Resolving Ambiguity in Pointing Gestures using Contextual Reasoning from Large Language Models*

Sumin Yeon, Minjae Lee, Jiho Bae, Sejik Park, Ilhwan Ha, and Suwon Lee†

Gyeongsang National University, Jinju-si, Gyeongsangnam-do, South Korea
{sumin.yeon, wjdchs0129, dream_cacao_jh, cafelena, mtslzx, leesuwon}@gnu.ac.kr

Abstract

In the field of Human-Computer Interaction (HCI), the ambiguity of pointing gestures remains a significant challenge in discerning user intent. This paper proposes an integrated system that combines a large language model (LLM), capable of understanding complex language, with pointing gestures to effectively process multimodal user commands. By synergistically leveraging spatial information from gestures and contextual reasoning from the LLM, our system accurately recognizes user intentions, even in complex environments.Experimental validation shows that our approach improves accuracy by over 12 %p compared to unimodal methods, demonstrating the potential for language-based spatial understanding within the field of HCI.

1 Introduction

With the increasing integration of computers into daily life, the field of Human-Computer Interaction (HCI) has seen a growing focus on technologies that emulate and support natural and intuitive modes of human communication. Interactions predicated solely on vocal or textual inputs are often insufficient for conveying complex user intent; social robots, in particular, require the capacity to process both linguistic and non-linguistic cues in tandem to achieve fluid interaction [5].

Humans frequently employ pointing gestures to refer to objects or areas of interest within their immediate environment. By coupling deictic pronouns, such as 'this' or 'that', with pointing, intent is expressed in a manner that is both simple and natural. This behavior serves as a critical non-linguistic tool for efficient communication, enabling an observer to swiftly identify the relevant object without requiring a detailed verbal description.

As depicted in Figure 1, the concurrent use of a verbal command and a pointing gesture facilitates natural comprehension for a human interlocutor. However, computational systems struggle to process these multimodal signals in a unified manner. This difficulty is exacerbated in real-world environments where the user's pointing gesture may be ambiguous or could reference one of several similar objects.

Recent advancements have demonstrated that Large Language Models (LLMs) exhibit remarkable performance in commonsense reasoning and contextual comprehension. This indicates that LLMs can interpret natural language by leveraging vast amounts of contextual information [10]. However, an LLM alone cannot directly process visual information, such as the precise location to which a person is pointing. Consequently, the spatial information obtained from the pointing gesture remains essential.

*Proceedings of The 2025 IFIP WG 8.4 International Symposium on E-Business Information Systems Evolution (EBISION 2025), Article No. 7, December 16-18, 2025, Sapporo, Japan.

†Corresponding Author

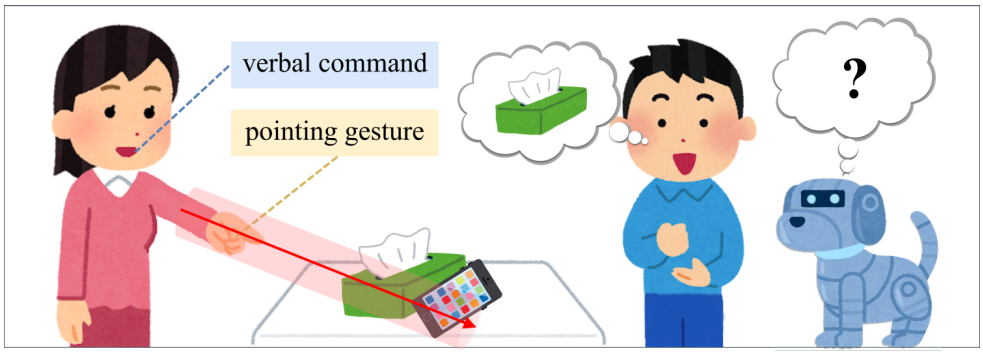

Figure 1: Problem Definition. When a voice command and a pointing gesture are presented concurrently, a human observer naturally integrates the two information streams, whereas computational systems are not yet capable of processing these cues in a comprehensive manner.

Furthermore, the proposed framework is closely related to E-Business Information Systems (EBIS), where multimodal interaction is increasingly important for enhancing user experience. Applications such as smart service interfaces, retail assistant systems, and collaborative platforms can benefit from our approach, as it enables systems to interpret both verbal and non-verbal cues for more intuitive and efficient communication.

This paper therefore proposes an integrated system for processing multimodal user commands, which combines a Large Language Model, capable of understanding higher-order human linguistic representations, with pointing gestures for spatial object designation. Within complex real-world settings, by fusing the contextual inference of an LLM with the spatial information from pointing gestures, our system resolves ambiguities in object recognition more effectively than unimodal approaches. The principal contributions of this paper are as follows:

- A novel framework that resolves the ambiguity of object reference by integrating the contextual reasoning capabilities of a Large Language Model with the spatial data from pointing gestures.
- A systematic evaluation in complex, realistic environments that validates the superior performance of our multimodal fusion approach when compared to unimodal methodologies.

2 Related Work

Pointing gestures, as one of the most fundamental and intuitive directive modes in human communication, have been the subject of sustained investigation within the field of HCI [2, 8]. For the estimation of pointing direction, we utilized the wrist-elbow vector, the accuracy of which has been established in previous research, and this vector was tracked via a Kinect RGB-D sensor [9]. Despite considerable research, when input is constrained to a single modality such as gestures or speech, the user's expression of intent is often not sufficiently rich, which can lead to confusion in complex scenarios that include multiple objects [11, 3].

To mitigate this, numerous studies have designed multimodal interfaces that combine various input channels. Seminal early research, such as the "Put-That-There" project, demonstrated the robustness of multimodal commands [1]. More recent investigations have enhanced accuracy

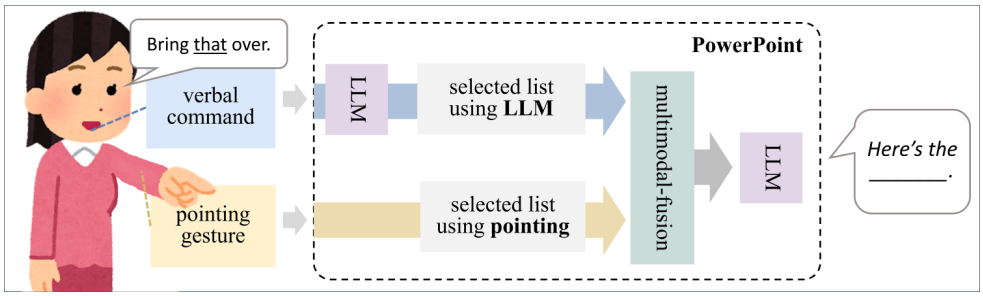

Figure 2: System Overview. The LLM interprets the user's voice command (e.g., "Bring that over") and pointing gesture to determine the correct selection and generate a response.

and usability by integrating gestures and speech in the contexts of virtual reality and industrial robotics [4, 6]. However, these systems frequently rely on rule-based methods that fail in complex, open-ended scenarios, or they require specialized hardware such as AR devices. Our work focuses on the challenge of resolving ambiguity in everyday, complex spaces without these constraints.

Large Language Models have been widely adopted within HCI for their impressive contextual reasoning abilities, enhancing human-robot communication through improved command execution and planning [12, 7]. However, because LLMs are not grounded in the physical world, they lack spatial awareness and risk producing "hallucinated" outputs. Our key contribution, therefore, is to combine the powerful reasoning capabilities of LLMs with real-world spatial data derived from user pointing gestures, thereby constraining their outputs to be both contextually and spatially relevant.

3 Methodology

As illustrated in Figure 2, our system processes a user's verbal command issued concurrently with a pointing gesture. The pointing information is interfaced with a 3D point cloud map to provide spatial context, while the LLM interprets the linguistic context. By integrating these two information streams, the system can identify the target object and generate an appropriate response, even in the presence of imprecise pointing or ambiguous language.

3.1 Pointing Gesture and Object Prioritization

We recognize pointing gestures by tracking the user's 3D skeleton with an RGB-D camera. The vector between the user's wrist and elbow serves as the pointing ray, a method known to provide high accuracy. To account for real-world inaccuracies arising from sensor noise or user posture, we extend this ray into a cylinder. Rather than requiring the ray to intersect the precise center of an object, all objects that collide with this cylinder are considered candidates.

To efficiently identify candidate objects, we employ a standard Axis-Aligned Bounding Box (AABB) collision detection technique on the environment's 3D point cloud map. Objects that collide with the pointing cylinder are prioritized based on their proximity to the central ray. To this end, we utilize five concentric cylinders of incrementally increasing radii. As illustrated in Figure 3, objects are prioritized according to which cylinder they first intersect. An object residing in the innermost zone (Q1) is assigned the highest priority score, followed sequentially

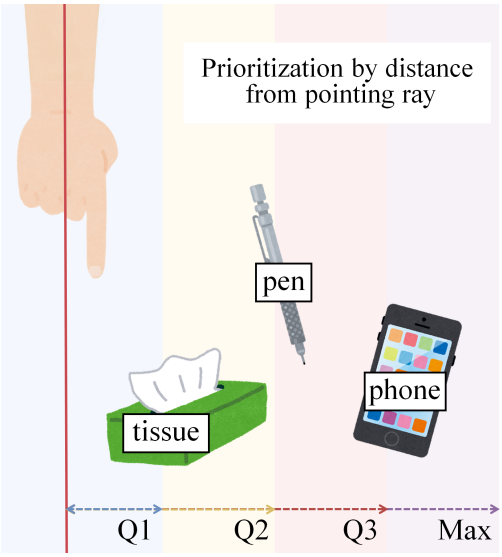

Figure 3: An example of object prioritization via a pointing gesture. Objects are prioritized based on their proximity to the pointing ray: tissue, pen, and phone.

by those in Q2 and Q3. This method facilitates the accurate selection of the intended object despite minor inaccuracies in the pointing gesture. Figure 4 shows the weighting process for the candidate list derived from the pointing gesture. The resulting scores for all candidate objects are then normalized to generate a final set of pointing-based weights.

3.2 Contextual Understanding via LLM

A user's verbal command, such as "Can you pass me this?" is captured and transcribed into text. The LLM then analyzes this text to infer which object is being referenced. For example, in the command "Can you give me that so I can listen to some music?" the LLM can leverage its general knowledge to assign a higher priority to objects like headphones or a mobile phone.

We use prompt engineering to guide the LLM's reasoning. As shown in Figure 5, the prompt provides the user's command, a list of all possible objects in the scene, and a reasoning process for the LLM to follow. The task for the LLM is to identify the most relevant object from the user's command and calculate a normalized relevance weight for each object in the scene. To ensure a stable and predictable output for post-processing, the LLM's response is constrained by a predefined JSON schema, forcing it to return a list of object names and a corresponding list of weights that sum to one.

3.3 Multimodal Fusion

The final stage involves combining the prioritized list of objects from the pointing gesture with the weighted list from the LLM. The fusion algorithm first finds the intersection of the two candidate lists. For each object present in both lists, the weights from the pointing and LLM modules are summed. The object with the highest aggregate weight is selected as the final target. If the intersection is empty (for instance, if the pointing gesture failed to identify any

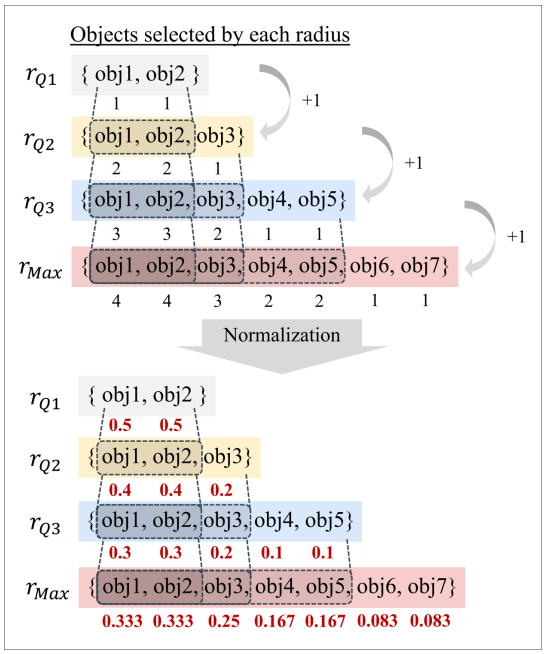

Figure 4: An example of calculating weights for a list selected by a pointing gesture. Scores are assigned based on radius ranges and then normalized to derive the final weights.

System: I will provide a command containing “this” or “that.” You need to infer what object “this” or “that” refers to.
The possible objects are as follows:
Objects = {‘mouse2’, ‘glasses’, ‘laptop1’, ⋯ ,‘headphones1’, ’tumbler2’, ‘pen’}
Respond with consideration of the priority order of the selected objects.
If the context of the sentence is unclear, return all possible matching objects instead of an answer like “unknown”.
Provide a weight for each object in your response. The sum of all weights in the list should equal 1.
The thought process should be as follows:
Command = “How much battery does this have left?”
1. This object must be an electronic device that uses a battery.
2. Among the items, the ones likely to contain a battery are laptop1, laptop2, mouse1, mouse2, phone1, phone2, headphones1, headphones2.
3. Let’s set a priority order, based on how often people tend to check the battery. And based on the priority, let’s assign a weight to each object.
4. Answer = [phone1, phone2, laptop1, laptop2, headphones1, headphones2, mouse1, mouse2], [0.2, 0.2, 0.15, 0.15, 0.1, 0.1, 0.05, 0.05]

Figure 5: An example of the task prompt defined for contextual reasoning. The system utilizes predefined objects, and the LLM learns the provided reasoning process to generate its response.

objects), the system defaults to selecting the object with the highest weight from the LLM’s list, thereby prioritizing the linguistic context in the event of spatial failure.

Once the final object is determined, the system queries the LLM again to generate a detailed, natural language response to the user’s original command. This multimodal approach allows the pointing gesture and the language model to function in a complementary manner, ensuring robust performance even when one modality is noisy or ambiguous.

Figure 6: The laboratory setup and its corresponding 3D point cloud map. A total of 19 objects are positioned on the desk and chair. Objects are as follows: laptops, mouses, headphones, phones, tumblers, tissue, wet wipes, diaries, pen, cushion, blanket, book, glasses

4 Evaluation

A total of 10 participants (3 female, 7 male, aged 23-26) took part in the experiment, which was conducted in a laboratory of approximately $26m^3$. A Kinect sensor was centrally positioned to capture the participants' full bodies from a distance of $1.5m$. Figure 6 shows the laboratory layout. The set of target objects for pointing consisted of 19 items, which were placed in a cluttered arrangement on a desk in the center of the laboratory. To simulate a realistic scenario, duplicate items such as two laptops and two phones were included. CTo simulate a realistic scenario, duplicate items such as two laptops and two phones were included, with each item treated as a distinct target object. To further enhance the naturalistic setting, some objects were positioned such that they overlapped.

A total of 350 unique voice commands were prepared, with 17 to 20 commands allocated to each object. Identical commands were used for objects within the same class, and context-free directives such as "Bring that over" were applied universally to all objects to assess scenarios where target identification is impossible without pointing. In total, we collected 3,500 data pairs. This was achieved by having each of the 10 participants perform one trial for each of the 350 unique command-object prompts. The system was implemented using an Azure Kinect sensor as the RGB-D sensor for body tracking, the Google Cloud Speech API for text transcription, and an external API for the Large Language Model for contextual inference.

4.1 Pointing Gesture and Object Prioritization Evaluation

Radius	R1	R2	R3	R4	R5
k = 1	9.77	11.31	11.97	**12.94**	11.30
k = 2	18.31	22.03	25.94	**26.40**	23.54
k = 3	19.94	25.17	33.34	**34.51**	30.26
k = 5	22.14	29.94	44.91	**52.69**	49.38

Table 1: Accuracy(%) of the pointing gesture for five radius options(R1-R5) using various top-k thresholds. R4 achieved the highest accuracy in all cases.

We first evaluated five different cylinder radius configurations(R1-R5) for the pointing gesture module. As shown in Table 1, the R4 configuration achieved the highest top-k accuracy, indicating it effectively balanced precision with tolerance for error; it was therefore adopted for all subsequent experiments.

- R1: [0.01, 5, 10, 15, 20]
- R2: [0.01, 12.5, 25, 37.5, 50]
- R3: [0.01, 25, 50, 75, 100]
- R4: [0.01, 50, 100, 150, 200]
- R5: [0.01, 125, 250, 375, 500]

4.2 Contextual Understanding via LLM Evaluation

Model	Gemini 1.5 flash	Gemini 2.0 flash	GPT 4o	GPT 4o mini
k = 1	17.71	**20.57**	14.57	17.14
k = 2	30.00	**36.00**	32.57	31.43
k = 3	36.00	**42.57**	39.71	39.71
k = 5	44.00	**52.00**	48.00	47.71

Table 2: Accuracy(%) of different language models under various top-k thresholds. Gemini 2.0 flash achieved the best performance in all cases.

Next, we compared 4 different LLMs for the contextual understanding module: Gemini 1.5 flash, Gemini 2.0 flash, GPT 4o, and GPT 4o mini. According to the results summarized in Table 2, Gemini 2.0 flash demonstrated superior performance across all top-k thresholds among the LLMs tested and was selected as the model for our system due to its robust contextual and commonsense reasoning capabilities.

4.3 Multimodal Fusion Evaluation

Pointing Gesture	Contextual Understanding	Multimodal Fusion
12.94	20.57	**32.83**

Table 3: Comparison of top-1 accuracy(%) for different methods. Our multimodal fusion approach significantly outperforms both unimodal baselines and random chance (5.3%, i.e., 1 out of 19).

Finally, we conducted a comparative evaluation of the complete multimodal fusion system against the individual modalities. The results in Table 3 clearly demonstrate the effectiveness of our approach. The multimodal fusion system (32.83%) accuracy significantly outperformed random chance ($1/19 \approx 5.3\%$), the pointing-only method (12.94%), and the LLM-only method (20.57%), achieving an improvement of more than 12%p over the best single modality. This confirms that the spatial cues from pointing and the contextual inference from language effectively complement each other to resolve user intent in complex environments.

5 Conclusion

In this paper, we have presented a multimodal system that combines pointing gestures with a Large Language Model for robust object referencing within complex indoor environments. By integrating the spatial cues from pointing with the contextual reasoning of an LLM, our system overcomes the limitations inherent in relying on a single modality. Our experiments have confirmed that the fusion of these components yields a substantial increase in accuracy—over 12%p—compared to unimodal baselines, which underscores the complementary nature of spatial and linguistic information. Even when a pointing gesture is misaligned, linguistic reasoning can narrow the potential candidates, while spatial cues serve to disambiguate otherwise vague commands.

The present study is, of course, subject to certain limitations. The participant pool was concentrated within a narrow age range of 23–26, which may limit the generalizability of our results. Additionally, the interaction was confined to the singular gesture of pointing, unlike in naturalistic communication. These limitations offer clear directions for future research. To ensure the generalizability of the findings, subsequent studies should recruit a more diverse participant group, one that includes varied age ranges and levels of technical proficiency. Future research could explore the addition of other modalities, such as gaze tracking, to further refine contextual awareness, or investigate the use of reinforcement learning strategies for continuous performance enhancement. By incorporating a broader range of gestures and contextual cues, it will be possible to develop systems that achieve a significantly more natural and intuitive level of human-computer interaction.

Acknowledgment

This research was supported by the Regional Innovation System & Education(RISE) program through the RISE Center, Gyeongsangnam-do, funded by the Ministry of Education(MOE) and the Gyeongsangnam-do Provincial Government, Republic of Korea. (2025-RISE-16-001)

References

[1] Richard A Bolt. “put-that-there” voice and gesture at the graphics interface. In *Proceedings of the 7th annual conference on Computer graphics and interactive techniques*, pages 262–270, 1980.

[2] Stefan Constantin, Fevziye Irem Eyiokur, Dogucan Yaman, Leonard Bärmann, and Alex Waibel. Multimodal error correction with natural language and pointing gestures. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1976–1986, 2023.

[3] Elias Dritsas, Maria Trigka, Christos Troussas, and Phivos Mylonas. Multimodal interaction, interfaces, and communication: a survey. *Multimodal Technologies and Interaction*, 9(1):6, 2025.

[4] Dan Fang, Jiangwei Chen, Yuliang Jiang, and Guoliang Zhang. A multimodal virtual reality system for switchgear operation training: Integration of dynamic gesture and speech recognition. In *2024 4th International Conference on Artificial Intelligence, Virtual Reality and Visualization*, pages 125–133. IEEE, 2024.

[5] Elisabeth Hildt. What sort of robots do we want to interact with? reflecting on the human side of human-artificial intelligence interaction. *Frontiers in Computer Science*, 3:671012, 2021.

[6] Jaewook Lee, Jun Wang, Elizabeth Brown, Liam Chu, Sebastian S Rodriguez, and Jon E Froehlich. Gazepointar: A context-aware multimodal voice assistant for pronoun disambiguation in wearable augmented reality. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems*, pages 1–20, 2024.

[7] Ruaridh Mon-Williams, Gen Li, Ran Long, Wenqian Du, and Christopher G Lucas. Embodied large language models enable robots to complete complex tasks in unpredictable environments. *Nature Machine Intelligence*, pages 1–10, 2025.

[8] Ángel-Gabriel Salinas-Martínez, Joaquín Cunillé-Rodríguez, Elías Aquino-López, and Angel-Iván García-Moreno. Multimodal human–robot interaction using gestures and speech: A case study for printed circuit board manufacturing. *Journal of Manufacturing and Materials Processing*, 8(6):274, 2024.

[9] Michal Tölgyessy, Martin Dekan, František Duchoň, Jozef Rodina, Peter Hubinskỳ, and L'uboš Chovanec. Foundations of visual linear human–robot interaction via pointing gesture navigation. *International Journal of Social Robotics*, 9(4):509–523, 2017.

[10] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.

[11] Ru Yao, Connie Qun Guan, Elaine R Smolen, Brian MacWhinney, Wanjin Meng, and Laura M Morett. Gesture–speech integration in typical and atypical adolescent readers. *Frontiers in Psychology*, 13:890962, 2022.

[12] Ceng Zhang, Junxin Chen, Jiatong Li, Yanhong Peng, and Zebing Mao. Large language models for human–robot interaction: A review. *Biomimetic Intelligence and Robotics*, 3(4):100131, 2023.