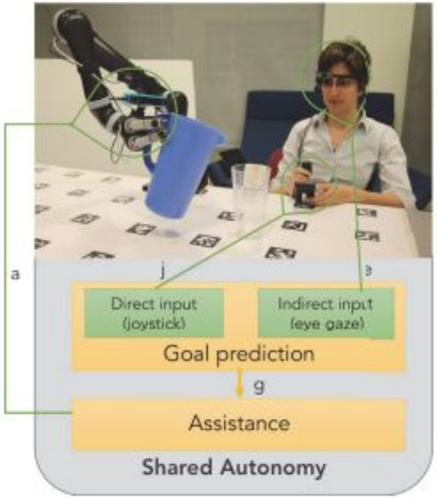
Improving Shared Control using Eye Gaze Estimation

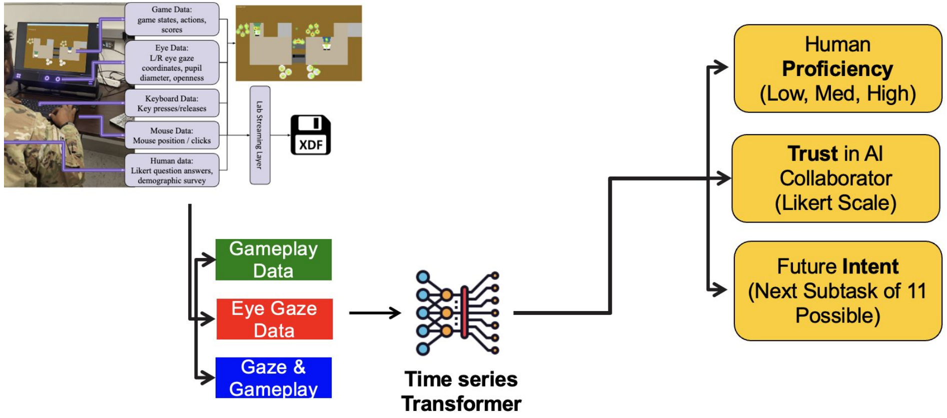
Gyanig, Jake, Yi-Shiuan, Himanshu, Shivendra,
Bradley Hayes, Alessandro Roncone

Motivation & Past Works

Eye Gaze is non-verbal, rich indicator for human intent



Predicting User Intent Through Eye Gaze for Shared Autonomy Henny Admoni, Siddhartha Srinivasa



Game Data: game states, actions, scores

Eye Data: L/R eye gaze coordinates, pupil diameter, openness

Keyboard Data: Key presses/releases

Mouse Data: Mouse position / clicks

Human data: Likert question answers, demographic survey

Lab Streaming Layer

XDF

Gameplay Data

Eye Gaze Data

Gaze & Gameplay

Time series Transformer

Human **Proficiency** (Low, Med, High)

Trust in AI Collaborator (Likert Scale)

Future **Intent** (Next Subtask of 11 Possible)

ROMAN’24 paper - Investigated the use of implicit signals (eye gaze + gameplay data) to model human collaborator

Found that Gaze + Gameplay produced most accurate predictions across the board

Next steps: Apply this to real world HRI task

Notable Prior Work - Opportunity

- The LAMS framework (Tao et al., HRI 2025)
    - LLM-driven automatic mode switching
        - Translates the current task context into a natural language description → prompts LLM for correct mode
        - Requires no task-specific training
    - Limitation: Assumes intended task and goal objects known in advance (Water pouring, book storage)
- Our ROMAN Paper: predict user intent from non-verbal signals

IDEA: Generalized, anticipatory shared control framework that:

- **Passive Intent Inference:**
  Infer user intent continuously from implicit signals such as eye gaze, motion patterns, and hesitation—without requiring explicit commands.
- **Cognitive Reasoning via VLM/LLM:**
  Use a vision–language/large-language reasoning module to interpret scene context, predict task goals, and adapt assistance for complex, real-world interactions.
- **Adaptive Autonomy Levels:**
  Seamlessly shift between teleoperation, shared control, and full autonomy based on user behavior, task demands, and inferred confidence.
- **Gaze-Based Success vs Failure Detection:**
  Identify intent success through stable, goal-directed gaze patterns and detect failures through hesitation, regressions, and prolonged search behavior.

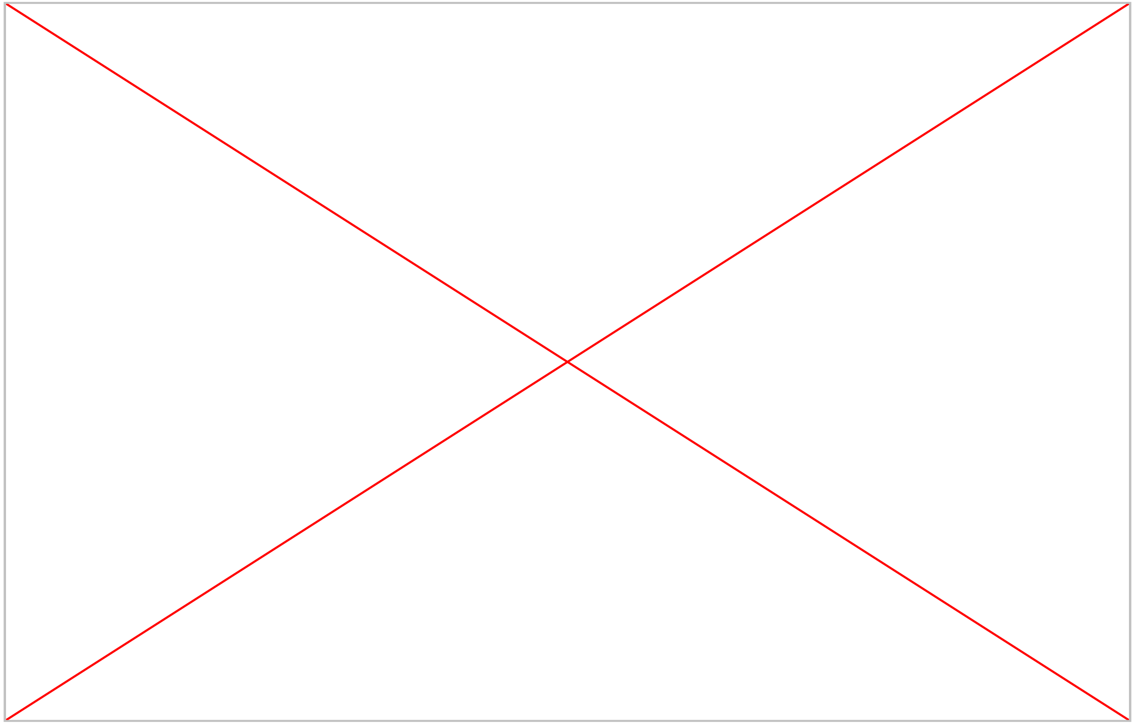
Passive Intent Inference (Our Current Demo)

Task:

Infer human intent by generating heatmap representations of eye-gaze behavior(shown in the left scene view) and using them to predict the user’s object of interest(shown on the right bottom in the video).

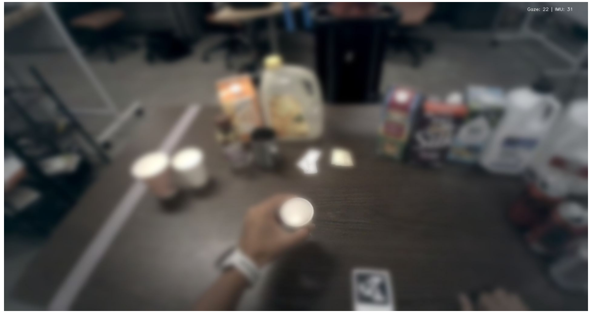
Implementation Details:

A Tobii Glasses 3–based Python pipeline sends gaze-annotated images to the OpenAI o4 VLM, where a task-specific prompt elicits intent predictions for objects in the scene.

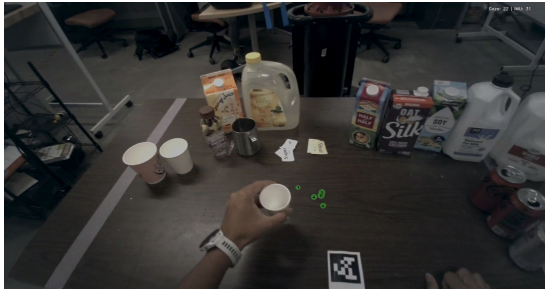


Experiments 1

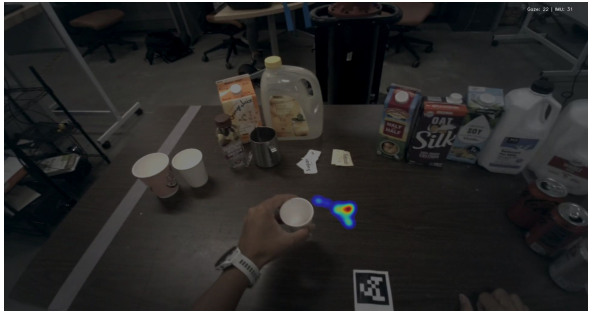
Raw



Foveated



Fixation



Heatmap

RQ: How do visual representations of eye gaze impact a (non-fine-tuned) VLMs prediction of user intent?

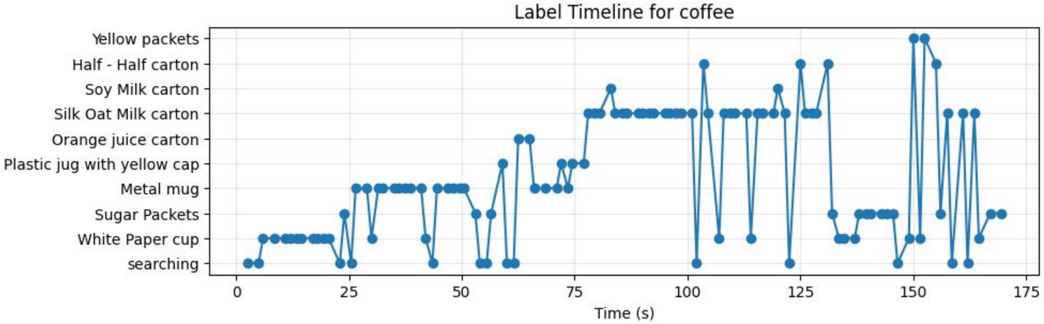
Experiment:

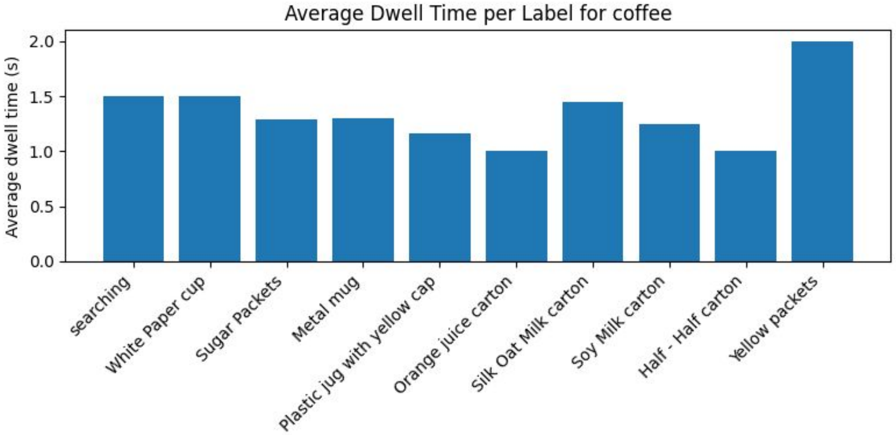
- Perform a mock barista task while wearing Tobii glasses
- Label frames of video w/ GT user intent
- Compare VLM predictions across common gaze reps + params

Ground-truth analysis of gaze behavior during the mock coffee-making task

Top: A timeline visualization showing how participants' gaze transitioned across task-relevant objects **over time**, revealing phases of **active searching, object verification, and focused attention** on specific ingredients (e.g., milk cartons, sugar packets, cups). Stable clusters **indicate moments of intent clarity**, while rapid switching **reflects uncertainty or exploration**.

Bottom: Average dwell time per label, highlighting which objects attracted longer or repeated fixations. Higher dwell durations (e.g., yellow packets) suggest **increased cognitive load, search difficulty, or deeper inspection**, whereas shorter durations indicate **quick recognition or low ambiguity**.

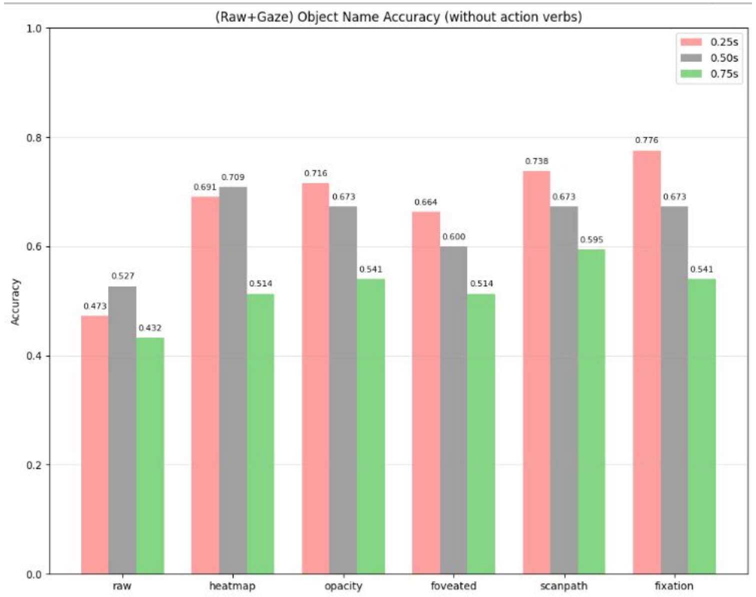




We tested the VLM across different gaze–scene timing windows, using individually tailored prompts with timestamps aligned to a 250 to 750 ms sampling rate.

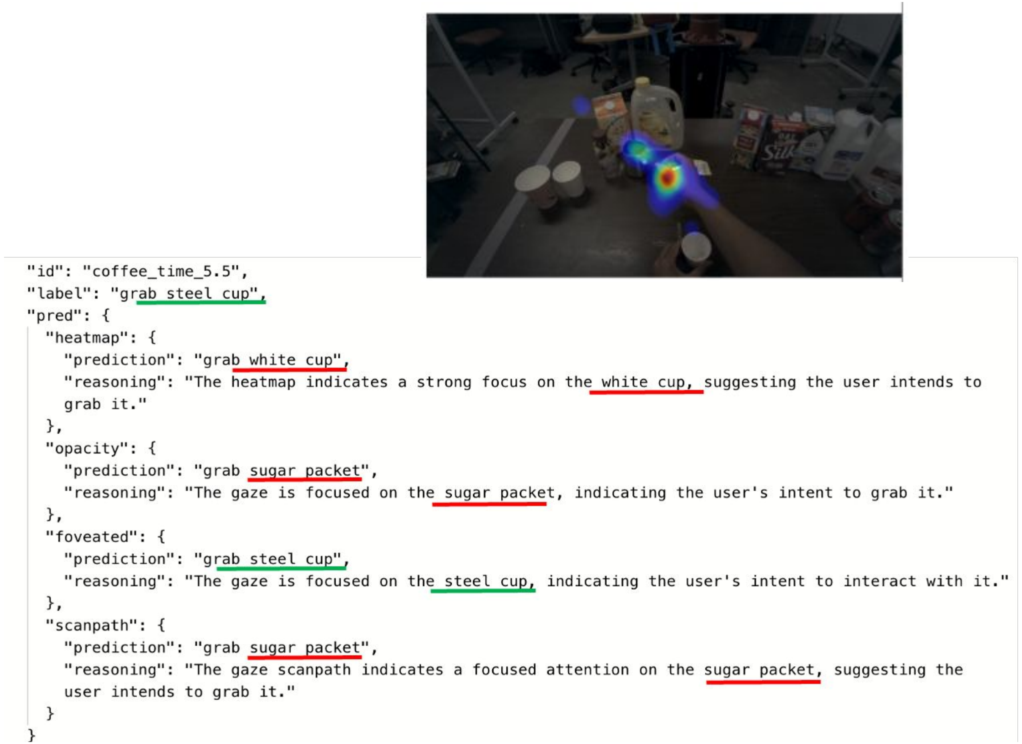
Gaze-enhanced inputs consistently improve VLM accuracy, and **longer timing windows (750 ms)** give the best performance.

- Tasks— **object-only inference**— representations that embed gaze (heatmap, opacity, foveated, scanpath, fixation) outperform raw scene images.
- **Scanpath and fixation** give the **highest accuracy**, especially in the object-only condition (up to ~0.70–0.80 range).
- **Raw frames perform the worst**, showing the VLM struggles when gaze cues are removed.
- Increasing the sampling window from **250** → **500** → **750 ms** steadily improves accuracy across all gaze representations.
- **VLM cannot reliably combine the action (“grab”) with the correct object, even when gaze clearly indicates the target.** The system latches onto salient regions but fails to interpret the *action–object pair* coherently.



Examples of each task written in two forms:
With action verb → *grab the pink cup,*
Without action verb → *pink cup(Above)*

The VLM output shows struggles to jointly infer both the human's intended action and the target object.



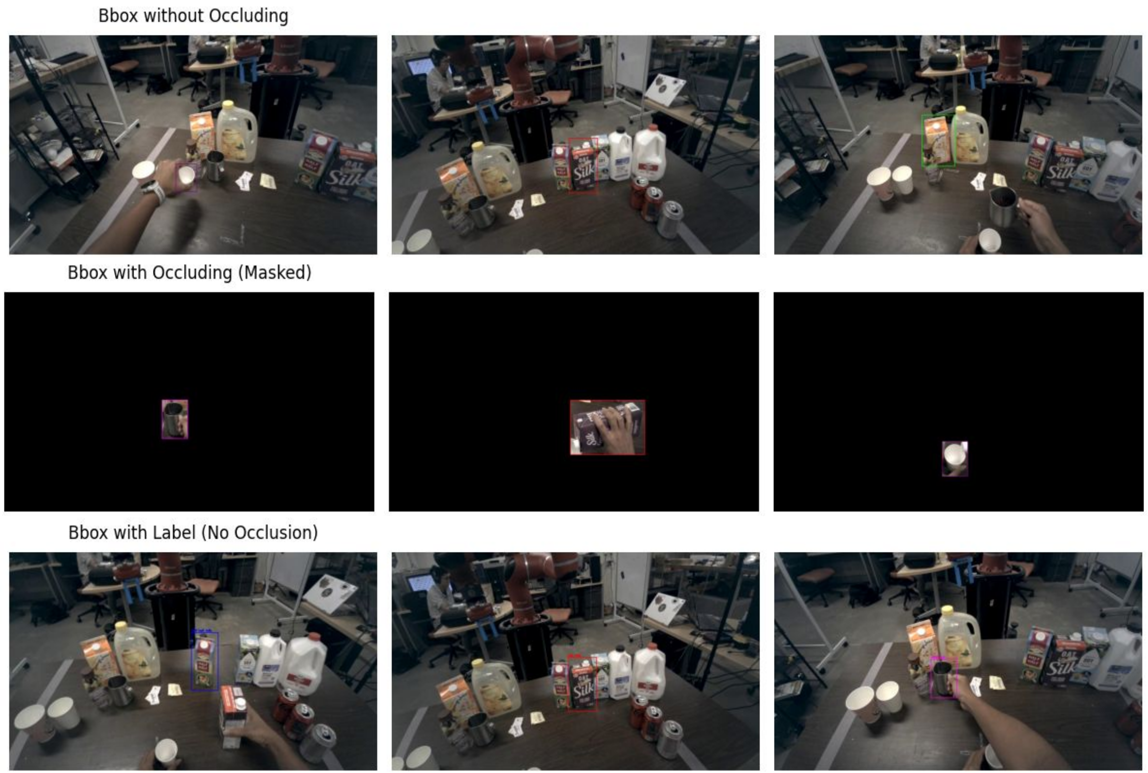
```
"id": "coffee_time_5.5",
"label": "grab steel cup",
"pred": {
  "heatmap": {
    "prediction": "grab white cup",
    "reasoning": "The heatmap indicates a strong focus on the white cup, suggesting the user intends to
    grab it."
  },
  "opacity": {
    "prediction": "grab sugar packet",
    "reasoning": "The gaze is focused on the sugar packet, indicating the user's intent to grab it."
  },
  "foveated": {
    "prediction": "grab steel cup",
    "reasoning": "The gaze is focused on the steel cup, indicating the user's intent to interact with it."
  },
  "scanpath": {
    "prediction": "grab sugar packet",
    "reasoning": "The gaze scanpath indicates a focused attention on the sugar packet, suggesting the
    user intends to grab it."
  }
}
```

The example reveals that **different gaze representations lead the VLM to make inconsistent and often incorrect intent predictions**, even when the ground-truth label is clear (*"grab steel cup"*).

- Heatmap predicts **white cup**
- Opacity predicts **sugar packet**
- Foveated gets it right: **steel cup**
- Scanpath again predicts **sugar packet**

This inconsistency shows a systematic limitation: **the VLM cannot reliably combine the action ("grab") with the correct object, even when gaze clearly indicates the target.** The system latches onto salient regions but fails to interpret the *action–object pair* coherently.

Experiments 2



Bbox without Occluding

Bbox with Occluding (Masked)

Bbox with Label (No Occlusion)

RQ: How well do standard object detection models perform on this setup, and what upper bound do they provide for VLM performance?

Experiment:

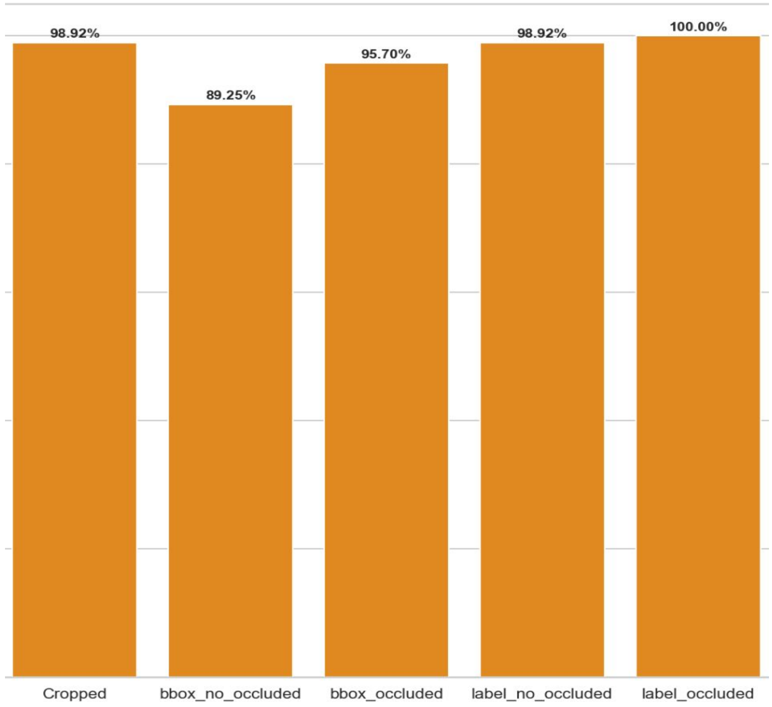
- Use the Mock Barista Task dataset,
- manually label all task-relevant objects with bounding boxes,
- evaluate VLM on this annotated data.

This plot shows the **maximum achievable accuracy** when the VLM is given *idealized inputs* such as perfectly cropped objects or bounding boxes (occluded or not).

Performance is extremely high (≈ 89–100%), demonstrating that **when the object is explicitly isolated**, the VLM can reliably identify it.

This acts as an **upper bound**: it reveals how well the model *could* perform if it had perfect perception of what the user is looking at.

The large gap between the Gaze and Object Detection Plots indicates that **the main challenge is not object recognition**, but rather **interpreting gaze, resolving ambiguity, and inferring intent from human visual behavior**.



Future Works

1. Conduct controlled experiments combining eye-gaze tracking and human teleoperation that distinguish intent success from failure by analyzing stable fixations versus hesitation and search-driven gaze patterns.

2. Understanding when the user's gaze reflects *confident intent* versus *struggle or indecision* allows the robot to **dynamically adjust its autonomy level**:
   a. **Intent Success → Higher confidence → Robot can increase assistance** (e.g., shift from teleop → shared control → partial autonomy)
   b. **Intent Failure → Low confidence → Robot should reduce autonomy** (e.g., give more user control, pause, request clarification)
   c. **Uncertain or noisy gaze → Maintain or lower autonomy** (avoids incorrect autonomous actions)

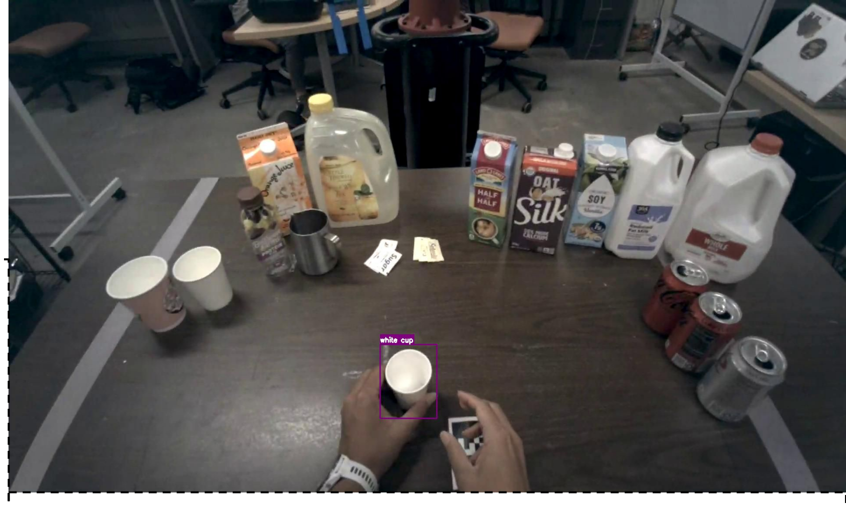
3. Other Areas for improvement - **Strengthening VLM inference through modular action–object prompting, Passing Gaze history for improving intent, Set-of-marks prompting, Gaze-retargeting strategies, and Synthetic fine-tuning of VLM model**.

Prompting for Scene Image + BoundingBox Image

GPT-40 model
Image resolution : 1920x1080
Average inference time :
Sampling rate : 0.25s

Sample prompt setup:

"<bbox_no_mask, bbox_mask, bbox_label, bbox_cropped>": “
You are an expert AI assistant specializing in human intent recognition. Your primary mission is to analyze a visual scene with bounding box annotations to predict which object the person is most likely to interact with next.
Your Input:
- (Optional) Image: An RGB image showing the <corresponding> raw scene
- Image: An RGB image showing the scene with <bounding box outlines drawn around objects>, indicating where the user's attention is focused.
- Candidate Objects: A predefined list of possible objects in the scene.
Your Task & Rules:
- Carefully analyze the image to identify which object the user is focusing on, using the bounding box outlines as the primary clue to their intent.
- From the provided candidates list, select the single most probable object they intend to interact with.
~~- If multiple bounding boxes are present without a clear primary focus, or no bounding boxes are drawn, it indicates the user is still observing or planning. In this case, you must ignore the object list and select the special candidate: "observing".~~
- Your final output must be a JSON object containing two keys:
  - "prediction": The string of the object you selected from the Candidate list (or "observing").
  - "reasoning": A brief, one-sentence explanation of why you made your choice, specifically referencing the bounding box annotations.
- You must output ONLY a valid JSON object, with no extra text, no explanations, and no code fences.
- Always return your response as a valid JSON string, e.g., {"prediction": "object", "reasoning": "explanation"}.
“



Candidates updated based on the GPT prompt from previous case on ground truth labels :

1. White Paper cup
2. Pink Paper cup
3. Metal mug
4. Sugar Packets
5. Yellow packets
6. Orange juice carton
7. Plastic jug with yellow cap
8. Half - Half carton
9. Silk Oat Milk carton
10. Oat Milk carton
11. Soy Milk carton
12. Milk jug with black cap
13. Milk jug with red cap
14. Coca-Cola can
15. Diet Coke can

Prompting for Scene Image + Gaze modality

VLM input

```
resp = client.chat.completions.create(
            model=model,
            temperature=TEMPERATURE,
            max_tokens=MAX_TOKENS,
            messages=[
               {"role": "system", "content": system_prompt},
               {
                  "role": "user",
                  "content": [
                     {"type": "text", "text": user_text},
                              {"type": "image_url", "image_url": {"url": f"data:image/png;raw64,{raw_b64}"}},
                     {"type": "image_url", "image_url": {"url": f"data:image/png;base64,{gaze_b64}"}},
                  ],
               },
            ],
         )
```

Sample User template for individual modalities

"heatmap": """
Analyze the provided scene and gaze data to predict the user's intended action.
Gaze Representation Type: Heatmap - This visualization shows where the user looked the longest or most frequently. Bright/hot areas (red, yellow, white) indicate high attention, while cooler areas (blue, green) indicate less attention.
[Image 1: Scene RGB]
[Image 2: Heatmap Visualization]
Candidate Actions:
{{
  "actions": {candidates}
}}
Provide your prediction in the required JSON format.
"""

Sample System template for individual modalities

"heatmap": """
You are an expert AI assistant specializing in human intent recognition. Your primary mission is to analyze a visual scene and its corresponding eye-gaze data to predict the most likely action a person will take next.
Your Inputs:
- Scene Image: An RGB image showing the environment and objects.
- Gaze Visualization Image: A heatmap visualization that shows where the user looked the longest or most frequently. Bright/hot areas (typically red, yellow, or white) indicate high attention, while cooler areas (blue, green) indicate less attention.
- Candidate Actions: A predefined list of possible actions.
Your Task & Rules:
- Carefully analyze both the scene and the heatmap to understand what the user is focusing on. The brightest/hottest areas in the heatmap are your primary clue to their intent.
- From the provided candidates list, select the single most probable action.
- If the heatmap shows broad, scattered attention across multiple objects without clear hotspots, or appears to be a general survey of the scene, it indicates the user is still observing or planning. In this case, you must ignore the action list and select the special candidate: "observing".
- Your final output must be a JSON object containing two keys:
  - "prediction": The string of the action you selected from the list (or "searching").
  - "reasoning": A brief, one-sentence explanation of why you made your choice, specifically referencing the heatmap data.
- Always return your response as a valid JSON string, e.g., {"prediction": "action", "reasoning": "explanation"}.
"""