

# Pointing-Guided Target Estimation via Transformer-Based Attention

Luca Müller<sup>1</sup>[0009–0000–7682–1786], Hassan Ali<sup>1</sup>[0000–0001–9907–1834], Philipp Allgeuer<sup>1</sup>[0000–0002–2355–0764], Lukáš Gajdošech<sup>2</sup>[0000–0002–8646–2147], and Stefan Wermter<sup>1</sup>[0000–0003–1343–4775]<sup>\*</sup>

<sup>1</sup> University of Hamburg, Department of Informatics, Knowledge Technology Research Group, Hamburg, Germany  
`{luca.mueller, hassan.ali, philipp.allgeuer, stefan.wermter}@uni-hamburg.de`

<sup>2</sup> Faculty of Mathematics, Physics and Informatics, Comenius University, Bratislava, Slovakia  
`lukas.gajdosech@fmph.uniba.sk`

**Abstract.** Deictic gestures, like pointing, are a fundamental form of non-verbal communication, enabling humans to direct attention to specific objects or locations. This capability is essential in Human-Robot Interaction (HRI), where robots should be able to predict human intent and anticipate appropriate responses. In this work, we propose the **Multi-Modality Inter-TransFormer (MM-ITF)**, a modular architecture to predict objects in a controlled tabletop scenario with the NICOL robot, where humans indicate targets through natural pointing gestures. Leveraging inter-modality attention, MM-ITF maps 2D pointing gestures to object locations, assigns a likelihood score to each, and identifies the most likely target. Our results demonstrate that the method can accurately predict the intended object using monocular RGB data, thus enabling intuitive and accessible human-robot collaboration. To evaluate the performance, we introduce a patch confusion matrix, providing insights into the model’s predictions across candidate object locations. **Code** available at: <https://github.com/lucamuellercode/MMITF>.

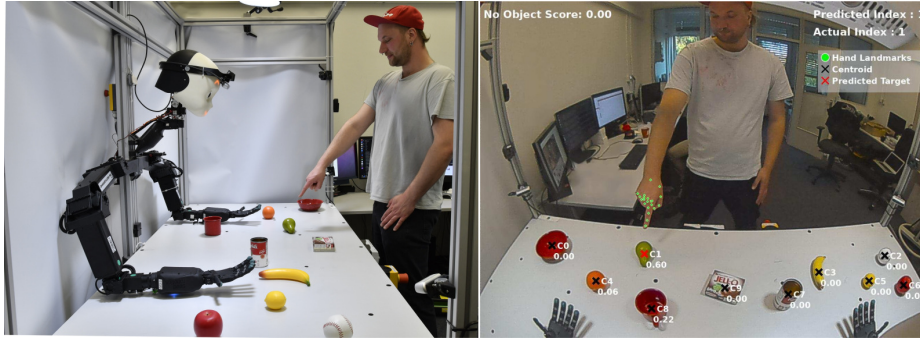
**Keywords:** Pose-object Matching · Attention-based Feature Fusion · Social Robotics

## 1 Introduction

The development of robots that collaborate with humans is increasing in various fields like industrial automation, healthcare, and domestic environments. As robots integrate further into society, their ability to react to human intent becomes a critical aspect of effective Human-Robot Interaction (HRI). Deictic gestures, such as pointing, provide a natural and intuitive means for humans to convey intent toward specific objects [16]. Due to the inherent ambiguity in natural language, pointing gestures offer a more precise spatial referral [27], and

---

<sup>\*</sup> This research was supported by Horizon Europe TERAIS (Grant 101079338), the DFG through the Crossmodal Learning (TRR-169), and the EU TRAIL. We thank Matthias Kerzel for his insightful comments that helped improve this manuscript.



**Fig. 1.** The participant interacts with the robot, pointing to an object. Objects are represented by centroids, with scores indicating their likelihood of being the target. The probability for the non-pointing case appears in the upper left corner.

bypass language barriers [11]. According to Lenz [18], objective pointing enables humans to direct attention to objects within the shared visual field of the pointer and receiver, relying on the pointer’s body pose alignment to the intended target. However, predicting a pointing target is challenging due to the need to detect hand poses, estimate direction, and identify the intended object. Traditional methods rely on measuring [10, 12, 4] or estimating [5] a pointing vector from 3D body representations, requiring costly hardware or extra processing, thus motivating the need for lightweight 2D approaches like ours.

The field of Human-Object Interaction (HOI) offers an alternative perspective on pointing gestures, emphasizing contextual relationships in predicting human intentions when interacting with objects. As such, a pointing gesture can be considered a form of HOI in itself. Encoder-decoder transformer architectures are a useful tool to learn the contextual representations of such interactions, producing robust attention maps [3]. Ji et al. extended this by introducing inter-modality attention [13], which fuses information across modalities, thus improving human-object reasoning. In our work, we propose the **Multi-Modality Inter-TransFormer (MM-ITF)**, which is a transformer-based encoder-decoder model that adapts inter-modality attention to capture the relationship between hand pose key points and object locations in a tabletop scenario (Figure 1). The proposed approach is RGB-based. Thus, it requires no extra equipment, wearable devices, or calibration. Our main contributions are as follows:

1. An **end-to-end transformer model** (Section 3.4), using inter-modal attention to predict in a single forward pass whether a human is pointing, and if so, which object is targeted.
2. An **evaluation with a social robot** (Section 4.2) in comparison to a 2D baseline in a controlled tabletop scenario, following prior setups [1, 2].
3. A **novel patch confusion matrix** (Section 4.3), constructed by mapping predicted object locations to discrete image regions, providing a structured visualization of the model’s predictions, and supporting interpretability.

## 2 Related Work

Many traditional approaches estimate a vector to determine a pointing gesture’s direction and project it into the scene to predict the target through intersection. This often requires 3D scene representations for precise estimation. Such representations can be obtained using sensors like IMUs [25] and EMG [10], offering high accuracy but requiring calibration and restricting movements. Multi-camera setups [17] and depth sensors [28] can reconstruct 3D human poses, but demand additional hardware and calibration. Moreover, single RGB camera approaches infer depth using models like MiDaS [24] and pose estimators [19, 7, 20], combining techniques as in [21], where skeleton data and ORB-SLAM [23] were integrated. However, a key challenge in pointing vector estimation is that the key points used to measure the vector—such as the forearm [10], index finger [12], or a vector formed from the nose to the index finger [4]—may not be collinear with the intended target. Approaches, like Bamani et al. [5], address this by learning a pointing vector directly from 2D input, yet their approach relies on multiple models, including depth estimation, arm segmentation, and pretrained pointing estimation with wearable sensor data. Ultimately, these methods determine the target object using geometric rules, typically by computing where the pointing vector intersects a plane in the scene’s representation [21, 5].

On the other hand, HOI detection localizes humans and objects in a scene and predicts their interactions [3]. Recent approaches leverage transformer-based encoder-decoder architectures [15, 9, 31, 26, 29] for end-to-end scene understanding, typically following a three-step strategy. First, a backbone network, like a CNN or DETR [8], extracts visual features. These feature maps are processed by the encoder for visual embedding. Then, the decoder attends to the encoded features using learnable queries representing interaction instances, followed by a simple prediction layer that generates the final HOI output: (human, interaction, object). Further, Ji et al. [13] introduce inter-modality attention by leveraging transformer-based attention to model dependencies between any pair of tokens. This allows human pose features to attend to object features within the encoder, enhancing the embedding and improving the model’s ability to capture human-object relationships. In this work, we adopt the encoder-decoder strategy from HOI methods to interpret human pointing gestures in robotic scenarios. By integrating inter-modality attention [13], we model the global scene context using hand pose and object location features, mapping them as hand-object pairs. Our work enables robots to infer human non-verbal pointing cues in a modular architecture, while eliminating the need for predefined geometric rules.

## 3 Methodology

We propose MM-ITF, an approach designed to predict target objects indicated by pointing gestures, using a multimodal integration of hand poses and object locations. The architecture consists of a pretrained backbone for feature extraction, an encoder to capture global context, and a decoder that maps this context

to hand-object pairs. Finally, a prediction layer assigns likelihood scores to each object. In the following, we describe the dataset, robotic platform, and architecture in detail.

### 3.1 Dataset and Robot Platform

Our work revolves around the Neuro-Inspired COLlaborator (NICOL) [14], earlier shown in Figure 1, which is a humanoid robotic platform designed to integrate both social interaction and physical collaboration. It features a stereo vision system, articulated arms, and an expressive face for multimodal interaction. The robot is fixed on a tabletop, creating a shared environment for collaboration between humans and the robot. The dataset consists of 30 videos, captured using the fisheye camera embedded in the robot’s left eye, showing 18 participants pointing at several objects in a controlled scenario. Each video features ten standard YCB objects [6] on the table, facilitating the scenario’s replication. Following the robot’s request to point at random objects, each participant performed nine pointing tasks—seven times with a single object and twice bi-manually with two objects simultaneously. In each frame in which pointing occurs, we locate hands and objects as 2D coordinates in the image space using pretrained models, resulting in a dataset of 572 samples with 356 pointing and 216 resting hands, each paired with a list of target objects.

Since transformers require large amounts of training data, we apply data augmentation by modifying the 2D coordinates of hand key points and object locations. Specifically, we introduce mirroring, eight random shifts along both the x- and y-axes, and eight rotations. These transformations are applied jointly to both the hand and the objects within a sample to ensure that the hand continues to point toward the intended target. Additionally, we apply Gaussian noise at four increasing levels to the hand key points and object locations to introduce robustness to minor input variations. The noise is sampled from a zero-mean normal distribution with standard deviations ranging up to 3 pixels and is randomly applied to 30% of the 2D coordinates. The augmented data significantly increased data variability, yielding 2,342,912 augmented samples.

### 3.2 Feature Extraction and Preprocessing

Our architecture has two input channels: hand pose and object location. Given an input frame, we extract hand key points and object bounding boxes. Also, we derive a third relationship feature, representing the angle between the index finger and each object location.<sup>3</sup> For hand pose estimation, we use MediaPipe [30] to detect 21 landmarks per hand. Each landmark  $\mathbf{lm}_i^p$  is a 2D coordinate:

$$\mathcal{P} = \{\mathbf{lm}_i^p\}_{i=1}^{21}, \quad \mathbf{lm}_i^p \in \mathbb{R}^2 \quad (1)$$

The spatial arrangement of these landmarks captures what we refer to as the *hand configuration*, which encodes information about the hand’s position, orientation, and gesture state, i.e., pointing or resting.

<sup>3</sup> We use superscripts to denote modalities:  $p$  (pose),  $o$  (object), and  $r$  (relationship).

For object detection, we employ OWLv2 [22], which generates a set of bounding boxes  $\{b_i^o\}$ , each defined as  $(x_{\min}, y_{\min}, x_{\max}, y_{\max})$ . From these, we compute their *centroids* as the center point of each bounding box, forming the sequence  $\{\mathbf{c}_i^o\}$ ,  $i \in \{1, \dots, N_t\}$ , where  $N_t$  denotes the number of detected objects. To account for cases where no object is being pointed at, we define a non-object token  $\mathbf{c}_{\text{non-object}} = (-1, -1)$ , choosing a value outside the valid image space for a clear distinction. With this token, the final sequence is:

$$\mathcal{O} = \{\mathbf{c}_i^o\}_{i=1}^{N_t} \cup \{\mathbf{c}_{\text{non-object}}\}, \quad \mathbf{c}_i^o, \mathbf{c}_{\text{non-object}} \in \mathbb{R}^2 \quad (2)$$

We generate a third feature as the angular alignment between the index finger landmarks and each *centroid*, reflecting the relationship between each hand-object pair. The *finger vector* is defined using the index fingertip and the top-most index finger joint,  $\mathbf{v}_{\text{finger}} = \mathbf{lm}_{\text{index\_finger\_tip}}^p - \mathbf{lm}_{\text{index\_finger\_dip}}^p$ .<sup>4</sup> For each detected object, we compute the vector from the index fingertip to the object centroid,  $\mathbf{v}_{\text{to\_centroid},i} = \mathbf{c}_i^o - \mathbf{lm}_{\text{index\_finger\_tip}}^p$ , and obtain the angle:

$$\theta_i = \arccos \left( \frac{\mathbf{v}_{\text{finger}} \cdot \mathbf{v}_{\text{to\_centroid},i}}{\|\mathbf{v}_{\text{finger}}\| \cdot \|\mathbf{v}_{\text{to\_centroid},i}\|} \right), \quad (3)$$

This results in the sequence  $\{\theta_i^r\}$ ,  $i \in \{1, \dots, N_t\}$ , where  $N_t$  denotes the number of objects. Similar to the object location sequence, we account for non-pointing hands by defining a non-relation token  $\theta_{\text{non-relation}} = -1$ , chosen outside the expected valid range for radians. With this token, the final sequence is:

$$\mathcal{R} = \{\theta_i^r\}_{i=1}^{N_t} \cup \{\theta_{\text{non-relation}}\}, \quad \theta_i^r \in [0, \pi], \quad \theta_{\text{non-relation}} \in \mathbb{R} \quad (4)$$

### 3.3 Embedding and Positional Encoding

The  $x$ - and  $y$ -values of the hand and object features are normalized to the interval  $[0, 1]$  using the image width  $W$  and height  $H$ :

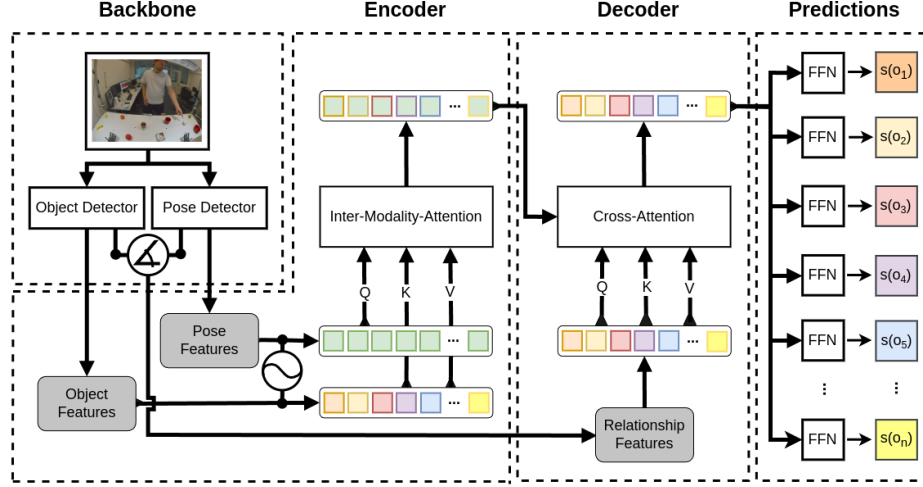
$$\tilde{x} = \frac{x}{W}, \quad \tilde{y} = \frac{y}{H} \quad (5)$$

Since relationship, pose, and object inputs have different dimensionalities, we project them to a common embedding space of dimension  $d_T$ . All  $(x, y)$  inputs, including centroid coordinates and hand landmarks, are embedded independently, with  $x$  and  $y$  projected to  $d_T/2$  dimensions each, and then concatenated. Angles are directly projected to  $d_T$ , ensuring a unified representation across all inputs.

For positional encoding, we follow [13, 8] and compute sinusoidal embeddings separately for  $x$  and  $y$ , concatenating them to form the final representation:

$$\mathcal{PE}(\tilde{x}, \tilde{y}) = \text{concat}(PE(\tilde{x}), PE(\tilde{y})), \quad (6)$$

<sup>4</sup> The *index\_finger\_dip* is the distal interphalangeal (DIP) joint of the index finger, the first joint below the fingertip.



**Fig. 2.** MM-ITF combines hand pose, object locations, and their angular relationship ( $\angle$ ) to predict pointing targets. The encoder uses hand pose features as queries ( $Q$ ) and object features as keys ( $K$ ) and values ( $V$ ), enabling inter-modality attention to capture global context. The decoder maps this context to hand-object pairs as relationship tokens, and a Feedforward Network (FFN) assigns scores  $s(o_i)$  to all objects.

$$PE(*)_{2i} = \sin(* / 10000^{2i/d_T}), \quad (7)$$

$$PE(*)_{2i+1} = \cos(* / 10000^{2i/d_T}), \quad (8)$$

where  $*$  represents either  $\tilde{x}$  or  $\tilde{y}$ . Since angles do not represent positional information in image space, we normalize them to  $[0, 2\pi]$ , project them directly to  $d_T$ , and exclude them from positional encoding. After embedding and encoding, the final transformer input is structured as follows:

$$\mathcal{P}' = \{\mathcal{PE}(W_h \mathbf{lm}_i^p)\}_{i=1}^{21}, \quad W_h \in \mathbb{R}^{d_T \times 2} \quad (9)$$

$$\mathcal{O}' = \{\mathcal{PE}(W_o \mathbf{c}_i^o)\}_{i=1}^{N_t+1}, \quad W_o \in \mathbb{R}^{d_T \times 2} \quad (10)$$

$$\mathcal{R}' = \{W_r \theta_i^r\}_{i=1}^{N_t+1}, \quad W_r \in \mathbb{R}^{d_T \times 1} \quad (11)$$

### 3.4 Inter-Modality Transformer

An overview of the architecture is shown in Figure 2. Our backbone produces an object sequence  $\mathcal{O}'$  and a pose sequence  $\mathcal{P}'$  as input to the encoder. In the inter-modality attention block, pose features act as *queries*, attending to object locations, which serve as *keys* and *values*. Each token in the pose sequence aggregates object location information, and the encoder outputs a *pose-object memory* that encodes the global context between the hand and detected objects.

Our decoder processes a sequence of relationship tokens  $\mathcal{R}'$ , constructed from pose and object location data. These tokens first undergo self-attention before attending, as *queries*, to the *pose-object memory* produced by the encoder. The cross-attention mechanism enables each relationship token to integrate scene-wide information, leveraging the global context captured by the encoder and

mapping it to its respective hand-object pair. The decoder outputs a sequence of tokens, each encoding pose-object information for a specific hand-object pair.

The decoder’s output sequence is processed by a Feedforward Network (FFN) with a sigmoid activation function, assigning a score to each token. The  $i$ -th decoder output token corresponds to the  $i$ -th input relationship token, representing a specific hand-object pair. These scores allow ranking the objects, where the model predicts the index  $j$  of the token with the highest score. Since each token retains its mapping to the original input relationship tokens, the predicted index identifies the hand-object pair most likely to fulfill the pointing relationship.

We frame our task as binary classification, where each output representation is evaluated based on whether it fulfills the pointing relationship. To optimize this, we use Binary Cross-Entropy (BCE) loss, encouraging the model to assign higher scores to hand-object pairs that align with the pointing relationship while reducing scores for those that do not. Since our goal is to generate scores for every object rather than make a strict binary decision, we do not apply a threshold to separate pointing and non-pointing tokens. Instead, we rank the raw scores, selecting the object with the highest likelihood as the predicted pointing target.

## 4 Experiments and Evaluation

We evaluate the MM-ITF architecture by comparing different channel configurations. Specifically, we compare a two-modality setup utilizing the hand pose and object location and a three-modality setup consisting of the same modalities in addition to the relationship feature (see Section 3.2). Our two-modality setup uses the same architecture described in Figure 2 where the relationship feature is replaced by the object location as input to the decoder. The results of both modality configurations are compared to a baseline that predicts objects based on proximity to a vector derived from the pointing gesture. In the following, we introduce the baseline, present the results, and conclude with a visual analysis of the model’s predictions using a patch confusion matrix.

### 4.1 Baseline for Evaluation

As baseline, we use a 2D method proposed by Ali et al. [1] to predict objects indicated by pointing hand gestures. This method was applied to the same tabletop scenario with the NICOL robot. It consists of a Multi-layer Perceptron (MLP) that uses hand landmarks to determine whether the user is pointing. The final target objects are predicted based on proximity of the object’s centroid to a continuous line that passes through the wrist and index finger key points. We choose this baseline because, similar to our approach, it relies completely on 2D data. Furthermore, its prior application within the same robotic setup ensures a fair and meaningful comparison. The baseline uses the same pretrained models for extracting hand key points and object locations, i.e., MediaPipe and OWLv2.

### 4.2 Experiment Results

We performed an eight-fold cross-validation, in which we trained eight models for both the baseline gesture classifier and our MM-ITF architecture. The 30

**Table 1.** Average object prediction results for the baseline method and our proposed MM-ITF architecture, with a two-modality and a three-modality configuration

Model	Accuracy	Precision	Recall	F1-Score	Top-2 Accuracy
Baseline [1]	0.89 $\pm$ 0.008	0.84 $\pm$ 0.007	0.90 $\pm$ 0.012	0.85 $\pm$ 0.008	<b>0.96</b> $\pm$ 0.004
MM-ITF*	0.71 $\pm$ 0.044	0.70 $\pm$ 0.037	0.68 $\pm$ 0.044	0.67 $\pm$ 0.041	0.92 $\pm$ 0.014
MM-ITF**	<b>0.90</b> $\pm$ 0.017	<b>0.88</b> $\pm$ 0.019	<b>0.92</b> $\pm$ 0.019	<b>0.90</b> $\pm$ 0.019	<b>0.96</b> $\pm$ 0.008

\* Our model trained with **two modalities** (hand pose and object locations).

\*\* Our model trained with all **three modalities**, including the relationship feature.

scenes from the dataset (see Section 3.1) were split for training, validation, and testing. Each model was trained on 21 scenes and validated on a unique subset of three, ensuring that all scenes were used for validation once, except for six scenes which were held out as a test set. Table 1 summarizes the test results, in which the values reflect the average performance across all models. MM-ITF in the three-modality setup (pose, object, and relationship data) achieves 90% accuracy, slightly above the baseline at 89%, showing that objects indicated by pointing gestures can be learned to a comparable performance without additional geometric post-processing. While the baseline relies on a two-stage geometric approach, our model jointly predicts both the gesture state and target object in a single step. Both methods similarly achieve high Top-2 accuracy, ranking the target object within the top two choices in 96% of cases.

Although the MM-ITF two-modality setup (pose and object) reaches only 71% accuracy, it achieves a high Top-2 accuracy of 92%, showing that it learns a meaningful link between the hand pose and object location but struggles to make a fine-grained correct final prediction. This suggests that while the model captures contextual relationships between hand pose and object location, it benefits noticeably from the relationship feature to improve object ranking precision.

Since the baseline selects the nearest object along a continuous line through the wrist and index finger, its predictions are straightforward to interpret. However, we need a confusion matrix to analyze the MM-ITF’s performance. A confusion matrix over predicted indices would offer limited insight, as object locations vary across samples, and constructing one based on centroids is impractical due to the extremely large number of possible positions. Therefore, we introduce a method to discretize our architecture’s outputs, mapping object centroids to fixed image regions for structured visual analysis.

### 4.3 Measuring Spatial Understanding

We further evaluate the performance of our architecture by visualizing the predicted objects. The table area in the image space is divided into evenly sized, non-overlapping patches, as illustrated in Figure 3. Since our model predicts an index corresponding to the object centroid  $(x, y)$  in image coordinates, each predicted and ground-truth centroid is mapped to a patch based on its coordinates. This mapping is achieved by dividing the  $x$ -coordinate by the patch width and the  $y$ -coordinate by the patch height. As a result, the centroid predictions are





**Fig. 3.** A visualization of the performance of our architecture using patches. The table space is divided into evenly sized, non-overlapping patches, and centroids (red dots) are assigned to patches by dividing their  $x, y$  coordinates by the patch width and height. The assigned patches are highlighted in purple.

discretized into predefined image regions, enabling structured spatial analysis through a confusion matrix over patches. Patches with no assigned predictions are filtered out, and row normalization is applied to enhance interpretability.

For a more detailed analysis of our MM-ITF model’s output, we construct a patch confusion matrix, which visualizes how predicted object centroid locations align with ground-truth centroids. Assigned patches for predicted centroids are shown along the x-axis, and those for ground-truth centroids along the y-axis. Each entry  $(i, j)$  represents how often a ground-truth centroid in patch  $i$  is predicted as patch  $j$ . Diagonal entries correspond to correct predictions, while off-diagonal values indicate spatial misclassifications. Additionally, the first row and last column of the matrix represent the non-object class, distinguishing non-pointing gestures from those associated with an object.

Examining the confusion matrix (Figure 4) for the three-modality setup reveals that correct predictions frequently receive high scores, demonstrating the model’s ability to distinguish target objects of a pointing gesture, even in close proximity.<sup>5</sup> This suggests that the relationship feature plays a key role in guiding target selection. However, when multiple objects align perfectly with the pointing direction, the model often predicts an object behind the actual target relative to the participant. For example, objects at  $(1, 12)$  are misclassified as  $(3, 12)$  in 88% of cases,  $(2, 10)$  as  $(3, 10)$  in 38%, and  $(1, 14)$  as  $(3, 15)$  in 50% of cases. Together, these findings indicate a strong reliance on hand-object alignment.

Beyond object predictions, the model distinguishes between pointing and non-pointing hands but prioritizes hand location over *hand configuration*. For example, it sometimes predicts no-object for gestures originating from positions typically associated with resting hands, as observed at patch  $(2, 11)$ . This suggests that hand position outweighs landmark arrangement, reinforcing the model’s reliance on hand-object alignment and proximity over hand articulation. The results for the two-modality setup reinforce our findings. The relationship

<sup>5</sup> The matrix shows average results over eight models; see Section 4.2 for details.



solely on human pose. By ranking likely target objects, our method serves as a building block for downstream tasks aimed at estimating human intent in collaborative scenarios. In this sense, our work contributes to a robot’s social skill set, enabling more intuitive and seamless interaction through the interpretation of pointing gestures. The architecture is modular and extendable, reflecting the multimodal nature of human communication.

While the current setup assumes fixed positions for the camera, table, and participants, future work will explore more dynamic and flexible interaction settings. We also aim to extend the hand features with modalities such as gaze, further enriching the global context between human and object features modeled by the encoder—and, in doing so, expanding the possibilities for humans to effortlessly indicate their intent to a robot.

**Disclosure of Interests.** The authors declare that they have no conflict of interest.

## References

1. Ali, H., Allgeuer, P., Wermter, S.: Comparing Apples to Oranges: LLM-powered Multimodal Intention Prediction in an Object Categorization Task. *ICSR* (2024)
2. Allgeuer, P., Ali, H., Wermter, S.: When Robots Get Chatty: Grounding Multimodal Human-Robot Conversation and Collaboration, p. 306–321. Springer (2024)
3. Antoun, M., Asmar, D.: Human Object Interaction Detection: Design and Survey. *Image Vision Computing* **130**(C) (2023)
4. Azari, B., Lim, A., Vaughan, R.T.: Commodifying Pointing in HRI: Simple and Fast Pointing Gesture Detection from RGB-D Images. *Conference on Computer and Robot Vision (CRV)* pp. 174–180 (2019)
5. Bamani, E., Nissinman, E., Koenigsberg, L., Meir, I., Matalon, Y., Sintov, A.: Recognition and Estimation of Human Finger Pointing with an RGB Camera for Robot Directive. *arXiv* **2307.02949** (2023)
6. Calli, B., Singh, A., Walsman, A., Srinivasa, S., Abbeel, P., Dollar, A.M.: The YCB object and Model set: Towards common benchmarks for manipulation research. In: *International Conference on Advanced Robotics ICAR*. pp. 510–517 (2015)
7. Cao, Z., Hidalgo, G., Simon, T., Wei, S.E., Sheikh, Y.: OpenPose: Realtime Multi-Person 2D Pose Estimation Using Part Affinity Fields. *IEEE Transactions on Pattern Analysis & Machine Int.* **43**(01), 172–186 (2021)
8. Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S.: End-to-End Object Detection with Transformers. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.M. (eds.) *ECCV*. pp. 213–229. Springer, Cham (2020)
9. Chen, M., Liao, Y., Liu, S., Chen, Z., Wang, F., Qian, C.: Reformulating HOI Detection As Adaptive Set Prediction. In: *CVPR*. pp. 9004–9013 (2021)
10. Haque, F., Nancel, M., Vogel, D.: Myopoint: Pointing and Clicking Using Forearm Mounted Electromyography and Inertial Motion Sensors. In: *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*. p. 3653–3656. CHI ’15, Association for Computing Machinery, NY, USA (2015)
11. Hewe, G.W.: Gesture Language in Culture Contact. In: *Sign Language Studies*, 3, vol. 4, pp. 1–34. Gallaudet University Press (1974)
12. Hu, K., Canavan, S., Yin, L.: Hand Pointing Estimation for Human Computer Interaction Based on Two Orthogonal-Views. In: *Proceedings of International Conference on Pattern Recognition*. pp. 3760–3763 (2010)

13. Ji, J., Desai, R., Niebles, J.C.: Detecting Human-Object Relationships in Videos. In: ICCV. pp. 8106–8116 (2021)
14. Kerzel, M., Allgeuer, P., Strahl, E., Frick, N., Habekost, J.G., Eppe, M., Wermter, S.: NICOL: A Neuro-Inspired Collaborative Semi-Humanoid Robot That Bridges Social Interaction and Reliable Manipulation. *IEEE Access* **11** (2023)
15. Kim, B., Lee, J., Kang, J., Kim, E.S., Kim, H.J.: HOTR: End-to-End Human-Object Interaction Detection With Transformers. In: CVPR. pp. 74–83 (2021)
16. Kita, S. (ed.): *Pointing: Where Language, Culture, and Cognition Meet*. Lawrence Erlbaum Associates, Mahwah, NJ (2003)
17. Kuramochi, A., Komuro, T.: *3D Hand Pointing Recognition over a Wide Area using Two Fisheye Cameras*, pp. 58–67. Springer International Publishing (2021)
18. Lenz, F.: *Deictic Conceptualisation of Space, Time and Person*. John Benjamins (2003)
19. Lugaresi, C., Tang, J., Nash, H., McClanahan, C., Uboweja, E., Hays, M., Zhang, F., Chang, C.L., Yong, M., Lee, J., Chang, W.T., Hua, W., Georg, M., Grundmann, M.: MediaPipe: A Framework for Perceiving and Processing Reality. In: Third Workshop on Computer Vision for AR/VR at IEEE CVPR 2019 (2019)
20. Maji, D., Nagori, S., Mathew, M., Poddar, D.: YOLO-Pose: Enhancing YOLO for Multi Person Pose Estimation Using Object Keypoint Similarity Loss. In: CVPR Workshops. pp. 2637–2646 (2022)
21. Medeiros, A.C.S., Ratsamee, P., Orlosky, J., Uranishi, Y., Higashida, M., Take-mura, H.: 3D pointing gestures as target selection tools: guiding monocular UAVs during window selection in an outdoor environment. *ROBOMECH* **8**(1), 14 (2021)
22. Minderer, M., Gritsenko, A., Hounsby, N.: Scaling Open-Vocabulary Object Detection. In: Proceedings of the 37th International Conference on Neural Information Processing Systems. NIPS ’23, Curran Associates Inc., Red Hook, NY, USA (2023)
23. Mur-Artal, R., Montiel, J.M.M., Tardos, J.D.: ORB-SLAM: A Versatile and Accurate Monocular SLAM System. *IEEE Trans. on Robotics* **31**(5), 1147–1163 (2015)
24. Ranftl, R., Lasinger, K., Hafner, D., Schindler, K., Koltun, V.: Towards Robust Monocular Depth Estimation: Mixing Datasets for Zero-Shot Cross-Dataset Transfer. *IEEE Trans. on Pattern Analysis & Machine Int.* **44**(03), 1623–1637 (2022)
25. Sikeridis, D., Antonakopoulos, T.: An IMU-Based Wearable System for Automatic Pointing During Presentations. *Image Processing & Communications* **21** (2017)
26. Tamura, M., Ohashi, H., Yoshinaga, T.: QPIC: Query-Based Pairwise Human-Object Interaction Detection with Image-Wide Contextual Information. In: CVPR. pp. 10405–10414 (2021)
27. Tomasello, M.: *Origins of Human Communication*. MIT Press (2008)
28. Tölgyessy, M., Dekan, M., Duchoň, F., Rodina, J., Hubinský, P., Chovanec, L.: Foundations of Visual Linear Human–Robot Interaction via Pointing Gesture Navigation. *International Journal of Social Robotics* **9**, 1–15 (2017)
29. Zhang, A., Liao, Y., Liu, S., Lu, M., Wang, Y., Gao, C., Li, X.: Mining the Benefits of Two-stage and One-stage HOI Detection. In: Ranzato, M., Beygelzimer, A., Dauphin, Y., Liang, P., Vaughan, J.W. (eds.) *Advances in Neural Information Processing Systems*. vol. 34, pp. 17209–17220. Curran Associates, Inc. (2021)
30. Zhang, F., Bazarevsky, V., Vakunov, A., Tkachenka, A., Sung, G., Chang, C.L., Grundmann, M.: MediaPipe Hands: On-device Real-time Hand Tracking. *ArXiv abs/2006.10214* (2020)
31. Zou, C., Wang, B., Hu, Y., Liu, J., Wu, Q., Zhao, Y., Li, B., Zhang, C., Zhang, C., Wei, Y., Sun, J.: End-to-End Human Object Interaction Detection with HOI Transformer. *CVPR* pp. 11820–11829 (2021)