YETI (YET to Intervene) Proactive Interventions by Multimodal AI Agents in Augmented Reality Tasks

Saptarashmi Bandyopadhyay
University of Maryland, College Park
College Park, MD, USA
saptab1@umd.edu

Vikas Bahirwani
Google
Mountain View, CA, USA
vrb@google.com

Lavisha Aggarwal
Google
Seattle, WA, USA
lavishaggarwal@google.com

Bhanu Guda
Google
Mountain View, CA, USA
bhanuguda@google.com

Lin Li
Google
Mountain View, CA, USA
linspeaking@google.com

Andrea Colaco
Google
Mountain View, CA, USA
andreacolaco@google.com

Abstract

Multimodal AI Agents are AI models that have the capability of interactively and cooperatively assisting human users to solve day-to-day tasks. Augmented Reality (AR) head worn devices can uniquely improve the user experience of solving procedural day-to-day tasks by providing egocentric multimodal (audio and video) observational capabilities to AI Agents. Such AR capabilities can help the AI Agents see and listen to actions that users take which can relate to multimodal capabilities of human users. Existing AI Agents, either Large Language Models (LLMs) or Multimodal Vision-Language Models (VLMs) are reactive in nature, which means that models cannot take an action without reading or listening to the human user's prompts. Proactivity of AI Agents on the other hand can help the human user detect and correct any mistakes in agent observed tasks, encourage users when they do tasks correctly or simply engage in conversation with the user - akin to a human teaching or assisting a user. Our proposed YET to Intervene (YETI) multimodal agent focuses on the research question of identifying circumstances that may require the agent to intervene proactively. This allows the agent to understand when it can intervene in a conversation with human users that can help the user correct mistakes on tasks, like cooking, using Augmented Reality. Our YETI Agent learns scene understanding signals based on interpretable notions of Structural Similarity (SSIM) on consecutive video frames. We also define the alignment signal which the AI Agent can learn to identify if the video frames corresponding to the user's actions on the task are consistent with expected actions. These signals are used by our AI Agent to determine when it should proactively intervene. We compare our results on the instances of proactive intervention in the HoloAssist multimodal benchmark for an expert agent guiding a user to complete procedural tasks.

1. Introduction

Recent advances in artificial intelligence have led to the widespread adoption of AI assistants across various platforms and modalities. While these systems, such as Siri for voice interaction and Gemini [18] for text-based communication, have demonstrated significant utility in task automation, they remain constrained by their single-modality architectures. This limitation presents a critical gap in human-AI interaction, particularly in scenarios requiring real-time, context-aware assistance.

Multimodal Vision-Language Models (VLMs) have emerged as a promising solution to bridge this modality gap, offering multimodal understanding that more closely aligns with human perception. However, current VLM-based assistive systems predominantly operate in a reactive paradigm, responding only to explicit user queries. This re-

Preprint.

active nature significantly limits their effectiveness in two critical scenarios: (1) novice learning environments, where users lack the domain knowledge to formulate appropriate queries, and (2) safety-critical operations, where immediate intervention may be necessary before user recognition of potential hazards.

To address these limitations, we propose **YET** to **I**ntervene (YETI), a novel framework (seen in Figure 1) for proactive AI intervention in augmented reality (AR) environments. Our approach leverages lightweight, real-time algorithmic signals to enable proactive assistance through AR interfaces such as smart glasses [19]. This system bridges the gap between cloud-based AI capabilities and real-world applications by enabling direct visual observation of user activities.

Our work builds upon recent developments in proactive AI assistance, particularly the HoloAssist dataset [21], which demonstrates the potential for real-time AI intervention in complex spatio-temporal tasks. While HoloAssist provides valuable insights into human-AI collaboration scenarios, such as computer assembly and coffee preparation, existing implementations face significant computational challenges.

Current state-of-the-art approaches for proactive intervention require extensive computational resources and multi-modal sensor data, including RGB streams, hand and head pose estimation, sensor readings like IMU (Inertial Measurement Unit), and depth information. The complexity of acquiring and processing this data in real-time presents a significant barrier to practical deployment. In contrast, YETI employs efficient algorithmic signals that can be computed on-the-fly, dramatically reducing the computational overhead while maintaining high intervention accuracy.

Features	Size (MB)	× SSIM	× CObj
Depth Estimation	137,408	6,543	6,870
Eye Gaze (E)	617	29	31
Hand Pose (H)	53,749	2,660	2,688
Head Pose	1,141	54	57
IMU (I)	1,132	54	57
SSIM (Ours)	**21**		
Alignment Cobj (Ours)	**20**		

Table 1. HoloAssist Feature sizes scaled with our Features

The YETI framework has comparable precision performance with different HoloAssist benchmark baseline models and shows better performance in some settings, specially higher recall and F-measure, all while using light-weight features that take 6500 times less memory, as seen in Table 1. This dramatic reduction in computational requirements enables real-time operation on resource-constrained AR devices, making proactive AI assistance practical for everyday use cases. Our framework thus represents a significant step toward deploying intelligent assistive systems in real-world applications, particularly in scenarios requiring immediate, context-aware intervention.

2. Related Works

2.1. Egocentric Interaction Datasets

Recent advances in egocentric vision have produced several datasets that capture human interactions and activities. HoloAssist [21] presents a large-scale egocentric dataset focusing on collaborative physical manipulation tasks between two people, providing detailed action and conversational annotations. This dataset offers valuable insights into how human assistants proactively and reactively intervene, correct mistakes, and ground their instructions in the environment.

Parse-Ego4D [1] introduces a benchmark for evaluating AI agents' capability to make unsolicited action suggestions based on user intent signals. We argue that as this benchmark evaluates the AI agents' response to user queries, it does not truly measure proactive behavior.

While Ego-Exo4D [11] provides a comprehensive multimodal, multiview dataset capturing both egocentric and exocentric perspectives in expert-learner scenarios, it primarily focuses on skilled single-person activities without addressing proactive communication. Similarly, existing datasets like Ego4D [10] and EPIC-Kitchens [7], while rich in activity and object annotations, lack direct mappings to actionable recommendations.

2.2. Proactive AI Agents and Communication

Proactive communication in AI agents encompasses several key aspects [8]: Intelligence (the ability to anticipate task developments), Adaptivity (dynamic adjustment of timing and interventions), and Civility (respect for user boundaries and ethical standards). Emerging research has demonstrated the value of proactive AI agents across various domains, including personal assistance, predictive maintenance, healthcare monitoring, and voice assistance [4, 5, 12].

2.3. Language Models for Proactive Assistance

Recent developments have shown promising results in using Multimodal Vision Language Models (VLMs) and LLMs for proactive assistance. ProAgent [23] introduces a framework that leverages LLMs to create agents capable of dynamically adapting their behavior and inferring teammate intentions. The effectiveness of these models has been further demonstrated through fine-tuning on ProactiveBench [13], which significantly enhances the proactive

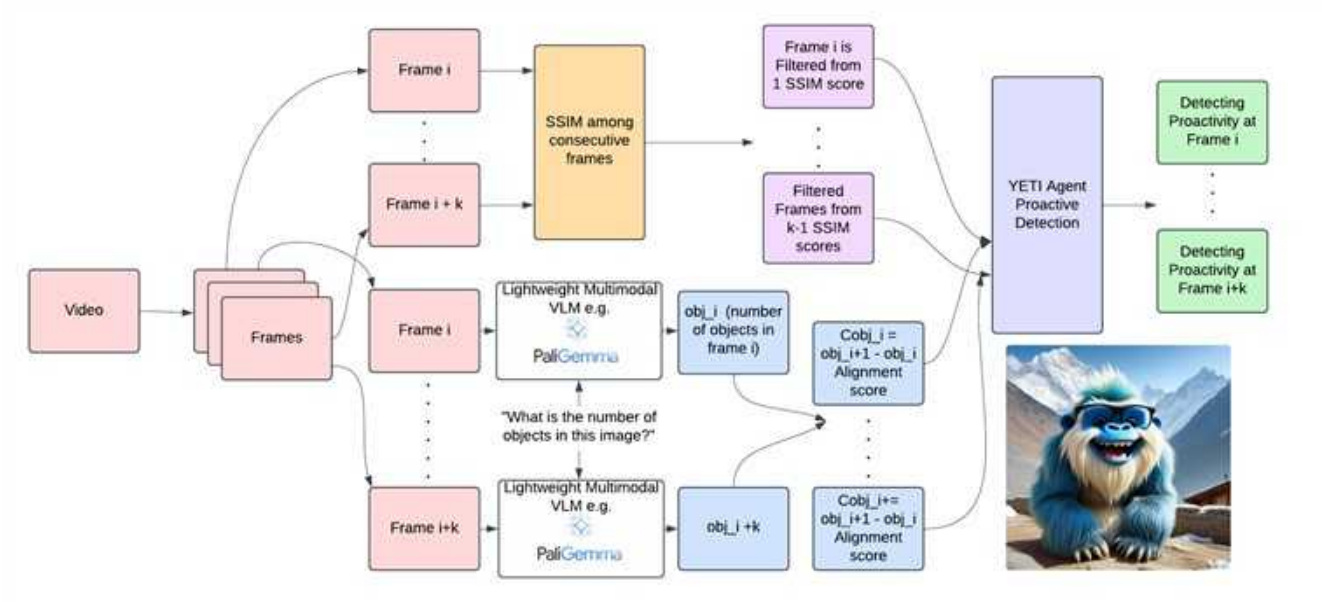


Figure 1. Overview of the YETI framework detecting the frames of proactive interaction or intervention by a Multimodal AI Agent. Our YETI Agent system generates lightweight features on-the-fly, enabling rapid decision-making for timely user assistance.

capabilities of LLM agents. In the context of assistive technology, Smart Help [6] demonstrates how proactive and adaptive support can be provided to users with diverse disabilities and dynamic goals across various tasks and environments.

Open-source VLMs are very popular as a starting point for AI assistants, especially Google's PaliGemma [3] Open-source VLM. Open-source VLMs do not have proactive interaction capabilities which is what we want to support in our research. PaliGemma generates a quick and accurate estimate of the number of objects in a given scene. PaliGemma was trained on a wide variety of datasets, including the TallyQA dataset [2], which is useful for taking a response to a question that asks for the number of objects in a given image.

3. Methodology

3.1. Proactive Augmented Reality Interaction Data of Cooperative Agents

The HoloAssist dataset [21] provides multimodal egocentric vision-language benchmarks focusing on Augmented Reality (AR)-based human-AI collaboration. AR devices are used to capture the Expert-User collaborative dynamics, recording the visual observations of an User Agent (human) collaborating with an Expert Agent (Instructor), which can be an AI Agent, on physical reasoning tasks, while documenting the dialogue between the two agents. The dataset comprises 482 unique Expert-User interaction sequences with videos and dialogues of the agents', spanning 20 diverse task domains, including but not limited to:

- Cooking procedures like making coffee
- Fixing items like motorcycles
- Assembling/Disassembling furniture
- Assembling Devices like Computers, Scanners, GPUs
- Maintaining Electrical systems like circuit breakers
- Configuring Devices like printers, cameras, switches

The User Agents wear the AR devices to record first-person perspective videos while executing procedural tasks. The AR devices simultaneously capture the Expert Agent's observations and guidance to the User Agents. The dataset's annotation schema encompasses a variety of interaction types. Some examples of interactions done by the Expert Agent include:

1. Proactive Interactions
   - High-level instructional guidance
   - Follow-up instructions without any user query
   - Interventional feedback
   - Error correction mechanisms
2. Reactive interactions:
   - Expert clarifications to user queries
   - User-initiated dialogues

The corpus encompasses 45.5 hours of video recordings generated by 350 distinct expert-user pairs, providing a rich foundation to study the spatio-temporal dynamics of when and how Proactive Multimodal AI Agents should engage in AR-assisted collaborative scenarios. This temporal aspect, when an AI Agent is yet to intervene, but should intervene, to improve collaborative task execution or correct mistakes

is crucial for developing AI agents which can guide humans in human-AI collaborative tasks.

3.2. Multimodal VLM Generation

We count objects in video sequences by leveraging the recent advancements in Multimodal Visual Language Models (VLMs). Our method processes videos by extracting frames at a rate of 1 frame per second (FPS), enabling efficient temporal analysis while maintaining sufficient granularity for accurate object counting.

Frame Extraction and Processing Given an input video V of duration T seconds, we extract a sequence of frames $\{f_1, f_2, ..., f_T\}$ at 1 FPS. This sampling rate balances computational efficiency with temporal resolution, ensuring that significant object state changes are captured while minimizing redundant processing.

Multimodal VLM Implementation We utilize PaliGemma-3b-mix-448, from the PaliGemma [3] family of lightweight multimodal VLMs created by Google, to process each frame independently. PaliGemma's ability to quickly leverage its visual and textual understanding capabilities makes it suitable for an AR / VR setting where there may not be much device compute available and a fast response is needed. For each frame f_t, we construct a prompt:

$$P_{obj} = \text{“The number of objects in this image is ”} \quad (1)$$

This prompt elicits a numerical response from the model, avoiding unwanted conversational output. We chose this prompt after thorough experimentation with PaliGemma's object detection capabilities. Some of the other prompts we explored as well as their output can be found in the supplementary material. The model processes each frame f_t with prompt P_{obj} to generate a count estimate:

$$C_t = \text{PaliGemma}(f_t, P_{obj}) \quad (2)$$

where C_t represents the predicted object count at time t. The change in object count between frames is heavily skewed towards zero as seen in Figure 2.

Implementation Details The PaliGemma model was used with its default configuration, maintaining the 448×448 pixel input resolution. The model's responses were post-processed to extract numerical values.

3.3. Alignment with Changing Object Count

The alignment signal taking the form of a change in object count for the scene observed by the AI Assistant is motivated by an intuition of how humans operate when listening

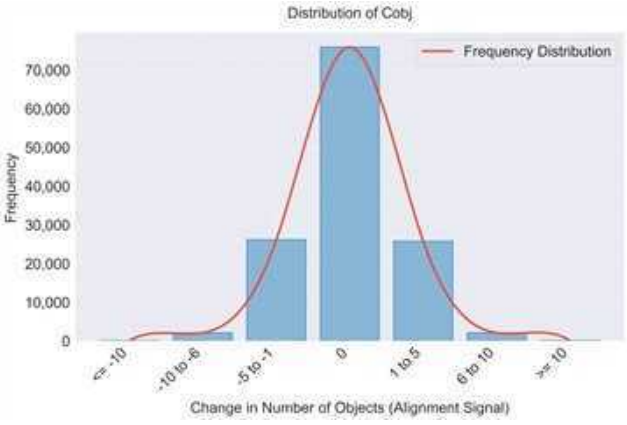
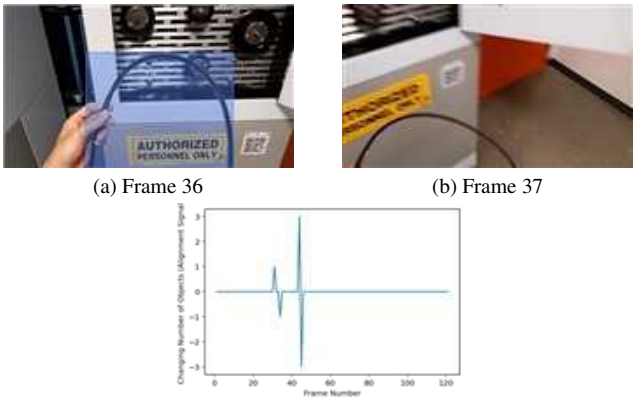


Figure 2. Distribution of Alignment Signal

to instructions. If a user agent is being guided on how to assemble a computer, they will not be moving objects around while they process the instructions. Rather, they will be listening so they know what to do next. Building upon this understanding, we can estimate when a proactive AI assistant should intervene by simply monitoring the change in object count from second to second. Figure 3 shows an example of frames where the object count changes.



(a) Frame 36 (b) Frame 37

(c) Alignment Signal over entire video.

Figure 3. Plot of Alignment Signal measuring changing object count along spatio-temporally changing image frames in a video for a procedural task on how to change a mechanical belt. The Expert Agent autonomously intervenes in the 37th second at Frame 37 based on the alignment signal with Frame 36

3.4. Scene Understanding

Spatio-temporal Signal Generation The frame-wise counting results are aggregated to create a temporal signal $\{\Delta C_1, \Delta C_2, ..., \Delta C_{T-1}\}$ where $\Delta C_i := C_{i+1} - C_i$. This signal captures how the number of objects change throughout the video sequence.

(a) SSIM Frame 98

(b) SSIM Frame 99

(c) SSIM Signal over entire video.

Figure 4. Plot of SSIM filtering proactive interventions by expert agents in an image frame (time instance) of a video capturing a procedural task on how to assemble a RAM computer. Autonomous intervention happens at the 98th second in Frame 98

To identify meaningful frames for proactive intervention, we employ the Structural Similarity Index Measure (SSIM) [22] to analyze temporal coherence between consecutive frames. This approach helps filter out redundant frames where the scene remains largely static, such as Figure 4, thus focusing interventions on moments of significant change.

Given two consecutive frames f_i and f_{i+1}, we compute their SSIM as:

$$\text{SSIM}(f_i, f_{i+1}) = \frac{(2\mu_i\mu_{i+1} + c_1)(2\sigma_{i,i+1} + c_2)}{(\mu_i^2 + \mu_{i+1}^2 + c_1)(\sigma_i^2 + \sigma_{i+1}^2 + c_2)} \tag{3}$$

where μ_i, μ_{i+1} denote the mean intensities of frames f_i and f_{i+1}, $\sigma_i^2, \sigma_{i+1}^2$ represent their respective variances, $\sigma_{i,i+1}$ is the covariance between the frames, $c_1 = (0.01L)^2$ and $c_2 = (0.03L)^2$ are stability constants, and $L = 255$ is the dynamic range of pixel values in grayscale.

The SSIM metric yields values in $[0, 1]$, where higher values indicate greater structural similarity between frames. We leverage this property to identify and filter out frames with SSIM values exceeding a threshold τ, effectively removing redundant temporal information. This filtering mechanism ensures that interventions are triggered only during meaningful scene changes, reducing unnecessary proactive interventions while maintaining responsiveness to significant environmental variations. Similarly to our temporal signal for changing object count, we consolidate the SSIM values for each frame into a set $\{s_1, s_2, ..., s_{T-1}\}$.

3.5. Proactive Interactions and Interventions

In the context of AI Agents, reactivity refers to the AI Agent's response to a user cue. In contrast, proactivity involves behaviors initiated by the AI Agent without user prompts. Proactive activities can be broadly categorized into proactive interaction and proactive intervention. Understanding the subtle but important distinction between these two is essential for leveraging the capabilities of a proactive agent. An AI Agent is considered to be *proactively interacting* with the user if it initiates any engagement without user cues. Conversely, an AI Agent is *proactively intervening* when it takes concrete steps to alter the user's behavior. By this definition, all proactive interventions are forms of proactive interactions, but not all proactive interactions qualify as interventions.

HoloAssist encompasses eight distinct categories of proactive behavior, of which three are classified as interventions, as detailed in Table 2.

3.6. YETI Proactive Agent Intervention Algorithm

The YETI algorithm incorporates several hyperparameters that determine when a Multimodal AI Agent should autonomously intervene proactively without any question or clarifications asked by the User Agent.

- **SSIM threshold (τ):** This parameter sets a filtering threshold. A frame's SSIM value with its corresponding frame must satisfy it to be considered in the YETI algorithm, to filter out highly similar frames where the user is not doing anything. In other words, if a frame and its proceeding frame have an SSIM of $\geq \tau$, the frame will not be considered for an autonomous intervention.
- **Conversation Interval (m):** This parameter enforces a minimum temporal gap between consecutive interventions, ensuring that the AI agent does not intervene too frequently. It defines the duration that must pass after an intervention before another can be initiated.
- **Local Extrema Range (r):** This parameter identifies the sensitivity of the algorithm to changes in object counts within the frames. It defines the range within which a change in object count must fall or rise to be considered significant.
- **Episode Interval (k):** This parameter limits the maximum rate at which interventions can occur by defining the length of an "episode." An episode is a consecutive sequence of frames within which only one intervention is permitted, thus preventing excessive and potentially disruptive interventions.

Conversation Type	Interaction	Intervention	Example
Follow-up Instruction	✓	✓	"Put the battery back."
Confirming Previous Action	✓	✓	"Perfect."
Correcting Wrong Action	✓	✓	"Nope, not that one."
Describe High-Level Instruction	✓	×	"You're going to validate, so we're going to move the shift lever through all of the settings."
Opening Remarks	✓	×	"Now for this task, we are removing the graphic cards from the PC slot."
Closing Remarks	✓	×	"You're all done. You can exit now."
Adjusting Video	✓	×	"Just keep your eyes on your hands."
Other	✓	×	"You can ground yourself again, but it's not really necessary."

Table 2. Examples of Proactive Interactions and Proactive Interventions in HoloAssist.

Algorithm 1 YETI Proactive Intervention Detection

Require: Frame sequence F with alignment scores $\{\Delta C_i\}$
Ensure: Set of intervention frames $\mathcal{I}$

Initialize empty sets: $\mathcal{I}$, E_{obj} where E_{obj} is the alignment scores per episode of frames
Set episode interval k, conversation interval m, local extrema range r
Initialize episode count $n = 0$, frame count $t = 0$
for each frame $f_i \in F$ **do**
  if $n > 0$ **and** $f_i \notin$ conversation interval **then**
    $t \leftarrow t + 1$
    Add ΔC_i to E_{obj}
    if $c_i \in$ local extrema range **then**
      $\mathcal{I} \leftarrow \mathcal{I} \cup \{f_i\}$
    end if
    if $t = k$ **then**
      Update conversation interval
      Reset episode metrics
      $n \leftarrow n + 1, t \leftarrow 0$
    end if
  else if $n = 0$ **then**
    $t \leftarrow t + 1$
    Add ΔC_i to E_{obj}
    if $t = k$ **then**
      $\Delta C_{min} \leftarrow \min(E_{obj})$
      $\Delta C_{max} \leftarrow \max(E_{obj})$
      Define local extrema range with tolerance r.
      $\mathcal{I} \leftarrow \mathcal{I} \cup \{f_{current}\}$
      Update conversation interval
      Reset episode metrics
      $n \leftarrow n + 1, t \leftarrow 0$
    end if
  end if
end for
  return $\mathcal{I}$

4. Experiments

To validate the results of our YETI algorithm, we conducted experiments with a wide variety of different settings. This also lets us see how each configuration of the algorithm and the value of each hyperparameter contributes to the evaluation metrics of our method compared to HoloAssist, the baseline for AI Agents proactively intervening with an user (student) task. An example of a proactive intervention being detected can be seen in Figure 5.

4.1. Experimental Settings

We carefully selected hyperparameters to balance the trade-off between timely interventions and avoiding excessive interruptions. The key parameters are summarized in Table 3. A comprehensive analysis of hyperparameter sensitivity is provided in the supplementary material.

Parameter	Value
SSIM threshold (τ)	0.9
Conversation interval (m)	1
Extrema range (r)	± 1
Minimum history (k)	5

Table 3. Hyperparameters used in our experiments. τ controls frame similarity filtering, m sets minimum gap between interventions, r defines the range for local minima detection, and k specifies required history length before intervention.

We evaluate two variants of our YETI algorithm:

- **Global YETI**: Uses the first detected local extrema as a fixed threshold throughout the sequence
- **Local YETI**: Continuously updates the extrema threshold based on recent history

This allows us to analyze the impact of adaptive versus fixed intervention thresholds on system performance. The full results of evaluating performance in this way can be seen in Tables 4 and 5.

For the comparative analysis between our YETI methods and HoloAssist, it is important to note we limited our evaluation to videos where the user and the expert agent both initiated conversations. This stipulation more closely aligns with the real-world assistive AI Agent scenario we had in mind, where the user may ask for help completing a task. This turns out to be 482 videos out of the total HoloAssist dataset, more than enough to be a representative sample. We also aggregated the HoloAssist results for each intervention class into one through averaging in order to have a head-to-head comparison with regards to predicting any kind of intervention. We also provide Supplemental results where the user agent does not communicate with any dialogue, however the expert agent proactively interacts and intervenes with the user, based on user's visual observations.

4.2. Metrics

We evaluate our YETI algorithm's effectiveness in detecting appropriate moments for proactive interaction and intervention using standard classification metrics:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{4}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{5}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{6}$$

$$\text{F-measure} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{7}$$

where TP, TN, FP, and FN denote true positives, true negatives, false positives, and false negatives, respectively. These metrics provide a comprehensive assessment of YETI's intervention capabilities, measuring both its ability to intervene at appropriate moments and its capacity to avoid unnecessary interruptions.

As there is no empiric measurement of when the "best" precise moment to intervene is, we use a window of five seconds around HoloAssist labelled ground-truth proactive intervention starting time-stamps in order to assess whether a detected intervention frame is a true positive or a false positive. This is consistent with the method used to evaluate the HoloAssist baseline model [21], which also determines true positives and false positives by measuring temporal proximity to the nearest labelled intervention and seeing if it is within a window of tolerance.

4.3. Results

We can see in Tables 5 and 1 that YETI achieves impressive results when detecting proactive interventions and detections. Furthermore, as PaliGemma's scale is on the order of three billion parameters, it is suitable to be used on board an Augmented Reality device. The small data size of SSIM and Object Count Alignment relative to other features, as seen in Table 1, means that AI Agents on the device can use the information while having a small response time.

5. Conclusion and Future Work

In this work, we tackle the significant gap in proactive AI assistance for everyday real-world tasks by introducing the Yet-to-Intervene (YETI) algorithm. Traditional AI agents, such as those exemplified by the HoloAssist baseline, predominantly operate reactively, responding only to explicit user prompts. This reactive nature limits their effectiveness in dynamic and context-sensitive environments where anticipatory intervention can substantially enhance user experience and task efficiency. Our YETI algorithm addresses this limitation by enabling AI agents to proactively identify and intervene in user actions without awaiting explicit cues.

Integrating YETI with lightweight Vision-Language Models (VLMs) like PaliGemma, our approach demonstrates remarkable improvements across key performance metrics, including Recall, Precision, Accuracy, and F-Measure. Notably, YETI achieves these enhancements while utilizing features that are up to 60,000 times more memory-efficient than those employed by state-of-the-art models such as HoloAssist. This drastic reduction in computational overhead not only makes YETI more accessible for deployment on resource-constrained devices but also paves the way for real-time, scalable AI assistance in diverse augmented reality (AR) applications.

Our extensive ablation studies in the paper and supplemental reveal the critical features that significantly influence the accurate detection of intervention moments. By systematically varying and analyzing different feature sets, we identify the most impactful components that empower YETI to discern when proactive interactions or interventions are warranted. This insight underscores the importance of feature selection in the design of efficient and effective AI assistance algorithms.

Looking ahead, there are several promising avenues for future research. First, we aim to enhance YETI's intervention capabilities by incorporating richer sensory data, including hand pose, eye gaze, head orientation, Inertial Measurement Unit (IMU) readings, and depth information. Integrating these modalities is expected to provide a more comprehensive understanding of the user's context and intentions, thereby enabling more nuanced and timely interventions. Second, we plan to evaluate YETI's performance across a broader spectrum of VLMs to assess its generalizability and identify optimal model architectures for proactive assistance. Additionally, fine-tuning VLMs on the HoloAssist dataset could further refine the agent's ability to anticipate user needs and improve intervention accuracy.

Furthermore, future implementations of YETI will ex-

	Method	Overall			Confirm Action			Correct Mistake			Follow Up		
		Prec.	Rec.	F-meas.	Prec.	Rec.	F-meas.	Prec.	Rec.	F-meas.	Prec.	Rec.	F-meas.
HoloAssist	(RGB)	13.93	33.33	19.65	0.00	0.00	0.00	0.00	0.00	0.00	41.79	100.00	58.95
	(R+H)	24.89	33.64	28.61	32.14	4.50	7.89	0.00	0.00	0.00	42.52	96.43	59.02
	(R+E)	25.55	33.73	29.08	33.90	10.36	15.87	0.00	0.00	0.00	42.76	90.83	58.15
	(R+H+E)	48.31	37.59	42.28	39.11	40.93	40.00	61.11	9.91	17.05	44.70	61.93	51.92
	(RGB[Pt])	37.54	37.74	37.64	42.31	27.50	33.33	27.33	36.61	31.30	42.97	49.11	45.84
YETI (Ours)	Global	41.86	88.31	56.17	22.54	91.51	36.17	11.68	90.29	20.69	27.55	89.69	42.15
	Local	46.88	60.38	52.77	26.55	68.62	38.29	14.71	68.02	24.18	30.07	62.05	40.51

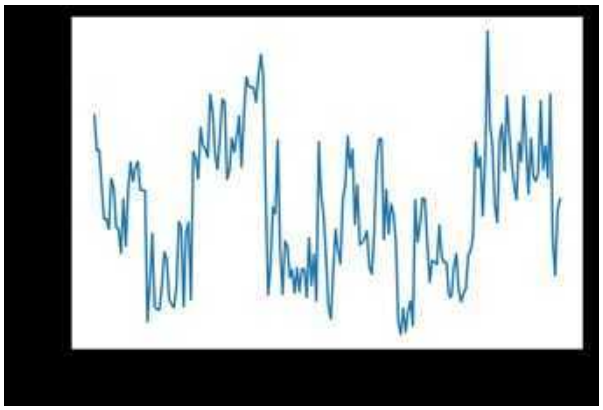
Table 4. Performance of YETI compared to HoloAssist in detecting proactive interventions.

(a) Intervention - 3 seconds

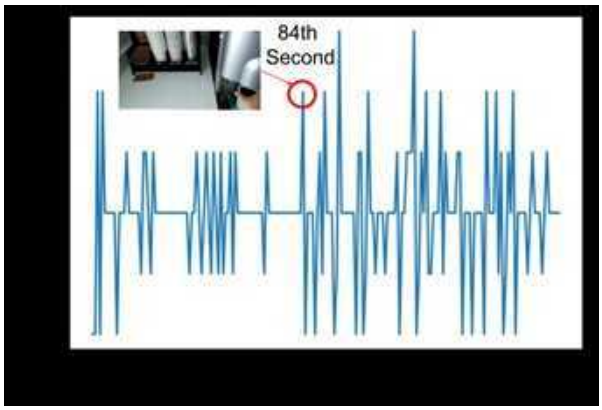
(b) Intervention - 2 seconds

(c) Intervention - 1 second

(d) AI Agent proactively intervenes.



(e) SSIM between consecutive frames.



(f) Changing Objects Alignment Signal

Figure 5. Intervention Detection for Coffee Making Task.

	Method	Overall			
		Acc.	Prec.	Rec.	F1
YETI (Ours)	Global	86.97	52.23	87.04	65.28
	Local	93.76	64.51	59.73	62.02

Table 5. Evaluation of YETI on our proactive interaction detection setting which encompasses the proactive intervention benchmark

plore adaptive learning mechanisms that allow the AI agent to continuously refine its intervention strategies based on user feedback and evolving task dynamics. This adaptive approach is anticipated to enhance the personalization and effectiveness of AI assistance, making it more attuned to individual user preferences and behaviors.

In summary, the YETI algorithm represents a significant advancement in the development of proactive AI assistants, offering enhanced performance with reduced computational demands. By enabling timely and context-aware interventions, YETI has the potential to transform human-AI collaboration in AR environments and beyond. Our ongoing and future work seeks to build upon these foundations, further pushing the boundaries of what proactive AI agents can achieve in facilitating and automating complex real-world tasks.

References

[1] Steven Abreu, Tiffany D. Do, Karan Ahuja, Eric J. Gonzalez, Lee Payne, Daniel McDuff, and Mar Gonzalez-Franco. Parse-ego4d: Personal action recommendation suggestions

for egocentric videos, 2024. 2

[2] Manoj Acharya, Kushal Kafle, and Christopher Kanan. Tallyqa: Answering complex counting questions. In *Proceedings of the AAAI conference on artificial intelligence*, pages 8076–8084, 2019. 3

[3] Lucas Beyer, Andreas Steiner, André Susano Pinto, Alexander Kolesnikov, Xiao Wang, Daniel Salz, Maxim Neumann, Ibrahim Alabdulmohsin, Michael Tschannen, Emanuele Bugliarello, Thomas Unterthiner, Daniel Keysers, Skanda Koppula, Fangyu Liu, Adam Grycner, Alexey Gritsenko, Neil Houlsby, Manoj Kumar, Keran Rong, Julian Eisenschlos, Rishabh Kabra, Matthias Bauer, Matko Bošnjak, Xi Chen, Matthias Minderer, Paul Voigtlaender, Ioana Bica, Ivana Balazevic, Joan Puigcerver, Pinelopi Papalampidi, Olivier Henaff, Xi Xiong, Radu Soricut, Jeremiah Harmsen, and Xiaohua Zhai. Paligemma: A versatile 3b vlm for transfer, 2024. 3, 4

[4] Yash Butala, Siddhant Garg, Pratyay Banerjee, and Amita Misra. ProMISe: A proactive multi-turn dialogue dataset for information-seeking intent resolution. In *Findings of the Association for Computational Linguistics: EACL 2024*, pages 1774–1789, St. Julian's, Malta, 2024. Association for Computational Linguistics. 2

[5] Caterina Bérubé, Marcia Nißen, Rasita Vinay, Alexa Geiger, Tobias Budig, Aashish Bhandari, Catherine Rachel Pe Benito, Nathan Ibarcena, Olivia Pistolese, Pan Li, Abdullah Bin Sawad, Elgar Fleisch, Christoph Stettler, Bronwyn Hemsley, Shlomo Berkovsky, Tobias Kowatsch, and A. Baki Kocaballi. Proactive behavior in voice assistants: A systematic review and conceptual model. *Computers in Human Behavior Reports*, 14:100411, 2024. 2

[6] Zhihao Cao, Zidong Wang, Siwen Xie, Anji Liu, and Lifeng Fan. Smart help: Strategic opponent modeling for proactive and adaptive robot assistance in households. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 18091–18101, 2024. 3

[7] Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Sanja Fidler, Antonino Furnari, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, and Michael Wray. Scaling egocentric vision: The epic-kitchens dataset. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018. 2

[8] Yang Deng, Lizi Liao, Zhonghua Zheng, Grace Hui Yang, and Tat-Seng Chua. Towards human-centered proactive conversational agents. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, page 807–818, New York, NY, USA, 2024. Association for Computing Machinery. 2

[9] Alessandro Flaborea, Guido Maria D'Amely di Melendugno, Leonardo Plini, Luca Scofano, Edoardo De Matteis, Antonino Furnari, Giovanni Maria Farinella, and Fabio Galasso. Prego: online mistake detection in procedural egocentric videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18483–18492, 2024. 2

[10] Kristen Grauman, Andrew Westbury, Eugene Byrne, Zachary Chavis, Antonino Furnari, Rohit Girdhar, Jackson Hamburger, Hao Jiang, Miao Liu, Xingyu Liu, Miguel Martin, Tushar Nagarajan, Ilija Radosavovic, Santhosh Kumar Ramakrishnan, Fiona Ryan, Jayant Sharma, Michael Wray, Mengmeng Xu, Eric Zhongcong Xu, Chen Zhao, Siddhant Bansal, Dhruv Batra, Vincent Cartillier, Sean Crane, Tien Do, Morrie Doulaty, Akshay Erapalli, Christoph Feichtenhofer, Adriano Fragomeni, Qichen Fu, Abrham Gebreselasie, Cristina González, James Hillis, Xuhua Huang, Yifei Huang, Wenqi Jia, Weslie Khoo, Jáchym Kolář, Satwik Kottur, Anurag Kumar, Federico Landini, Chao Li, Yanghao Li, Zhenqiang Li, Karttikeya Mangalam, Raghava Modhugu, Jonathan Munro, Tullie Murrell, Takumi Nishiyasu, Will Price, Paola Ruiz, Merey Ramazanova, Leda Sari, Kiran Somasundaram, Audrey Southerland, Yusuke Sugano, Ruijie Tao, Minh Vo, Yuchen Wang, Xindi Wu, Takuma Yagi, Ziwei Zhao, Yunyi Zhu, Pablo Arbeláez, David Crandall, Dima Damen, Giovanni Maria Farinella, Christian Fuegen, Bernard Ghanem, Vamsi Krishna Ithapu, C. V. Jawahar, Hanbyul Joo, Kris Kitani, Haizhou Li, Richard Newcombe, Aude Oliva, Hyun Soo Park, James M. Rehg, Yoichi Sato, Jianbo Shi, Mike Zheng Shou, Antonio Torralba, Lorenzo Torresani, Mingfei Yan, and Jitendra Malik. Ego4d: Around the world in 3,000 hours of egocentric video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 18995–19012, 2022. 2

[11] Kristen Grauman, Andrew Westbury, Lorenzo Torresani, Kris Kitani, Jitendra Malik, Triantafyllos Afouras, Kumar Ashutosh, Vijay Baiyya, Siddhant Bansal, Bikram Boote, Eugene Byrne, Zach Chavis, Joya Chen, Feng Cheng, Fu-Jen Chu, Sean Crane, Avijit Dasgupta, Jing Dong, Maria Escobar, Cristhian Forigua, Abrham Gebreselasie, Sanjay Haresh, Jing Huang, Md Mohaiminul Islam, Suyog Jain, Rawal Khirodkar, Devansh Kukreja, Kevin J Liang, Jia-Wei Liu, Sagnik Majumder, Yongsen Mao, Miguel Martin, Effrosyni Mavroudi, Tushar Nagarajan, Francesco Ragusa, Santhosh Kumar Ramakrishnan, Luigi Seminara, Arjun Somayazulu, Yale Song, Shan Su, Zihui Xue, Edward Zhang, Jinxu Zhang, Angela Castillo, Changan Chen, Xinzhu Fu, Ryosuke Furuta, Cristina Gonzalez, Prince Gupta, Jiabo Hu, Yifei Huang, Yiming Huang, Weslie Khoo, Anush Kumar, Robert Kuo, Sach Lakhavani, Miao Liu, Mi Luo, Zhengyi Luo, Brighid Meredith, Austin Miller, Oluwatumininu Oguntola, Xiaqing Pan, Penny Peng, Shraman Pramanick, Merey Ramazanova, Fiona Ryan, Wei Shan, Kiran Somasundaram, Chenan Song, Audrey Southerland, Masatoshi Tateno, Huiyu Wang, Yuchen Wang, Takuma Yagi, Mingfei Yan, Xitong Yang, Zecheng Yu, Shengxin Cindy Zha, Chen Zhao, Ziwei Zhao, Zhifan Zhu, Jeff Zhuo, Pablo Arbelaez, Gedas Bertasius, David Crandall, Dima Damen, Jakob Engel, Giovanni Maria Farinella, Antonino Furnari, Bernard Ghanem, Judy Hoffman, C. V. Jawahar, Richard Newcombe, Hyun Soo Park, James M. Rehg, Yoichi Sato, Manolis Savva, Jianbo Shi, Mike Zheng Shou, and Michael Wray. Ego-exo4d: Understanding skilled human activity from first- and third-person perspectives, 2024. 2

[12] Lizi Liao, Grace Hui Yang, and Chirag Shah. Proactive conversational agents in the post-chatgpt world. In *Proceedings of the 46th International ACM SIGIR Conference on*

Research and Development in Information Retrieval, page 3452–3455, New York, NY, USA, 2023. Association for Computing Machinery. 2

[13] Yaxi Lu, Shenzhi Yang, Cheng Qian, Guirong Chen, Qinyu Luo, Yesai Wu, Huadong Wang, Xin Cong, Zhong Zhang, Yankai Lin, Weiwen Liu, Yasheng Wang, Zhiyuan Liu, Fangming Liu, and Maosong Sun. Proactive agent: Shifting llm agents from reactive responses to active assistance, 2024. 2

[14] Michele Mazzamuto, Antonino Furnari, and Giovanni Maria Farinella. Eyes wide unshut: Unsupervised mistake detection in egocentric video by detecting unpredictable gaze. *arXiv preprint arXiv:2406.08379*, 2024. 3

[15] Leonardo Plini, Luca Scofano, Edoardo De Matteis, Guido Maria D'Amely di Melendugno, Alessandro Flaborea, Andrea Sanchietti, Giovanni Maria Farinella, Fabio Galasso, and Antonino Furnari. Ti-prego: Chain of thought and in-context learning for online mistake detection in procedural egocentric videos. *arXiv preprint arXiv:2411.02570*, 2024. 2

[16] Rosario Scavo, Francesco Ragusa, Giovanni Maria Farinella, and Antonino Furnari. Quasi-online detection of take and release actions from egocentric videos. In *International Conference on Image Analysis and Processing*, pages 13–24. Springer, 2023. 3

[17] Luigi Seminara, Giovanni Maria Farinella, and Antonino Furnari. Differentiable task graph learning: Procedural activity representation and online mistake detection from egocentric videos. *arXiv preprint arXiv:2406.01486*, 2024. 2

[18] Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023. 1

[19] Ethan Waisberg, Joshua Ong, Mouayad Masalkhi, Nasif Zaman, Prithul Sarker, Andrew G Lee, and Alireza Tavakkoli. Meta smart glasses—large language models and the future for assistive glasses for individuals with vision impairments. *Eye*, 38(6):1036–1038, 2024. 2

[20] Xiang Wang, Shiwei Zhang, Zhiwu Qing, Yuanjie Shao, Zhengrong Zuo, Changxin Gao, and Nong Sang. Oadtr: Online action detection with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7565–7575, 2021. 3

[21] Xin Wang, Taein Kwon, Mahdi Rad, Bowen Pan, Ishani Chakraborty, Sean Andrist, Dan Bohus, Ashley Feniello, Bugra Tekin, Felipe Vieira Frujeri, et al. Holoassist: an egocentric human interaction dataset for interactive ai assistants in the real world. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 20270–20281, 2023. 2, 3, 7, 1

[22] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004. 5

[23] Ceyao Zhang, Kaijie Yang, Siyi Hu, Zihao Wang, Guanghe Li, Yihang Sun, Cheng Zhang, Zhaowei Zhang, Anji Liu, Song-Chun Zhu, Xiaojun Chang, Junge Zhang, Feng Yin, Yitao Liang, and Yaodong Yang. Proagent: Building proactive cooperative agents with large language models. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(16):17591–17599, 2024. 2

YETI (YET to Intervene) Proactive Interventions by Multimodal AI Agents in Augmented Reality Tasks

Supplementary Material

6. Detailed Analysis of Results

This section provides a more in-depth analysis of the results presented in Tables 1, 4, and 5 with a special focus on Table 4 comparing the performance of YETI to the HoloAssist baseline in detecting proactive interventions.

Overall Performance:

YETI consistently outperforms HoloAssist [21] overall, demonstrating a significant improvement in accurately detecting proactive interventions in most scenarios. This superior performance is observed in both the Global and Local variants of YETI. Notably, YETI achieves substantially higher recall in all categories, indicating its effectiveness in identifying a greater proportion of actual proactive interventions. This improvement is crucial for ensuring that the AI agent can effectively assist users by recognizing and responding to their needs in a timely manner.

Specific Intervention Types:

- **Overall:** When taking into account all intervention types, both Local and Global variants of YETI substantially outperform the HoloAssist baseline on recall and F-measure, while surpassing precision in every modality of HoloAssist except R+H+E (RGB video + Hand pose + Eye gaze), where it still remains competitive.
- **Confirm Action:** In only considering interventions where the AI Agent should step in to confirm the action the user is taking, YETI continues to achieve substantially higher recall than HoloAssist. The F-measure, which seeks to balance out precision and recall, is also higher than every combination of modalities used for HoloAssist besides R+H+E.
- **Correct Mistake:** While both YETI and HoloAssist struggle to detect when to intervene to correct a mistake by the user, many modalities for HoloAssist fail to detect them entirely, and for the ones that can detect some of these interventions, the recall is very poor. YETI, on the other hand, has extremely high recall but comparatively poor precision. This contributes to YETI having a higher F-measure than all HoloAssist modalities besides HoloAssist RGB[Pt] (model pretrained on ImageNet).
- **Follow Up:** Following up with the user after the user performs an action is where YETI struggles the most compared to the HoloAssist baselines, but it still remains competitive, outperforming the R+H+E and RGB[Pt] baselines on recall. Ironically, even though YETI and R+H+E and RGB[Pt] performed better on every other category, they had the worst results in following up. This indicates that more sophisticated features may complicate the process of detecting Follow-up interventions.

Further Comparison with HoloAssist:

The performance gains of YETI over HoloAssist can be attributed to several factors:

- **Global / Local Context:** YETI's ability to leverage its history contributes to its improved accuracy in detecting proactive interventions. By considering both the broader context of the interaction, as in the case of Global YETI, and the specific details of the current situation, as in the case of local YETI, YETI can better understand the user's intentions and needs.
- **Advanced Feature Representation:** YETI likely employs more streamlined feature representations compared to HoloAssist, allowing it to capture more nuanced aspects of the interaction and make more informed decisions about potential interventions.
- **Data Efficiency:** The smaller data size of SSIM and Object Count Alignment features, as highlighted in Table 1, enables YETI to process information and respond to user actions with minimal latency. This efficiency is crucial for real-time applications in augmented reality environments.

YETI's superior performance in detecting proactive interventions, coupled with its efficiency and suitability for deployment on augmented reality devices, makes it a promising approach for enhancing human-computer interaction in various domains. Its ability to accurately detect when to proactively intervene has the potential to significantly improve user experiences and facilitate more effective collaboration between humans and AI agents.

7. Comparative Analysis of YETI

7.1. Comparison with other Classifier Models

In order to assess the efficacy of our YETI algorithm (Algorithm 1), we trained a Random Forest Classifier, a Decision Tree, and a Multi-Layer Perceptron (MLP) in order to have a comparative analysis. Statistics about the data used to train and evaluate the Random Forest Classifier can be seen in Table 7. Due to the imbalance in labels, it was more statistically safe for the models to assume that every single frame was a negative example of a proactive interaction or a proactive intervention, resulting in accuracy being high but precision, recall, and F-measure being zero. This justifies the use of our algorithm to detect when to intervene rather than relying solely on a model such as these ones, which are sensitive to skewed data distributions.

	Interactions	Interventions	Confirm Action	Correct Mistake	Follow Up	Total
Train	7285 (6.77%)	4562 (4.24%)	2222 (2.07%)	526 (0.49%)	1814 (1.37 %)	107592 (81.43%)
Test	1583 (6.45%)	1005 (4.1%)	474 (1.93%)	142 (0.58%)	389 (1.59%)	24537 (18.57%)
Total	8868 (6.72%)	5567 (4.21%)	2696 (2.04%)	668 (0.51%)	2203 (1.67%)	132129

Table 6. Training and Test splits of Image Frames with Distributions of Proactive Interactions, Proactive Interventions and 3 Proactive Intervention types

	Interactions	Interventions	Confirm Action	Correct Mistake	Follow Up
Random Forest Classifier Decision Tree Classifier MLP Classifier	93.55	95.9	98.41	99.42	98.07
Global YETI (Ours)	86.97	84.85	80.75	79.97	82.36
Local YETI (Ours)	93.76	93.36	93.05	93.32	93.31

Table 7. Accuracies of different classification models in proactivity prediction. Random Forest (RF), Decision Tree (DT) and Multi-Layer Perceptron (MLP) Classifiers have high accuracies as they can predict when the AI Agent should not be proactively interacting or intervening. The same trend is observed in the 3 different proactive intervention type predictions like confirming actions, correcting mistakes, following up instructions. However, RF, DT and MLP classifiers cannot classify any true positive (actual proactive instances) due to the extremely sparse and skewed distribution of proactivity labels in Table 7, thus has 0 precision and recall. These classifiers cannot detect proactivity on-the-fly in real time. Our YETI Proactive Agent Detection Algorithm has good precision and high recall model architectures in detecting proactivity on-the-fly as shown in Table 4.

7.2. Comparison with Implicit Expert-User Agent Setting

In Table 4, we only presented results using data from HoloAssist examples with both student-led and instructor-led conversations. In Tables 8 and 9 we present the results for the 1191 HoloAssist examples where there was not a student-led conversation, instead relying on implicit student-instructor interactions where, for example, the instructor tells the student to do something and the student complies without further comment.

8. Additional Related Works

8.1. Procedural Mistake Detection

Previous studies have explored the analysis of egocentric data to assist with procedural tasks; however, none have approached this challenge in the same comprehensive manner as YETI. YETI is designed to detect optimal moments for proactive intervention. In contrast, existing works primarily focus on mistake detection, which limits direct comparisons with YETI's broader intervention detection capabilities.

PREGO [9] (Mistake Detection in **PR**ocedural **EGO**centric Videos) targets online mistake detection in a manner akin to YETI's real-time intervention detection. Nevertheless, PREGO is confined to the Mistake Detection intervention type and does not address other scenarios where AI intervention could be beneficial. Specifically, PREGO lacks mechanisms for proactive interventions, requiring users to make errors before the AI can respond. Although PREGO incorporates step anticipation, it only detects deviations from predefined plans to identify mistakes. Additionally, PREGO necessitates a symbolic description of the task for mistake identification, whereas YETI operates directly on video frames without such annotations.

TI-PREGO [15] extends the work of PREGO with a more comprehensive integration of Large Language Models (LLMs) for action anticipation and detection modules, incorporating chain-of-thought reasoning and in-context learning. They evaluate the ability of several LLaMA, Mistral, Gemma, and GPT models to perform step anticipation. Similar to PREGO, TI-PREGO remains focused solely on mistake detection within procedural tasks and does not expand into proactive intervention detection.

Differentiable Task Graph Learning [17] also addresses online mistake detection but diverges by utilizing Task Graphs instead of LLM-based symbolic reasoning. Task Graphs model procedures as sequences of steps with directed dependencies, ensuring certain steps precede others. However, this approach requires pre-segmented key-step sequences from input videos, rendering it unsuitable for real-time Augmented Reality applications. In contrast, YETI processes continuous actions without necessitating pre-annotated data, allowing for real-time operation.

	Method	Overall			Confirm Action			Correct Mistake			Follow Up		
		Prec.	Rec.	F-meas.	Prec.	Rec.	F-meas.	Prec.	Rec.	F-meas.	Prec.	Rec.	F-meas.
HoloAssist	(RGB)	13.93	33.33	19.65	0.00	0.00	0.00	0.00	0.00	0.00	41.79	100.00	58.95
	(R+H)	24.89	33.64	28.61	32.14	4.50	7.89	0.00	0.00	0.00	42.52	96.43	59.02
	(R+E)	25.55	33.73	29.08	33.90	10.36	15.87	0.00	0.00	0.00	42.76	90.83	58.15
	(R+H+E)	48.31	37.59	42.28	39.11	40.93	40.00	61.11	9.91	17.05	44.70	61.93	51.92
	(RGB[Pt])	37.54	37.74	37.64	42.31	27.50	33.33	27.33	36.61	31.30	42.97	49.11	45.84
YETI (Ours)	Global	41.86	88.31	56.17	22.54	91.51	36.17	11.68	90.29	20.69	27.55	89.69	42.15
	Local	46.88	60.38	52.77	26.55	68.62	38.29	14.71	68.02	24.18	30.07	62.05	40.51
Implicit	Global	28.96	92.59	44.11	15.68	94.76	26.90	11.28	93.58	20.12	20.47	92.56	33.53
	Local	34.55	71.71	46.63	18.4	77.71	29.76	14.28	74.21	23.95	24.51	72.48	36.64

Table 8. Evaluation of YETI compared to HoloAssist and Implicit Interventions

	Method	Overall			
		Acc.	Prec.	Rec.	F1
YETI (Ours)	Global	86.97	52.23	87.04	65.28
	Local	93.76	64.51	59.73	62.02
Implicit	Global	81.37	35.68	91.9	51.42
	Local	92.62	45.67	71.28	55.67

Table 9. Evaluation of YETI compared to HoloAssist and Implicit Interactions

Eyes Wide Unshut [14] mitigates the reliance on supervised learning observed in Differentiable Task Graph Learning by predicting mistakes based on eye gaze trajectories instead of manual annotations. This method forecasts the user's next gaze position during task execution and compares it with actual gaze data to detect discrepancies. Unlike YETI, Eyes Wide Unshut heavily depends on the availability of eye gaze data, which may not always be accessible. Furthermore, many tasks do not require significant gaze shifts, limiting the method's applicability and increasing the potential for false positives when gaze changes are unrelated to task performance errors.

8.2. Action Detection

Other works have also explored broad action detection, classifying actions detected in videos instead of detecting when to proactively intervene or when a user makes a mistake.

Quasi-Online Detection of Take and Release Actions [16] focuses on near real-time detection, allowing a slight delay between an action occurring and its detection. This work concentrates on identifying "take" and "release" actions—instances where the user's hands interact with objects—rather than on mistake or intervention detection. While such action detection could support downstream tasks like intervention or mistake identification, it does not directly address these areas, thereby limiting its overall scope and applicability for use as an AI assistant.

OadTR [20] is one of the first works to use Transformer-based models for online action detection, pivoting from Recurrent Neural Networks (RNNs), which exhibit less parallelism. Unlike the other works in this list, however, the video in the datasets used to evaluate and OadTR are not egocentric, so it is not clear whether the models would generalize to detect actions from video with a different perspective.

9. Ablation Studies

9.1. Agent Conversation Interval

The Agent Conversation Interval is a parameter for how long we suppose it will take a user to respond to an intervention by the AI Agent. In Table 3 we use a value of one, indicating the user will take about one second to act on the advice of the proactive AI Agent. Here we explore alternative intervals where a user might respond to the intervention within up to five seconds rather than one second.

Table 10 shows precision, recall, and F-measure for detecting each kind of proactive intervention, with varying values for the conversation interval. We see that as we increase the conversation interval, the F-score decreases, showing that overall, a conversation interval of one is the best choice. The same logic follows for detection proactive interactions, shown in Table 11, where we also see that F-measure decreases as the conversation interval increases.

9.2. Extrema Range

The extrema range refers to how close the alignment score needs to be to a minima or maxima (whether local or global) in order for a proactive intervention to be detected. A larger extrema range makes YETI more sensitive, resulting in more proactive detections, while a narrower extrema

Method	Conv. Interval	Overall			Confirm Action			Correct Mistake			Follow Up		
		Prec.	Rec.	F-meas.	Prec.	Rec.	F-meas.	Prec.	Rec.	F-meas.	Prec.	Rec.	F-meas.
Global YETI	1	41.86	88.31	56.17	22.54	91.51	36.17	11.68	90.29	20.69	27.55	89.69	42.15
	2	41.06	84.09	55.18	22.52	88.33	35.89	11.49	86.52	20.29	27.44	85.87	41.59
	3	41.13	80.49	54.44	22.41	85.42	35.50	11.50	83.28	20.22	27.31	82.53	41.04
	4	41.08	77.09	53.61	22.53	82.81	35.43	11.54	80.46	20.19	27.33	79.34	40.65
	5	40.95	73.91	52.70	22.31	80.03	34.89	11.59	77.92	20.19	26.96	76.22	39.83
Local YETI	1	46.88	60.38	52.77	26.55	68.62	38.29	14.71	68.02	24.18	30.07	62.05	40.51
	2	47.01	56.4	51.27	26.54	64.97	37.69	13.92	63.37	22.82	30.39	58.54	40.01
	3	47.15	56.39	51.36	25.93	64.46	36.99	14.35	63.52	23.41	30.75	59.26	40.49
	4	46.43	53.66	49.78	25.78	62.39	36.49	13.32	58.98	21.73	30.41	56.45	39.53
	5	45.81	51.01	48.27	26.05	60.58	36.44	13.37	56.71	21.64	29.47	52.99	37.88

Table 10. Evaluation of YETI for Agent Conversation Intervals for proactive interventions

Method	Conv. Interval	Overall			
		Acc.	Prec.	Rec.	F1
Global YETI	1	86.97	52.23	87.04	65.28
	2	90.29	52.30	82.51	64.02
	3	92.01	52.42	78.63	62.90
	4	93.11	52.69	75.20	61.97
	5	92.65	51.15	76.22	61.22
Local YETI	1	93.76	64.51	59.73	62.02
	2	94.25	63.79	55.47	59.34
	3	94.59	63.43	55.12	58.98
	4	94.77	62.23	52.3	56.84
	5	94.98	61.61	49.79	55.08

Table 11. Comparison over different Agent Conversation interval sizes for proactive interactions.

range leads to less proactive detections. We see in Tables 12 and 13 that an extrema range of zero results in a much lower recall, while an extrema range of two results in a lower precision. Since overall, YETI struggles more with precision than recall, our chosen extrema range is one, which maximizes precision.

9.3. SSIM Thresholding

When we filter out certain frames based on their SSIM score relative to the previous frame, we are abstracting the meaning of what it means to have structural similarity. This abstraction is a form of reasoning.

9.4. Episode Length

Varying the initial history (i.e. Episode Length) needed in order to make the first proactive intervention/interaction did not influence the results from those in Tables 4 and 5. This could be due to the fact that the Episode Length is only relevant for the first few frames of the video, and thus has an extremely narrow window to influence the ultimate output.

10. Additional Use-Cases for Proactive Intervention Detection

The HoloAssist paper presents many additional use cases for an AI Assistant that can proactively intervene when the user is trying to accomplish a task. An exhaustive list of all the uses included in the dataset is as follows:

- Assembly tasks
  - Assembling a computer
  - Assembling a laser scanner
  - Assembling a nightstand
  - Assembling a stool
  - Assembling a tray table
  - Assembling a utility cart
  - Disassembling a nightstand
  - Disassembling a tray table
  - Disassembling a utility cart
- Maintenance and Repair
  - Changing a mechanical belt
  - Changing a circuit breaker
  - Fixing a motorcycle
- Consumer Electronics
  - Making coffee with an espresso machine
  - Making coffee with an espresso machine
  - Setting up a big printer
  - Setting up a camera
  - Setting up a GoPro
  - Setting up a small printer
  - Setting up a Nintendo Switch

Assistive AI Agents that can proactively intervene are particularly useful for tasks such as these, where they can help with:

1. **Complexity Management:** Furniture assembly is a good example of a task that seems straightforward at first

Method	Extrema Ranges	Overall			Confirm Action			Correct Mistake			Follow Up		
		Prec.	Rec.	F-meas.	Prec.	Rec.	F-meas.	Prec.	Rec.	F-meas.	Prec.	Rec.	F-meas.
Global YETI	0	41.34	70.61	52.15	22.52	76.96	34.85	11.09	73.87	19.28	27.13	73.92	39.69
	±1	41.86	88.31	56.17	22.54	91.51	36.17	11.68	90.29	20.69	27.55	89.69	42.15
	±2	41.05	91.95	56.75	22.19	94.09	35.91	11.48	93.08	20.44	27.43	93.04	42.37
Local YETI	0	43.42	26.02	32.54	24.83	33.16	28.39	11.12	28.28	15.96	27.35	27.17	27.26
	±1	46.88	60.38	52.77	26.55	68.62	38.29	14.71	68.02	24.18	30.07	62.05	40.51
	±2	46.74	77.23	58.24	25.98	83.11	39.59	13.34	79.83	28.86	30.63	79.22	44.18

Table 12. Evaluation of YETI for Extrema Ranges for proactive interventions

Method	Extrema Ranges	Overall			
		Acc.	Prec.	Rec.	F1
Global YETI	0	92.14	53.94	68.52	60.36
	±1	86.97	52.23	87.04	65.28
	±2	84.82	51.76	91.09	66.02
Local YETI	0	94.40	68.19	28.17	39.87
	±1	93.76	64.51	59.73	62.02
	±2	91.82	61.39	76.09	67.95

Table 13. Comparison over different Extrema ranges.

but can quickly get out of hand, especially if a mistake made early on is only noticed by the user much later on in the process. A proactive AI Agent intervention could help catch these mistakes early on before they become a larger problem.

2. **Safety Considerations:** Tasks such as changing a mechanical belt or a circuit breaker can open the user up to serious harm if they are not careful. If the user interacts with a live wire while the power is on they could be seriously injured. An AI assistant could potentially proactively intervene before the user makes this mistake.
3. **Expertise Gap:** If a user is new to completing a task that benefits from learned experience, the proactive AI Agent could take on the role of an instructor guiding a student. This would be especially useful for fixing a motorcycle, which is often done by expert mechanics.

The diversity of these use-cases demonstrates that proactive AI assistance has broad applicability in scenarios where expert oversight would traditionally be beneficial. This is particularly valuable in contexts where users might not possess sufficient domain knowledge to self-identify potential issues or formulate appropriate queries for assistance.

A Proactive AI Agent can help in many industrial or home maintenance tasks like how to change an electrical task in Figure 6. The AI Agent helps to guide the user on how to change the circuit breaker and proactively intervenes without any questions from the user when the AI Agent observes that the user may touch the electric circuit which will be a safety worry. The Extrema of the changing object count for the corresponding frame is helpful to determine when the AI Agent should intervene proactively, post filtering by the SSIM signal. Filtering helps to obtain an abstract scene understanding capability to the Proactive AI Agent. The expectation is that the frames where the AI Agent should intervene are dissimilar and can be observed by SSIM while pruning the edge cases by filtering.

Method	SSIM	Overall			Confirm Action			Correct Mistake			Follow Up		
		Prec.	Rec.	F-meas.	Prec.	Rec.	F-meas.	Prec.	Rec.	F-meas.	Prec.	Rec.	F-meas.
Global YETI	0.5	43.20	71.83	53.95	23.86	77.14	36.45	11.40	71.87	19.68	30.01	76.74	43.14
	0.6	42.55	79.60	55.46	23.07	83.90	36.19	11.70	81.33	20.46	29.53	83.02	43.56
	0.7	41.71	85.49	56.06	22.56	88.93	35.99	11.88	87.77	20.93	28.81	87.69	43.36
	0.8	41.29	87.38	56.08	22.44	90.64	35.98	11.87	89.37	20.96	27.99	89.05	40.59
	0.9	41.86	88.31	56.17	22.54	91.51	36.17	11.68	90.29	20.69	27.55	89.69	42.15
Local YETI	0.5	47.35	43.05	45.09	25.88	50.31	34.18	14.23	45.22	21.65	32.66	48.00	38.87
	0.6	45.55	49.19	47.30	24.60	56.28	34.24	14.00	54.62	22.29	30.84	53.22	39.05
	0.7	45.93	55.86	50.41	25.09	63.09	35.91	14.50	63.14	23.59	30.44	58.83	40.12
	0.8	47.09	59.52	52.58	26.5	67.61	38.08	14.98	67.39	24.51	30.45	61.47	40.73
	0.9	46.88	60.38	52.77	26.55	68.62	38.29	14.71	68.02	24.18	30.07	62.05	40.51

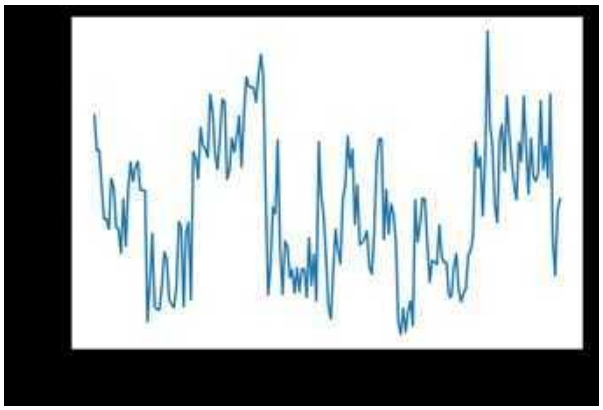
Table 14. Evaluation of YETI for SSIM Thresholding for proactive interventions

(a) Intervention - 3 seconds

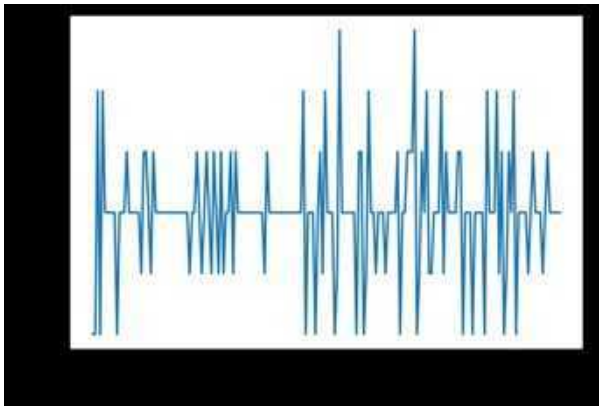
(b) Intervention - 2 seconds

(c) Intervention - 1 second

(d) AI Agent proactively intervenes.



(e) SSIM between consecutive frames.



(f) Changing Objects Alignment Signal

Figure 6. Intervention Detection for Changing Electric Circuit Task.

Method	SSIM	Overall			
		Acc.	Prec.	Rec.	F1
Global YETI	0.5	91.49	55.38	68.88	61.39
	0.6	90.20	54.14	77.36	63.69
	0.7	88.62	53.06	83.92	65.01
	0.8	87.55	52.39	85.99	65.11
	0.9	86.97	52.23	87.04	65.28
Local YETI	0.5	94.13	64.25	41.12	50.14
	0.6	93.98	62.75	48.05	54.42
	0.7	93.88	63.46	55.14	59.01
	0.8	93.81	64.42	58.65	61.40
	0.9	93.76	64.51	59.73	62.02

Table 15. Comparison over different SSIM Thresholding levels for proactive interactions.