu, Z. Zhang, D. Tang, Q. Nie, L. Zhang, and J. Song, "A mixed perception-based human-robot collaborative maintenance approach driven by augmented reality and online deep reinforcement learning," vol. 83, p. 102568, Oct. 2023.

[18] A. Tellaeche Iglesias, I. Pastor-López, B. Sanz Urquijo, and P. García-Bringas, "A real time vision system based on deep learning for gesture based human machine interaction," vol. 12344 LNAI, p. 561–572, 2020.

[19] M. Serj, M. Asgari, B. Lavi, D. Valls, and M. Garcia, "A time-distributed convolutional long short-term memory for hand gesture recognition," p. 478–482, 2021.

[20] H. Dutta, K. Manivas, M. Bhuyan, and M. Bhuyan, "An end-to-end anchorless approach to recognize hand gestures using centernet," p. 1–6, 2023.

[21] N. Santavas, I. Kansizoglou, L. Bampis, E. Karakasis, and A. Gasteratos, "Attention! a lightweight 2d hand pose estimation approach," *IEEE SENSORS JOURNAL*, vol. 21, p. 11488–11496, May 2021.

[22] B. Chanda and H. Nyeem, "Automatic hand gesture recognition with semantic segmentation and deep learning," 2022.

[23] Z. Liu, Q. Liu, W. Xu, Z. Liu, Z. Zhou, and J. Chen, "Deep learning-based human motion prediction considering context awareness for human-robot collaboration in manufacturing," *Procedia CIRP*, vol. 83, p. 272–278, 2019.

[24] X. Zhang and F. Zhang, "Differentiable spatial regression: A novel method for 3d hand pose estimation," *IEEE TRANSACTIONS ON MULTIMEDIA*, vol. 24, p. 166–176, 2022.

[25] Q. Gao, Y. Chen, Z. Ju, and Y. Liang, "Dynamic hand gesture recognition based on 3d hand pose estimation for human-robot interaction," *IEEE SENSORS JOURNAL*, vol. 22, p. 17421–17430, Sept. 2022.

[26] "Gesture recognition with complex background based on improved convolutional neural network," p. 1345–1349, 2021.

[27] O. Güler and Yücedağ, "Hand gesture recognition from 2d images by using convolutional capsule neural networks," vol. 47, p. 1211–1225, 2022.

[28] M. Zhang, Z. Zhou, X. Tao, N. Zhang, and M. Deng, "Hand pose estimation based on fish skeleton cnn: application in gesture recognition," vol. 44, p. 8029–8042, 2023.

[29] S. Adebayo, S. McLoone, and J. C. Dessing, "Hand-eye-object tracking for human intention inference," *IFAC-PapersOnLine*, vol. 55, no. 15, p. 174–179, 2022.

[30] C. Tan, K. Lim, R. Chang, C. Lee, and A. Alqahtani, "Hgr-vit: Hand gesture recognition with vision transformer," *Sensors*, vol. 23, no. 12, 2023.

[31] S. Korkmaz, "Integrated deep learning structures for hand gesture recognition," vol. 896, p. 129–136, 2019.

[32] H. Wang, S. Pirk, E. Yumer, V. Kim, O. Sener, S. Sridhar, and L. Guibas, "Learning a generative model for multi-step human-object interactions from videos," vol. 38, p. 367–378, May 2019.

[33] J. Tang, X. Yao, X. Kang, S. Nishide, and F. Ren, "Position-free hand gesture recognition using single shot multibox detector based neural network," p. 2251–2256, 2019.

[34] M. Amit, A. Fajardo, and R. Medina, "Recognition of real-time hand gestures using mediapipe holistic model and lstm with mlp architecture," p. 292–295, 2022.

[35] A. Mohammed, J. Lv, and M. Islam, "Small deep learning models for hand gesture recognition," p. 1429–1435, 2019.

[36] H. Baumgartl, D. Sauter, C. Schenk, C. Atik, and R. Buettner, "Vision-based hand gesture recognition for human-computer interaction using mobilenetv2," p. 1667–1674, 2021.

[37] Y. Hu, J. Gao, and C. Xu, "Learning scene-aware spatio-temporal gnns for few-shot early action prediction," *IEEE Transactions on Multimedia*, vol. 25, p. 2061–2073, 2023.

[38] F. S. Khan, M. N. H. Mohd, D. M. Soomro, S. Bagchi, and M. D. Khan, "3d hand gestures segmentation and optimized classification using deep learning," *IEEE Access*, vol. 9, pp. 131614–131624, 2021.

[39] Z. Bo, H. Zhang, J. Yong, H. Gao, and F. Xu, "Denseattentionseg: Segment hands from interacted objects using depth input," *APPLIED SOFT COMPUTING*, vol. 92, July 2020.

[40] J. P. Sahoo, S. P. Sahoo, S. Ari, and S. K. Patra, "Hand gesture recognition using densely connected deep residual network and channel attention module for mobile robot control," *IEEE Transactions on Instrumentation and Measurement*, vol. 72, p. 1–11, 2023.

[41] P. Ting, E. Chou, Y. Tang, and L. Fu, "Hand pose estimation based on 3d residual network with data padding and skeleton steadying," vol. 11365, p. 293–307, 2019.

[42] F. Jafari and A. Basu, "Saliency-driven hand gesture recognition incorporating histogram of oriented gradients (hog) and deep learning," *SENSORS*, vol. 23, Sept. 2023.

[43] C. Zhu, B. Hu, J. Chen, X. Ai, and S. Agrawal, "Sarn: Shifted attention regression network for 3d hand pose estimation," *BIOENGINEERING-BASEL*, vol. 10, Feb. 2023.

[44] J. Hou, G. Wang, X. Chen, J. Xue, R. Zhu, and H. Yang, "Spatial-temporal attention res-tcn for skeleton-based dynamic hand gesture recognition," vol. 11134, p. 273–286, 2019.

[45] F. Noroozi, C. A. Corneanu, D. Kamińska, T. Sapiński, S. Escalera, and G. Anbarjafari, "Survey on emotional body gesture recognition," *IEEE transactions on affective computing*, vol. 12, no. 2, pp. 505–523, 2018.

[46] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*, pp. 234–241, Springer, 2015.

[47] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs," *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 4, pp. 834–848, 2017.

[48] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

[49] F. N. Iandola, S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally, and K. Keutzer, "Squeezenet: Alexnet-level accuracy with 50x fewer parameters and¡ 0.5 mb model size," *arXiv preprint arXiv:1602.07360*, 2016.

[50] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 779–788, 2016.

[51] X. Zhang, X. Zhou, M. Lin, and J. Sun, "Shufflenet: An extremely efficient convolutional neural network for mobile devices," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 6848–6856, 2018.

[52] R. Girshick, "Fast r-cnn," in *Proceedings of the IEEE international conference on computer vision*, pp. 1440–1448, 2015.

[53] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "Ssd: Single shot multibox detector," in *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14*, pp. 21–37, Springer, 2016.

[54] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[55] C. Zheng, W. Wu, C. Chen, T. Yang, S. Zhu, J. Shen, N. Kehtarnavaz, and M. Shah, "Deep learning-based human pose estimation: A survey," *ACM Computing Surveys*, vol. 56, no. 1, pp. 1–37, 2023.

[56] A. Newell, K. Yang, and J. Deng, "Stacked hourglass networks for human pose estimation," in *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part VIII 14*, pp. 483–499, Springer, 2016.

[57] P. Rawat, L. Kane, M. Goswami, A. Jindal, and S. Sehgal, "A review on vision-based hand gesture recognition targeting rgb-depth sensors," *International Journal of Information Technology & Decision Making*, vol. 22, no. 01, pp. 115–156, 2023.
[58] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4700–4708, 2017.
[59] K. Sun, B. Xiao, D. Liu, and J. Wang, "Deep high-resolution representation learning for human pose estimation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 5693–5703, 2019.
[60] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," *arXiv preprint arXiv:1704.04861*, 2017.
[61] S. Sabour, N. Frosst, and G. E. Hinton, "Dynamic routing between capsules," *Advances in neural information processing systems*, vol. 30, 2017.
[62] K. Duan, S. Bai, L. Xie, H. Qi, Q. Huang, and Q. Tian, "Centernet: Keypoint triplets for object detection," in *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 6569–6578, 2019.
[63] 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.
[64] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

B

Acknowledgement

First and foremost, I extend my deepest gratitude to my thesis supervisors, Prof. Arno Stienen and Prof. Xucong Zhang. I feel incredibly fortunate to have been guided by such knowledgeable and supportive mentors. Your expert advice not only steered me throughout my thesis project but also led me into the world of science. Beyond this, your patience and support were invaluable to me. As someone who struggles with self-confidence, your encouragement helped me recognize my own potential. I am profoundly grateful for your belief in me.

I must express my heartfelt thanks to my grandmother, who has been an extraordinary influence in my life. I have a lot of thankfulness for you, and I also carry a deep sense of regret. Your love and care have shaped who I am today, and for that, I am eternally grateful. I apologize for the times I was away when I should have been by your side, and I also apologize for the words I have never said to your face. 谢谢您。我爱您。

My gratitude also goes to my parents, my friends, and my boyfriend. Thank you all for your love, support, and understanding.