Toward Human Deictic Gesture Target Estimation

Xu Cao[1], Pranav Virupaksha[2], Sangmin Lee[3], Bolin Lai[2], Wenqi Jia[1], Jintai Chen[4], James M. Rehg[1]
[1]University of Illinois Urbana-Champaign
[2]Georgia Institute of Technology
[3]Korea University
[4]The Hong Kong University of Science and Technology (Guangzhou)
{xucao2,jrehg}@illinois.edu

Abstract

Humans have a remarkable ability to use co-speech deictic gestures, such as pointing and showing, to enrich verbal communication and support social interaction. These gestures are so fundamental that infants begin to use them even before they acquire spoken language, which highlights their central role in human communication. Understanding the intended targets of another individual's deictic gestures enables inference of their intentions, comprehension of their current actions, and prediction of upcoming behaviors. Despite its significance, gesture target estimation remains an underexplored task within the computer vision community. In this paper, we introduce **GestureTarget**, a novel task designed specifically for comprehensive evaluation of social deictic gesture semantic target estimation. To address this task, we propose **TransGesture**, a set of Transformer-based gesture target prediction models. Given an input image and the spatial location of a person, our models predict the intended target of their gesture within the scene. Critically, our gaze-aware joint cross attention fusion model demonstrates how incorporating gaze-following cues significantly improves gesture target mask prediction IoU by 6% and gesture existence prediction accuracy by 10%. Our results underscore the complexity and importance of integrating gaze cues into deictic gesture intention understanding, advocating for increased research attention to this emerging area. All data, code will be made publicly available upon acceptance. Code of TransGesture is available at GitHub.com/IrohXu/TransGesture.

1 Introduction

Human social communication is profoundly multidimensional and multimodal. Beyond speech, gestures and body language play an important role in conveying intent, directing attention, and facilitating social interaction [1, 2, 3]. For example, a simple pointing gesture can silently direct someone's gaze toward a distant object, implying the intention of "look at that"; a simple giving gesture can pass a business card to someone, implying the intention of "remember me". Such deictic gestures (e.g., pointing with a finger or transferring an object with hands) enable people to share focus and coordinate actions without speaking. The ability to interpret these gestures is fundamental to human social capability – we can easily infer what another person is referring to when they present deictic gestures [4, 5]. Endowing machines with a similar understanding of human gestures would significantly advance human–AI communication and embodied intelligence, enabling applications from assistive robotics to mental monitoring tools where AI agents must respond to non-verbal gesturing cues [6, 7, 8, 9, 10, 11].

However, despite decades of progress in object detection and action recognition, the specific task of estimating gesture targets - inferring what a person is gesturing to - has remained largely unexplored.

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

Figure 1: Gesture Target Estimation: the first row is the input image and the location of subject person; the second row is the groundtruth mask of the target; the third row visualizes the gesture interaction between subject person and target. The fourth row is the text description of subject person.

Prior work has primarily focused on gesture classification [12, 13, 14] or synthesis [15, 16, 17, 18], rather than on interpreting their referential intent. Additionally, gesture target estimation is particularly challenging. Unlike standard object detection or segmentation, it requires reasoning across multiple pieces of information, including body pose, gaze direction, spatial context, and occlusions. A pointing gesture might refer to a nearby object or a distant landmark, and the intended referent may be partially hidden. Compounding the challenge, there exist no public datasets for this task, limiting progress in the field.

In this paper, we address the previously mentioned gaps by introducing **GestureTarget**, a novel benchmark dataset for deictic gesture target estimation, and a Transformer-based gaze-aware gesture model, **TransGesture**, that learn to predict the targets of human gestures by leveraging multimodal cues. GestureTarget is, to our knowledge, the first large-scale dataset focused on images of people engaged in deictic gestures (such as pointing, reaching, or showing an object) with annotations of the intended target of each gesture. Each data instance in GestureTarget consists of an image containing a specified person-of-interest who is performing a gesture, along with bounding box and segmentation mask annotation indicating the target of that gesture. By collecting and curating this benchmark, we aim to facilitate the quantitative evaluation of models on the gesture target estimation task and drive progress on this underexplored problem.

Building on this dataset, we propose TransGesture, a group of Transformer-based baseline models designed for gesture target prediction. Inspired by the psycholinguistic and cognitive theory in the relationship between gaze and deictic gesture [19, 20, 21, 22], we incorporate gaze target prediction to guide the estimation of gesture target. By fusing gaze direction via multi-layer joint cross attention, the model can disambiguate the intended gesture target even in complex scenes. We term TransGesture as "gaze-aware gesture models" because they have a pre-trained gaze target estimation branch (on the GazeFollow [23] dataset). They are then finetuned with a large amount of gestural data from the GestureTarget dataset. TransGesture models are designed to be adaptable, serving as a basis for various deictic gesture understanding tasks. Experimental results show that our approach of incorporating gaze cues significantly improves performance in identifying deictic gesture targets, validating the intuition that "where they look" helps convey "what they point at." Our contributions can be summarized as:

(1) We propose TransGesture, a group of Transformer models that integrates human gesture and gaze social cues through large-scale frozen visual encoder and applys joint cross attention fusion mechanisms to accurately infer gesture targets in complex visual scenes.

(2) We demonstrate that incorporating gaze target estimation as an auxiliary modality can significantly improve understanding of deictic gesture targets, highlighting the importance of gaze as a critical cue in understanding nonverbal human communication.

(3) We introduce GestureTarget, a new task and dataset designed for deictic gesture target estimation, containing over 20K annotated instances of pointing, reaching, showing, and giving gestures with corresponding target mask annotations.

2 Related Works

Hand and Mind Gestures are a core channel of human communication, continuously shaping and enriching spoken language. Developmental studies show that infants' early spontaneous gesticulation can reliably predict the timing of later linguistic milestones [24, 25, 26, 27]. McNeill argues that gestures externalize imagery that words alone cannot always capture, so speech and gesture must be analyzed together to expose the workings of thought [28, 29]. To distinguish different gestures, Kendon situates gesture on a functional continuum that runs from spontaneous co-speech gesticulation, through language-like gestures, to pantomimic emblems, and finally fully conventionalised sign languages [30, 31]. Moving rightward along Kendon's scale: (1) Dependence on accompanying speech diminishes. (2) Linguistic structure becomes more explicit. (3) Idiosyncratic movements give way to socially regulated symbols [32]. Most computer vision research has focused on the latter two categories — emblematic gesture [33, 34, 13, 35] and sign language recognition [36, 37, 38]. In contrast, spontaneous co-speech gestures, especially the deictic gestures, remain largely unexplored, leaving a crucial gap in the computational understanding of human communication.

Gesture & Gaze Modeling Gesture recognition and understanding is a well-known and widely explored task in the computer vision community, with many introduced datasets and models. Notable datasets for gesture recognition include LD-ConGR, featuring 542 RGB-D videos representing 10 gestures [12], HaGRID, which focuses on high-resolution image based gesture recognition across 18 hand gesture classes [13], and EgoGesture, with over 2 million frames from 50 subjects, meant for egocentric interactions with wearable devices [39]. Recently in social gesture understanding, SocialGesture introduces a large-scale multi-person gesture dataset with more than 42,000 instances of social gestures [40]. Almost all gesture models focus on classification, including CNN-based models [41, 42] and Transformer-based models [43, 44, 45].

Similar to gesture analysis, interpreting gaze is important for human behavior understanding. GazeFollow [23], VideoAttentionTarget [46] and ChildPlay [47] introduce the gaze-following task, where the task requires models to predict the location in a provided scene that a target person is looking at. Most approaches, including the original GazeFollow model, have taken a fusion-based approach, processing elements such as depth, pose, head position and head keypoints separately and then combining them to yield a gaze prediction [48, 49, 50, 51, 52]. More recently, Gaze-LLE [53] moved away from multi-branch designs by utilizing a frozen, pretrained visual encoder and a decoder conditioned with head position information to make individual-specific gaze predictions. However, there is limited exploration of the relationship between gesture and gaze in model development [54, 55].

3 Gesture Target Estimation

We introduce **GestureTarget**, a new benchmark under CC BY-NC 3.0 License for simultaneous localization and recognition of deictic gesture targets. To overcome the sparsity of deictic gestures in natural images, we repurpose four existing multi-person interaction datasets—VCR [56], Werewolf Among Us [57], SocialGesture [40], and Social-IQ [58] — each of which already provide human and object bounding box annotations. The annotation process is described as follows: (i) Frame selection. We manually identify representative frames containing clear deictic gestures from videos. (ii) Pseudo-annotation. We run YOLOv11's segmentation and a human keypoint detector [59] to produce initial masks and bounding boxes. (iii) Human verification. Our author-annotators review each image, confirm or correct the subject person's body and head boxes, and select the correct semantic mask of the gesture target. Images with incorrect human boxes or keypoints are discarded. (iv) Dataset finalization. After removing roughly 30% of wrong-labeled frames during verification, GestureTarget comprises 20K instances. Each instance includes: Body and head bounding boxes of the gesturing subject; Bounding box and semantic segmentation mask for the gesture target; A paired textual description linking the subject person to their target. Our annotation pipeline leverages existing resources to efficiently create a rich dataset for deictic gesture target estimation.

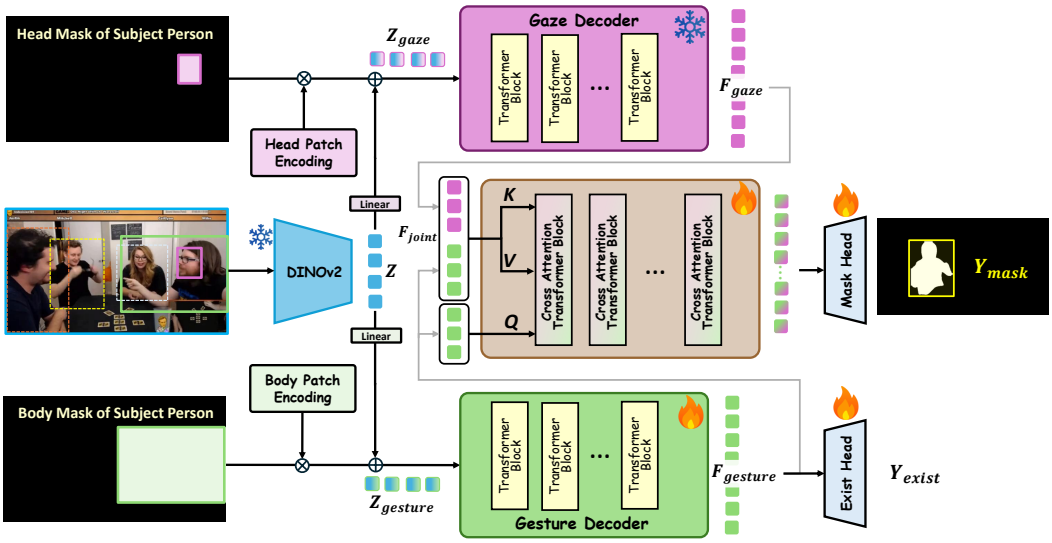

Figure 2: We introduce **TransGesture**, a new framework for gesture target estimation under the guidance of human gaze. We first use a frozen DINOv2 to extract scene tokens and then project them to Z_{gaze} and Z_{gesture} with two linear layers. After adding head or body patch encodings, Z_{gaze} and Z_{gesture} are updated by two separate transformer-based decoders. The outputs are fused by a multi-block joint cross attention transformer modules and finally upsample and decode by a convolution mask head.

Figure 1 illustrates an example annotation from GestureTarget. Given an input RGB image $X \in \mathbb{R}^{3\times H\times W}$, our annotations include: the bounding box of the subject's body ($X_{\text{subject}} \in \mathbb{R}^4$); eye keypoints (left eye $X_{\text{leye}} \in \mathbb{R}^2$, right eye $X_{\text{reye}} \in \mathbb{R}^2$); hand keypoints (left hand $X_{\text{lhand}} \in \mathbb{R}^2$, right hand $X_{\text{rhand}} \in \mathbb{R}^2$); and a textual description (X_{text}) detailing the subject's appearance, location, and action. For gesture target annotations, we provide a bounding box ($Y_{\text{target}} \in \mathbb{R}^4$), a semantic segmentation mask ($Y_{\text{mask}} \in \mathbb{R}^{H\times W}$), a text description ($Y_{\text{text}}$) including the target's attributes and location, and a categorical existence label (Y_{exist}) indicating whether the gesture target is present, absent, or out of frame. The goal of the task is predicting Y_{exist} and Y_{mask}.

How Do SOTA Vision-Language Models Perform on GestureTarget? Additionally, to demonstrate the utility of GestureTarget, we evaluate the zero-shot performance of state-of-the-art Multimodal Large Language Models (MLLMs) on predicting the existence of deictic gestures. GPT-4o [60] achieves a deictic gesture existence accuracy of 70.21%, while Claude-3.7 [61] obtains 68.48%. State-of-the-art open-vocabulary segmentation VLMs (GLaMM [62], OMG-LLaVA [63]) all achieve <5.0% IoU for the gesture target mask estimation task. These results indicate that even current SOTA MLLMs and VLMs struggle with reliably identifying deictic gestures and their target, highlighting the importance of designing new foundation model for this task.

4 Method

4.1 Model Architecture

Figure 2 illustrates our transformer-based, gaze-aware gesture target estimation model **TransGesture**. The framework consists of a large-scale frozen visual encoder and two transformer-based decoders dedicated to gaze and gesture representation learning, respectively. Their outputs are subsequently fused through a gesture-gaze joint cross-attention module to produce a segmentation mask of the gesture target and a binary existence prediction.

Shared Backbone. Inspired by recent advances in VLMs [64, 65] and prior work on human pose and gaze estimation [66, 53], we utilize visual representations from a large-scale, frozen pretrained feature extractor as input to our downstream gesture target estimation model. While such extractors encode rich semantic information, gesture target estimation fundamentally relies on fine-grained

spatial reasoning across instances. We empirically found that DINOv2 [67] outperforms CLIP-based encoders in preserving intra-instance token coherence, leading to more accurate gesture target estimation. While CLIP excels at semantic discrimination, its token embeddings are less spatially consistent across instances in our task setting.

Specifically, for an input RGB image $X \in \mathbb{R}^{H_{\text{in}} \times W_{\text{in}} \times 3}$, the frozen visual encoder provide the visual feature $Z = f_\phi(X)$, where $Z \in \mathbb{R}^{H \times W \times d_{\text{model}}}$. H and W is the height and width of the output feature map, defined by $H = H_{\text{in}}/P$ and $W = W_{\text{in}}/P$, where P is the patch size of the token.

Head and Body Patch Encoding. To incorporate the head position and body positions of a subject person who perform gestures, we construct two downsampled, binarized masks $M_{\text{head}} \in \mathbb{R}^{H \times W}$ and $M_{\text{body}} \in \mathbb{R}^{H \times W}$. While M_{body} is extracted by the subject's person bounding box X_{subject}, RetinaFace-ResNet50 [68] predicts M_{head} and uses the eye key points X_{leye} and X_{reye} to match the head to extracted subject bounding boxes. Two learnable patch encoding $\text{PE}_{\text{head}} \in d_{\text{gaze}}$ and $\text{PE}_{\text{body}} \in d_{\text{gesture}}$ are used to enhance selected region for downstream gaze and gesture decoders. Consequently, we have:

$$Z_{\text{gaze}} = Z + M_{\text{head}} \cdot \text{PE}_{\text{head}} \quad (1) \qquad Z_{\text{gesture}} = Z + M_{\text{body}} \cdot \text{PE}_{\text{body}} \quad (2)$$

where $Z_{\text{gaze}} \in \mathbb{R}^{H \times W \times d_{\text{gaze}}}$ and $Z_{\text{gesture}} \in \mathbb{R}^{H \times W \times d_{\text{gesture}}}$ are feature maps projected from Z, additional absolute 2D sinusoidal position embeddings [69] are added to each visual feature before adding the position encoding.

Transformer-based Gaze Decoder. We employ a frozen gaze decoder, pre-trained on the GazeFollow dataset[23], to extract feature representations of gaze targets. The decoder consists of a stack of Transformer layers designed to model spatial dependencies relevant to gaze estimation. It takes as input the head-conditioned feature map Z_{gaze}, which encodes the positional and contextual visual information centered on the subject's head. Leveraging self-attention, the decoder captures fine-grained relationships between different spatial regions in Z_{gaze}, enabling it to infer the likely direction and location of the person's gaze. We freeze the decoder to preserve its pre-learned self-attention patterns and prevent overfitting during training on our gesture dataset.

Transformer-based Gesture Decoder. To refine the feature representation for gesture understanding, we introduce other lightweight learnable Transformer layers that capture spatial dependencies critical for gesture existence classification and target mask prediction. The input Z_{gesture} is first reshaped from $H \times W \times d_{\text{gesture}}$ to $HW \times d_{\text{gesture}}$, where HW is the length of visual tokens and d_{gesture} is a smaller feature dimension. Then a learnable gesture existence token $t_{\text{exist}} \in \mathbb{R}^{d_{\text{gesture}}}$ is concatenated to Z_{gesture}:

$$Z'_{\text{gesture}} = [t_{\text{exist}}, \underbrace{t_1, t_2, t_3, \cdots, t_{H \times W}}_{Z_{\text{gesture}}}] \in \mathbb{R}^{HW \times d_{\text{gesture}}} \quad (3)$$

Z'_{gesture} is used as the input of the gesture transformer decoder. A 2-layer MLP takes t_{exist} of the last layer as the input and output the existence score of deictic gesture.

Gesture-Gaze Joint Cross Attention (JCA). To enhance the feature integration of gaze and gesture, we apply a joint cross attention layer to fuse tokens. Assume $F_{\text{gaze}} \in \mathbb{R}^{HW \times d_{\text{gaze}}}$ and $F_{\text{gesture}} \in \mathbb{R}^{HW \times d_{\text{gesture}}}$ are the output of the Transformer-based gaze and gesture decoder. A projection function f^g is used to alignment dimension of F_{gaze} to F_{gesture}. Thus, the gesture-gaze joint feature is defined by:

$$F_{\text{joint}} = [\underbrace{s_1, s_2, s_3, \cdots, s_{H \times W}}_{f^g(F_{\text{gaze}})}, \underbrace{t_1, t_2, t_3, \cdots, t_{H \times W}}_{F_{\text{gesture}}}] \in \mathbb{R}^{2HW \times d_{\text{gesture}}} \quad (4)$$

Then, the module performs cross-attention between F_{gesture} and F_{joint}:

$$\begin{aligned} Q = F_{\text{gesture}} W_q, \qquad K = F_{\text{joint}} W_k, \qquad V = F_{\text{joint}} W_v, \\ A = \text{Softmax}(QK^T / d_{\text{gesture}}), \qquad \text{JointCrossAttention}(F_{\text{gesture}}, F_{\text{joint}}) = AV \end{aligned} \quad (5)$$

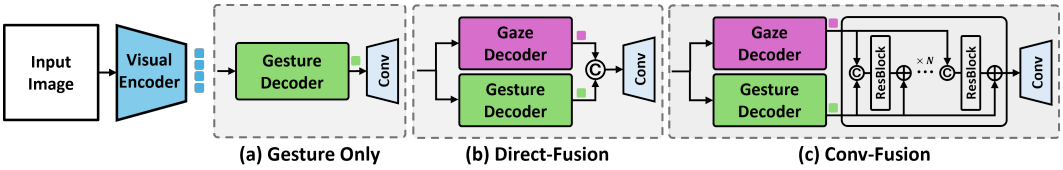

Figure 3: Baseline models for gesture target estimation.

Moreover, as in self-attention, we also use multiple heads when we apply gesture-gaze joint cross attention in the Transformer model. Layer Normalization and a feed-forward network is also connected after the joint cross attention. The design simplifies the fusion process without weakening the interaction between gaze tokens and gesture tokens.

Joint Learning Objective. We train our model using a joint multitask objective that combines pixel-wise binary cross-entropy loss for gesture target segmentation and focal loss for gesture existence prediction. The ground truth for the segmentation task is a binary mask $Y_{\text{mask}} \in \mathbb{R}^{H \times W}$, while the gesture existence is supervised with a binary label Y_{exist}. We use focal loss to address class imbalance in the existence prediction task. The overall training loss is defined as:

$$\mathcal{L}_{\text{total}} = (1 - \beta) \cdot \mathcal{L}_{\text{BCE}}(Y_{\text{mask}}, \hat{Y}_{\text{mask}}) + \beta \cdot \mathcal{L}_{\text{focal}}(Y_{\text{exist}}, \hat{Y}_{\text{exist}}) \tag{6}$$

4.2 Model Training

Stage 1: Pre-training for Gaze Decoder. We first pre-train the gaze decoder on the GazeFollow dataset[23] to precisely extract gaze-related features. Specifically, we modify the original model architecture by removing all gesture-related branches and appending a dedicated gaze prediction head after the gaze decoder. The training objective is formulated as minimizing the pixel-wise binary cross-entropy loss for gaze target heatmap prediction. An auxiliary head regression loss [52] for 2D spatial guidance is also used. During pre-training, we freeze the visual encoder parameters and exclusively optimize all weights associated with the gaze decoder, including the head positional encoding [53].

Stage 2: Fine-tuning End-to-End. In this stage, we always keep the visual encoder weights and gaze decoder frozen, and continue to update the weights in the gesture decoder and fusion modules. The training objective is the joint multitask loss function defined in Formula 6.

4.3 Baselines

Apart from the gesture-gaze joint cross attention fusion module, we also introduce several baseline fusion strategies designed for gesture target estimation. These baseline models serve as initial benchmarks and can be considered as different varient of TransGesture, providing insights into the effectiveness of incorporating multimodal information such as gaze direction and visual context. Each baseline employs a shared frozen visual encoder and separate decoders for gaze and gesture targets, but they differ in their approach to fusing gesture and gaze cues.

Direct Fusion. Direct fusion integrates gaze and gesture visual features via straightforward concatenation. Specifically, the visual encoder extracts visual features, and the gaze decoder provides gaze features. These two feature vectors are concatenated along the channel dimension and then passed to the gesture decoder. Formally, the fusion can be described as: $F_{\text{fusion}} = \textbf{Concat}(F_{\text{gaze}}, F_{\text{gesture}})$.

This simple concatenation facilitates the basic integration of multimodal information but does not explicitly model interactions between modalities.

Conv Fusion. Deep convolution fusion method employs a more sophisticated convolution-based fusion strategy designed to capture richer multimodal interactions. After obtaining visual features and gaze features , these features are concatenated and subsequently processed through convolutional layers (denoted as Conv) in residual blocks (ResBlock) [70] to better integrate and refine multimodal interactions. The single step fusion operation can be formally represented as:

$$F_{\text{fusion}} = \textbf{ResBlock}(\textbf{Concat}(F_{\text{gaze}}, F_{\text{gesture}})) + F_{\text{gesture}} \tag{7}$$

The Conv Fusion approach thus explicitly learns interactions between gaze and visual contexts, enhancing the ability of the model to accurately predict gesture targets in complex scenes.

5 Experiments and Results

5.1 Implementation Details

All models are trained and evaluated on the proposed GestureTarget dataset, using an 80:20 split for training and testing. Following practices from PASCAL VOC Action Recognition [71], where ground-truth bounding boxes for subjects are provided during both training and evaluation, we similarly assume availability of the subject person's bounding box at both train and test time. For gaze decoder pre-training, we use the official GazeFollow training dataset [23].

Evaluation Metrics We evaluate the gesture target estimation performance by two main metrics: Existence Accuracy, whether the intended target is correctly detected as present in the scene, and IoU, the localization overlap with the true target.

Technical Details All models in our study generate gesture target masks with a resolution of 128×128. We freeze the visual encoder during training, using its default image and patch sizes. For the gesture-gaze fusion module, we employ three Transformer layers featuring joint cross-attention, each configured with 16 attention heads and a 1024-dimensional MLP. To enhance model robustness and generalization, we apply diverse augmentation techniques during training, including head/body bounding box jittering, color jittering, random resizing and cropping, random horizontal flipping, random rotations, and random masking of scene patches. All models are trained for 25 epochs using the Adam optimizer and a cosine learning rate scheduler with an initial rate of 1e-3 and batch size 32, followed by an additional 5 epochs with a reduced learning rate of 1e-5. All experiments are conducted with 1 NVIDIA H100 GPU.

5.2 Main Results

Table 1 compares the performance of different fusion strategies and visual encoders for gesture target estimation, reporting Existence Accuracy, which indicates whether the intended target is correctly detected as present in the scene, and IoU, which measures the localization overlap with the true target. We analyze the effectiveness of the proposed **Gesture-Gaze Joint Cross-Attention (JCA)** fusion module across various visual backbones. JCA fusion consistently improves performance for all tested image encoders, outperforming other fusion strategies in both existence accuracy and IoU. For instance, with the CLIP-Large encoder, the baseline (gesture-only, no gaze) achieves an existence accuracy of 79.21% and an IoU of 54.37%. Incorporating JCA improves them to 87.66% and 56.65%, respectively – an absolute gain of +8.5% in accuracy and +2.3% in IoU. A similar trend is observed with the SigLIP-Base model. Notably, the DINOv2-Base backbone, which starts with a relatively low-performing baseline, sees the largest improvement: JCA raises its existence accuracy from 79.19% to 89.01% and IoU from 52.14% to 58.48% (a +9.8% and +6.3% jump, respectively). Across all encoders, JCA yields the best performance among the fusion modules. In comparison, convolution-based fusion techniques ***Direct Fusion*** and ***Conv Fusion*** offer only modest gains, and can even degrade target estimation performance if not carefully designed. These results highlight the effectiveness of the joint cross-attention module in integrating gaze and gesture cues, leading to higher accuracy and more precise localization across all visual encoders.

In addition, we also recruited 10 volunteer participants and asked them to perform two tasks on a randomly selected subset of 50 images from our test set: (1) Gesture existence detection (binary classification): determining whether the subject in the image is performing a deictic gesture; and (2) Target identification: if a gesture is present, identifying the intended target in the image by clicking on the corresponding object or person mask. For the gesture existence detection task, the average human performance was 87.00%, with a maximum of 90.00%. For the target estimation task, we measured performance by the IoU between the region selected by each participant and the ground-truth target

TransGesture series Models						
Visual Encoder	**#Params**	**Resolution/Patch Size**	**Fusion Method**	**Gaze Guidance**	**Exist Acc (%) ↑**	IoU (%) ↑
CLIP-Large* [72]	303M	336/14	Gesture-Only	✗	79.21	54.37
			Direct Fusion	✓	83.23	53.70
			Conv Fusion	✓	85.19	56.07
			JCA Fusion	✓	**87.66**	**56.65**
SigLIP-Base [73]	86.0M	512/16	Gesture-Only	✗	75.96	54.58
			Simple-Fusion	✓	83.52	54.24
			Conv Fusion	✓	84.32	55.68
			JCA Fusion	✓	**84.48**	**57.54**
SigLIP2-Base [74]	86.0M	512/16	Gesture-Only	✗	76.70	55.34
			Direct Fusion	✓	82.86	53.76
			Conv Fusion	✓	84.64	54.78
			JCA Fusion	✓	**85.13**	**56.13**
DINOv2-Base [67]	86.6M	518/14	Gesture-Only	✗	79.19	52.14
			Direct Fusion	✓	80.19	53.66
			Conv Fusion	✓	88.10	54.63
			JCA Fusion	✓	**89.01**	**58.48**
Human Baselines						
Human Groups					**Exist Acc (%) ↑**	IoU (%) ↑
Average of all subjects					87.00	66.40
Maximum of all subjects					90.00	80.00

Table 1: Exploring the influence of CLIP/SigLIP-based visual encoder and DINOv2 in gesture target estimation with different gaze and gesture feature fusion strategies. In human baselines, we compare the average human performance and the maximum human performance.

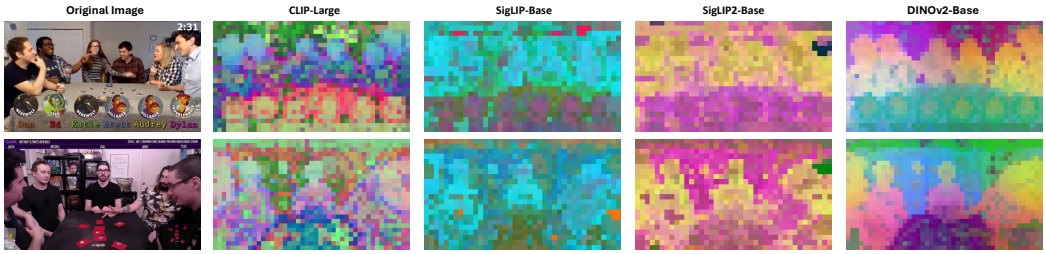

Figure 4: Compared with CLIP, SigLIP, SigLIP2, DINOv2 in token-wise affinity map visualization.

mask; the average human Target IoU was 66.40, with a maximum of 80.00%. This human study provides an upper bound on achievable performance.

Comparison of Different Visual Backbones. Table 1 compares the performance of several pre-trained visual encoders, including CLIP [72], SigLIP [73], SigLIP2 [74], and DINOv2 [67], all trained on large-scale datasets. Among them, DINOv2 achieves the best performance on our task, demonstrating strong general-purpose feature extraction capabilities. As shown in the token affinity visualizations in Figure 4, DINOv2 preserves notably stronger intra-instance token affinity, resulting in more coherent and spatially structured representations. In contrast, CLIP- and SigLIP-based models exhibit weaker spatial coherence, indicating limitations in modeling spatial relationships in complex social scenes involving multiple humans.

5.3 Ablation Study

Impact of Model Scale. To assess how backbone size influences gesture target estimation, we evaluated visual encoders with progressively larger parameter counts, as shown in Table 2. Increasing the encoder size leads to modest but consistent improvements in both existence accuracy and mask IoU. However, this trend plateaus at DINOv2-Large.

Image Encoder	#Params	Exist Acc (%) ↑	IoU (%) ↑
SigLIP2-Base	86.0M	85.13	56.13
SigLIP2-Large	303M	88.36	57.27
DINOv2-Small	22.1M	87.52	56.77
DINOv2-Base	86.6M	89.01	**58.48**
DINOv2-Large	304M	**89.85**	58.26

Table 2: Ablation of different scale of pretrained Vision Transformer Encoder.

β	Exist Acc (%) ↑	IoU (%) ↑
0.01	84.74	56.24
0.1	86.44	**58.62**
0.5	**89.01**	58.48
0.9	87.36	55.28
0.99	85.50	56.49

Table 3: Ablation of β in the loss function (use DINOv2-Base as image encoder).

Impact of Loss Hyperparameter. In Table 3, we study the impact of hyperparameter β in the total loss function (Formula 6). We observe that both decreasing and increasing β lets the model perform poorly in the existence classification and target mask prediction tasks, thus demonstrating the complementary nature of these two training objectives.

Impact of Freezing Individual Modules. We ablated TransGesture by selectively freezing its constituent components; the results are summarised in Table 4. Finetuning the DINOv2 visual backbone yields only a marginal uplift in existence accuracy and no measurable gain in target mask IoU, yet it inflates the number of trainable parameters by an order of magnitude and lengthens training time substantially. These findings suggest that DINOv2's pretrained representations are already near-optimal for gesture-target estimation, and further finetuning offers a poor cost-benefit trade-off.

Frozen Image Encoder	Frozen Gaze Decoder	Frozen Gesture Decoder	Trainable #Params	Exist Acc (%) ↑	IoU (%) ↑
✗	✗	✗	94.21M	**91.07**	57.22
✓	✗	✗	7.63M	87.48	57.92
✗	✓	✗	91.65M	85.56	57.52
✓	✓	✗	5.07M	89.01	**58.48**

Table 4: Ablation experiment on frozen modules in the proposed TransGesture framework.

Inference Efficiency. We also measured the inference runtime of the TransGesture model (using a DINOv2 encoder) to compare the computational cost of its different fusion architectures on 518×518 images. The average runtimes were 0.0108s (Gesture-Only), 0.0111s (Direct Fusion), 0.0131s (Conv Fusion), and 0.0110s (JCA Fusion). This demonstrates that incorporating JCA fusion adds a negligible overhead of only 0.0002s compared to the gesture-only baseline. In practice, the computations for the gaze branch, which primarily involve a few additional cross-attention layers, are lightweight relative to the overall cost of the Transformer-based visual encoder.

5.4 Visualization

Figure 5 contrasts three variants of our system: the gesture-only baseline, Conv Fusion, and the proposed gesture-gaze JCA Fusion. For gesture-only baseline, predictions frequently blur across large image regions or drift to distracting objects, and breakdowns are common when scenes contain more than three people. For conv fusion, the additional convolutional fusion layer suppresses some noise, yet masks remain coarse and often bleed beyond true object boundaries—especially in cluttered, multi-person interactions. For our proposed method that can jointly aligning gaze and gesture cues, it pinpoints the intended target with compact masks that closely trace the ground-truth silhouettes while ignoring irrelevant actors, even under occlusion, low light, or dense layouts. Failures arise only in the most complex scenes that demand high-level spatial reasoning (the last row). These qualitative results highlight our methods' clear advantage in precise gesture target estimation.

5.5 Demo

We implement an MLLM workflow that uses TransGesture as a tool to support complex text queries from users and can integrate with different other social AI models. Details of this agent is showed in Appendix C.

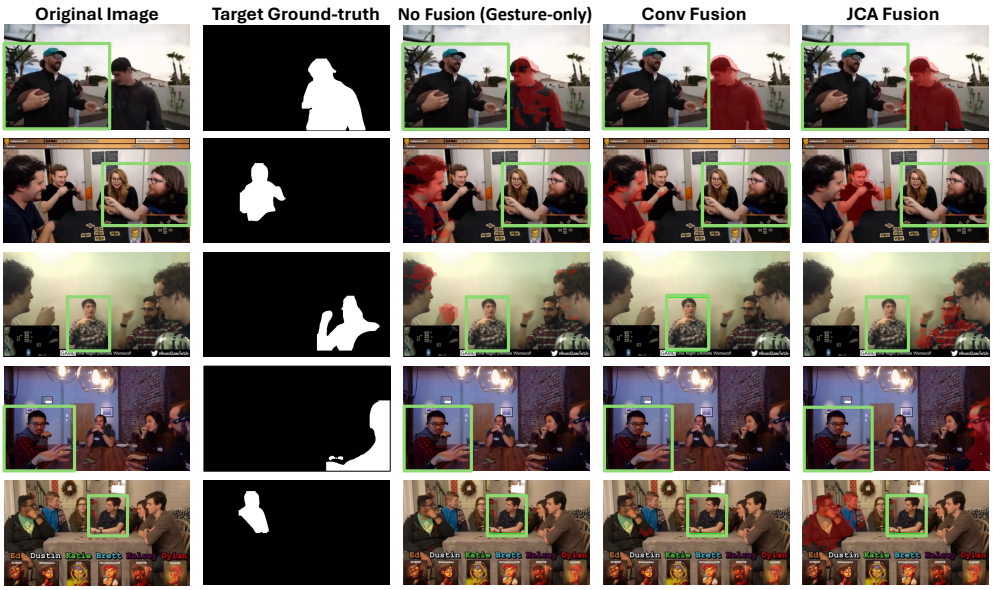

Figure 5: Qualitative examples of gesture target estimation under different fusion strategies. Green bounding boxes indicate the gesture initiator, and red masks show the predicted target person.

6 Discussion

The practical applications of deictic gesture target estimation are diverse and impactful. In healthcare, this technology can significantly enhance the analysis of non-verbal cues in clinical settings, particularly for early autism diagnosis. Since individuals with autism spectrum disorder often display differences in gesture use and interpretation, an accurate model for detecting these deictic cues could assist clinicians in identifying subtle communication deficits in young children, enabling earlier interventions [75]. Beyond the clinical domain, this work is crucial for improving human–AI interaction. Because deictic gestures are fundamental to natural communication, enhancing a machine's ability to understand them facilitates smoother, more intuitive collaborations [76]. This is especially valuable for assistive robotics, where robots must interpret human intent to provide effective support, and for virtual assistants operating in complex social or professional environments.

As with many activity recognition tasks, our work carries potential privacy implications. Therefore, while our dataset and models will be publicly released for research purposes, careful ethical consideration is essential—especially when utilizing these resources in contexts involving accusations or consequential decision-making. A more detailed discussion is provided in the Appendix D.

7 Conclusion

We investigated a novel problem: detecting deictic gestures and associating them with their intended targets. To address this challenge, we introduced a new Transformer-based architecture featuring joint cross-attention between gesture and gaze cues. Additionally, we contributed a large-scale annotated dataset, providing images with labeled bounding boxes for subject persons, corresponding body locations, and semantic masks of gesture targets. Through extensive experiments, we demonstrated the effectiveness and importance of our approach across two key tasks: deictic gesture existence prediction and gesture target estimation.

Acknowledgments

We acknowledge partial support from the Health Care Engineering Systems Center (Grainger College of Engineering, UIUC). We also extend our thanks to Prof. Jianguo Cao (Shenzhen Children's Hospital) for his valuable discussion regarding clinical practice.

References

[1] Michael A Arbib. *How the brain got language: The mirror system hypothesis*, volume 16. Oxford University Press, 2012.

[2] Holger Diessel. Where does language come from? some reflections on the role of deictic gesture and demonstratives in the evolution of language. *Language and Cognition*, 5(2-3):239–249, 2013.

[3] Sangmin Lee, Minzhi Li, Bolin Lai, Wenqi Jia, Fiona Ryan, Xu Cao, Ozgur Kara, Bikram Boote, Weiyan Shi, Diyi Yang, et al. Towards social ai: A survey on understanding social interactions. *arXiv preprint arXiv:2409.15316*, 2024.

[4] Adam Kendon. Do gestures communicate? a review. *Research on language and social interaction*, 27(3):175–200, 1994.

[5] Oliver Herbort and Wilfried Kunde. How to point and to interpret pointing gestures? instructions can reduce pointer–observer misunderstandings. *Psychological research*, 82(2):395–406, 2018.

[6] Allison Sauppé and Bilge Mutlu. Robot deictics: How gesture and context shape referential communication. In *Proceedings of the 2014 ACM/IEEE international conference on Human-robot interaction*, pages 342–349, 2014.

[7] Qi Wu, Cheng-Ju Wu, Yixin Zhu, and Jungseock Joo. Communicative learning with natural gestures for embodied navigation agents with human-in-the-scene. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4095–4102. IEEE, 2021.

[8] Martha W Alibali, Rebecca Boncoddo, and Autumn B Hostetter. Gesture in reasoning: An embodied perspective. In *The Routledge handbook of embodied cognition*, pages 233–245. Routledge, 2024.

[9] Hannah VanderHoeven, Nathaniel Blanchard, and Nikhil Krishnaswamy. Point target detection for multimodal communication. In *International Conference on Human-Computer Interaction*, pages 356–373. Springer, 2024.

[10] Yihao Liu, Xu Cao, Tingting Chen, Yankai Jiang, Junjie You, Minghua Wu, Xiaosong Wang, Mengling Feng, Yaochu Jin, and Jintai Chen. From screens to scenes: A survey of embodied ai in healthcare. *Information Fusion*, 119:103033, 2025.

[11] Abdul Mohaimen Al Radi, Xu Cao, Fanyang Yu, Yuyuan Liu, Fengbei Liu, Chong Wang, Yuanhong Chen, Jintai Chen, Hu Wang, Yanda Meng, et al. Agentic large-language-model systems in medicine: A systematic review and taxonomy. *Authorea Preprints*, 2025.

[12] Dan Liu, Libo Zhang, and Yanjun Wu. Ld-congr: A large rgb-d video dataset for long-distance continuous gesture recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3304–3312, 2022.

[13] Alexander Kapitanov, Karina Kvanchiani, Alexander Nagaev, Roman Kraynov, and Andrei Makhliarchuk. Hagrid–hand gesture recognition image dataset. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 4572–4581, 2024.

[14] Anton Nuzhdin, Alexander Nagaev, Alexander Sautin, Alexander Kapitanov, and Karina Kvanchiani. Hagridv2: 1m images for static and dynamic hand gesture recognition. *arXiv preprint arXiv:2412.01508*, 2024.

[15] Zeyi Zhang, Tenglong Ao, Yuyao Zhang, Qingzhe Gao, Chuan Lin, Baoquan Chen, and Libin Liu. Semantic gesticulator: Semantics-aware co-speech gesture synthesis. *ACM Transactions on Graphics (TOG)*, 43(4):1–17, 2024.

[16] Xingqun Qi, Hengyuan Zhang, Yatian Wang, Jiahao Pan, Chen Liu, Peng Li, Xiaowei Chi, Mengfei Li, Wei Xue, Shanghang Zhang, et al. Cocogesture: Toward coherent co-speech 3d gesture generation in the wild. *arXiv preprint arXiv:2405.16874*, 2024.

[17] Supreeth Narasimhaswamy, Uttaran Bhattacharya, Xiang Chen, Ishita Dasgupta, Saayan Mitra, and Minh Hoai. Handiffuser: Text-to-image generation with realistic hand appearances. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2468–2479, 2024.

[18] Haozhou Pang, Tianwei Ding, Lanshan He, Ming Tao, Lu Zhang, and Qi Gan. Llm gesticulator: leveraging large language models for scalable and controllable co-speech gesture synthesis. In *Eighth International Conference on Computer Graphics and Virtuality (ICCGV 2025)*, volume 13557, page 1355702. SPIE, 2025.

[19] David McNeill. Gesture, gaze, and ground. In *International workshop on machine learning for multimodal interaction*, pages 1–14. Springer, 2005.

[20] Kristine Lund. The importance of gaze and gesture in interactive multimodal explanation. *Language Resources and Evaluation*, 41:289–303, 2007.

[21] Marlou Rasenberg, Asli Özyürek, and Mark Dingemanse. Alignment in multimodal interaction: An integrative framework. *Cognitive science*, 44(11):e12911, 2020.

[22] Gozdem Arikan, Peter Boddy, and Kenny R Coventry. The relative importance of language, gaze, and gesture in deictic reference. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 2025.

[23] Adria Recasens, Aditya Khosla, Carl Vondrick, and Antonio Torralba. Where are they looking? *Advances in neural information processing systems*, 28, 2015.

[24] Daniel S Messinger and Alan Fogel. Give and take: The development of conventional infant gestures. *Merrill-Palmer Quarterly (1982-)*, pages 566–590, 1998.

[25] Amanda L Woodward and Jose J Guajardo. Infants' understanding of the point gesture as an object-directed action. *Cognitive Development*, 17(1):1061–1084, 2002.

[26] Colwyn Trevarthen. Form, significance and psychological potential of hand gestures of infants. In *The Biological Foundations of Gesture*, pages 149–202. Psychology Press, 2014.

[27] Irene Guevara, Cintia Rodríguez, and María Núñez. Developing gestures in the infant classroom: from showing and giving to pointing. *European Journal of Psychology of Education*, 39(4):4671–4702, 2024.

[28] David McNeill. *Hand and mind: What gestures reveal about thought*. University of Chicago press, 1992.

[29] David McNeill. *Gesture and thought*. University of Chicago press, 2019.

[30] Adam Kendon. How gestures can become like words. In *This paper is a revision of a paper presented to the American Anthropological Association, Chicago, Dec 1983.* Hogrefe & Huber Publishers, 1988.

[31] Adam Kendon. *Gesture: Visible action as utterance*. Cambridge University Press, 2004.

[32] Susan Goldin-Meadow and Diane Brentari. Gesture, sign, and language: The coming of age of sign language and gesture studies. *Behavioral and brain sciences*, 40:e46, 2017.

[33] Jun Wan, Yibing Zhao, Shuai Zhou, Isabelle Guyon, Sergio Escalera, and Stan Z Li. Chalearn looking at people rgb-d isolated and continuous datasets for gesture recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 56–64, 2016.

[34] Liang Zhang, Guangming Zhu, Lin Mei, Peiyi Shen, Syed Afaq Ali Shah, and Mohammed Bennamoun. Attention in convolutional lstm for gesture recognition. *Advances in neural information processing systems*, 31, 2018.

[35] Xin Zeng, Xiaoyu Wang, Tengxiang Zhang, Chun Yu, Shengdong Zhao, and Yiqiang Chen. Gesturegpt: Toward zero-shot free-form hand gesture understanding with large language model agents. *Proceedings of the ACM on Human-Computer Interaction*, 8(ISS):462–499, 2024.

[36] Lionel Pigou, Sander Dieleman, Pieter-Jan Kindermans, and Benjamin Schrauwen. Sign language recognition using convolutional neural networks. In *Computer Vision-ECCV 2014 Workshops: Zurich, Switzerland, September 6-7 and 12, 2014, Proceedings, Part I 13*, pages 572–578. Springer, 2015.

[37] Necati Cihan Camgoz, Oscar Koller, Simon Hadfield, and Richard Bowden. Sign language transformers: Joint end-to-end sign language recognition and translation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10023–10033, 2020.

[38] Ronglai Zuo, Fangyun Wei, and Brian Mak. Natural language-assisted sign language recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14890–14900, 2023.

[39] Yifan Zhang, Congqi Cao, Jian Cheng, and Hanqing Lu. Egogesture: A new dataset and benchmark for egocentric hand gesture recognition. *IEEE Transactions on Multimedia*, 20(5):1038–1050, 2018.

[40] Xu Cao, Pranav Virupaksha, Wenqi Jia, Bolin Lai, Fiona Ryan, Sangmin Lee, and James M Rehg. Socialgesture: Delving into multi-person gesture understanding. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 19509–19519, 2025.

[41] Pavlo Molchanov, Shalini Gupta, Kihwan Kim, and Jan Kautz. Hand gesture recognition with 3d convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 1–7, 2015.

[42] Pradyumna Narayana, Ross Beveridge, and Bruce A Draper. Gesture recognition: Focus on the hands. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5235–5244, 2018.

[43] Andrea D'Eusanio, Alessandro Simoni, Stefano Pini, Guido Borghi, Roberto Vezzani, and Rita Cucchiara. A transformer-based network for dynamic hand gesture recognition. In *2020 International Conference on 3D Vision (3DV)*, pages 623–632. IEEE, 2020.

[44] Mallika Garg, Debashis Ghosh, and Pyari Mohan Pradhan. Gestformer: Multiscale wavelet pooling transformer network for dynamic hand gesture recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2473–2483, 2024.

[45] Sindhu B Hegde, KR Prajwal, Taein Kwon, and Andrew Zisserman. Understanding co-speech gestures in-the-wild. *arXiv preprint arXiv:2503.22668*, 2025.

[46] Eunji Chong, Yongxin Wang, Nataniel Ruiz, and James M Rehg. Detecting attended visual targets in video. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5396–5406, 2020.

[47] Samy Tafasca, Anshul Gupta, and Jean-Marc Odobez. Childplay: A new benchmark for understanding children's gaze behaviour. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 20935–20946, 2023.

[48] Xucong Zhang, Yusuke Sugano, Mario Fritz, and Andreas Bulling. Appearance-based gaze estimation in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4511–4520, 2015.

[49] Yihua Cheng and Feng Lu. Gaze estimation using transformer. In *2022 26th International Conference on Pattern Recognition (ICPR)*, pages 3341–3347. IEEE, 2022.

[50] Mingfang Zhang, Yunfei Liu, and Feng Lu. Gazeonce: Real-time multi-person gaze estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4197–4206, 2022.

[51] Samy Tafasca, Anshul Gupta, Victor Bros, and Jean-Marc Odobez. Toward semantic gaze target detection. *Advances in Neural Information Processing Systems*, 37:121422–121448, 2024.

[52] Yuehao Song, Xinggang Wang, Jingfeng Yao, Wenyu Liu, Jinglin Zhang, and Xiangmin Xu. Vitgaze: gaze following with interaction features in vision transformers. *Visual Intelligence*, 2(1):1–15, 2024.

[53] Fiona Ryan, Ajay Bati, Sangmin Lee, Daniel Bolya, Judy Hoffman, and James M Rehg. Gaze-lle: Gaze target estimation via large-scale learned encoders. 2025.

[54] Anshul Gupta, Pierre Vuillecard, Arya Farkhondeh, and Jean-Marc Odobez. Exploring the zero-shot capabilities of vision-language models for improving gaze following. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 615–624, 2024.

[55] Anshul Gupta, Samy Tafasca, Arya Farkhondeh, Pierre Vuillecard, and Jean-marc Odobez. Mtgs: A novel framework for multi-person temporal gaze following and social gaze prediction. *Advances in Neural Information Processing Systems*, 37:15646–15673, 2024.

[56] Rowan Zellers, Yonatan Bisk, Ali Farhadi, and Yejin Choi. From recognition to cognition: Visual commonsense reasoning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6720–6731, 2019.

[57] Bolin Lai, Hongxin Zhang, Miao Liu, Aryan Pariani, Fiona Ryan, Wenqi Jia, Shirley Anugrah Hayati, James M Rehg, and Diyi Yang. Werewolf among us: A multimodal dataset for modeling persuasion behaviors in social deduction games. *arXiv preprint arXiv:2212.08279*, 2022.

[58] Amir Zadeh, Michael Chan, Paul Pu Liang, Edmund Tong, and Louis-Philippe Morency. Social-iq: A question answering benchmark for artificial social intelligence. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8807–8817, 2019.

[59] Glenn Jocher, Jing Qiu, and Ayush Chaurasia. Ultralytics YOLO, January 2023.

[60] OpenAI. Gpt-4o system card. `https://openai.com/index/gpt-4o-system-card/`, 2025.

[61] Anthropic. Introducing the next generation of claude. `https://www.anthropic.com/news/claude-3-family`, 2024.

[62] Hanoona Rasheed, Muhammad Maaz, Sahal Shaji, Abdelrahman Shaker, Salman Khan, Hisham Cholakkal, Rao M Anwer, Eric Xing, Ming-Hsuan Yang, and Fahad S Khan. Glamm: Pixel grounding large multimodal model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13009–13018, 2024.

[63] Tao Zhang, Xiangtai Li, Hao Fei, Haobo Yuan, Shengqiong Wu, Shunping Ji, Chen Change Loy, and Shuicheng Yan. Omg-llava: Bridging image-level, object-level, pixel-level reasoning and understanding. *Advances in Neural Information Processing Systems*, 37:71737–71767, 2024.

[64] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, et al. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025.

[65] Jinguo Zhu, Weiyun Wang, Zhe Chen, Zhaoyang Liu, Shenglong Ye, Lixin Gu, Yuchen Duan, Hao Tian, Weijie Su, Jie Shao, et al. Internvl3: Exploring advanced training and test-time recipes for open-source multimodal models. *arXiv preprint arXiv:2504.10479*, 2025.

[66] Yufei Xu, Jing Zhang, Qiming Zhang, and Dacheng Tao. Vitpose: Simple vision transformer baselines for human pose estimation. *Advances in neural information processing systems*, 35:38571–38584, 2022.

[67] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

[68] Jiankang Deng, Jia Guo, Yuxiang Zhou, Jinke Yu, Irene Kotsia, and Stefanos Zafeiriou. Retinaface: Single-stage dense face localisation in the wild. *arXiv preprint arXiv:1905.00641*, 2019.

[69] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[70] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[71] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88:303–338, 2010.

[72] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021.

[73] Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 11975–11986, 2023.

[74] Michael Tschannen, Alexey Gritsenko, Xiao Wang, Muhammad Ferjad Naeem, Ibrahim Alabdulmohsin, Nikhil Parthasarathy, Talfan Evans, Lucas Beyer, Ye Xia, Basil Mustafa, et al. Siglip 2: Multilingual vision-language encoders with improved semantic understanding, localization, and dense features. *arXiv preprint arXiv:2502.14786*, 2025.

[75] Şeyda Özçalışkan, Lauren B Adamson, and Nevena Dimitrova. Early deictic but not other gestures predict later vocabulary in both typical development and autism. *Autism*, 20(6):754–763, 2016.

[76] Cynthia Matuszek, Liefeng Bo, Luke Zettlemoyer, and Dieter Fox. Learning from unscripted deictic gesture and language for human-robot interactions. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 28, 2014.

[77] Samy Tafasca, Anshul Gupta, and Jean-Marc Odobez. Sharingan: A transformer architecture for multi-person gaze following. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2008–2017, 2024.

[78] Aymeric Roucher, Albert Villanova del Moral, Thomas Wolf, Leandro von Werra, and Erik Kaunismäki. 'smolagents': a smol library to build great agentic systems. `https://github.com/huggingface/smolagents`, 2025.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: Yes, and see abstract and Section 1. We claim to design a new task for human deictic gesture target estimation and propose the first model for this task.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: See Section 7 Conclusion.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

Justification: This is a task-driven application-based paper, thus we do not have any theoretical results or proofs.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The code and a subset of GestureTarget dataset is available in the supplementary material. They will be made publicly available upon acceptance.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: We will provide access to the data and the code upon acceptance, in accordance to the NeurIPS code and data submission guidelines. For paper review, we attach our code and a subset of the GestureTarget in the supplementary material.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: See Section 5.1 Technical Details.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: We run the experiments multiple times, with different train-validation splits, and take the average.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: See Section 5.1 - Technical Details.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: All authors in the paper reviewed and strictly followed NeurIPS Code of ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: Positive impacts are discussed in Section 1 Introduction. Negative impacts are discussed in Section 7 Conclusion. We will include a more detailed discussion of these impacts in the supplementary material.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.

- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: Our work poses no such risks, as all data used was publicly available.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We use the cc-by-nc-3.0 license for our work. (Code and Dataset)

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.

- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: We provide sufficient documentation along with our new assets (dataset, code and models).

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: We do not have a crowdsourcing component in the data annotation process.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [Yes]

    Justification: The dataset has IRB approval. We will include this approval and any necessary description in our supplementary material.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.

- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [NA]

    Justification: We do not use LLMs to help in our methodology. The extent to which we use them is zero-shot evaluations on our dataset to underscore the impact of our dataset.

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.

A Notations

Data and Indices	
H_{in}	Height of input image
W_{in}	Width of input image
$X \in \mathbb{R}^{3\times H_{in}\times W_{in}}$	Input RGB image
$X_{\text{subject}} \in \mathbb{R}^4$	Bounding box of subject's body
$X_{\text{leye}} \in \mathbb{R}^2$	Left eye keypoints
$X_{\text{reye}} \in \mathbb{R}^2$	Right eye keypoints
$X_{\text{lhand}} \in \mathbb{R}^2$	Left hand keypoints
$X_{\text{rhand}} \in \mathbb{R}^2$	Right hand keypoints
X_{text}	Text describing subject's appearance, location, action
$Y_{\text{target}} \in \mathbb{R}^4$	Bounding box target
$Y_{\text{mask}} \in \mathbb{R}^{H\times W}$	Target semantic segmentation mask
Y_{text}	Target text describing subject's appearance, location, action
Y_{exist}	Existence label
Embeddings and Positional Encodings	
P	Patch size of the token
$H = H_{\text{in}}/P$	Height of feature map
$W = W_{\text{in}}/P$	Width of feature map
$Z \in \mathbb{R}^{H\times W\times d_{\text{model}}}$	Feature map outputted by visual encoder
$M_{\text{head}} \in \mathbb{R}^{H\times W}$	Downsampled binary mask for head
$M_{\text{body}} \in \mathbb{R}^{H\times W}$	Downsampled binary mask for body
$\text{PE}_{\text{head}} \in d_{\text{gaze}}$	Patch encoding for head
$\text{PE}_{\text{body}} \in d_{\text{gesture}}$	Patch encoding for body
$Z_{\text{gaze}} \in \mathbb{R}^{H\times W\times d_{\text{gaze}}}$	Feature map adding gaze positional embeddings to Z
$Z_{\text{head}} \in \mathbb{R}^{H\times W\times d_{\text{head}}}$	Feature map adding head positional embeddings to Z
Attention Mechanism and Transformer Components	
$t_{\text{exist}} \in \mathbb{R}^{d_{\text{gesture}}}$	Learnable gesture existence token
Q, K, V	Query, key, value matrices
$F_{\text{gaze}} \in \mathbb{R}^{HW\times d_{\text{gaze}}}$	Output of the gaze decoder
$F_{\text{gesture}} \in \mathbb{R}^{HW\times d_{\text{gest.}}}$	Output of the gesture decoder
$F_{\text{joint}} \in \mathbb{R}^{2HW\times d_{\text{gesture}}}$	Joint feature used for joint cross attention
$\text{JointCrossAttention}(\cdot)$	Joint cross-attention
Loss Functions and Parameters	
$\mathcal{L}_{\text{BCE}}$	Binary crossentropy loss
$\mathcal{L}_{\text{focal}}$	Focal loss
$\mathcal{L}_{\text{total}}$	Total loss
β	Loss weighting parameter

B Additional Experiment

Results on Stage 1 - Pre-training for Gaze Decoder In model training Stage 1, we freeze the visual encoder to focus solely on pre-training the gaze decoder. Table 6 presents the evaluation outcomes on the GazeFollow test set, highlighting the effectiveness of our pre-trained gaze decoder. Our approach demonstrates highly competitive performance, achieving results comparable to state-of-the-art gaze estimation models. Specifically, we report an AUC of 0.959, Average L2 distance of 0.096, and Minimum L2 distance of 0.045. These metrics place our method closely alongside leading models such as Sharingan [77] (AUC: 0.938, Avg L2: 0.108, Min L2: 0.054), Gaze-LLE [53] (AUC: 0.958, Avg L2: 0.099, Min L2: 0.041), and ViTGaze [52] (AUC: 0.949, Avg L2: 0.105, Min L2: 0.047). These results validate the strength of our gaze decoder pre-training strategy in effectively capturing precise gaze targets and maintaining competitive accuracy.

Image Encoder	#Params	Learnable #Params	AUC ↑	Avg L2 ↓	Min L2 ↓
CLIP-Large	303M	2.93M	0.9550	0.0965	0.0455
SigLIP-Base	86.0M	2.86M	0.9484	0.1119	0.0559
SigLIP2-Base	86.0M	2.86M	0.9529	0.1052	0.0508
DINOv2-Small	22.1M	2.76M	0.9502	0.1146	0.0583
DINOv2-Base	86.6M	2.86M	0.9562	0.1021	0.0484
DINOv2-Large	304M	2.93M	**0.9591**	**0.0957**	**0.0446**

Table 6: Performance on Gazefollow in the gaze decoder pretraining stage.

Head Bounding Box	Exist Acc (%) ↑	IoU (%) ↑
✓	89.76	58.92
✗	90.04	50.57

Table 7: Comparison in the test set: existence of head bounding box.

Impact of Head Bounding Box (Gaze Guidance) To understand the influence of head bounding boxes as a form of gaze guidance, we conducted an ablation study using TransGesture with the DINOv2-Large visual encoder, summarized in Table 7. We divided the test dataset into two subsets based on the visibility of head bounding boxes. The first subset includes samples where the subject's head bounding box is visible, providing explicit gaze cues, while the second subset consists of cases without a visible head bounding box, typically due to occlusion or the subject's head being out of the camera's field of view.

Our findings reveal a significant performance disparity between these two subsets. Specifically, when head bounding boxes are present, the IoU scores for gesture target localization improve substantially compared to cases lacking explicit head gaze cues. This indicates that head bounding boxes (gaze guidance) serve as critical visual signals, enabling more accurate gesture targets estimation by effectively aligning gaze intention with the subject's gestures.

C Application: Integrate TransGesture into LLM Toolbox

To demonstrate the utility of TransGesture in downstream applications, we integrate the models as part of a larger toolbox for an LLM-based agent supported by GPT-4o. In our experiment, we implement a social gesture comprehension-focused agent by smolagent [78], with access to TransGesture (for gesture target estimation), object detection model, head detection model, and a gaze target estimation model. Figure 6 shows the proposed workflow for the agent.

The ideal query consists of at least one image along with a social gesture comprehension question posed by the user.

For stability, the tools are designed to be self-contained – that is, they depend on minimally processed raw inputs and compute any other required inputs such as head or subject bounding box information.

We define the inputs to the tools to be an image from the user's query, as well as an optional description of a human subject to be analyzed. This description is parsed from the user's query and is returned only if present. Optionally, if a user inputs an image of the target person instead of a text description, a vision-language model can be used to generate the description automatically.

If a description is included, for the TransGesture tool specifically, we first use YOLOv11 to detect persons in the image and yield respective bounding boxes. We then pass cropped regions of the image corresponding to individuals to a large vision-language model, which determines which region (if any) corresponds to the described target person. This is then used to extract head bounding box information via RetinaFace. The tool now has all the required inputs for our TransGesture model. The output of our model is overlaid on the original image, and is returned to the LLM. Figure 7 shows an example integration of TransGesture as a tool.

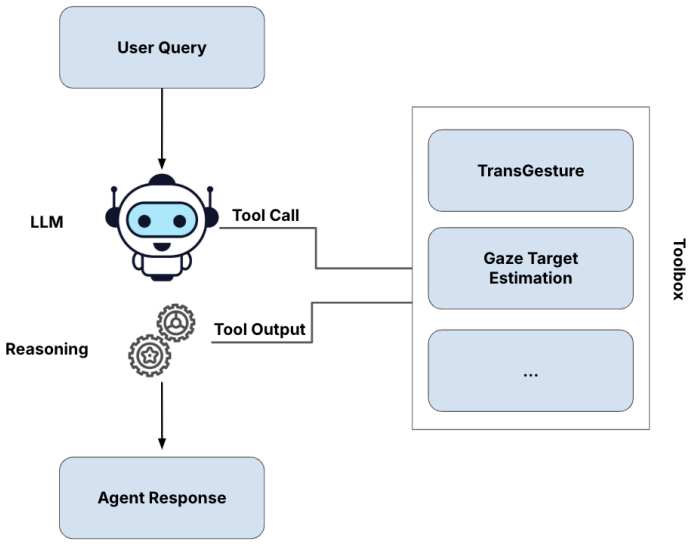

Figure 6: Proposed Agent Workflow.

As shown in Figure 7, TransGesture can be easily integrated as part of an agent's toolbox. We will publicly release all APIs of the agent demo.

D Broader Impacts

D.1 Positive Societal Impacts

The ability to estimate human deictic gesture targets has promising benefits for embodied intelligence and human–AI interaction. By empowering AI systems to understand where a person is pointing/reaching/showing, TransGesture can facilitate more natural, intuitive communication between humans and embodied AI agents such as robots or AR assistants. This is because deictic gesture can direct attention to relevant objects and aiding collaboration. Likewise, gaze cues are a fundamental part of non-verbal communication, and combining both modalities can substantially improve an AI's understanding of human intent. In assistive technology contexts, this capability could enable point-and-look interfaces for people with speech or mobility impairments, allowing them to refer to objects or locations and receive assistance without complex verbal commands. Overall, our transformer-based TransGesture model (trained on the GestureTarget dataset) can enhance embodied AI systems by making them more context-aware and responsive to natural human signals, potentially improving ease-of-use and inclusivity in everyday human–AI interactions.

D.2 Negative Societal Impacts

However, we recognize that this technology also raises important ethical and societal concerns. First, any system that relies on continuous visual and gaze monitoring inherently implicates privacy – even seemingly innocuous gaze data can reveal far more personal information than users intend (including aspects of identity, health, or interests). Without safeguards, a gesture-target estimator could be misused for unwarranted surveillance or attention tracking, monitoring where people look and point without their consent. This calls for careful dataset curation, and inclusive design. We stress the importance of deploying this technology with user consent, transparency, and context-appropriate limitations. By proactively addressing privacy (e.g. on-device processing, data anonymization), bias mitigation, and clear use policies, we aim to ensure that the benefits of our system are realized responsibly, without compromising individual rights or societal trust.

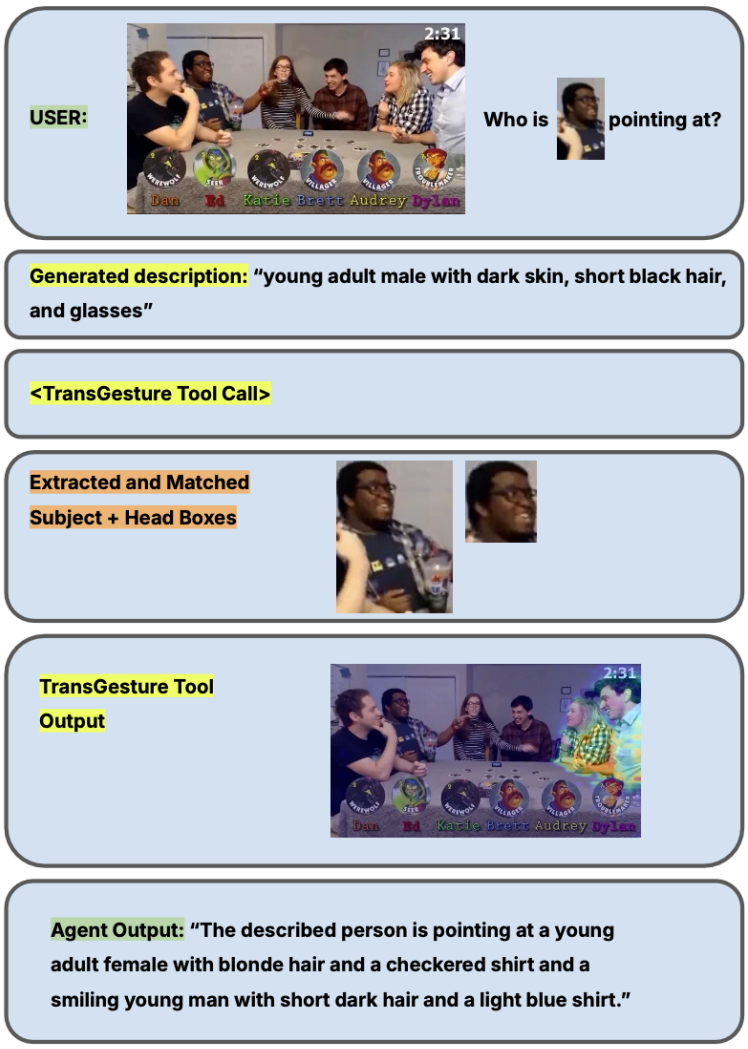

Figure 7: Example Agent Query and Response. Green marks content visible to the user. Yellow marks general internal agent outputs. Orange indicates tool internal outputs.