

SeeIntent: A Real-Time Multimodal Intent Recognition System in XR Using Spatio-Temporal LLM^{*}

Gathik Jindal^{[0009–0005–1999–920X]**}, Jayant Sharma^{[0009–0001–1006–4492]**},
Beryl Gnanaraj^[0000–0003–0314–5929], and
Jaya Sreevalsan-Nair^{[0000–0001–6333–4161]***}

¹Graphics-Visualization-Computing Lab (GVCL), IIIT Bangalore (IIIT-B),
26/C, Electronics City, Karnataka 560100, India
<http://www.iiitb.ac.in/gvcl>

Abstract. Recognizing user intent in Mixed Reality (MR) environments is essential for HCI and assistive technologies, where human intent drives adaptive systems. However, it remains a challenging open problem due to the multimodal and highly dynamic nature of such interactions. We present SeeIntent, a novel end-to-end real-time multimodal emotionally-aware intent recognition system from an egocentric user perspective, which fuses gaze descriptors and head rotation, streamed from the XR headset, and the object-level visual context, emotional state, and spatial focus, derived using the Facial Action Coding System. The aggregated state is encoded into a rolling temporal history and submitted to a locally deployed open-weight Large Language Model (LLM) backend for action-intent generation. We introduce a temporal context-window architecture, the Intent Recognition Module (IRM), to maintain a history of multimodal state vectors for conditioning the LLM inference. This integrates with the spatial context of local surroundings from the gaze data. Our framework offers a replicable, generalized methodology for the cognitive analysis of user behavior in First Person Vision research. To validate the proposed egocentric emotion context, we introduce a novel annotated pilot dataset collected under naturalistic, in-the-wild conditions, providing ground-truth labels for scene-level objects, fixation targets, egocentric emotional state, and intended action. Evaluation on EPIC-KITCHENS and our pilot dataset shows that while the higher-parameter LLM models achieve the highest semantic matching accuracy, smaller model variants offer competitive performance at lower inference latency (under 300 ms), making them better suited for real-time interactive deployment.

Keywords: Mixed Reality · Egocentric Behavior · Intent Recognition · Action Recognition · Spatio-temporal Context · Large Language Models · Eye Tracking · Object Detection · Emotion Classification

^{*} Supported by Machine Intelligence & Robotics Center (MINRO) grant by the Government of Karnataka and the E-Health Research Center (EHRC).

^{**} Both authors contributed equally to this work.

^{***} Corresponding author: jnair@iiitb.ac.in

1 Introduction

Mixed Reality (MR) systems are rapidly evolving from passive display technologies into active, context-aware assistants capable of understanding and anticipating user needs [3]. A key enabler of proactive MR assistance is accurate, low-latency *intent recognition*, i.e., determining what action a user is about to perform, based on available contextual information. Achieving this requires the system to simultaneously interpret what the user sees, how they feel, and where their attention is directed — signals that are heterogeneous in modality, asynchronous in arrival, and temporally correlated in meaning.

Prior work on intent and action recognition has largely focused on unimodal cues: egocentric vision [6, 12], physiological signals [19], or gaze [1]. Multimodal fusion approaches have emerged more recently but typically operate offline on pre-segmented clips, rather than in real-time [16]. Action recognition has also been rephrased as a language generation problem over noun and verb phrases [23]. With the advent of Large Language Models (LLMs), natural language reasoning over real-time multimodal data becomes tractable without the need for task-specific classifiers.

Gaze provides the spatial context of the user’s surroundings. To contextualize it temporally, we track the sequence of objects to which the user directs their attention over time, maintaining a chronological record within a historical buffer. Together, the spatial and temporal dimensions of gaze constitute the primary contextual signal from which intent is inferred. Our contributions are as follows:

- A modularized methodology to capture, process, and analyze multimodal XR data streams for action and intent prediction.
- A proof-of-concept end-to-end egocentric system, SeeIntent, that performs real-time intent recognition using multimodal contextual information.
- A pilot dataset for system validation, comprising annotations for egocentric emotion, fixation target, scene-level objects, and intent.

We also present an evaluation of five open-weight, locally deployable LLM configurations for processing the spatio-temporal context.

2 Related Work

Egocentric video understanding has advanced significantly with large-scale datasets such as EPIC-KITCHENS [6], Ego4D [12], egoEmotion [15], EGTEA-Gaze+ [18], and EgoIntrospect [22]. Transformer architectures [16] now achieve strong performance on third-person action recognition, and recent work extends these to anticipation tasks that predict the *next* action from a short observation window [11]. However, these approaches are computationally heavy and require offline processing, making them unsuitable for real-time MR deployment.

Gaze is a strong predictor of imminent interaction [7]. Modern commercial mixed-reality headsets natively expose per-frame head rotation quaternions and eye-tracking vectors, enabling fine-grained attention tracking without external

hardware. Several studies fuse gaze with scene understanding for object selection in VR [7], but the integration of gaze into intent-recognition pipelines that concurrently incorporate egocentric user emotion, scene-level objects, and language remains largely unexplored.

Beyond gaze, Facial Action Coding System (FACS)-derived blendshapes provide a compact, hardware-agnostic representation of facial expressions [8]. Public datasets, *e.g.*, *emoji-hero-vr-db* [9], include labeled VR-users’ blendshape recordings for training emotion classifiers without external cameras. While emotion recognition from raw blendshapes is studied [20], there are gaps in the integration of the resulting affective states in downstream intent-recognition tasks.

Instruction-tuned LLMs, meanwhile, demonstrate strong zero-shot reasoning over structured context [4, 21]. Recent systems exploit LLMs as planners or intent interpreters in embodied AI settings [2], but these typically operate on clean, pre-parsed inputs. Integrating noisy, high-frequency sensor streams into LLM prompts, while strictly managing latency, remains an open challenge. Recent advances in local inference frameworks make sub-second LLM calls feasible on commodity hardware, yet a systematic evaluation of open-weight models for structured intent generation in MR is needed.

Late-fusion aggregation of unimodal predictions is effective for asynchronous modalities [16]. SeeIntent adopts a text-serialization approach where each modality is converted into a natural-language token (*emotion label*, *object list*, *gaze fixation target*), concatenated into a structured temporal prompt, and reasoned over by the LLM. This design bypasses the need for computationally expensive cross-modal alignment layers while directly exploiting the LLM’s pre-trained commonsense knowledge. Our architecture trades long-range predictive depth for computationally lightweight, real-time inference expected in wearables.

3 Methodology: System Design of SeeIntent

SeeIntent aggregates various camera feeds and sensor inputs from commercially available XR headsets for the task of intent recognition, as shown in Figure 1. At each time step, the system captures multimodal camera and sensor feeds and processes them using object detection, fixation target estimation, and emotion classification modules in parallel. The combined observation is encoded into a state vector, which is appended to a rolling history, while simultaneously triggering an asynchronous LLM query for intent recognition. The modular design of the architecture allows the selection of optimal models for each task.

3.1 Telemetry

Telemetry aids in time-synchronizing and centralized processing of multimodal data. A threaded UDP server captures the user’s movements and facial expressions in real time by listening for JSON packets from a real-time 3D engine client running on the commercial MR headset. UDP is chosen over TCP to eliminate

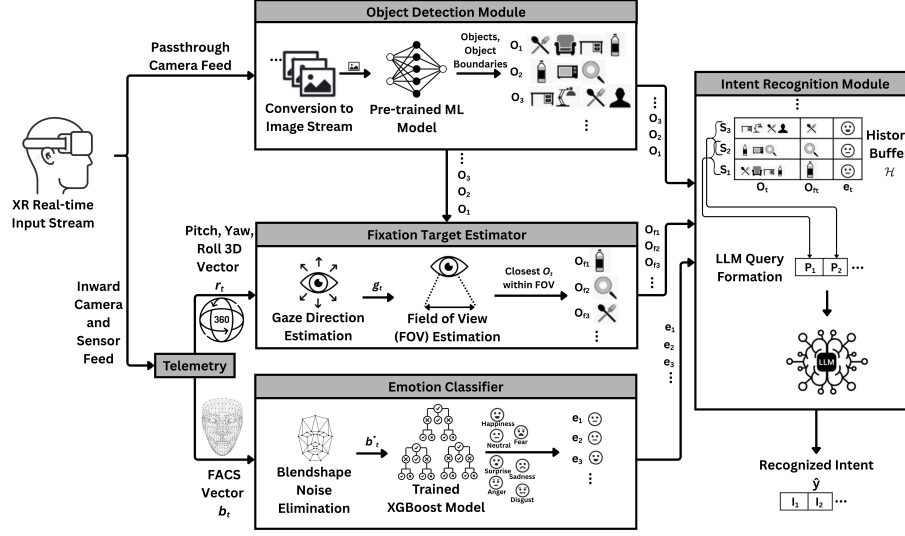


Fig. 1: End-to-end architecture of SeeIntent with five modules has various camera and sensor feeds as the input, and the egocentric user’s intent as the output.

handshake overhead and prevent head-of-line blocking [17], prioritizing real-time state updates and low latency over guaranteed packet delivery.

Head rotation is tracked as a 3D vector from the headset’s cameras. The inward camera gives the *Facial Expression Activation Code (FACS)* values for each facial muscle to compute the FACS vector. Thus, at each timestep t , the payload computed from the eye and lower-face tracking camera feed contains:

- A 3-DOF head *rotation vector*, $\mathbf{r}_t \in \mathbb{R}^3$, with pitch, yaw, and roll in degrees.
- A 63-dimensional *FACS vector*, $\mathbf{b}_t \in [0, 1]^{63}$. These blendshapes track specific facial muscle movements (e.g., a jaw drop, an eyebrow-raise) on a 0 (neutral) to 1 (fully flexed) scale.

3.2 Emotion Classifier

We propose a novel integration of the egocentric user’s emotion into the traditional intent recognition pipeline [7, 10], which utilizes multimodal data consisting of gaze and scene perception. While recent efforts have shown that the egocentric user’s emotion reveals user behavior and personality [15], we use emotion as an input modality for intent recognition. As this research area is nascent [22] and available information is access-controlled, we use freely available public datasets and repositories to propose suitable models that fit the device’s hardware constraints. Since the inward camera feed is access-restricted due to privacy policies on most headsets, we use the XR headset’s inward camera to provide a FACS vector. Due to the resource constraints of locally running

ML models, we allocate the CPU to the FACS-based emotion detection and the GPU to the object detection (Section 3.3) modules.

Through empirical testing, we selected an *Extreme Gradient Boosting (XGBoost)* model [5], pre-trained on a suitable Facial Emotion Recognition (FER) dataset using FACS coding. XGBoost is apt owing to its high efficiency on tabular data and its ability to learn complex patterns across facial expressions without significant computational overhead [13].

Dimensionality Reduction: The 63D FACS vector poses a computational burden on the already resource-constrained XR headset; hence, we propose a feature selection step to eliminate noisy data and any blendshape data occluded due to camera position. The latency incurred by feature selection is a trade-off for the faster performance of the XGBoost model using a smaller subset b_t^* of the high-dimensional FACS vector.

We propose a two-step feature selection approach to determine the optimal subset $b_t^* \in b_t$ to reduce inter-class misclassification. High-dimensional facial tracking data often contains overlapping signals and low discriminatory features between closely related classes, leading to misclassification, *e.g.*, ‘Fear’ and ‘Surprise’. The top- k features are identified using the information gain metric in XGBoost. In the first step, an exploratory global classifier is trained across all emotion classes to extract the top-20 most influential blendshapes, capturing the primary variations necessary for generalized classification. Second, we isolate the data for known related classes, which are ‘Fear’ and ‘Surprise’ in our case, and train a targeted binary classifier. This extracts an additional top-10 features specifically optimized to disambiguate expressions that are narrowly separated. The final selected subset b_t^* is the union set of these two feature sets. This feature set ensures the selection of both the macro-level features required for general emotion classification and the micro-level nuances necessary for edge cases. Our model then takes the corresponding b_t^* reliable blendshapes and classifies them into one of seven emotions defined by FACS [8], such that $e_t = \text{XGB}(\mathbf{b}_t^*)$ with: $e_t \in \{\text{‘Fear’}, \text{‘Happiness’}, \text{‘Anger’}, \text{‘Disgust’}, \text{‘Neutral’}, \text{‘Sadness’}, \text{‘Surprise’}\}$.

3.3 Object Detection Module (ODM)

For real-time scene understanding, a lightweight object detection model is deployed locally on available GPU hardware. The module receives sequential frames from the egocentric passthrough camera feed and produces an array of object class labels $\mathcal{O}_t = \{o_1, o_2, \dots, o_k\}$ for each timestep t , corresponding to entities present in the observed scene. A propagation latency of approximately 20–40 ms is expected from the passthrough camera pipeline for most commercially available XR headsets. However, the GPU-accelerated inference ensures that model execution latency is low for real-time interactivity.

3.4 Fixation Target Estimator (FTE)

The module uses the scene-level object array, along with the objects’ bounding boxes and depth, from ODM and the head rotation sensor output from Telemetry

to predict the probable gaze target. The discrete gaze direction g_t is obtained from the yaw and pitch components of $\mathbf{r}_t$, of the telemetry data:

$$g_t = \text{encode}(\text{pitch}_t, \text{yaw}_t), \text{ where } g_t \in \{\text{'Centre'}, \text{'Left'}, \text{'Right'}, \text{'Up'}, \text{'Down'}\}.$$

Since foveated rendering [14] is used in XR devices, the ‘Centre’ of the world-view is considered the primary gaze vector, with thresholds for the visual field as reported in the literature. Thresholds are $\pm 15^\circ$ for the horizontal axis and $\pm 10^\circ$ for the vertical axis, which reflect natural head-movement granularity in MR tasks. From the scene-level objects $\mathcal{O}_t$, the fixation target is computed as $o_f \in \mathcal{O}_t$ whose bounding box intersects with the visual field. In the case of multiple objects present within the field of view, the object with the lowest depth, *i.e.*, directly in front of the user, is designated as the fixation target.

3.5 Intent Recognition Module (IRM)

The IRM maintains a fixed-length history buffer of w state vectors:

$$\mathcal{H} = [\mathbf{s}_{t-w}, \dots, \mathbf{s}_t], \text{ where } \mathbf{s}_t = (e_t, \mathcal{O}_t, o_{f_t}).$$

When an LLM query is triggered, the system serializes $\mathcal{H}$ into a structured natural-language prompt using Algorithm 1. The prompt instructs the model to predict the user’s immediate next action as a single verb-noun phrase. Inference is dispatched asynchronously to either a local or cloud-based LLM backend, preventing the prediction loop from blocking frame captures.

Standard text-instructed LLMs struggle to natively reason over streaming, high-dimensional floating-point arrays without specialized projection layers. Furthermore, representing raw FACS in text drastically increases the token count, violating our strict sub-300 ms latency constraint. By utilizing the upstream XGBoost classifier to collapse the facial state into a semantic token (*e.g.*, ‘Happiness’ or ‘Hostile’), we temporally synchronize the user’s affective state with their spatial fixation and provide the LLM with a highly token-efficient contextual cue. This allows the LLM to differentiate intent in physically identical scenes (*e.g.*, fixating on a tool while ‘Neutral’ versus ‘Hostile’). Hence, we serialize the discrete emotion category e_t instead of the raw 63D-FACS vector b_t .

Algorithm 1: Prompt construction in the Intent Recognition Module.

Input: History $\mathcal{H}$, window size w
Output: LLM intent prediction $\hat{y}_t$

```

1  $P \leftarrow$  "You are an intent predictor in a Mixed Reality scene. ";
2 for  $i = t - w$  to  $t$  do
3    $P \leftarrow P +$  "Step  $i$ : Emotion= $e_i$ , Objects= $\mathcal{O}_i$ , Fixation Target= $o_{f_i}$ . ";
4 end
5  $P \leftarrow P +$  "Predict the next action as a verb-noun phrase." ;
6  $\hat{y}_t \leftarrow \text{LLM}(P)$ ;
```

3.6 System Integration

In most XR headset configurations, the passthrough camera and display operate at 60-90 Hz, while the eye-tracking camera runs at a lower frame rate of approximately 30-60 FPS. We also consider this system for casting to a remote monitoring device, which may have lower refresh rates. To ensure reliable data synchronization across these asynchronous streams, we run the main orchestration loop by sampling at the lowest sampling rate among all active devices.

At this baseline sampling rate, the multimodal observations are continuously captured and stored in our rolling history window. While the LLM inference is inherently the slowest process in our pipeline, this latency does not slacken our data collection, which is independently populating the sampling window with the streaming telemetry, object data, and emotion states. To maximize predictive throughput without dropping frames, we query the LLM continuously in an asynchronous loop. As soon as the backend finishes processing one intent prediction query, the next query is immediately dispatched using the most up-to-date state from the rolling history buffer.

3.7 Dataset for Evaluation

Currently, no dataset in the public domain contains egocentric user emotion, gaze information, scene-level objects, and user actions or intent. To validate the full multimodal pipeline as a proof-of-concept, we generated a small-scale dataset from a pilot study, recording four video segments from a Meta Quest Pro headset worn by the two primary authors, performing naturalistic daily tasks.

Annotation: Each clip is about one minute long, and there were four clips in total. Each clip depicted one continuous set of logical actions, culminating in five to six intents per clip. Ground-truth action labels are manually annotated by the authors, specifying the intent as a verb–noun pair for specific temporal segments of the video, based on the duration of distinct atomic actions (*e.g.*, “grabbing a notebook,” “cutting the paper,” etc.). For each temporal segment, annotators provide a standardised verb–noun pair (*e.g.*, verb: ‘grab’, noun: ‘notebook’), the duration in seconds, and a self-annotated predominant facial expression selected from the FACS categories.

Unlike other intent recognition evaluation datasets [6, 15], full multimodal state vectors (emotion and objects) are provided for each segment. Consequently, each discrete time window t is represented by a unified ground-truth state vector:

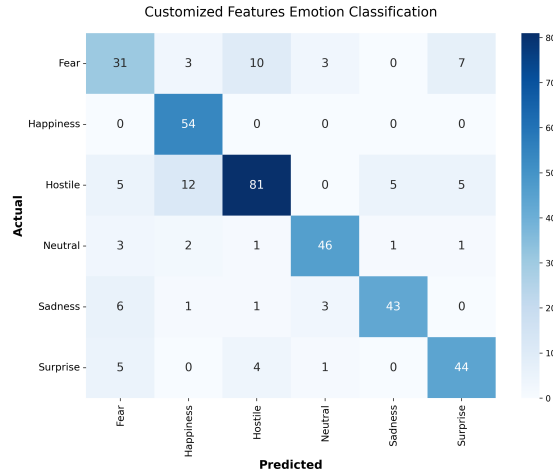
$$\mathbf{v}_t = [\text{emotion}_t, \text{duration}_t, \text{verb}_t, \text{noun}_t].$$

3.8 Evaluation Metrics

Since we annotated our own emotion dataset for validation, we use classification metrics, namely, F1-score, accuracy, precision, and recall, to measure the performance of the emotion classifier. The object detector is not separately evaluated as pre-trained models are used. We evaluate the IRM using metrics that cover lexical quality (BLEU-1, ROUGE-L, METEOR), task-specific taxonomy (noun- and verb-matches), hallucination rate, and time efficiency (latency in seconds).

Table 1: Performance of Emotion Classifier across datasets.

Dataset	Evaluation	Accuracy↑	Precision↑	Recall↑	F1↑
Emoji-Hero	Offline Baseline	78%	0.79	0.78	0.78
VR DB	(6 target emotions)				
Pilot Study	Frame-Level	73%	1.00	0.73	0.84
	(Raw millisecond snapshots)				
Pilot Study	Clip-Level	86%	1.00	0.86	0.93
	(Majority vote per action)				

**Fig. 2:** Confusion matrix for the optimized 22-feature XGBoost emotion classifier. The matrix highlights high accuracy in detecting ‘Happiness’, ‘Neutral’, and ‘Hostile’, while revealing the hardware-induced overlap between ‘Fear’ and ‘Surprise’.

4 Experiments and Results

We empirically tested various ML models for object detection, emotion classification, and various lightweight LLM models. We report on suitable evaluation metrics at each step. Our experiments are performed on two datasets, namely, the EPIC-KITCHENS [6], an egocentric video dataset of kitchen activities, and our own pilot dataset that contains everyday desk-based activities. The code for the system implementation and our complete annotated dataset, including the JSON-formatted ground-truth state vectors, is available at <https://github.com/GVCL/EgocentricIntentPrediction>.

4.1 Module-Level Evaluation

Before evaluating the end-to-end intent prediction pipeline, we validated the optimized 22-feature XGBoost emotion classifier, where ‘Hostile’ combines labels ‘Anger’ and ‘Disgust’:

$$e_t = \text{XGB}(\mathbf{b}_t^*) \in \{\text{‘Fear’}, \text{‘Happiness’}, \text{‘Hostile’}, \text{‘Neutral’}, \text{‘Sadness’}, \text{‘Surprise’}\}.$$

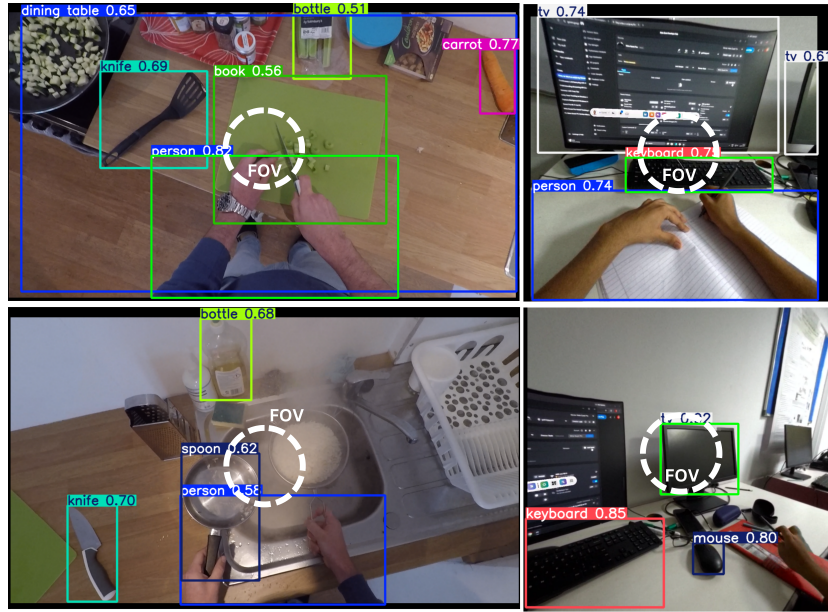


Fig. 3: Real-time bounding-box visualizations, along with “field-of-view” (FOV) markings, generated by the object detection module and fixation target estimator for two sample frames of (Left) EPIC-KITCHENS [6], and (Right) our pilot datasets. The frames show objects detected in the scene, along with the object detection model’s confidence estimates and the FOV indicated by the white dashed circle.

We tested the emotion classifier offline with the Emoji-Hero VR database as a baseline and then in the wild against our streaming pilot study dataset. As the raw facial tracking in VR is susceptible to physical artifacts (*e.g.*, headset shifting or squinting), we evaluated the pilot data at both the instantaneous frame-level and with a temporal smoothing technique (taking the majority vote across the duration of a user action). As shown in Table 1, the baseline model achieved 78% accuracy across all six emotions. ‘Hostile’ is often misclassified as ‘Happiness’ and ‘Fear’ as ‘Hostile’ (Figure 2). During the live pilot study, raw frame-level accuracy dipped to 73% due to transient hardware noise. However, applying the clip-level majority vote successfully filtered out this noise, boosting real-world accuracy to **86%**. The emotion classifier has a 1-2 millisecond latency on the CPU, but is still efficient and reliable with an accuracy of 86%.

We deployed YOLOv11n for object detection on the NVIDIA GPU, while the Python scripts for the fixation target estimator ran on the CPU. The IRM uses locally deployed LLMs, served via Ollama on the GPU. No fine-tuning is applied; all models are evaluated zero-shot. We evaluate five LLMs for each dataset, namely llama3.2, mistral, gemma3:4b, phi4-mini, and qwen2.5:3b. Tables 2 and 3 show the performance measurements of the IRM for EPIC-KITCHENS

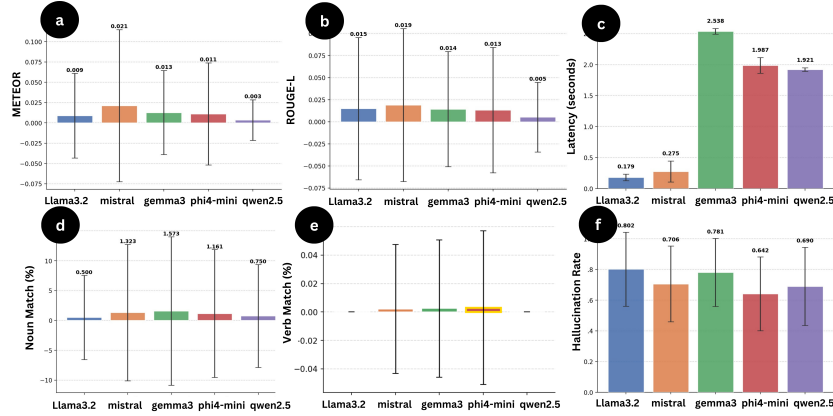


Fig. 4: Bar plots, with error bars, comparing the LLM model performance on the EPIC-KITCHENS [6] dataset. Standard evaluation metrics, such as (a) METEOR score (for synonym evaluation), (b) ROUGE-L, (c) Latency in seconds, (d) Noun match percentage, (e) Verb match percentage, and (f) Hallucination rate, are shown.

Table 2: Performance of the Intent Recognition Module for the EPIC-KITCHENS dataset using the mean values. Best numbers are shown in **bold**.

Model	BLEU-1↑	ROUGE-L↑	METEOR↑	Verb%↑	Noun%↑	Halluc.↓	Latency (s)↓
Llama3.2	0.004	0.015	0.009	0.2%	0.5%	0.80	0.18
Mistral	0.010	0.019	0.021	0.5%	1.3%	0.71	0.27
Gemma3 (4B)	0.007	0.014	0.013	1.4%	1.6%	0.78	2.54
Phi-4 Mini	0.008	0.013	0.011	0.7%	1.2%	0.64	1.99
Qwen2.5 (3B)	0.003	0.005	0.003	0.0%	0.7%	0.69	1.92

and our pilot dataset, respectively. Figure 3 shows the detected objects and the Field-of-View (FOV) markings on sample frames.

EPIC-KITCHENS Dataset: We sampled 17 participant videos from the EPIC-KITCHENS-100 [6] validation split. At each annotation midpoint, three frames are extracted and processed by YOLOv11n. The resulting object state is passed to the IRM with an empty emotion state in the absence of headset telemetry. Table 2 and Figure 4 summarize the performance. All models achieve modest BLEU-1 and ROUGE-L scores (below 0.025), reflecting the inherent difficulty of predicting fine-grained kitchen actions from object lists alone.

Regarding response times, Table 2 reveals a clear split across the models. llama3.2 and mistral respond in under 300 ms (mean ≈ 0.18 s and 0.27 s, respectively). On the other hand, gemma3:4b, phi4-mini, and qwen2.5:3b require approximately 1.9–2.6 s per query. The inference time should be given careful consideration for real-time interactivity.

In-the-Wild Pilot Study: All three models achieve substantially higher scores on the pilot dataset (BLEU-1 up to 0.08; ROUGE-L up to 0.10) compared to EPIC-KITCHENS, confirming that emotion and gaze signals provide meaningful

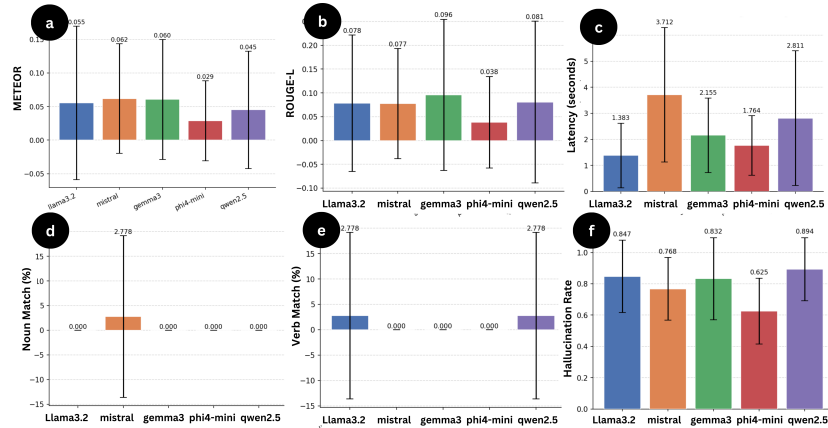


Fig. 5: Bar plots, with error bars, comparing the LLM model performance in our in-the-wild pilot study. Standard evaluation metrics, such as (a) METEOR score, (b) ROUGE-L, (c) Latency in seconds, (d) Noun match, (e) Verb match, and (f) Hallucination rate, are shown.

Table 3: Performance of the Intent Recognition Module for our pilot dataset using the mean values. Best numbers are shown in **bold**.

Model	BLEU-1↑	ROUGE-L↑	METEOR↑	Verb%↑	Noun%↑	Halluc. ↓	Latency (s)↓
llama3.2	0.069	0.078	0.055	2.8%	0.0%	0.85	1.38
mistral	0.058	0.077	0.062	0.0%	2.8%	0.77	3.71
gemma3	0.084	0.096	0.060	0.0%	0.0%	0.83	2.15
phi4-mini	0.033	0.038	0.029	0.0%	0.0%	0.63	1.76
qwen2.5	0.079	0.081	0.045	2.8%	0.0%	0.89	2.81

context. As shown in Figure 5, **gemma3:4b** dominates on all NLP overlap metrics, notably peaking at a ROUGE-L of 0.55 on one highly predictable segment. However, the verb and noun exact-match scores remain near zero for all models, indicating that predicted phrases are semantically related but lexically diverse.

As shown in Table 3, the inference latency was uniformly higher on this benchmark, at approximately 1.3–2.8s mean, compared to the fastest EPIC-KITCHENS models. This overhead is directly attributable to the richer, longer prompts generated from the full multimodal state histories.

5 Discussion

SeeIntent provides a real-time decision-support layer for MR assistants rather than a fully autonomous planner. Its most immediate applications are adaptive guidance, safety-aware cueing, and post-hoc interaction analytics in headset-driven workflows. As the current implementation depends on a constrained detector vocabulary and a small pilot study, the deployment is limited to controlled or semi-controlled environments until broader validation is available.

5.1 Pilot Study

The improvement from the vision-only EPIC-KITCHENS evaluation to the fully multimodal pilot study is substantial in absolute terms, yet overall scores remain low across all NLP metrics. This indicates that while each additional modality contributes to a useful context, the overall task of predicting a specific verb-noun action phrase from noisy, unstructured sensor inputs requires advanced prompt engineering. Incorporating explicit scene-graph representations or object-relation tuples into the prompt may bridge this semantic gap.

5.2 System Design: Use of Zero-Shot LLMs

The non-trivial zero-shot LLM performance in this study is interpreted as a consequence of the context encoding available in the instruction-tuned LLMs. The model is not using the headset data from memory, nor is it accessing an external knowledge base in our implementation. Rather, the prompt activates parametric knowledge learned during pre-training and instruction tuning: object affordances (*e.g.*, cups are commonly held, filled, washed, or drunk from), verb-noun co-occurrence statistics, short everyday action scripts, and linguistic paraphrase relations between semantically similar labels. When the prompt contains a temporally ordered list of detected objects, gaze/fixation targets, and affective state, the LLM can map this structured context onto these learned regularities and select a plausible action phrase without task-specific fine-tuning.

This mechanism is particularly useful for intent recognition because many first-person activities are governed by strong commonsense priors. For example, the sequence “hand near cup”, “fixation on sink”, and “neutral emotion” makes actions such as “wash cup” or “pick cup” more likely than unrelated actions, even if the exact ground-truth phrase was never seen during training. The LLM, therefore, functions as a lightweight semantic integrator, *i.e.*, it retrieves relations among objects, locations, human actions, and natural-language descriptions, and then compresses the multimodal state into a verb-noun intent. This also explains both the strength and the weakness of the zero-shot setting. The same learned priors that allow plausible predictions can also produce context deviation when the perception layer provides incomplete evidence or a generic object label. Thus, zero-shot success should be understood as commonsense semantic completion over a structured prompt, not as reliable visual understanding by the LLM itself.

Verb and noun match accuracies remain near zero across both evaluation domains, suggesting that zero-shot LLMs cannot reliably reproduce the specific lexical choices used in ground-truth annotations. Fine-tuning on a small set of annotated MR interaction pairs would likely yield large gains on exact-match metrics while concurrently reducing the hallucination rate.

5.3 System Evaluation

Our results reveal a clear Pareto frontier, *i.e.*, `llama3.2` and `mistral` occupy the low-latency, moderate-accuracy region, while `gemma3:4b` achieves the high-

est accuracy at the cost of nearly $10\times$ longer response times. For real-time MR applications, where user actions unfold on the scale of 1–2 seconds, sub-300-millisecond latency is a strict practical requirement, making `llama3.2` and `mistral` the pragmatic choices. Speculative decoding or quantized model variants could potentially close the latency gap for larger architectures.

Evaluating SeeIntent reveals a *semantic bottleneck* introduced by the object detection layer, in the case of both datasets. YOLOv11n maps the visual space to a highly constrained, generic vocabulary. The LLMs lack the visual granularity required to exactly match specific ground-truth annotations (*e.g.*, predicting “use cup” instead of “wash glass”). This strictly penalizes n-gram overlap scores, even when predictions are semantically plausible.

While our in-the-wild pilot study yields higher performance scores than the EPIC-KITCHENS evaluation, the limited sample size precludes definitive claims regarding the absolute superiority of multimodal fusion. Overall, the pilot successfully validates the operational viability of SeeIntent, proving that the integration of concurrent telemetry, vision, and LLM executes reliably in real-time.

5.4 System Implementation

All models exhibit a higher hallucination rate (above 60%), meaning the majority of predictions reference nouns not explicitly provided by the YOLO vision layer. However, a qualitative review suggests this is heavily conflated with the semantic bottleneck described above. For example, if the vision layer detects a “cup” but the LLM naturally predicts the action “wash mug,” the rigid metric flags this as a hallucination. While true confabulations (inventing actions entirely unsupported by the scene) remain a safety concern for MR assistants, a significant portion of this 60% represents the LLM attempting to compensate for the detector’s limited vocabulary. Future iterations must employ semantic similarity thresholds (*e.g.*, cosine similarity of word embeddings) instead of strict string-matching to isolate true hallucinations.

Model parameter count does not strictly correlate with zero-shot spatial reasoning. The 7B-parameter `mistral` model outperformed larger architectures on several metrics for the EPIC-KITCHENS dataset, while responding nearly $10\times$ faster than `gemma3:4b`. However, uniformly high hallucination rates across all models indicate that unconstrained LLMs frequently confabulate actions beyond the observed object context. This suggests that the main novelty of the paper lies in the multimodal temporal fusion and real-time orchestration, while the object detector itself should be interpreted as a standard perception baseline.

The system currently relies on a Windows-specific Application Programming Interface (API) for MR frame capture, which limits cross-platform portability. Also, while the XGBoost emotion classifier is rigorously validated offline, its robustness to dynamic real-world factors during live deployment is to be studied, *e.g.*, for physical Head-Mounted Display (HMD) slippage or diverse facial morphologies. Finally, the pilot dataset needs to be expanded for a large-scale evaluation to fully assess pipeline generalization and optimal system settings.

6 Conclusions

We presented a real-time multimodal intent prediction system, SeeIntent, for MR environments that combines object-level visual context, facial blend-shape emotion classification, and head-rotation gaze encoding into a temporally structured LLM prompt, thus enhancing it to a spatio-temporal LLM. We experimented with SeeIntent using the standardized EPIC-KITCHENS egocentric dataset and an in-the-wild pilot study of annotated MR recordings, which provides a comparison of five open-weight LLMs across accuracy, context deviation, and latency.

Key findings indicate that (i) multimodal fusion improves semantic matching over vision-only baselines, (ii) **gemma3:4b** achieves the highest overlap scores, and (iii) **llama3.2** and **mistral** offer competitive predictive capability at sub-300-millisecond latency for real-time MR applications. Remaining errors are dominated by the limited object vocabulary and the difficulty of producing exact lexical matches from noisy sensor streams.

Future work will focus on (i) extending the gaze module to leverage per-eye vergence data for precise 3D spatial grounding, (ii) fine-tuning lightweight LLMs on domain-specific MR interaction datasets, (iii) integrating a retrieval-augmented action ontology and open-vocabulary detectors to suppress context deviation, and (iv) conducting a comprehensive user study to assess how accurate intent prediction translates to perceived utility in MR assistants.

References

1. Agtzidis, I., Startsev, M., Dorr, M.: 360-Degree Gaze Estimation in the Wild using Graph Convolutional Networks. In: 2019 IEEE International Conference on Image Processing (ICIP) (2019)
2. Ahn, M., Brohan, A., Brown, N., Chebotar, Y., Cortes, O., David, B., Finn, C., Fu, C., Gopalakrishnan, K., Hausman, K., et al.: Do as I Can, Not as I Say: Grounding Language in Robotic Affordances. In: arXiv preprint arXiv:2204.01691 (2022)
3. Billinghurst, M., Clark, A., Lee, G.: A Survey of Augmented Reality. *Foundations and Trends in Human-Computer Interaction* **8**(2–3) (2015)
4. Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J.D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al.: Language Models are Few-Shot Learners. *Advances in Neural Information Processing Systems* **33** (2020)
5. Chen, T., Guestrin, C.: XGBoost: A Scalable Tree Boosting System. In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. pp. 785–794 (2016)
6. Damen, D., Doughty, H., Farinella, G.M., Fidler, S., Furnari, A., Kazakos, E., Moltisanti, D., Munro, J., Perrett, T., Price, W., Wray, M.: Scaling Egocentric Vision: The EPIC-KITCHENS Dataset. In: *Proceedings of the European Conference on Computer Vision (ECCV)* (2018)
7. David-John, B., Peacock, C., Zhang, T., Murdison, T.S., Benko, H., Jonker, T.R.: Towards Gaze-based Prediction of the Intent to Interact in Virtual Reality. In: *ACM Symposium on Eye Tracking Research and Applications*. pp. 1–7 (2021)
8. Ekman, P., Friesen, W.V.: *Facial Action Coding System: A Technique for the Measurement of Facial Movement*. Consulting Psychologists Press (1978)

9. Emoji Hero VR Contributors: Emoji Hero VR Dataset (2023), available at <https://github.com/emoji-hero-vr>
10. Furnari, A., Farinella, G.M.: What Would You Expect? Anticipating Egocentric Actions with Rolling-Unrolling LSTMs and Modality Attention. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 6252–6261 (2019)
11. Girdhar, R., Grauman, K.: Anticipative Video Transformer. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (2021)
12. Grauman, K., Westbury, A., Byrne, E., Chavis, Z., Furnari, A., Girdhar, R., Hamburger, J., Jiang, H., Liu, M., Liu, X., et al.: Ego4D: Around the World in 3,000 Hours of Egocentric Video. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (2022)
13. Grinsztajn, L., Oyallon, E., Varoquaux, G.: Why do Tree-Based Models Still Outperform Deep Learning on Typical Tabular Data? *Advances in Neural Information Processing Systems* **35**, 507–520 (2022)
14. Guenter, B., Finch, M., Drucker, S., Tan, D., Snyder, J.: Foveated 3D Graphics. *ACM Transactions on Graphics (TOG)* **31**(6), 1–10 (2012)
15. Jammot, M., Braun, B., Streli, P., Wampfler, R., Holz, C.: egoEMOTION: Egocentric Vision and Physiological Signals for Emotion and Personality Recognition in Real-World Tasks. *Advances in Neural Information Processing Systems* **38** (2026)
16. Kazakos, E., Nagrani, A., Zisserman, A., Damen, D.: EPIC Fusion: Audio-Visual Temporal Binding for Egocentric Action Recognition. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (2019)
17. Langley, A., Riddoch, A., Wilk, A., Vicente, A., Krasic, C., Zhang, D., Yang, F., Kouranov, F., Swett, I., Iyengar, J., et al.: The Quic Transport Protocol: Design and Internet-Scale Deployment. In: Proceedings of the Conference of the ACM Special Interest Group on Data Communication. pp. 183–196 (2017)
18. Li, Y., Liu, M., Rehg, J.M.: In the eye of the beholder: Gaze and actions in first person video. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **45**(6), 6731–6747 (2021)
19. Nojavanasghari, B., Gopinath, D., Koushik, J., Baltrušaitis, T., Morency, L.P.: Deep Multimodal Fusion for Persuasiveness Prediction. In: Proceedings of the 18th ACM International Conference on Multimodal Interaction (2016)
20. Tian, Y.L., Kanade, T., Cohn, J.F.: Recognizing Action Units for Facial Expression Analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **23**(2) (2001)
21. Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.A., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., Azhar, F., et al.: LLaMA: Open and Efficient Foundation Language Models. *arXiv preprint arXiv:2302.13971* (2023)
22. Wang, Z., Liu, C., Tjitrahardja, E., Wang, Y., Pavlov, B., Gou, F., Davila, J.M., Shi, D., Xu, R., Pan, Y., Tan, J., Chang, S., Wang, Q., Li, J., Hua, J., Huang, Y., Sun, J., Zhang, Y., Zhang, L., Yao, G., Jia, J., Li, Y., Wang, Q., Shi, Y., Liu, M.: EgoIntrospect: An Egocentric Dataset and Benchmark for User-Centric Internal State Reasoning (2026), <https://arxiv.org/abs/2605.17262>
23. Xu, H., Ghosh, G., Huang, P.Y., Okhonko, D., Aghajanyan, A., Metze, F., Zettlemoyer, L., Feichtenhofer, C.: VideoCLIP: Contrastive Pre-Training for Zero-Shot Video-Text Understanding. In: Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing. pp. 6787–6800 (2021)