Pedagogical Agent Responsive to Eye Tracking in Educational VR

Adil Khokhar Andrew Yoshimura Christoph W. Borst

University of Louisiana at Lafayette

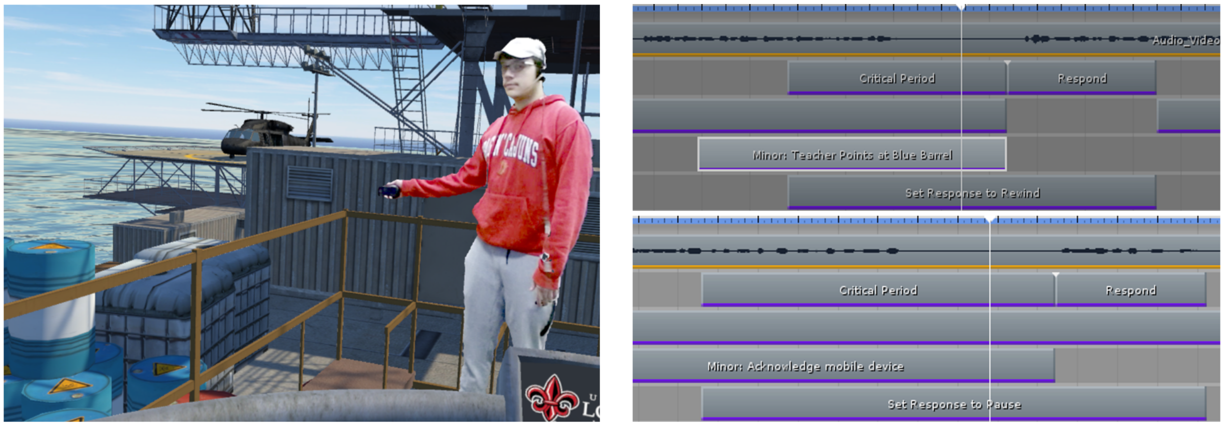

Figure 1: Left: Teacher agent points at a barrel. If the student does not look, the agent may pause or replay a phrase, depending on response ranks. Right: Two timelines with different responses. The student is required to fulfill certain conditions in critical periods.

Abstract

We present an architecture to make a VR pedagogical agent responsive to shifts in user attention monitored by eye tracking. The behavior-based AI includes low-level sensor elements, sensor combiners that compute attention metrics for higher-level sensors called generalized hotspots, an annotation system for arranging scene elements and responses, and its response selection system. We show that the techniques can control the playback of teacher avatar clips that point out and explain objects in a VR oil rig for training.

Index Terms: Human-centered computing—Visualization

1 Introduction

We previously showed virtual field trips with both a live teacher and sequenced prerecorded teacher clips [1]. Both approaches were well-received by students, but the live teacher resulted in better test scores. However, the live approach does not scale well, due to extra networking, equipment, planning, and teacher time. We therefore consider improved environment and clip responses based on student attention or distraction. Distractions can cause students to miss critical information. For example, a student's attention drifts as they think about lunch while a teacher explains how a turbine works, and the student does not look at turbine parts pointed to by the teacher.

Eye gaze is a useful indicator for attention and real live teachers may adapt their instruction to guide student attention. Eye tracking [5] can provide a mechanism for a system to monitor and respond to focus shifts. Pedagogical agents provide an educational benefit [10] over [3] [6] [9] [5] [7] [4] a fixed presentation sequence by being more interesting and could respond to misunderstandings or attentional shifts.

We describe an architecture for a pedagogical agent, that uses animated clips, to be responsive to student eye tracking. This allows it to behave more appropriately during distraction. We illustrate initial results in a VR oil rig where trainees receive a rig overview.

2 Sensing System

The agent senses attentional shifts with generalized hotspots that are built from components combining low-level sensor information.

Low-level sensors retrieve device data, for example, from an eye tracker. Eye movements, pupil dilation, and gaze patterns are useful in detecting user attention [10]. Each sensor receives a single input and passes it to a combiner system. Other sensor types are included to support specific sequences in educational activities, e.g., controller input and aim, agent pose, and the state of a game object.

Combiners receive multiple inputs and produce a single output based on a math or logic operator. Mappers transform an input by a function, for example, to apply a nonlinear response curve or apply a temporal filter. These components compose low-level sensors to output a single inattention score for a higher-level sensor that we refer to as a generalized hotspot.

We generalize from standard response-triggering hotspots (e.g., circular gaze targets in Google Expeditions). First, whereas standard hotspots directly trigger a response, our generalized hotspot is a conceptual high-level sensor that sets up candidate behaviors for further consideration by a response selection mechanism (Section 4). Additionally, a generalized hotspot may be associated with multiple scene areas (objects) and be used to respond to gaze drifting away as well as to targeted focus. Some hotspots can be automatically created based on analyzed teacher pointing (e.g., to estimate where students should look), and a director can later adjust these. Activation and control of generalized hotspots are based on annotations described in Section 3. For example: a generalized hotspot is annotated to set up pause and replay as candidate responses when gaze drifts.

Suppose we want the agent to be affected when a student is neither focused on the agent nor on the object being described. Combiners receive low-level sensor data and compute eye gaze angles away from relevant objects (e.g., teacher and described devices). The minimum angle will be mapped by a sigmoid function to produce an inattention score between 0 and 1, with 1 corresponding to a very distant gaze. A low-pass temporal filter and a thresholding mapper or accumulator could be added to only affect behavior when the student looks far away for some time. The resulting generalized hotspot could be used to set up pause, replay, and an attention-getting clip as

This is an author-formatted version. Original publication: A. Khokhar, A. Yoshimura, and C .W. Borst, "Pedagogical Agent Responsive to Eye Tracking in Educational VR", IEEE Virtual Reality 2019; Osaka, Japan, 2019 candidate responses. The agent could point at a crane and pause if the student is looking slightly away. If the student is looking very far away, an audio clip could play to ask if the student needs assistance. The system could include histories to choose different clips based on what the agent or student have done before (Section 4).

3 Timeline Metadata

Unity's Timeline feature is used to coordinate audio, animation, and game object activations. We extended this to support annotations (metadata) about teacher content. Annotation tracks differ from standard timeline tracks in that annotations provide contextual information to the agent and an adjustable mechanism to control how prerecorded content on the timeline is sequenced by agent responses. The annotations also provide a way to handle playback, seeking, rewinding, and stopping teacher clips in timeline tracks.

We included 5 annotation types:

Major: Marks an interval of contextual information representing a topic and its associated default sequence.

Minor: A sentence or action that affects a response. For example, it can let an agent find the start of the current sentence for replay, pause while pointing, or know when an interruption is appropriate.

Critical Period: A time interval during which a generalized hotspot is in effect, i.e., when the agent monitors the hotspot score.

Respond: A marker where a response may occur.

Promote Responses: Promotes a set of responses to be candidate responses based on associated hotspots and an optional time interval, affecting ranks in the response selection mechanism (Section 4). Omitting the interval makes a hotspot always-on. For example: an always-on hotspot detects the student's gaze away from the current educational object for display of an attention-guiding arrow.

The timing annotations allow adjustments to the agent's behavior based on when it is appropriate in the educational activity. In Figure 1, the top timeline shows a sequence where a teacher points to a barrel and a critical period activates certain hotspots. A replay response is set to subsume other responses if the student does not meet the requirement of fixating at the barrel for a second within the critical period (then the teacher will repeat from the beginning of the sentence, for example). If the student does look then the teacher continues the default behavior of following the normal linear sequence. This default behavior will be subsumed by a promoted response if the critical period requirement is not met. The bottom timeline is a sequence where the agent requires the student to acknowledge a text on a mobile device. A pause response is promoted: if the student does not look at the phone and press a button within a critical period, the agent will pause until the requirement is met.

So, a timeline specifies which responses may subsume the default agent behavior, what are the critical points where an agent responds, and what a student needs to do to progress the sequence.

4 Teacher Response Selection

We integrated our approaches into an established behavior-based AI framework to allow more general control of agent and environment responses. Behavior-based AI methods, such as Utility AI, resemble the subsumption architecture from robotics [2]. The main idea of subsumption is that behaviors are selected using sensor-based conditions and behavior priority. We follow the GAIA architecture for agent response [4]. Responses are associated with ranks and the highest ranked category is selected based on responses generated. The response description gives rank to the responses by composing a combined inattention score with contextual metadata from the timeline. Differing from the original architecture, we do not use weighted random determination of responses.

The agent's response description uses the techniques described earlier to calculate a rank and choose the response with the highest utility. The default behavior for the teacher is to continue a linear presentation, so this behavior's rank is initialized to 0. Other responses' ranks are initialized to -1. Subsequently, ranks are computed based on the combined inattention score and contextual information. Contextual information may include timeline annotations, player history, timers, candidate responses, and agent execution history.

For example, suppose the agent points out a specific button on a handheld controller, to be pressed by the student as practice. The timeline will include the Critical Period, Minor, Promote Responses, and Respond information. A hotspot detects if the student has not pressed the button. The Promote Responses annotation associated with the hotspot promotes candidate responses by elevating their ranks. A candidate response will subsume the default behavior if the requirement is not met when the teacher points at the controller and the Respond annotation is reached. Suppose, optionally, the responses are to be constrained in order: pause, replay, or play a different clip. The constraint is applied using response ranks. The teacher will first pause while pointing and a cooldown timer will prevent other responses from activating by temporarily demoting their ranks so the student has time to meet the requirement. After the timer ends, the execution history will demote pause's rank to -1. Then the next response "Replay" can be selected and another cooldown timer starts. Replay causes the teacher repeat the sentence and point once again. If the student again doesn't meet the requirement, the execution history demotes replay and the next response will make the teacher play a different clip acknowledging the student's inability to complete the task and perhaps offering help or moving on to a different activity. The inattention score will also affect the sequence. If the inattention score is very high, e.g., a very distant gaze or a history of extraneous inputs, then the last response from the order above, playing a different clip, can be promoted in rank to subsume other responses immediately. With increasingly complex examples, the benefit of the AI architecture is that the agent can be more dynamic and extensible without an extensive set of explicit if-then conditions.

5 Conclusion

Future studies will investigate the subjective suitability and desirable timing of different responses such as the pause, replay, and a combination with attention-restoring visual cues [8]. We will gather eye tracking data, analyze the data for insights into users' attention, and determine the feasibility of various eye tracking sensors in their role in attention and how eyes behave in different attentional states.

Acknowledgments

This material is based upon work supported by the National Science Foundation under Grant Nos. 1815976 and 1451833.

References

[1] C. W. Borst, N. G. Lipari, and J. W. Woodworth. Teacher-guided educational vr: Assessment of live and prerecorded teachers guiding virtual field trips. In *2018 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pages 467–474, March 2018.

[2] R. Brooks. A robust layered control system for a mobile robot. *IEEE Journal on Robotics and Automation*, 2(1):14–23, March 1986.

[3] J. J. Bryson. The behavior-oriented design of modular agent intelligence. In *Proceedings of the NODe 2002 Agent-related Conference on Agent Technologies, Infrastructures, Tools, and Applications for E-services*, NODe'02, pages 61–76, Berlin, Heidelberg, 2003. Springer-Verlag.

[4] K. Dill, E. R. Pursel, P. Garrity, G. Fragomeni, and V. Quantico. Design patterns for the configuration of utility-based ai. In *Interservice/Industry Training, Simulation, and Education Conference (I/ITSEC)*, number 12146, pages 1–12, 2012.

[5] S. D'Mello, A. Olney, C. Williams, and P. Hays. Gaze tutor: A gaze-reactive intelligent tutoring system. *Int. J. Hum.-Comput. Stud.*, 70(5):377–398, May 2012.

[6] S. Hutt, C. Mills, N. Bosch, K. Krasich, J. Brockmole, and S. D'Mello. "out of the fr-eye-ing pan": Towards gaze-based models of attention

during learning with technology in the classroom. In *Proceedings of the 25th Conference on User Modeling, Adaptation and Personalization*, UMAP '17, pages 94–103, New York, NY, USA, 2017. ACM.

[7] W. L. Johnson, J. Rickel, R. Stiles, and A. Munro. Integrating pedagogical agents into virtual environments. *Presence*, 7(6):523–546, Dec 1998.

[8] A. Khokhar, A. Yoshimura, and C. W. Borst. Eye-gaze-triggered visual cues to restore attention in educational vr. In *2019 IEEE Conference on Virtual Reality and 3D User Interfaces (VR), Poster*, March 2019.

[9] A. Poole and L. J. Ball. Eye tracking in human-computer interaction and usability research: Current status and future. In *Prospects, Chapter in C. Ghaoui (Ed.): Encyclopedia of Human-Computer Interaction. Pennsylvania: Idea Group, Inc*, 2005.

[10] H. Wang, M. Chignell, and M. Ishizuka. Empathic tutoring software agents using real-time eye tracking. In *Proceedings of the 2006 Symposium on Eye Tracking Research & Applications*, ETRA '06, pages 73–78, New York, NY, USA, 2006. ACM.