http://dx.doi.org/10.14236/ewic/BRITCHI26.13

RPD-Grounded Simulated Users for Pre-screening AR Training Interface Designs: A Unity ML-Agents Approach

Thusithanjana Thilakarathna
Auckland University of Technology
Auckland, New Zealand
thusithanjana.thilakarathna@autuni.ac.nz

B. L. William Wong
Auckland University of Technology
Auckland, New Zealand
william.wong@aut.ac.nz

Minh Nguyen
Auckland University of Technology
Auckland, New Zealand
minh.nguyen@aut.ac.nz

Evaluating augmented reality (AR) training interfaces typically requires recruiting domain practitioners, coordinating controlled studies, and incurring the logistical and ethical costs of human-subject testing. This paper presents a *pre-screening* framework that uses simulated users to surface interface design trade-offs before human trials begin. Agents are built with Unity ML-Agents and grounded in the Recognition-Primed Decision (RPD) model, with distinct reward structures derived from Cognitive Task Analysis of paramedic diagnostic reasoning encoding novice and expert decision strategies. We evaluate three interface configurations varying in button density, spatial layout, and *labelling* across both agent types, and characterize behaviour using behavioural interaction indicators computed from interaction logs and conceptually organized using a structure inspired by the NASA Task Load Index (NASA-TLX). Results show that dense dispersed layouts impose the highest temporal and search cost on novice agents, while deceptive clustering imposes the highest temporal and effort cost on expert agents. Search Complexity Indicator diverged between cohorts in a pattern consistent with RPD-predicted selective attention: expert agents narrowed search as complexity increased while novice agents maintained broad attention throughout. These patterns were consistent across behavioural indicator scores and raw interaction metrics including path entropy, inter-click intervals, and idle ratios. The framework requires no human participants, produces reproducible *behavioural* traces across conditions, and is intended as a complement to human-subject evaluation rather than a replacement. Validation against human self-reported ratings and objective performance data is identified as the essential next step.

Simulated users, Recognition-Primed Decision, augmented reality training, Unity ML-Agents, AR UI evaluation

1. INTRODUCTION

Evaluating augmented reality (AR) training interfaces typically requires recruiting domain experts, coordinating access to high-fidelity environments, and running controlled user studies under realistic conditions. In high-stakes training domains such as emergency medicine, these requirements are not merely logistical inconveniences: paramedic students and practitioners are scarce participants, training scenarios carry ethical constraints, and iterating on interface designs between studies is slow and costly. As a result, many AR training interfaces reach user evaluation only once, at a late stage of development, when the cost of discovering fundamental design problems is highest.

Simulated users offer a way to move some of this evaluation earlier. The idea of replacing or supplementing human participants with computational models has a substantial history in HCI. Analytical models such as GOMS and KLM have long enabled designers to predict expert task times from interface specifications alone (Card, 1983). More recently, the computational rationality framework has produced agent-based models capable of simulating cognitively and motorically plausible interaction behavior across a range of interface types, including pointing tasks, menu selection, and visual search (OulasvirtaJokinen and Howes, 2022). Reinforcement learning agents have also been applied to VR interfaces to probe usability properties without human involvement (Fischer et al., 2024). These contributions establish that principled, participant-free evaluation is both theoretically defensible and practically useful.

However, existing approaches share a limitation that matters specifically for AR training evaluation.

© Thilakarathna et al. Published by BCS Learning and Development Ltd.

They model interaction as a general cognitive-motor process, optimizing task completion under physical and temporal constraints. In high-stakes training contexts, the more important question is not whether a user can complete a task efficiently, but whether the interface supports the kind of decision-making the training domain requires. A paramedic trainee interacting with a diagnostic AR interface must not only locate and select elements; they must recognize clinically relevant cues, prioritize competing information, and act under time pressure with incomplete data. These are the behavioral characteristics described by the Recognition-Primed Decision (RPD) model (Klein, 1993b), which accounts for how experienced practitioners make rapid, pattern-based decisions in naturalistic settings. No prior simulation work has grounded agent behavior in RPD for AR training interface evaluation.

This paper addresses that gap. We present a prescreening framework in which simulated users, built with Unity ML-Agents and grounded in the RPD model, interact with AR training interface configurations to surface design trade-offs before human trials begin. Agents are trained with distinct reward structures reflecting novice and expert decision strategies derived from Cognitive Task Analysis (CTA) of paramedic diagnostic reasoning(Smith et al., 2013). Novice agents attend to all available interface elements without prioritization; expert agents selectively target clinically relevant cues first. We evaluate three interface configurations varying in button density, layout, and labeling, and characterize agent behavior using NASA TLX-inspired behavioral interaction indicators computed from interaction logs. The paramedic scenario, adapted from Smith et al.'s CTA study of experienced and novice paramedics, provides the empirical basis for the cognitive profiles encoded in each agent type.

Our results show that the two agent types produce meaningfully different behavioral profiles across interface conditions, and that these differences shift systematically as interface complexity increases. Dense layouts raise novice temporal load while expert agents maintain efficiency; deceptive clustering raises effort and temporal demand for expert agents while novices distribute attention broadly. These patterns are consistent with the recognition-driven strategies RPD predicts and suggest that the framework can surface interface trade-offs that matter for training effectiveness.

We make the following contributions. First, a prescreening framework for AR training interface evaluation using RPD-grounded simulated users implemented in Unity ML-Agents, requiring no human participants. Second, an operational mapping from CTA-derived cognitive profiles to reinforcement learning reward structures, enabling reproducible novice and expert agent behavior. Third, a demonstration on a paramedic diagnostic reasoning scenario showing simulated agents surface expertise-dependent workload differences across interface configurations.

We position this work as a complement to, not a replacement for human-subject evaluation. The framework is intended to identify promising interface designs and eliminate poor ones before the logistical and ethical costs of human trials are incurred. Limitations and the path toward validation against human data are discussed in the final section.

2. RELATED WORK

The following review covers five areas directly relevant to the proposed framework: automated and model-based UI evaluation, agent-based and simulated user approaches in HCI, Recognition-Primed Decision making and cognitive modelling, AR interface evaluation methods, and statistical approaches to interface performance measurement. Foundational works predating the primary search window are retained where their theoretical centrality justifies inclusion.

2.1. Automated and Model-Based UI Evaluation

The ambition to evaluate user interfaces without recruiting live participants has a deep and well-documented lineage in HCI research. The foundational contribution is the GOMS family of analytical models introduced by Card (1983). GOMS decomposes user activity into Goals, Operators, Methods, and Selection rules, enabling analysts to derive quantitative predictions of expert task-execution time from interface specifications alone, without any empirical data collection. The Keystroke-Level Model (KLM), introduced by CardMoran and Newell (1980), represents the most practically accessible variant, predicting task time as the sum of operator times across a fixed taxonomy of physical and cognitive actions. KLM's predictive accuracy for expert, error-free performance on desktop systems is well established, and the model has been extended to touchscreen interactions via the Touch Level Model (Rice & Lartigue, as discussed in John and Kieras (1996)).

The CogTool system (John et al., as reviewed in John (2011)) employed KLM and CPM-GOMS within an automated prototyping and prediction environment, using ACT-R as its underlying computational engine. CogTool allows designers to construct storyboarded interface prototypes and receive quantitative performance predictions without user involvement, demonstrating that model-based evaluation can yield actionable

design guidance within iterative development cycles. However, a persistent limitation of GOMS-derived approaches is that they model only expert, error-free performance (Card, 1983); they cannot represent exploratory behaviour, affective dimensions, learning trajectories, or the situated adaptation that characterizes expert use in genuine operational contexts.

More recent automated evaluation work has explored machine-learning techniques for quality prediction and reinforcement-learning agents for interface crawling, but these approaches are predominantly oriented towards functional fault detection rather than usability quality per se, and they rarely invoke explicit cognitive theory. Ritter and Young (2001) articulated the conceptual case for *embodied cognitive models as simulated users*. They argued that models equipped with simulated perception and action capable of interacting directly with interface software represent a fundamentally more powerful and ecologically grounded approach to automated evaluation than scripted or symbolic models operating on abstracted interface specifications. This position underlies the use of Unity ML-Agents as a simulation substrate, extending the embodied-model tradition into a three-dimensional spatial interface context.

2.2. Agent-Based and Simulated User Approaches in HCI

The use of autonomous agents as simulated human users in interface evaluation has been pursued through two largely parallel traditions: cognitive architecture-based modelling and agent-oriented software engineering. In the cognitive architecture tradition, the ACT-R architecture developed by Anderson et al. (1998) has served as the primary platform. Byrne (2001) demonstrated the application of ACT-R/PM: an extension of ACT-R with perceptual and motor modules to menu selection tasks, generating a priori predictions of interaction times that corresponded well with empirical user data. Salvucci (2001), in the same special issue, showed that integrated ACT-R models could compare car phone interfaces in terms of their effects on driving performance, producing behavioural data without human-participant testing. Ritter et al. (2000), presented the Cognitive Model Interface Management System (CMIMS) concept, demonstrating that cognitive models could be connected to real interface software via standardized interaction mechanisms, enabling broad applicability as automated interface testers.

A key insight from this body of work, synthesized by Ritter and Young (2001), is that cognitive models used as simulated users must be *embodied*, equipped with simulated perceptual and motor capabilities in order to interact with interfaces as users do, rather than as scripted access mechanisms operating on internal interface representations. This requirement has important implications for the design of the simulated agents in this study, which must perceive and act within a three-dimensional AR environment rather than a planar GUI. In the Soar and EPIC tradition, parallel work has demonstrated simulated agents capable of multitasking and parallel cognitive processing (Kieras and Meyer, 1997), with applications to air traffic control and command-and-control interfaces. However, these cognitive-architecture agents require substantial parameterization and are difficult to extend to novel interface modalities such as spatial AR without significant architectural modification.

Unity ML-Agents, as a reinforcement-learning platform, offers a pragmatic alternative infrastructure for agent-based simulation in three-dimensional interactive environments. While its application to HCI evaluation specifically remains rare in the peer-reviewed literature, it has been used extensively in game AI research and is increasingly employed in robot training and simulation (Juliani et al., 2018). The key methodological challenge for agent-based HCI simulation has identified across both the cognitive architecture and the RL-agent traditions, is the specification of the agent's decision process: purely behavioural agents that reproduce performance distributions without modelling the underlying cognitive mechanisms may generate evaluatively misleading data in complex, expert-user contexts. This study addresses this challenge by grounding its agent's decision process in Recognition-Primed Decision theory.

2.3. Recognition-Primed Decision Making and Cognitive Modelling in UI Contexts

The Recognition-Primed Decision (RPD) model was first articulated by KleinCalderwood and Clinton-Cirocco (1986) in studies of fireground commanders and was subsequently formalized by Klein (1993a) in *Decision Making in Action: Models and Methods* (Klein, Orasanu, Calderwood, & Zsambok, Eds.). The model describes three levels of decision-making: a simple match in which the situation is immediately recognized and a standard response implemented; a diagnosed situation in which the decision-maker must reconcile competing interpretations before selecting an action; and a mental simulation level in which the selected action is mentally simulated to evaluate its feasibility before commitment(Klein, 1993b). All three levels involve pattern-matching against experiential repertoires rather than exhaustive comparison of alternatives: a characterization empirically validated across firefighting, military command, neonatal nursing, chess, and aviation (EndsleyZsambok and Klein, 1997).

The theoretical importance of RPD for HCI lies in its descriptive adequacy for expert users operating in time-pressured, high-stakes interface contexts, precisely the contexts in which classical cognitive models such as GOMS and KLM are least applicable. As Lipshitz et al. (2001) noted in their decade-long review of NDM research, the NDM paradigm has generated substantial applied insight into interface and system design for expert decision-making, but has struggled to translate this insight into prescriptive computational models that can be used in evaluation tools. Applications of NDM to interface design have tended to produce guidelines for instance, supporting situation assessment rather than option enumeration in command-and-control interfaces rather than formal computational specifications of agent behaviour (Klein, 2008).

Computational implementations of RPD are rare. Work by Zacharias and colleagues in the early 2000s produced RPD-consistent agent architectures for military simulation contexts, and intelligent tutoring systems research has incorporated recognition-based decision mechanisms in student and expert modelling, though rarely with explicit RPD formalization(Ritter et al., 2000). Byrne (2001) and St. AmantHorton and Ritter (2004) demonstrated cognitive-model evaluation of sequential menu and cell phone interactions, but these involve analytical models of novice-to-expert sequential task execution rather than recognition-based pattern matching. This study's use of RPD as the decision-process substrate for a Unity ML-Agent represents a novel contribution to the computational modelling literature, extending recognition-based decision simulation from military and tutoring contexts into AR interface evaluation.

2.4. AR/XR Interface Evaluation: Methods and Challenges

Augmented reality interfaces present evaluators with methodological challenges that are qualitatively distinct from those of desktop or mobile HCI. The co-presence of digital overlays and physical environments introduces perceptual complexity, spatial registration demands, depth perception ambiguity, and attentional competition that standard usability instruments including the System Usability Scale (SUS), NASA Task Load Index (NASA-TLX), and think-aloud protocols were not designed to address. Gabbard and Swan II (2008) conducted the first comprehensive survey of AR user studies, reviewing over 1,100 AR papers published between 1992 and 2004 and finding only 21 that reported formal user studies, classified into perceptual, interaction technique, and collaborative categories. Dey et al. (2018) in a systematic review of ten years of AR usability studies (2005–2014), found that objective task performance measures dominated AR evaluation practice, with subjective instruments employed inconsistently and rarely validated for spatial interface contexts.

Training-focused AR systems have been evaluated across several application domains. Gavish et al. (2015), in a study, evaluated VR and AR platforms for industrial maintenance and assembly training with forty expert technicians, finding that AR training reduced unsolved errors relative to control conditions. This study is exemplary in its use of objective performance metrics and experimental design, but it also illustrates the logistical demands of AR training evaluation: specialized participant populations, controlled industrial environments, and expert-user recruitment. These demands are precisely what the present paper's participant-free methodology seeks to mitigate.

A recurring observation across the AR evaluation literature is that the standard instruments of usability research exhibit reduced validity in immersive and spatial contexts. Gabbard and Swan II (2008), argued that AR-specific evaluation frameworks grounded in empirical user studies were necessary to inform design, but acknowledged the logistical barriers to conducting such studies at scale. The present paper does not propose to replace this empirical tradition but to complement it with a participant-free, agent-based alternative suited to iterative interface comparison in the AR training domain.

2.5. Statistical Approaches to Interface Performance Measurement

Quantitative evaluation of interface performance data has seen significant methodological development in HCI. The ISO 9241-11 framework defines usability in terms of effectiveness, efficiency, and satisfaction, providing a widely adopted structure for measurement. Task completion time and error rate, the most reported metrics in AR evaluation studies, map onto the efficiency and effectiveness components of this taxonomy and motivate the use of behavioral measures as evaluation currency.

Robertson and Kaptein (2016) argue that HCI's historical reliance on null hypothesis significance testing has produced an inflated rate of false positives and an underpowered body of evidence. This concern is directly relevant to participant-free evaluation, where sample sizes and variance differ fundamentally from human-participant designs. The present paper responds to this concern by reporting behavioral indicators descriptively rather than inferentially, making no statistical claims beyond what the data can support, and identifying inferential analysis as part of the planned human-participant follow-up.

2.6. Synthesis: Gaps That Justify the Current Study

The five thematic areas reviewed above individually represent active and productive research traditions. Automated and model-based evaluation, from GOMS (Card, 1983) through CogTool to embodied cognitive models(Ritter and Young, 2001), has established that principled, participant-free interface evaluation is both theoretically defensible and practically feasible for a range of interface types and task contexts. Agent-based HCI simulation, exemplified by Byrne (2001)'s ACT-R/PM work and the CMIMS framework of Ritter et al. (2000), has demonstrated that cognitively grounded simulated users can generate evaluatively meaningful behavioural data without live participants. NDM theory, anchored in Klein's (Klein, 1997, Klein, 1993b) RPD model and reviewed by Lipshitz et al. (2001), provides a theoretically validated account of expert decision-making under uncertainty that is directly relevant to the professional, time-pressured contexts in which AR training interfaces are deployed. AR interface evaluation (Dey et al., 2018, Gavish et al., 2015) has established that meaningful performance differences between interface variants can be detected using objective metrics, but that human-participant evaluation in this domain carries substantial logistical costs. And modern statistical methods for HCI (Kaptein, 2016, Wobbrock and Kay, 2016) provide the inferential infrastructure necessary to draw rigorous conclusions from repeated-measures behavioural data, regardless of whether that data was generated by human participants or simulated agents. No prior work in any of these traditions has combined them: RPD-based cognitive modelling of simulated agent behaviour, implemented in a three-dimensional agent platform such as Unity ML-Agents, applied to the comparative evaluation of AR training interfaces, and analysed using mixed-effects statistical models. This convergence of existing methodological strands into a unified, participant-free evaluation pipeline constitutes the primary contribution of the present paper.

3. METHODOLOGY

3.1 Overview

The methodology follows four steps. First, cognitive task analysis of paramedic diagnostic reasoning informs the construction of novice and expert agent profiles. Second, these profiles are mapped to reward structures within the PERE framework (Policy, Environment, Reward signal, Episode) and implemented in Unity ML-Agents. Third, agents are trained using Proximal Policy Optimization and evaluated across three interface configurations. Fourth, interaction logs are used to compute behavioral interaction indicators.

3.2 From RPD to Reward Structure

The Recognition-Primed Decision model distinguishes expert from novice behavior primarily through selective attention to relevant cues. Experts rapidly recognize patterns and act on the most diagnostically relevant information first; novices attend broadly without prioritization. This distinction was employed directly in the reward function.

Agent profiles were derived from Smith et al.'s CTA study of paramedic diagnostic reasoning (Smith et al., 2013), which identified systematic differences between experienced and novice paramedics in cue detection, hypothesis generation, and adaptability. From this study, three novice and three expert fictional personas were constructed, capturing behavioral patterns such as anchoring to initial impressions, selective cue prioritization, and dual-pathway reasoning. These profiles informed two distinct reward structures rather than a single unified agent.

In novice mode, the agent receives a reward for clicking any available button, modeling broad, undiscriminating attention. In expert mode, the agent receives a bonus for clicking expert-labeled buttons before novice-labeled ones, and a penalty for clicking novice buttons prematurely, modeling selective prioritization of clinically relevant cues. Both modes share additional shaping terms: a small penalty for unnecessary pointer movement, a progress reward for approaching relevant targets, and a completion bonus for finishing all required targets without errors. This design means the two agents learn genuinely different interaction strategies from the same interface, rather than simply performing the same task at different speeds.

The mapping from RPD outputs to reward components is summarized in Table 1.

Table 1: *The mapping from RPD outputs to reward components*

RPD Component	Novice Reward Term	Expert Reward Term
Cue recognition	*Reward any click*	*Bonus for expert-labelled clicks*
Goal prioritization	*None*	*Penalty for premature novice clicks*
Action efficiency	*Movement penalty*	*Movement penalty*
Task completion	*Completion bonus*	*Completion bonus*

3.3 Agent Implementation

Agents were implemented using Unity ML-Agents with a Proximal Policy Optimization (PPO) policy. The MDP was formalized as follows.

The observation space at each step consisted of the agent's normalized pointer position on the canvas and structured information about all visible interface elements, including their normalized positions, sizes, and labels (expert or novice). This gave the agent full knowledge of the interface layout at each step, ensuring that behavioral differences between conditions arose from the layout itself rather than perceptual limitations.

The action space had three discrete branches: move pointer left or right (horizontal), move pointer up or down (vertical), and click. This kept the action space tractable while allowing the agent to learn sequential navigation and selection strategies rather than treating each click as an isolated decision.

Training began with a simplified environment containing a single target button and was progressively extended to multiple buttons in varied positions. Button positions and sizes were randomized at the start of each episode using a normalized distribution, so that extreme sizes remained rare while mid-range values were more frequent. This prevented the agent from overfitting to fixed layouts and encouraged generalization across varied interface configurations. The pointer was initialized to a random position at the start of each episode. An episode ended when all targets were clicked or the maximum step limit was reached.

Ten novice agents and three expert agents were trained by independent random seeds. Training was dominated by policy updates, which accounted for approximately 70.9% of the time for novice agents and 88.3% for expert agents.

3.4 Interface Configurations

Three interface configurations were designed for the paramedic training scenario, each varying button size, number, placement, and labeling. In all configurations, 30% of buttons were labeled expert and the remainder novice.

UI-1 (Baseline) used four buttons arranged in a grid with a large button size, providing a low-demand reference condition. UI-2 (Dense Search) used ten smaller buttons in a dispersed layout, targeting mental, temporal, and physical demand. UI-3 (Deceptive Cluster) used ten medium buttons in a clustered layout with a mix of expert, novice, and untagged buttons, targeting frustration and effort through spatial ambiguity. Both novice and expert agents were evaluated across all three interface configurations, and their behavioral interaction indicators were analyzed across six experimental conditions.

3.5 Behavioural Interaction Indicators

Because the agents in this study are computational entities rather than human participants, self-report instruments such as the NASA Task Load Index (NASA-TLX) cannot be directly applied. We therefore derive six behavioral interaction indicators from interaction logs. These indicators are based on observable interaction behavior and are not NASA-TLX scores, do not estimate subjective workload, and should not be interpreted as substitutes for human self-report measures. NASA-TLX is used only as a conceptual framework for organizing the indicators and supporting future comparisons with human-participant studies.

The indicators were organized using a NASA-TLX-inspired structure consisting of Search Complexity Indicator (SCI), Movement Cost Indicator (MCI), Temporal Interaction Cost (TIC), Task Success Indicator (TSI), Composite Interaction Effort (CIE), and Error Recovery Indicator (ERI). SCI was computed from path entropy, click hesitation time, and direction changes; MCI from path length and pointer speed; TIC from inter-click interval, completion time, and idle ratio; TSI from completion ratio, order adherence, and error count; CIE as a composite z-score of movement distance, angle change, dwell time, and inter-click variability; and ERI from mis-click rate, repeated clicks, and pointer jitter.

All indicators were linearly rescaled to a 0–10 range across agents and interface conditions. Scores are intended only for relative comparison between conditions within this study. TSI and ERI exhibited ceiling effects and are therefore interpreted with caution. TIC and CIE provided the strongest differentiation between interface configurations.

4. RESULTS

4.1 Training Cost

Ten novice agents and three expert agents were trained by independent random seeds. Median wall-clock training time was 0.135 hours for novice agents (mean 0.401 h, range 0.017 to 0.900 h) and 0.046 hours for expert agents (mean 0.298 h, range 0.005 to 0.844 h). Training was dominated by policy updates in both cohorts, accounting for approximately 70.9% of wall-clock time for novice agents and 88.3% for expert agents, with experience collection accounting for the remainder. Inference time was negligible. These figures suggest that training cost is modest and that optimizer and batch settings are the primary levers for reducing it further. The small expert cohort

(n=3) means the training time comparison should be read as directional rather than conclusive.

4.2 Agent Trajectories

Representative agent trajectories across the three interface configurations are shown in Figure 1. In UI-1, both agent types produced relatively compact paths given the simple grid layout. In UI-2, expert paths remained directed while novice paths showed broader spatial coverage, consistent with the undiscriminating attention encoded in the novice reward structure. In UI-3, novice agents showed visible cluster tunnelling repeatedly visiting the clustered button region, while expert agents produced longer paths as they worked through the deceptive layout to locate expert-labelled targets first. These qualitative patterns are consistent with the workload proxy results described below.

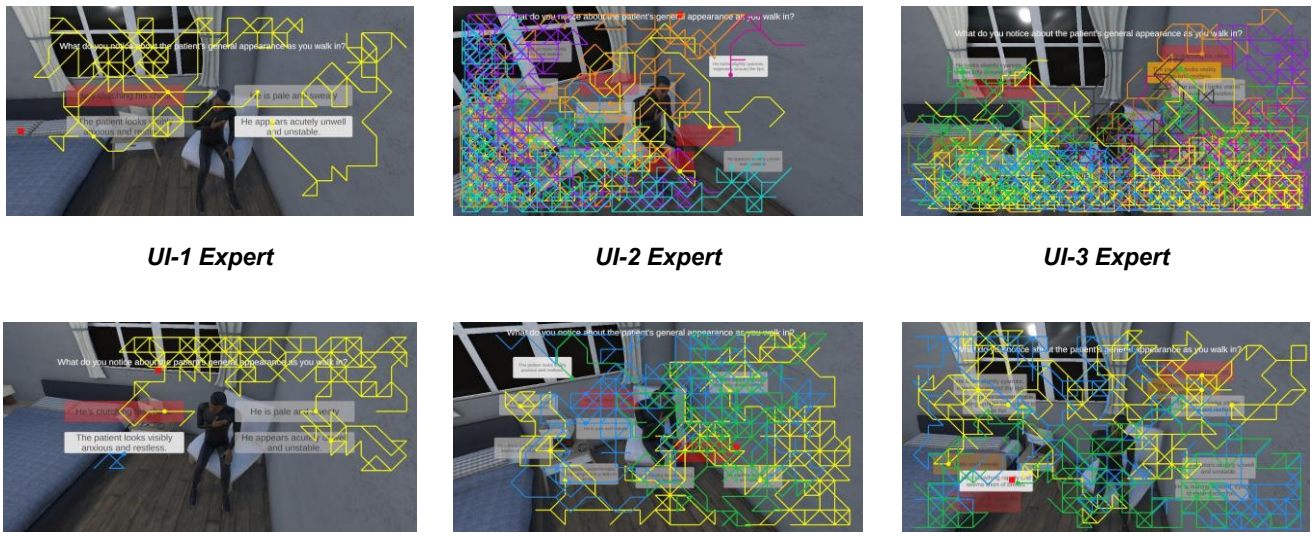
4.3 Behavioural Interaction Indicators

Table 2 reports mean behavioural indicator scores for each dimension across the six conditions, organized using a NASA-TLX-inspired structure. Two dimensions, Task Success Indicator and Error Recovery Indicator, showed ceiling effects across all conditions (scores at or near 5.0) and could not differentiate interface configurations. The remaining four dimensions produced interpretable patterns, described below. Raw behavioural metrics from the episode logs corroborate the proxy scores and are reported alongside them.

TIC showed the strongest differentiation. For expert agents, TIC was stable across UI-1 (5.00) and UI-2 (4.47), then rose sharply in UI-3 to 6.85, the highest value observed across any condition. This reflects the additional time expert agents required to work through the deceptive clustering while maintaining expert-first prioritization. For novice agents, Temporal Interaction Cost peaked in UI-2 at 5.52, reflecting the search cost imposed by the dispersed dense layout, before dropping slightly in UI-3 to 5.33. These patterns are corroborated by episode counts: the expert agent required 15 episodes in UI-3 compared to 2 in UI-1, indicating the deceptive layout substantially increased task difficulty for expert-strategy agents.

SCI revealed a clear expertise divergence across conditions. Expert SCI declined progressively from UI-1 (5.00) to UI-2 (4.25) to UI-3 (3.50), consistent with selective attention: as the expert agent focused on a narrowing set of relevant cues, the overall search effort decreased. Novice SCI remained stable across all three conditions (4.83, 4.85, 4.91), consistent with broad, undiscriminating attention that does not adapt to layout changes. This pattern is corroborated by path entropy in the raw logs: novice entropy rose from 2.91 in UI-1 to 3.97 in UI-3, indicating increasingly dispersed spatial coverage, while Expert entropy remained stable at approximately 3.5 across all conditions

CIE followed a pattern that mirrors TIC for experts. Expert CIE was stable in UI-1 (5.00) and UI-2 (4.10), then rose in UI-3 (5.34), reflecting the cost of resisting deceptive clustering while maintaining prioritization.



UI-1 Expert ***UI-2 Expert*** ***UI-3 Expert***

UI-1 Novice ***UI-2 Novice*** ***UI-3 Novice***

Figure 1: *Agent trajectories across interface conditions and cohorts*

Table 2: *Behavioural interaction indicator scores across conditions and cohorts (0–10 scale; higher values indicate greater interaction cost).*

Condition	Agent	SCI	MCI	TIS	TSI	CIE	ERI
UI-1 Baseline	*Expert*	*5.00*	*5.00*	*5.00*	*5.00*	*5.00*	*5.00*
UI-1 Baseline	*Novice*	*4.83*	*4.47*	*4.32*	*5.00*	*4.19*	*5.00*
UI-2 Dense	*Expert*	*4.25*	*5.40*	*4.47*	*5.00*	*4.10*	*5.00*
UI-2 Dense	*Novice*	*4.85*	*4.76*	*5.52*	*5.00*	*4.58*	*5.00*
UI-3 Deceptive	*Expert*	*3.50*	*4.51*	*6.85*	*5.00*	*5.34*	*5.00*
UI-3 Deceptive	*Novice*	*4.91*	*3.90*	*5.33*	*5.00*	*4.76*	*5.00*

Novice CIE increased gradually across all three conditions (4.19, 4.58, 4.76), suggesting a steady rise in interaction cost as layout complexity increased. Mean inter-click intervals from the raw logs reinforce this: novice agents waited between 7 and 9 seconds between clicks on average across conditions, compared to 1.5 seconds for expert agents, indicating that novices expended considerably more time locating each successive target.

MCI showed less consistent patterns. Expert MCI rose in UI-2 (5.40) compared to UI-1 (5.00) and dropped in UI-3 (4.51), while novice MCI slightly across conditions. Path length data from the raw logs shows both cohorts increasing from UI-1 (Expert 12.6, Novice 10.4) to UI-2 and UI-3 (Expert 22-23, Novice 20-22), which is broadly consistent with the denser interfaces requiring more movement. The MCI proxy was less sensitive to the novice-expert distinction than temporal or Search Complexity Indicator.

5. DISCUSSION

5.1 What the Results Support

The central claim of this paper is that RPD-grounded simulated users can surface interface design trade-offs in AR training environments before human trials begin. The results support this claim in a specific and bounded way. The two agent types, novice and expert, produced systematically different behavioral profiles across the three interface configurations, and those differences shifted in ways that are consistent with the decision-making strategies each agent was designed to model.

The most interpretable finding is the divergence in temporal interaction cost between UI-2 and UI-3. Novice agents were most affected by the dense dispersed layout, where undiscriminating attention across ten scattered buttons produced the highest temporal cost. Expert agents handled density relatively well but were most affected by deceptive clustering, where maintaining expert-first prioritization against spatially misleading button groups extended task time substantially. This distinction matters for interface design: density and deception impose costs on different user types, and a design that reduces one may not reduce the other. A simulation framework that cannot model this distinction cannot surface it. This one does.

The Search Complexity Indicator (SCI) reinforces this interpretation. Expert SCI declined across conditions as the agent narrowed its attention to relevant cues, while novice SCI remained stable. This mirrors the selective attention characteristic of RPD-based decision-making: experienced practitioners reduce cognitive load by filtering irrelevant information, while novices maintain broad attention regardless of layout. The path entropy data from the raw logs corroborates this directly, with novice entropy rising steadily from UI-1 to UI-3 while expert entropy remained stable.

The inter-click interval difference is also worth noting. Novice agents waited between seven and nine seconds between clicks on average, compared to approximately 1.5 seconds for expert agents. This gap reflects not just speed but strategy: expert agents moved directly between targets while novice agents searched broadly between selections. This kind of behavioral differentiation is exactly what a prescreening framework needs to produce to be useful.

One concern with reward-shaped agents is circularity: if behaviour is encoded in the reward function, observing that behaviour in the results proves nothing beyond the reward design working as intended. This concern deserves a direct response. The reward function encoded one thing: prioritization order. Expert agents were penalized for clicking novice buttons before expert ones. What the reward function did not encode was how that prioritization would play out across different spatial layouts. The SCI divergence, where expert entropy remained stable while novice entropy rose steadily across conditions, was not specified in the reward. Neither was the differential response to deception, where expert temporal cost rose sharply in UI-3 while novice temporal cost peaked in UI-2. These are emergent consequences of a prioritization strategy meeting different layout structure. The distinction between encoded behaviour and emergent behaviour is the relevant test for circularity, and on that test the results are meaningful.

5.2 What the Results Do Not Support

Several findings are limited in ways that matter for interpretation.

Task Success Indicator(TSI) and Error Recovery Indicator (ERI) showed ceiling effects across all conditions for both agent types. These two dimensions produced no useful differentiation and should not be interpreted as evidence that the interface configurations were uniformly easy or frustration-free. More likely, the proxy formulas for these dimensions lack the dynamic range needed to capture variation at the scale of this task. Mis-click rates and rage click counts, which feed the Error Recovery Indicator proxy, may simply not occur frequently enough in a button-clicking task with a well-trained agent to produce meaningful variance.

The episode counts per condition are also low, particularly for UI-1, which had only two expert and three novice episodes. Conclusions drawn from UI-1 comparisons should be treated as illustrative rather than definitive. The stronger comparisons are between UI-2 and UI-3, where episode counts were higher and the patterns more consistent.

Most importantly, the agents in this study are trained on a 2D canvas interface, not a full AR environment. The button positions, sizes, and labels are the only interface properties the agent perceives. Real paramedic trainees would also perceive spatial depth, physical embodiment, ambient scene context, and social dynamics. The workload proxies capture behavioral traces of the agent's interaction with the 2D representation, not the full perceptual and cognitive load of immersive AR use. This is a real limitation, and it means the framework should be understood as a tool for comparing interface layout variants, not for predicting absolute workload levels in deployed AR systems.

5.3 Implications for Interface Design

Despite these limitations, the results point to several practical design implications for AR training interfaces in paramedic contexts.

Dense layouts impose temporal and search costs primarily on users without established cue prioritization strategies. For novice trainees who have not yet developed recognition-based decision patterns, dispersing interface elements increases the effort required to locate and process each cue. Designers should consider whether density is intentional, serving as a deliberate challenge, or incidental, arising from information requirements that could be managed through progressive disclosure or layered presentation.

Deceptive spatial clustering, where visually similar elements are grouped regardless of their clinical relevance, imposes a different cost on experienced users. Expert agents spent more time and effort in the deceptive condition not because they could not find targets but because maintaining correct prioritization against misleading spatial groupings required additional work. For AR training interfaces intended to develop expert-level cue recognition, this suggests that spatial arrangement of interface elements is not neutral. Grouping cues by visual similarity rather than clinical relevance may inadvertently train incorrect prioritization habits.

Taken together, these findings suggest that the spatial organization of AR training interface elements should be considered a design variable with measurable behavioural consequences, not simply a layout decision. The simulation framework presented here offers a way to test those consequences before committing to a design.

A recurring concern with simulation-based frameworks is that value cannot be established without prior validation against human data. We argue this mischaracterizes the role of pre-screening. The function of a pre-screening tool is to narrow the design space before human trials begin, not to predict absolute performance. This does not require prior validation; it requires a sound theoretical basis and a clear validation pathway. The theoretical basis here is RPD and CTA-derived profiles grounded in empirical paramedic research. The validation pathway is explicit: a follow-up human-participant study using the same three interface configurations, collecting self-reported TLX ratings for direct comparison with the behavioural indicators reported here. GOMS and KLM were used productively in interface evaluation for years before systematic empirical validation was completed. We make the same claim for this framework.

5.4 Positioning Against Prior Work

The computational rationality tradition has produced powerful models of human-computer interaction that optimize behavior under cognitive and physical constraints. SIM2VR demonstrated that biomechanical simulation can predict user movements and ergonomic variables in VR without human participants (Fischer et al., 2024). This study differs in its motivation and its grounding. Rather than modeling general cognitive-motor behavior, it encodes domain-specific decision strategies derived from naturalistic decision-making research. The RPD model is not a general model of efficient interaction; it is a model of how practitioners in high-stakes fields make decisions under uncertainty. Grounding agent behavior in RPD rather than general rationality assumptions makes the simulated users more relevant to training interface evaluation specifically, even if it limits their generalizability to other domains.

This distinction also addresses a limitation of GOMS and KLM-based approaches. Those models predict expert, error-free performance under well-defined task conditions. The present agents are not error-free: they can fail to complete tasks within the episode limit, produce inefficient paths, and show sensitivity to layout manipulations that GOMS would not capture. This behavioral variability is a feature rather than a limitation, because it is what makes the framework useful for comparing interface conditions.

6. LIMITATIONS

The behavioral indicators are unvalidated against human data. Two dimensions showed ceiling effects and produced no signal. This tells us something useful: mis click-based and rage-click-based metrics are poorly suited to well-trained agents completing a button-clicking task. They may perform better with human participants who make genuine errors under cognitive load. The dimensions that did differentiate conditions, Temporal Interaction Cost and Search Complexity Indicator, are grounded in observable interaction patterns that have theoretical connections to real workload. Whether those connections hold empirically is the question the follow-up study will answer.

The agents operate with full state observability. They know the position, size, and label of every button at every step. Real AR users do not. This simplification removes perceptual uncertainty from the simulation, which means the behavioral indicators reflect pure interaction strategy rather than strategy plus perception. Future work should introduce partial observability to bring the simulation closer to real AR conditions.

The task runs on a 2D canvas rather than a full immersive AR environment. This is a deliberate simplification for a first demonstration, not an oversight. The claim is that layout properties, density, clustering, and labeling, affect interaction cost in ways that a simulation can detect. That claim does not require full AR fidelity to be tested. It does require human validation to be confirmed, which is the next step.

7. CONCLUSION

This paper presented a pre-screening framework for AR training interface evaluation using RPD-grounded simulated users implemented in Unity ML-Agents. Novice and expert agents were derived from CTA findings and evaluated across interface configurations varying in density, layout, and labelling.

The results showed that the agents produced distinct behavioural profiles consistent with their intended decision strategies. Dense layouts primarily affected novice agents, while deceptive clustering imposed greater temporal and interaction costs on expert agents. These findings suggest that simulated users can reveal expertise-dependent design trade-offs before human evaluation begins.

The proposed behavioural interaction indicators provide a structured way to compare interface alternatives without requiring human participants. Although the indicators are not intended to estimate subjective workload and remain unvalidated against human data, they offer a practical mechanism for narrowing the design space before expensive user studies. Future work will evaluate the same interface configurations with human participants and compare behavioural indicators with self-reported and objective performance measures.

8. Acknowledgments:

Claude (Anthropic) was used to assist with improving the writing, data analysis, and formatting throughout this work. The authors verified all content and take full responsibility for the claims, analysis, and conclusions presented.

9. REFERENCES

ANDERSON, J. R., BOTHELL, D., LEBIERE, C. & MATESSA, M. 1998. An integrated theory of list memory. Journal of Memory and Language, 38, 341-380.

BYRNE, M. D. 2001. ACT-R/PM and menu selection: Applying a cognitive architecture to HCI. International Journal of Human-Computer Studies, 55, 41-84.

CARD, S. K., MORAN, T. P. & NEWELL, A. 1980. The keystroke-level model for user performance time with interactive systems. Communications of the ACM, 23, 396-410.

CARD, S. K. M., THOMAS P.; NEWELL, ALLEN 1983. The Psychology of Human-Computer Interaction, Hillsdale, NJ, Lawrence Erlbaum Associates.

DEY, A., BILLINGHURST, M., LINDEMAN, R. W. & SWAN, J. E. 2018. A systematic review of 10 years of augmented reality usability studies: 2005 to 2014. Frontiers in Robotics and AI, 5, 37.

ENDSLEY, M. R., ZSAMBOK, C. & KLEIN, G. 1997. Naturalistic decision making, Mahwah, NJ, Lawrence Erlbaum Associates.

FISCHER, F., IKKALA, A., KLAR, M., FLEIG, A., BACHINSKI, M., MURRAY-SMITH, R.,

HÄMÄLÄINEN, P., OULASVIRTA, A. & MÜLLER, J. Sim2vr: Towards automated biomechanical testing in vr. Proceedings of the 37th Annual ACM Symposium on User Interface Software and Technology, 2024. 1-15.

GABBARD, J. L. & SWAN II, J. E. 2008. Usability engineering for augmented reality: Employing user-based studies to inform design. IEEE Transactions on visualization and computer graphics, 14, 513-525.

GAVISH, N., GUTIÉRREZ, T., WEBEL, S., RODRÍGUEZ, J., PEVERI, M., BOCKHOLT, U. & TECCHIA, F. 2015. Evaluating virtual reality and augmented reality training for industrial maintenance and assembly tasks. Interactive Learning Environments, 23, 778-798.

JOHN, B. E. Using predictive human performance models to inspire and support UI design recommendations. Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, 2011. 983-986.

JOHN, B. E. & KIERAS, D. E. 1996. The GOMS family of user interface analysis techniques: Comparison and contrast. ACM Transactions on Computer-Human Interaction (TOCHI), 3, 320-351.

JULIANI, A., BERGES, V.-P., TENG, E., COHEN, A., HARPER, J., ELION, C., GOY, C., GAO, Y., HENRY, H. & MATTAR, M. 2018. Unity: A general platform for intelligent agents. arXiv preprint arXiv:1809.02627.

KAPTEIN, M. 2016. Using generalized linear (mixed) models in HCI. Modern Statistical Methods for HCI. Springer.

KIERAS, D. E. & MEYER, D. E. 1997. An overview of the EPIC architecture for cognition and performance with application to human-computer interaction. Human–Computer Interaction, 12, 391-438.

KLEIN, G. 1997. Developing Expertise in Decision Making. Informa UK Limited.

KLEIN, G. 2008. Naturalistic Decision Making. Human Factors, 50, 456-460.

KLEIN, G. A. 1993a. Decision making in action: Models and methods, Ablex Norwood, NJ.

KLEIN, G. A. 1993b. A recognition-primed decision (RPD) model of rapid decision making. In: KLEIN, G. A. O., J.; CALDERWOOD, R.; ZSAMBOK, C.E. (ed.) Decision Making in Action: Models and Methods. Ablex.

KLEIN, G. A., CALDERWOOD, R. & CLINTON-CIROCCO, A. Rapid decision making on the fire ground. Proceedings of the human factors society annual meeting, 1986. Sage Publications Sage CA: Los Angeles, CA, 576-580.

LIPSHITZ, R., KLEIN, G., ORASANU, J. & SALAS, E. 2001. Taking stock of naturalistic decision making. Journal of Behavioral Decision Making, 14, 331-352.

OULASVIRTA, A., JOKINEN, J. P. & HOWES, A. Computational rationality as a theory of interaction. Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems, 2022. 1-14.

RITTER, F. E., BAXTER, G. D., JONES, G. & YOUNG, R. M. 2000. Supporting cognitive models as users. ACM Transactions on Computer-Human Interaction (TOCHI), 7, 141-173.

RITTER, F. E. & YOUNG, R. M. 2001. Embodied models as simulated users: Introduction to this special issue on using cognitive models to improve interface design. International Journal of Human-Computer Studies, 55, 1-14.

ROBERTSON, J. & KAPTEIN, M. 2016. An introduction to modern statistical methods in HCI. Modern Statistical Methods for HCI. Springer.

SALVUCCI, D. D. 2001. Predicting the effects of in-car interface use on driver performance: An integrated model approach. International Journal of Human-Computer Studies, 55, 85-107.

SMITH, M. W., BENTLEY, M. A., FERNANDEZ, A. R., GIBSON, G., SCHWEIKHART, S. B. & WOODS, D. D. 2013. Performance of experienced versus less experienced paramedics in managing challenging scenarios: a cognitive task analysis study. Annals of Emergency Medicine, 62, 367-79.

ST. AMANT, R., HORTON, T. E. & RITTER, F. E. Model-based evaluation of cell phone menu interaction. Proceedings of the SIGCHI conference on Human factors in computing systems, 2004. 343-350.

WOBBROCK, J. O. & KAY, M. 2016. Nonparametric statistics in human–computer interaction. *Modern statistical methods for HCI.* Springer.