Wasta, C., Wu, S., & Ritter, F. E. (in press, 2025). CogDriver: The longest-running autonomous driving cognitive model exhibits human factors. *16th International Conference on Applied Human Factors and Ergonomics (AHFE 2025).*

CogDriver: The Longest-Running Autonomous Driving Cognitive Model Exhibits Human Factors

Christian Wasta (ccw5294@psu.edu), Siyu Wu (sfw5621@psu.edu),
and Frank E. Ritter, (frank.ritter@psu.edu)
College of Information Sciences and Technology,
Pennsylvania State University, University Park, PA 16802 USA

2 February 2025
venues: https://ahfe.org/submissions.html, abstract due Dec 16th
https://cognitivesciencesociety.org/cogsci-2025/, full paper due Feb 1st, 2025

Abstract

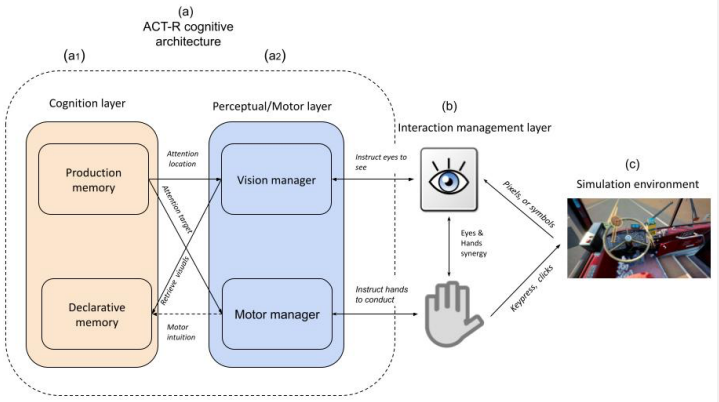
We are exploring how models can use models of human perception and motor control to interact directly with interfaces. This opens new issues. We present CogDriver, a cognitive driving model capable of performing a long-duration autonomous driving task in a virtual simulation environment. This model, built using the ACT-R cognitive architecture and enhanced with robotic hands and eyes, supports the cognitive-perceptual-motor knowledge essential for simple human driving. It has two main strengths compared to other autonomous driving models: (a) it built upon human-observed driving behavior, incorporating error-making and learning, and (b) it leverages a cognitive architecture to provide insights into psychological driving behavior. Compared to our previous version, this model shows improved endurance, maintaining its driving state for over 16 hours from Tucson to Las Vegas, even under nighttime conditions. The enhancements were realized through incorporating human-like driving knowledge representations, and actions. It now includes a model of error handling and several logical visual cue strategies. The model's predictions can match certain aspects of human behavior in fine detail, such as the number of course corrections, average speed, learning rate, and adaptation to low visibility conditions. This model demonstrates that (a) perception and action loops with fallback handling provide a very accessible testbed for examining further aspects of behavior and (b) the model-task combination supports exploring aspects of human behavior that remain missing from ACT-R.

Keywords: Cognitive autonomous driving; ACT-R cognitive architecture

Introduction

Current autonomous driving has focused on real-time simulation with artificial sensor systems (CITE). However, our approach through cognitive modeling provides the opportunity to add human factors until a human-like autonomous simulation is made. Since cognitive architectures (CA) can develop cognitive models of various psychological phenomena and tasks (Newell, 1990), they also provide procedures and structures that align with human behavior, such as reaction times, error rates, and fMRI results (Anderson, 2007; Laird, 2019). In this study, we developed a human-like cognitive model that can perform an autonomous driving task in a virtual simulation environment. The model has two strengths compared to other autonomous driving models on similar tasks: 1) it built upon human-observed driving behavior, incorporating error-making and learning, and (2) it leverages cognitive architecture to provide insights into psychological driving behavior.

CogDriver architecture is shown in figure1.



This CogDriver architecture represents a closed-loop system (Trapp, Schroll, Hamker, 2012) where cognitive reasoning (through production and declarative memory) interacts with perceptual and motor processes. It begins with the (a) ACT-R Cognitive Architecture, which serves as the foundation for building a cognitive model. This architecture is composed of a (a1) cognition layer with production memory, which encodes human subjects' procedural knowledge for decision-making, actions, and directing attention to specific targets within the environment. And declarative memory, which stores subjects' factual knowledge, retrieved visual information, and provides motor intuition to guide task execution. (a2) The Perceptual/Motor Layer includes a vision manager, which manages visual attention and perception by instructing the eyes to focus on specific locations in the environment. The visual system processes chunks of

information about an object's location in the "where" buffer and information about objects in the visual scene in the "what" buffer. The motor manager coordinates motor actions, such as steering and keypresses, based on instructions from the cognition layer. The cognition and perceptual/motor layers are tightly integrated. The central production system can reason about chunks of information stored in the visual buffers to guide behavior. In a driving context, this enables the model to move forward or steer based on position data retrieved from the visual buffer (Ritter et al., 2019). However, the ACT-R model is not complete with the restriction that interaction knowledge cannot work on unaltered tasks. In this work, we extend it to include new types of interaction knowledge and the capability to interact with all tasks that have a computer interface represented by a screen and can be interacted with using a keyboard and a mouse.

To extend the interaction knowledge of the cognitive model, we use the (b) Interaction Management Layer which facilitates the synergy between visual and motor functions. This layer allows the cognitive model to process inputs and execute outputs through visual functions (e.g., whatIsOnScreen to identify visual patterns, whereIs to locate patterns, getMouseLocation to track the mouse's position) and motor functions (e.g., click to simulate mouse clicks, Keypress to replicate key presses, moveCursorTo to move the mouse to a location). These functions enable the system to interact directly with (c) unaltered simulation environment, using the screen's bitmap to detect objects and respond accordingly. For example, the system analyzes pixels or symbols to identify objects and locations on the screen, guiding motor responses like steering or clicking. The integration of these components ensures the model can dynamically adapt to and interact with its environment in a human-like manner guided by the cognitive model. Compared to the previous cognitive driving model for the same simulation environment (Wu et al., 2023), the model has demonstrated improved endurance, maintaining its driving state for over 16 hours, even under dynamic changes in the simulation environment (from daytime to nighttime), achieved through incorporating human factors into the ACT-R model and interaction layer.

The following section introduces the task, the model design and development, the model evaluation, and the implications in the field of autonomous driving.

The Task

Penn and Teller created the video game *Desert Bus* with the intention of making a statement about video games. The game is deliberately monotonous and lengthy, with the player driving a bus in real-time at a maximum speed of 45 mph from Tucson, AZ to Las Vegas, NV. Each leg takes 360 miles to complete, or at least eight hours at maximum speed, and the bus continuously drifts to the right. If the player swerves off the road, the engine will stall, and they will need to start over from Tucson. The game has no virtual passengers or other cars on the road. Once the player completes the 360-mile journey, the screen fades to black, and they return to the starting point to play again indefinitely. At night, the road is dark. Figure 1 (c) provides a screenshot of the game available through Steam (there are other versions available now).

This game offers the player a first-person view as they carry out tasks, and the surroundings change dynamically based on their actions. The specific edition that we use was created by Dinosaur Games and released by Gearbox Software, based on the unreleased "Smoke and Mirrors" Sega CD game. The game's driving environment, Desert Bus, was obtained from Steam (https://store.steampowered.com/app/638110/Desert_Bus_VR/) and can be downloaded for free on Windows machines. It should be noted that the game supports both virtual reality (VR) and 2D (non-VR) environments. All testing was done in the non-VR environment although future work could expand support for virtual reality headsets. There were no alterations made to the game to support the model.

Related Work

Cognition Architecture and Cognitive Models

To create a cognitive driving model, we bring a suite of tools rooted in cognitive architecture (CAs) (C. Breazeal, A. Edsinger, P. Fitzpatrick and B. Scassellati, 2001). CAs are computational frameworks designed to capture the invariant mechanisms of human cognition. These mechanisms include functions related to attention, control, learning, memory, adaptivity, perception, and action. Cognitive architectures propose a set of fixed mechanisms to model human behavior, functioning akin to agents and aiming for a unified representation of the mind. By using task-specific knowledge, these architectures not only simulate but also explain behavior through direct examination and real-time reasoning tracing. One representative cognitive architecture is ACT-R. ACT-R is a theory of simulating and understanding human cognition (Anderson, 2007; Ritter, Tehranchi, & Oury, 2019). Its theory is embodied in the ACT-R software, through which we can construct models that can store, retrieve, and process knowledge, as well as explain and predict performance (Anderson, 1996; Bothell, 2017). There are currently two kinds of knowledge representations in ACT-R, and they are declarative knowledge and procedural knowledge. Declarative knowledge consists of chunks of memory (e.g., apple is a kind of fruit), while procedural knowledge performs basic operations, moves data among buffers, and identifies the next instructions to be executed (e.g., to submit your answer, you have to click the submit bottom). When the model is driving a bus in a first-person perspective, these pieces of information will contain information such as what visual items to look at and what tasks to do next. ACT-R is a cognitive architecture and a

theory of simulating and understanding human cognition (Anderson, 2007; Ritter, Tehranchi, & Oury, 2019). Its theory is embodied in the ACT-R software, through which we can construct models that can store, retrieve, and process knowledge, as well as explain and predict performance (Bothell, 2017).

ACT-R is not complete, like all models. In this work we extend it to include new types of interaction knowledge and the capability to interact with all tasks that have a computer interface that is represented with a screen and that can be interacted with a keyboard and a mouse.

The Architecture of Interaction

Models interact with the world through their visual and motor systems. The interaction includes processing visual items presented (visual systems), pressing keys, and moving and clicking the mouse (motor systems).

Specifically, the visual system holds chunks of information about an object's location in the "where" buffer and chunks of information about objects in the visual scene in the "what" buffer. A central production system can reason about and lead to behavior based on these chunks. For example, the driving model may move forward, or steer based on the position data retrieved from the visual buffer (Ritter et al., 2019).

Models can interact with the simulation, but the approach we will use is to use the screen's bitmap directly to find objects. Motor output can be put on the USB bus and appear as if a user at the keyboard typed characters or moved the mouse. In Table 2, we list previous models' history of interaction using this approach.

Table 2: Previous models history of interaction.

Name of model	Interaction tool	Reference
Eyes and Hands	ESegman	(Tehranchi & Ritter, 2017)
Biased coin	JSegman	(Tehranchi & Ritter, 2020)
Spreadsheet	JSegman	(Tehranchi & Ritter, 2020)
Desertbus 1	JSegman	(Schwartz et al., 2020)
Heads and Tails	VisiTor	(Bagherzadehkhorasani & Tehranchi, 2022)
Desertbus 2	VisiTor	(Wu, Bagherzadeh, & Ritter, 2023)
Desertbus 3	VisiTor	(This paper)

VisiTor (Bagherzadeh & Tehranchi, 2022) is a Python software package stored on a public GitHub that has been developed to provide simulated hands and eyes. It is comprised of two types of functions—motor and visual. The visual functions include "whatIsOnScreen", which checks if certain visual patterns are present in the environment, "whereIs", which locates a pattern within a defined module, and "getMouseLocation", which retrieves the mouse's location. The motor functions consist of "click", which imitates a single mouse click, "Keypress," which replicates the pressing a key, "moveCursorTo", which emulates mouse movement to a specific screen location, and "moveCursorToPattern", which replicates mouse movement to a specific visual pattern.

CogDriver

This section starts with capturing intuition and domain knowledge from the human subjects, followed by the model structure and learning mechanism, and concludes by examining a model's driving performance.

Incorporating Human Factors into Model Design

The model, built upon human factors distilled from the behavior of human subjects in driving simulations, incorporates how cognitive models are designed for human-like driving simulations. Data collection... Data analysis, and how the cognitive models map the collected data. (to be continued, Siyu)

Declarative Chunks

The model has two types of chunks, and a total of 12 declarative memories, which are working memories that tell the model to make the action based on the visual cues it saw. The first chunk is named "drive" and has two slots, "strategy" and "state", with state having parameters as object items. Another chunk type is "encoding", which has slots for the screen-x locations of the two visual cues and a deviation slot.

Procedural Memories

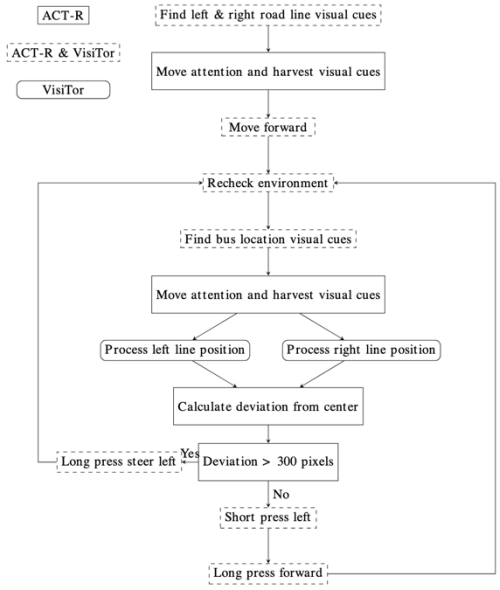


Figure NEW: Control loop of the model.

To improve the model's lack of performance in nighttime environments, the revised cognitive model marked improvements through several key architectural and behavioral areas. The most fundamental change involves the transition from a single-reference visual system (by monitoring the center yellow line) to a dual-reference system that tracks both left and right white road lines. These road lines both create a greater contrast against the road's pavement during the night and allow for greater control over steering outputs depending on where the bus is located within the road. These modifications can be seen with the updated production rules named "whereisdanger" and "whereiscenter".

Talk about FDUCS book here.

Colors at lower illumination lose hue and cause road lines to blend into one another. For this reason, VisiTor naturally recognizes the color white, with a higher contrast ratio, than yellow. [Add in the human centered factor to this]

Two additional procedures were added to make the model drive more like a human. With some additional error handling and continuous operation mechanisms, the "continue-cycle" and "handle-missing-cue" production ensure the model maintains forward progress even when visual cues are temporarily lost.

More significant enhancements involve the model's motor control system, which now includes both long and short duration key presses for a more nuanced vehicle control experience. Demonstrated in the "consider-ahead" production, deviations from the left road line of more than 300 pixels now result in a "short-keypress" which keeps the model in the right road lane.

Using the knowledge found from Foundations for Designing User Centered Systems [FDUCS citation], transitioning from centerline tracking to edge-line detection would most likely significantly improve nighttime performance, as white edge lines maintain better visibility in low-light conditions than yellow.

This model uses an explicit goal state to control the model. It contained 13 production rules and now contains an additional 2 which significantly advance the model's autonomous longevity through those added procedures. Table 3 lists the high-level descriptions of the steps the model performs and the corresponding production rules.

Step 4 handles missing cues by adding error recovery by providing a fallback mechanism for unclear or undetectable road lines. It also ensures the model does not halt if visual information is not immediately found when started. Finally, this procedure bridges the gap between step 3 and 5 by providing an alternative path when "where-is-danger" and "where-is-center" can't find cues.

Step 7 ensures the model always has a valid state transition from the end of the action sequence to the beginning of the perception sequence. "Continue-cycle" prevents dead-end states where no production can fire due to "consider-steer" not firing. Step 7 allows for a reassessment of the current environment, meaning the model no longer freezes if it is unable to find a road line during the nighttime environment at around 8 hours into the game.

Table 3: High Level Description of the Steps and the Production Rules. Remove fill

High level descriptions of steps	**Corresponding productions**
1. When it detects a start visual cue, attend it, and press the "W" key using the manual buffer	Go PerceiveEnvironment Move-attention Ahead
2. Clear the visual buffer and attend to the bus location	Recheck-environment Danger Finding-danger Move-attention-danger
3. Calculate the bus deviation from the center lane	Where-is-danger Where-is-center Calculate-deviation
4. Checks if both cues are missing and continues to move forward if true	Handle-missing-cue
5. Use the manual buffer by pressing "w" if the deviation is less than 200 pixels	Consider-ahead
6. Clear the manual buffer if the deviation > 200 pixels. Using the manual buffer, align the bus by pressing the key for 6 seconds.	Consider-steer
7. Sets state to perceive and resets goals	Continue-cycle

The Mechanism of Interaction

Here we can talk in details on how the extended Visitor help ACT-R model interacts with the environment. Most of the previous wrote parts can be reused here.

This research advances the capabilities of cognitive modeling for extended driving tasks by developing enhancements to the model created by Schwartz et al. (2020). [Desertbus 2] built on ACT-R7's architecture, utilizing an enhanced version of the Perceptual-Motor module (MCL). This was done by creating VisiTor (Bagherzadeh & Tehranchi, 2022). VisiTor functions as a vision manager tool that receives motor commands from the ACT-R PM module and sends them to the environment through an Emacs/slime link. By using this tool, ACT-R can engage with any environment while

maintaining operations that are as similar as possible to those of the user. When visual patterns are detected, ACT-R executes production rules that control the bus through a combination of continuous forward movements (through the "W" key) and steering controls (the "A" key).

ACT-R instructs VisiTor to scan the screen for particular pixel patterns that activate a production rule to initiate the program. As opposed to previous iterations, ACT-R now detects both left and right road lines, of which, coordinates are sent to the Emacs/slime link to initiate the driving productions. ACT-R productions still include only a left and forward steering control, since the bus has a wheel misalignment causing it to always steer right. The primary control loop begins with a continuous forward keep press via "W", while simultaneously monitoring the deviations between both coordinates from the road lines. The model calculates these deviations by measuring the distance between left road line (DrivingCueDanger.png) and right road line (drivingCueTest.png). When this deviation exceeds 300 pixels, indicating the bus is drifting to the right, the model runs a short keep press of "A" to steer the bus to the left.

To undertake this task, VisiTor requires one minor extension. [Desertbus 3] now incorporates both short and long-duration key presses, allowing for specific control based on deviation thresholds. To support extending the overall driving duration to more than 4 hours, we implemented several key enhancements to ACT-R including improved visual object processing, more duration-based motor commands, and error handling and logging mechanisms.

With added error logging, handling, and a continuously looping event procedure, the model successfully completes the goal of creating the longest driving cognitive model. While the model successfully completed this extended task, its behavior revealed new insights into cognitive modeling, particularly regarding sustained driving performance and decision-making during changing environmental conditions.

Model Performance Evaluation

The updated model's design relies on two improved aspects: the cognitive architecture and behavioral components.

The experiment involved running the added behavioral factors to assess its performance and collect ACT-R output data using the new error handling and complementary mechanisms.

The introduction of the "handle-missing-cue" production and dual reference visual cue systems allowed the model to maintain operation even when visual cues become temporarily out of sight. Furthermore, the improvements made to the model allowed the bus to drive at night-time for the first time.

As seen in Figure 4, the screenshot shows the bus operating at low light conditions with only headlights illuminating the road. Checking for two reference images instead of one allows the model to compensate for either road line being too dim or undetectable. In Figure 4, this is shown by the headlights slightly illuminating the right side of the road more than the other.

Figure 4: Screenshot of first nighttime test successfully working.

As seen in Table 5, The ACT-R output data revealed that the total decision-making time the model took to detect the first visual cue to action execution was 0.450 seconds. This is an improvement of 0.45 seconds from the previous model, which slowly makes the model's reaction time equivalent to the average of a human, which is around 250 milliseconds.

"On the order of hours, we will see the model will outperform humans. This allows the model to accomplish a task in driving the bus that surpasses human capability, as it does not experience fatigue or mistakes (Gunzelmann, Moore, Salvucci, & Gluck, 2011)." (Wu et al., 2023). At this point, the bus can now reliably drive in little to no light nighttime conditions, and at this point, was ready to drive from Tucson to Las Vegas for the entire 360-mile journey.

Table 5: UPDATED Model running output

```
CL-USER>  (run 10)
          0.000  GOAL           SET-BUFFER-CHUNK GOAL GOER NIL
          0.000  VISION         SET-BUFFER-CHUNK VISUAL-LOCATION CHUNKO NIL
          0.050  PROCEDURAL     PRODUCTION-FIRED GO
          Ready to go
          0.100  PROCEDURAL     PRODUCTION-FIRED CONTINUE-CYCLE
          Continuing cycle — choose new strategy
          0.150  PROCEDURAL     PRODUCTION-FIRED DANGER
          0.200  PROCEDURAL     PRODUCTION-FIRED FINDING DANGER
          0.200  VISION         SET-BUFFER-CHUNK VISUAL-LOCATION CHUNKO
          0.335  VISION         SET-BUFFER-CHUNK VISUAL CHUNK2
          0.385  PROCEDURAL     PRODUCTION-FIRED MOVE-ATTENTION-DANGER
          0.435  VISION         SET-BUFFER-CHUNK WHEREISDANGER
          Executing command: [Python file directory]
          Python script error output: [Returns blank if successful]
          Full path being searched: [Visual cue #1 file directory]
          File not found: [Output directory if file not found]
          File found: [Output directory when file is found]
          Pattern found at: (497.5, 671.0)
          Python script exit code: 0
          0.470  VISION         SET-BUFFER-CHUNK VISUAL CHUNK3
          0.635  IMAGINAL       SET-BUFFER-CHUNK-FROM-SPEC IMAGINAL
          0.685  PROCEDURAL     PRODUCTION-FIRED WHEREISCENTER
          Executing command: [Python file directory]
          Python script error output: [Returns blank if successful]
          Full path being searched: [Visual cue #2 file directory]
          File not found: [Output directory if file not found]
          File found: [Output directory when file is found]
          Pattern found at: (1344.5, 732.5)
          Python script exit code: 0
          0.785   VISION         PRODUCTION-FIRED CALCULATE-DEVIATION
          0.785   VISION.        PRODUCTION-FIRED CONSIDER-AHEAD
          Decision made: (COND ((= 497.5 -1) MOVE FORWARD
                   ((= 1344.5 -1) STEER LEFT
                   ((> 847.0 300) (T move forward)
```

Results

As with any cognitive model, the results vary on a case-by-case basis. After trialing the model by running the bus in the middle of the night, a second test was run from the game's start to finish. At approximately 8 hours of driving, the game's night cycle has already started. It is at this time in which the game has yet to turn on the bus's headlights and the lack of sunlight causes the model to be unable to distinguish the difference between yellow and white road lines due to how similar they look.

Because the model relies on the rightmost white road line to steer left and stay centered, the middle yellow road line is mistakenly taken as the rightmost whenever it crosses over the yellow median. Around this time, the bus enters the oncoming lane and is unable to steer back, resulting in the bus steering too far into the left dirt shoulder and ending the game. Figure 6 shows a comparison between the yellow and white road lines after 8 hours of gameplay.

Philosophy of Software Design Ch. 20 Measure talks about how the two best ways to solve this include a coded approach or a human centered approach. Because of this, the model is not to blame for making such a mistake. After hours of driving and a low visibility road environment, human drivers may mistake which road line is which. Two solutions to such a problem were to either create a "short term memory" model that remembers the approximate last location the rightmost white line was at. A second, and albeit easier solution, was to always keep the model in the correct road lane.

The final cognitive model significantly improves the driving behaviors of all previous model versions. The core enhancements to achieve this included a continue-cycle procedure which reruns the model regardless of failed visual cue operations, and a variable-duration steering system that mimics human keystroke actuation disparities and adjustments when steering the bus. The model's production cycle consistently executes in 0.45 seconds from visual detection to action, demonstrating a streamlined stability regardless of environmental differences. To show this, the new production code demonstrates more minute steering to the left which continues at full throttle forwards. Most players drive as fast as possible while tapping left and right steering controls (Wu, et al., 2023).

The introduction of the 'my-short-keypress' function represents a more human-like driving behavior that makes it easier for the model to differentiate yellow from white road lines. Instead of relying solely on continuous steering inputs, the model now makes brief, corrective adjustments when steering left.

Testing the model revealed that the bus could now be driven for the entire 360-mile distance. So far, the model has been able to achieve one point after driving for 18 hours and 30 minutes at an average speed of 20 miles per. While this is lower than the busses top speed of 45 miles per hour, future adjustments and enhancements to the model's visual cue recognition will naturally make the bus drive faster, since the model always move forward after successfully steering left.

Discussion

The aim of this study was to employ ACT-R 7 and its architecture of interaction to successfully complete a demanding cognitive modeling task. The model runs for an average of 18 and a half hours, with slower versions running more than 24 hours.

Instead of changing the model's procedural framework or adding unnecessary functions to VisiTor, addressing the problem through a human-centered lens provides valuable insights for autonomous vehicle design from computational cognitive perspectives.

The implementation of dual-reference visual tracking and adaptive error handling mechanisms offers a blueprint for autonomous vehicle perception systems. Current autonomous vehicle algorithms suffer from poor visibility conditions and low light environments. Our model shows one way in which this may happen and additionally demonstrates how cognitive architectures can maintain reliable performance while adapting to environmental changes.

Conclusions

Contribution

CogDriver makes a leap forward in developing an autonomous driving cognitive model. Adding human behaviors to the model through cognitive architecture is achieved by adding behavioral error making and learning improvements to the ACT-R model. This is demonstrated by the model's maintained 18 hour driving record and was achieved by adding human-like driving knowledge representations, error handling mechanisms, and new visual cues. This improvement showcases the capability to establish human behavioral models by examining human perception, action loops, and fallback procedures.

These contributions advance both autonomous driving research and cognitive modeling, showing how incorporating human factors and psychological insights can improve the performance and reliability of autonomous systems, particularly in challenging conditions that have traditionally been difficult for cognitive models to handle.

Limitations and future work

However, the model's performance during the transition to nighttime conditions reveals limitations. While successfully navigating most nighttime conditions, the model encounters difficulties when yellow and white road lines are indistinguishable around 8 hours into the

day-night cycle. While circumventing this problem by driving the bus in the correct lane solves this, further development in the spatial memory system of the model would aid edge cases in which the bus is too far to the left side of the road

There are still limitations with the central modules in this environment. These limitations include the lack of consideration for physiological factors such as fatigue or decreasing correction rates over time. In a study by Schwartz et al. (2020), it was suggested that incorporating physiology with ACT-R could make the model more realistic. We agree with this point and add that future work should test the compound effects of fatigue and learning rate on the model (Wu, et al., 2023).

This platform, ACT-R + VisiTor playing *Drive the Bus* provides an excellent platform for studying the interaction of vision, attention, errors, and fatigue. It is a more naturalistic task than the PsychoMotor Vigilance task (PVT, Dinges & Powell, 1985). We can now go and insert an existing fatigue model (Gunzelmann, Gross, Gluck, & Dinges, 2009), and examine fatigued driving (e.g., Gunzelmann, Moore, Salvucci, & Gluck, 2011), visual attention, the need for micuration, and modeling the details of interaction.

We could gain understanding about how long-term and repetitive physical activities, like driving a bus for an extended period, affect human performance. It remains to be seen if this task is more like the PVT or like motor control (Bolkhovsky, Ritter, Chon, Qin, 2018). This task would also allow us to determine whether psychological factors could potentially harm or the increasing of learning rate due to the practice would enhance driving skills. We could also introduce additional variables, such as caffeine consumption, to examine their combined impact.

References (christian maintains)

Anderson, J. R. (1996). ACT: A simple theory of complex cognition. *American Psychologist*, *51*(4), 355.

Anderson, J. R. (2007). *How can the human mind occur in the physical universe?* New York, NY: Oxford University Press.

Anderson, J. R., & Douglass, S. (2001). Tower of Hanoi: Evidence for the cost of goal retrieval. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *27*, 1331–1346. https://doi.org/10.1037/0278-7393.27.6.1331

Bagherzadeh, A., & Tehranchi, F. (2022). Comparing cognitive, cognitive instance-based, and reinforcement learning models in an interactive task. *Proceedings of ICCM, The 20th International Conference on Cognitive Modeling.* 1-7.

Bolkhovsky, J. B., Ritter, F. E., Chon, K. H., & Qin, M. (2018). Performance trends during sleep deprivation. *Aerospace Medicine and Human Performance, 89*(7), 626-633(8).

Bothell, D. (2017). ACT-R 7 reference manual. Available at actr.psy.cmu edu/wordpress/wpcontent/themes/ACT-R/actr7/reference-manual pdf, Accessed February 2023.

Byrne, M. D., O'Malley, M. K., Gallagher, M. A., Purkayastha, S. N., Howie, N., & Huegel, J. C. (2010). A preliminary ACT-R model of a continuous motor task. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, *54*(13), 1037–1041. https://doi.org/10.1177/154193121005401308

C. Breazeal, A. Edsinger, P. Fitzpatrick and B. Scassellati, "Active vision for sociable robots," in *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, vol. 31, no. 5, pp. 443-453, Sept. 2001, doi: 10.1109/3468.952718. keywords: {Robot vision systems;Robot kinematics;Humanoid robots;Robot sensing systems;Human robot interaction;Machine vision;Animation;Visual system;Cameras;Artificial intelligence},

Dinges, D. I., & Powell, J. W. (1985). Microcomputer analysis of performance on a portable, simple visual RT task sustained operations. *Behavior Research Methods, Instrumentation, and Computers, 17*, 652–655.

Fleetwood, M. D., & Byrne, M. D. (2002). Modeling icon search in ACT-R/PM. *Cognitive Systems Research, 3*(1), 25–33. https://doi.org/10.1016/S1389-0417(01)00041-9

Gunzelmann, G., Gross, J. B., Gluck, K. A., & Dinges, D. F. (2009). Sleep deprivation and sustained attention performance: Integrating mathematical and cognitive modeling. *Cognitive Science, 33*(5), 880-910.

Gunzelmann, G., Moore, L. R., Salvucci, D. D., & Gluck, K. A. (2011). Sleep loss and driver performance: Quantitative predictions with zero free parameters. *Cognitive Systems Research, 12*, 154-163.

Laird, J. E. (2019). *The Soar cognitive architecture*. Cambridge, MA, MIT Press.

Morrison, G. R., Ross, S. M., Kahlman, H. K., & Kemp, J. E. (2010). *Designing effective instruction* (6th Edition ed.). Hoboken, NJ: John Wiley & Sons.

Pew, R. W., & Mavor, A. S. (Eds.). (2007). *Human-system integration in the system development process: A new look.* Washington, DC: National Academy Press. books.nap.edu/catalog/11893, checked May 2019.

Newell, A. (1990). *Unified Theories of Cognition.* Harvard University Press.

Ritter, F. E., Baxter, G. D., Jones, G., & Young, R. M. (2000). Supporting cognitive models as users. *ACM Transactions on Computer-Human Interaction, 7*(2), 141–173. https://doi.org/10.1145/353485.353486

Ritter, F. E., Tehranchi, F., & Oury, J. D. (2019). ACT-R: A cognitive architecture for modeling cognition. *WIREs Cognitive Science, 10*(3), e1488. https://doi.org/10.1002/wcs.1488

Ritter, F. E., Van Rooy, D., Amant, R. St., & Simpson, K. (2006). Providing user models direct access to interfaces: An exploratory study of a simple interface with implications for HRI and HCI. *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans, 36*(3), 592–601. https://doi.org/10.1109/TSMCA.2005.853482

Schwartz, D. M., Tehranchi, F., & Ritter, F. E. (2020). Drive the bus: Extending JSegMan to drive a virtual long-range bus. In *Proceedings of the 18th International Conference on Cognitive Modeling (ICCM 2020)*. 241-246.

Tehranchi, F., & Ritter, F. E. (2017). An eyes and hands model for cognitive architectures to interact with user interfaces. In *MAICS* (pp. 15-20).

Tehranchi, F., & Ritter, F. E. (2020). Extending JSegMan to interact with a biased coin task and a spreadsheet task. In *17th International Conference on Cognitive Modeling, ICCM 2019* (pp. 259-260). Applied Cognitive Science Lab, Penn State.

Tehranchi, F., & Ritter, F. E. (2018). Modeling visual search in interactive graphic interfaces: Adding visual pattern matching algorithms to ACT-R. In *Proceedings of ICCM-2018-16th International Conference on Cognitive Modeling* (pp. 162-167)

Trapp S, Schroll H, Hamker FH. Open and closed loops: A computational approach to attention and consciousness. Adv Cogn Psychol. 2012 Feb 3;8(1):1-8. doi: 10.2478/v10053-008-0096-y. PMID: 23853675; PMCID: PMC3709102.

Wilson, D., & Sicart, M. (2010). Now it's personal: On abusive game design. *Proceedings of the International Academic Conference on the Future of Game Design and Technology*, 40–47. https://doi.org/10.1145/1920778.1920785

Wu, Siyu & Bagherzadeh, Amirreza & Ritter, Frank & Tehranchi, Farnaz. (2023). Long Road Ahead: Lessons Learned from the (soon to be) Longest Running Cognitive Model.