Augmenting Human Performance with an XR Agent Learning from Online Behavior and BCI Evidence

Anonymous Author(s)

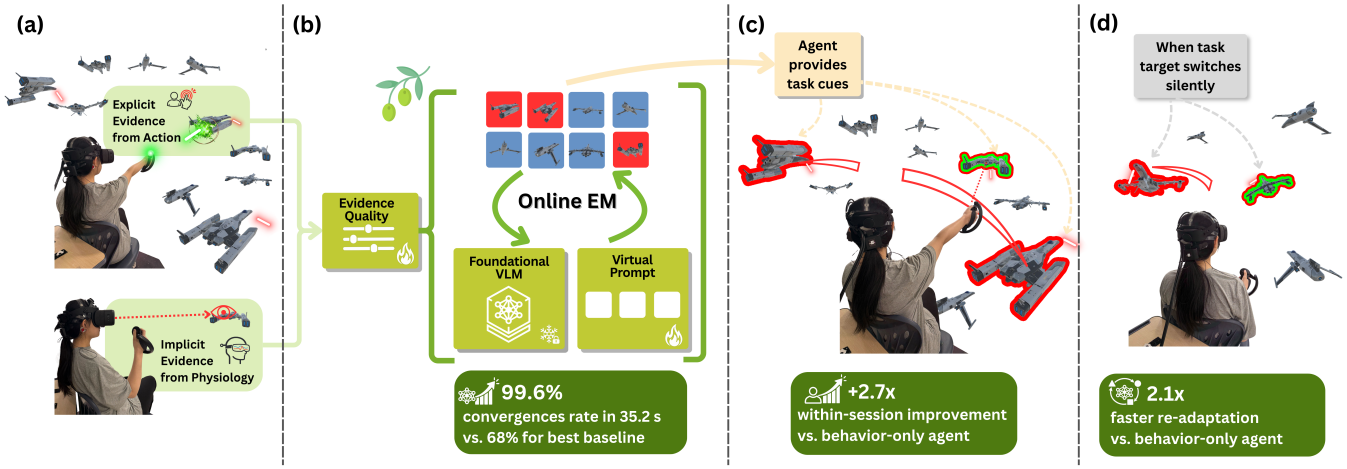


Figure 1: OLIVE deployed in SpaceShooter, an XR target-identification game. (a) Two passive evidence streams (trigger pulls; fixation-locked EEG) feed into an online EM loop. (b) The EM update adapts a virtual prompt atop a frozen VLM, weighting each channel by estimated reliability. (c) Live deployment: OLIVE renders directional line cues toward high-belief targets outside the user's field of view. (d) Silent target switch: OLIVE-IE guidance reconverges faster than the behavior-only agent, reducing the window of misaligned support.

Abstract

We present **OLIVE**, a framework for adapting a foundation model to provide real-time assistance in temporally demanding, high-stakes, and dynamic tasks. We show that passive EEG, fused online with behavioral evidence, can meaningfully extend the number of targets users detect and engage beyond their unaided action bandwidth. OLIVE learns from both *explicit* behavioral signals (trigger pulls, hereafter "shots") and *implicit* physiological signals (fixation-locked EEG) to provide timely guidance, continuously adapting a frozen vision-language model's belief about which items are task-relevant by jointly estimating per-source reliability without manual labels or offline retraining. Through three user studies deploying an assistive agent driven by OLIVE in XR, we show that OLIVE outperforms prior test-time adaptation frameworks in convergence speed and rate. Combining implicit and explicit signals, the agent significantly improves every user's ability to detect and engage targets across the session , regardless of individual shooting skill, while behavioral-only learning produces no reliable throughput gain. When targets switch silently mid-block, OLIVE-IE's guidance reconverges 11.8 s faster on average ($p = 0.061$), shortening the window in which user lack reliable agent support.

CCS Concepts

• **Human-centered computing** → **Mixed / augmented reality**; *Accessibility*; *Interaction techniques*; Collaborative interaction; • **Computing methodologies** → **Machine learning**; *Machine learning*; Machine learning.

Keywords

brain-computer interface, performance augmentation, online adaptation, EEG, fixation-related potentials, attention-limited tasks, shared autonomy, extended reality, AI Agent, OLIVE

ACM Reference Format:

Anonymous Author(s). 2018. Augmenting Human Performance with an XR Agent Learning from Online Behavior and BCI Evidence. In *Proceedings of Make sure to enter the correct conference title from your rights confirmation email (Conference acronym 'XX)*. ACM, New York, NY, USA, 27 pages. https://doi.org/XXXXXXX.XXXXXXX

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

Conference acronym 'XX, Woodstock, NY

© 2018 Copyright held by the owner/author(s). Publication rights licensed to ACM.

ACM ISBN 978-1-4503-XXXX-X/2018/06
https://doi.org/XXXXXXX.XXXXXXX

1 Introduction

Modern work increasingly places people in settings where attention, rather than raw computation, is the bottleneck. Controllers monitor dense trajectories for emerging conflicts; radiologists scan images for subtle but consequential signals; security operators must distinguish true threats from distractors under time pressure. Attention failures in these settings carry real costs: vigilance decrements

under sustained load are implicated in mid-air collisions, undetected cancers, and catastrophic process-control failures [45, 68]. What unifies these domains is a common computational structure: candidates exceed what a human can inspect in time, actions must be taken within expiring windows, and the target class may shift without warning. We call these *operational vigilance (OV) tasks* (defined formally in §2).

Human visual search is fundamentally capacity-limited: recognition degrades under clutter [70], search is guided by selective prioritization rather than exhaustive inspection [71], and even noticed items may go unacted upon due to downstream attentional bottlenecks [63]. A natural response is to add automation, but decades of supervisory control research show the challenge is not simply to automate more; the challenge is to coordinate human and machine behavior without undermining judgment or creating inappropriate reliance [40, 58]. Static alarms and one-shot recommendations are poor fits; what is needed is an assistant that adapts as the task evolves.

Passive brain–computer interfaces (BCI) offer a promising route. Rather than requiring explicit user commands, passive BCIs infer task-relevant states from naturally occurring neural and physiological signals during ongoing interaction [23, 74]. Among these, fixation-aligned neurophysiology is particularly attractive for time-critical search-and-act tasks: the N2pc and P300 EEG signals emerge during visual search and are reliably enhanced for task-relevant targets [14, 50]; pupillometry provides a complementary index of effort and uncertainty [4]. Because these signals can be aligned to natural eye movements during free viewing [12], they provide an always-on pre-action readout of what the user may have recognized before they had a chance to act [25, 59].

Yet BCI research has, for decades, framed these interfaces almost exclusively as assistive technologies. There is a different, largely unexplored application: BCI as a *performance augmentation* tool, acting as an intelligent co-pilot that reads neural signals of attention and intent to handle more than any operator could manage alone, without requiring deliberate BCI commands.

Building such a system requires solving a difficult online learning problem. Explicit user behavior is sparse: users act on only a fraction of what they recognize, and absent actions are ambiguous. Implicit physiological signals are denser but noisy. Test-time adaptation of foundation models assumes unlabeled visual streams, not human-contingent evidence [17, 37, 60]. Crowd-learning methods handle reliability-weighted annotations but not online visual foundation model adaptation [9, 54]. Reinforcement learning sidesteps the label problem but is far too sample-hungry for settings where opportunities expire in seconds [36, 49]. None of these frameworks addresses what we actually need: an assistant that learns what *this* user is currently treating as relevant, from mixed and noisy evidence, and acts on that belief within the task's own time budget.

To address this, we introduce **OLIVE** (**O**nline **L**atent **I**nference from **V**ariable **E**vidence), the algorithmic framework underlying such an agent. OLIVE maintains an evolving posterior over item target-states by fusing behavioral and fixation-locked EEG evidence as they arrive, jointly estimating per-source reliability and adapting a frozen vision–language model (VLM), all within the time budget of the task itself. We instantiate and evaluate this framework across three user studies in a demanding eXtended Reality (XR) setting, asking whether OLIVE converges reliably within an active task block (US1), whether the resulting agent improves live task performance (US2), and whether it readapts quickly when the target class shifts silently (US3).

Our contributions are threefold:

- We introduce **OLIVE**, a general online latent-inference framework that fuses sparse explicit behavioral signals and noisy implicit EEG into a frozen VLM during task execution, jointly estimating per-source reliability weights without manual labels or offline retraining. OLIVE serves as the backbone of an AI agent for *performance augmentation*: operating passively as an intelligent co-pilot, extending operator capability during normal task execution without requiring deliberate BCI commands.
- We show that fixation-locked EEG, incorporated as an implicit evidence channel in OLIVE, achieves 96.8% guidance convergence in 28.3 s (the first third of each 90 s block), Pareto-dominating all single-modality variants and algorithmic baselines in simulation, and remains robust under cognitive overload that degrades explicit-only and implicit-only methods.
- We demonstrate across two live-deployment studies that OLIVE-IE is the only condition in which every participant improves within the session: under sustained operational load (US2), all 8 OLIVE-IE operators increased target throughput ($\Delta = +0.034$ kills/s, $p = 0.004$), independently of shooting skill; and after repeated silent target switches (US3), all 8 OLIVE-IE operators improved on the new target class ($\Delta = +0.084$ kills/s, $p = 0.002$), with guidance reconverging 11.8 s faster than the behavior-only agent ($p = 0.061$, directional).

2 Background

Attention-Limited Tasks and Computational Assistance. Many real-world domains share a common structure: the environment produces a continuous stream of candidates (aircraft trajectories, radiological images, security feeds), the number of candidates exceeds what the operator can inspect before action windows expire, and the relevant target class may shift without explicit notification. We term these *operational vigilance (OV) tasks* [43]: they extend classical vigilance by requiring not just detection but timely execution. OV tasks recur in air traffic control [39], radiology [13], and security triage [8], and share three structural properties:

- **Competing concurrency**: the number of candidates simultaneously demanding attention exceeds the user's inspection bandwidth [15, 45, 68].
- **Limited action window**: the user must act on a target within an expiring window; unlike classical vigilance [43].
- **Contextual target shift**: what constitutes a target can change mid-task without explicit notification, requiring the user (and any assisting system) to readapt.

Rule-based systems hardcode relevance and produce alert fatigue [33, 35, 45, 58]; deep learning matches expert accuracy [6, 8, 53] but adapts to populations, not individuals, and stales silently when task definitions shift [21, 40]. This leaves an open question: *can an assistant learn, in real time, what **this** user is currently treating as relevant?*

Online Foundation Model Adaptation. Foundation VLMs such as CLIP [52] provide open-vocabulary representations that generalize across categories without task-specific retraining. Test-time adaptation (TTA) personalizes such models during inference via entropy minimization, prompt tuning, and consistency regularization [17, 37, 60, 76], but assumes the adaptation signal derives from the unlabeled visual stream. A more fitting lineage comes from *learning from crowds*: Dawid and Skene's Expectation-Maximization (EM) framework jointly estimates annotator reliability and the latent true label rather than pooling labels [9], and was extended to ML by Raykar et al. who parameterized per-annotator sensitivity and specificity [54]. The mapping to our setting is direct: one annotator is the user's explicit action (high specificity, positive-only), another is the user's physiological response at each fixation (denser but noisier, soft-labeled [11]); yet existing crowd-learning models do not account for this structural heterogeneity or the need to update a VLM.

Reinforcement Learning (RL) offers a third framework: update a policy from task-outcome rewards [10, 20, 34, 36, 46, 49, 75]. But our regime is too latency-constrained for RL's sample requirements: opportunities expire within seconds, requiring useful guidance quickly, and EEG non-stationarity further destabilizes the reward signal [18, 24, 66]. Human-centered work on interactive AI has established guidelines for systems that adapt under user feedback [2, 7, 38], but these approaches require deliberate, discrete user input rather than the continuous passive evidence streams (behavioral and physiological) that OLIVE exploits. OLIVE occupies the space none of these frameworks covers: online adaptation of a foundation model from heterogeneous, human-contingent evidence using an EM update that runs within the task's own time budget.

Physiological Signals as Implicit Evidence. That a user acts on only a fraction of the targets they recognize provides a concrete opportunity: the physiological record of a fixation carries evidence of recognition even when no action follows. Fixation-related potentials (FRPs), EEG components time-locked to natural fixation onsets during free viewing, are the primary vehicle, with principled frameworks now available for separating FRP components from overlapping activity of rapid eye movements [11, 12]. Three signals have well-established links to target detection: the N2pc (contralateral posterior negativity, 180–300 ms) marks covert attentional deployment to task-relevant items [14, 41]; the P300 (300–600 ms, parietal) reflects target-category confirmation and tracks the subjective probability that a fixated item is a target [50]; and task-evoked pupillary responses (TEPR, 500–2000 ms) index cognitive effort and evaluative uncertainty [4, 65]. All three are extractable during free viewing without interrupting primary task performance (though TEPR is excluded from our implementation due to luminance contamination in XR; see Appendix S).

Prior work shows fixation-locked EEG distinguishes target from distractor fixations above chance from single trials [25, 59]; more broadly, fNIRS and EEG have been applied to workload-adaptive interfaces [31, 47, 61, 62, 67], cognitive state sensing in VR [56], and closed-loop deployment in ATC [3] and threat monitoring [51, 55]. What these systems share is a decode-then-act architecture: none feeds physiological posteriors into an online EM loop that jointly estimates source reliability and updates a visual foundation model — the missing link OLIVE provides.

Shared Autonomy and Task Design. Knowing which items are task-relevant is necessary but not sufficient for a useful assistant. Parasuraman and Riley's automation framework [45], Sheridan's supervisory-control hierarchy [58], and Horvitz's mixed-initiative principles [32] establish that effective human–machine teaming requires matching machine initiative to user capacity and system confidence; Lee and See's trust-calibration model adds that trust must track demonstrated reliability over time, or users will over-rely on a confident-but-wrong system [40]. Shared-autonomy systems have operationalized these principles, including in robotics via BCI-driven shared control [57, 64] and in HCI via passive neural sensing for adaptation [1, 73]; yet all assume a fixed machine policy and do not accommodate an assistant whose model of relevance is itself evolving in real time.

The form of assistance also matters ethically: hard automation erodes situational awareness [58], and covert neural sensing adds the obligation that users must form accurate mental models of what the system infers and when it will act, or implicit use of brain data risks manipulation that undermines the autonomy it is meant to protect [5, 27, 28]; sensing-based approaches to attention and interruptibility management using physiological signals [19, 77] illustrate the design challenges that arise when implicit biosignals govern system behavior. Micro-assists (brief, subtle attentional cues) preserve the user as the final decision-maker while reducing the cost of target localization. We evaluate OLIVE using a controlled XR surrogate task following established methodology in supervisory control and passive-BCI research [3, 42, 51, 55], affording ground truth and experimental control unavailable in real ATC or clinical settings.

3 Methods

3.1 Problem Statement

We study assistance for operational vigilance (OV) tasks (§2), defined by three structural properties: competing concurrency, limited action windows, and contextual target shift. Given a user can act on only a fraction of recognized items, labeling is structurally *incomplete*, *asymmetric* (absent action does not imply irrelevance since the user may not have reached the item in time), and *heterogeneous*. Signals differ in timing, confidence, and semantics; action events are sparse but consequential; fixation-locked EEG arrives earlier but noisier. At the same time, all adaptation must occur online and within the task's own time budget. Our goal is an assistant that maintains an evolving belief over which visible items are likely targets and integrates visual, behavioral, and physiological evidence continuously to support the user in real time.

3.2 SpaceShooter: A Surrogate Operational Vigilance Task

We instantiate OV as an XR SpaceShooter game in which participants defend a mothership against a mixed fleet of enemy and friendly ships. Targets are enemies at an initial prevalence of $\pi_0 \approx 0.30$. Shooting a friendly ship incurs a disproportionately heavier score penalty than missing an enemy (see Appendix Q), creating

an asymmetric cost structure that mirrors real OV domains such as ATC and medical triage. OLIVE observes visual crops from two scene cameras, receives explicit shot events, and EEG-derived fixation labels per item; the agent renders assistance as a subtle highlight only when its beliefs warrant it, keeping the user in control. Evidence stream details are in Appendix A. Full task design parameters (fleet composition, scoring, shooting mechanics) are in Appendix V; system implementation details are in Appendix U.

To maintain operator engagement throughout the session, block difficulty d is adapted using a QUEST+-style Bayesian staircase [69] targeting ≈70% block score: challenging enough to sustain attention, easy enough that participants remain motivated to engage. Difficulty directly controls fleet size: each block spawns $(1+d)\times 16$ ships, scaling the inference problem alongside task demand (see Appendix O). At the start of each SpaceShooter block, OLIVE's item-level beliefs are cleared and a fresh EM sequence begins, while annotator reliability parameters and the VLM prompt are warm-started from the previous block.

3.3 Visual Search: Calibrating the Physiological Decoder

Before each SpaceShooter session, participants complete a *visual search* (VS) calibration phase in which they locate targets in static arrays under a time budget. Because ground-truth labels are known per array, fixation-locked EEG epochs are labeled and used to train a participant-specific physiological decoder offline. OLIVE uses this decoder as a black-box: given a fixation, it returns a soft label $\hat{y}_i^{\text{physio}} \in (0,1)$. Because OLIVE's M-step A continuously estimates the channel's reliability from session evidence, the framework self-compensates for decoder imprecision without requiring a perfect classifier (see damage-control analysis, Table 1). Decoder training, FRP characterization, and the rationale for excluding pupil dilation are detailed in Appendix S.

3.4 OLIVE: Online Latent Inference from Variable Evidence

Overview. Our core contribution is OLIVE, an online adaptation layer that converts a frozen VLM into a task-personalized assistant by fusing mixed supervision streams during task execution. OLIVE operates at the item level. For each item currently visible in the scene, it maintains a latent posterior: how likely this item is a target. This posterior is updated as new evidence arrives, which may not be as accurate as traditional supervised labels. Instead, OLIVE models each evidence channel as a noisy annotator with its own semantics and reliability, and jointly updates the posterior over item "targetness", the reliability of each evidence source, and the VLM that processes visual evidence. The joint update is what makes OLIVE more than a label aggregator: the latent inference layer and the foundational VLM co-adapt, using mixed evidence to refine beliefs, which in turn refine the visual decision boundary.

Task latent formulation. Let $i \in \{1,\dots,N\}$ index the items seen so far in the current task. Each item has a latent binary target state $z_i \in \{0,1\}$ with prior $P(z_i{=}1) = \pi$ ($\pi \approx 0.30$ in SpaceShooter). OLIVE accumulates three evidence types per item: visual crops $\mathcal{C}_i$, an action label $\hat{y}_i^{\text{act}}$, and a physiological soft label $\hat{y}_i^{\text{physio}}$, and maintains the posterior $\mu_i = P(z_i{=}1 \mid \mathcal{C}_i, \hat{y}_i^{\text{act}}, \hat{y}_i^{\text{physio}})$; these posteriors, representing how likely each item is a target, are the primary output consumed by the downstream assistance layer. We describe each evidence type in turn:

Prompt-tuning of a foundation model as the visual scorer. The visual channel provides appearance-based evidence via a frozen foundational VLM (we use CLIP in our experiments for its fast inference speed) with a learnable virtual prompt [52, 76]. Let $\phi_{\text{img}}(\cdot)$ and $\phi_{\text{txt}}(\cdot)$ denote the frozen unit-norm image and text encoders, respectively. A virtual prompt $\theta_{\text{prompt}} \in \mathbb{R}^{M\times d}$: M free token embeddings of dimension d replacing any natural-language description of the target, passes through ϕ_{txt} to produce a unit-norm target concept embedding $\phi_{\text{txt}}(\theta_{\text{prompt}})$. The visual scorer for a cropped image c of item i is

$$f_{\text{vis}}(c;\theta_{\text{prompt}}) = \sigma\left(\tau\langle\phi_{\text{img}}(c),\, \phi_{\text{txt}}(\theta_{\text{prompt}})\rangle + b\right), \tag{1}$$

where τ and b are fixed logit scale and bias. Per-item crops $\mathcal{C}_i$ are collected over the task with quality weights $\gamma_{ic} \in [0,1]$ (lower for blurry or occluded crops); f_{vis} scores are pooled as a quality-weighted average and converted to a log-likelihood ratio $\Lambda_{\text{vis}}(i) = \gamma_i \,\text{logit}(p_i^{\text{vis}})$. Only θ_{prompt} is adapted; the VLM backbone is frozen throughout.

Explicit evidence from positive-only user actions. For each item i, $\hat{y}_i^{\text{act}} \in \{0,1\}$ records whether the user engaged it. Non-actions are treated as *unlabeled*: in a high-load task the user may simply not have reached a true target in time, so absence of action never implies irrelevance. The channel is parameterized by sensitivity $\alpha = P(\hat{y}_i^{\text{act}}{=}1 \mid z_i{=}1)$ and false-positive rate $\beta = P(\hat{y}_i^{\text{act}}{=}1 \mid z_i{=}0)$; since non-actions are unlabeled, β directly encodes how likely the user acts on non-targets [54].

$$\Lambda_{\text{act}}(\hat{y}_i^{\text{act}}) = \mathbf{1}[\hat{y}_i^{\text{act}}{=}1] \cdot \log\frac{\alpha}{\beta}, \tag{2}$$

and $\hat{y}_i^{\text{act}}{=}0$ contributes zero.

Implicit evidence from a soft physiological annotator. On the first visual fixation of item i, an already-trained physiological decoder returns a soft label $\hat{y}_i^{\text{physio}} \in \mathbb{R}$ (the output of a classifier applied to a fixation-locked EEG epoch) and an optional quality weight $q_i \in [0,1]$. Only the first fixation is used, preventing repeated re-observation of the same item from contributing multiple times to the posterior (although we can consider utilizing multiple fixations with additional parameterization; this is discussed in section 8).

OLIVE models $\hat{y}_i^{\text{physio}}$ under an equal-variance Gaussian likelihood:

$$\hat{y}_i^{\text{physio}} \mid z_i{=}1 \;\sim\; \mathcal{N}(\nu_+, \sigma^2), \quad \hat{y}_i^{\text{physio}} \mid z_i{=}0 \;\sim\; \mathcal{N}(\nu_-, \sigma^2), \tag{3}$$

where $\nu_+ > \nu_-$ are class-conditional means and σ^2 is a shared variance. Under this model, the log-likelihood ratio represents how likely an item is a target, as read from the user's implicit neural response when they fixate it. It simplifies to a linear function of $\hat{y}_i^{\text{physio}}$:

$$\Lambda_{\text{physio}}(\hat{y}_i^{\text{physio}}) = q_i\,\frac{\nu_+ - \nu_-}{\sigma^2}\left(\hat{y}_i^{\text{physio}} - \tfrac{\nu_+ + \nu_-}{2}\right). \tag{4}$$

The three factors capture the key properties of physiological evidence: the sign depends on which class mean $\hat{y}_i^{\text{physio}}$ is closer to;

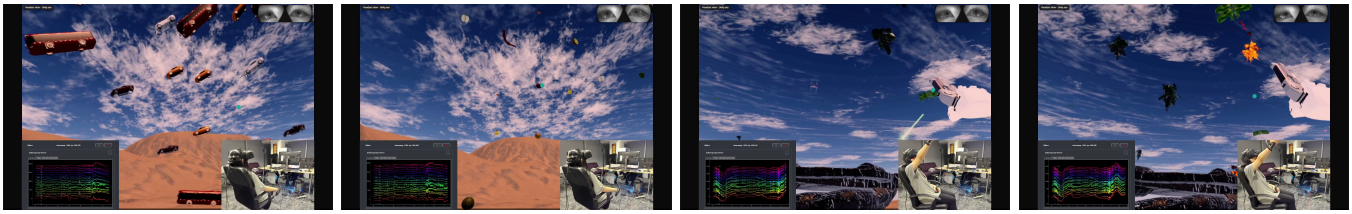


Figure 2: Task environment. (a,b) Visual Search: participants locate the target type; EEG labels train the physio decoder. (c) SpaceShooter: participant aims while OLIVE renders guidance cues toward high-belief targets. (d) Shooting yields an explicit action label for OLIVE's EM update. See Appendix B for an annotated view of the interface elements and experiment apparatus.

the magnitude scales with the channel separation $(\nu_+ - \nu_-)$ and the per-fixation quality weight q_i (initialized in practice to the offline-trained AUC of the user's physio decoder). Adapting ν_+ and ν_- online (M-step A, below) lets OLIVE continuously reestimate how informative the physiological channel is for this user in this session.

Why this parameterization. OLIVE requires only four scalars: (α, β) for the action channel and (ν_+, ν_-) for physiology. Each is estimated online, so imprecise operators and noisy EEG are absorbed as higher inferred uncertainty rather than corrupted evidence; OLIVE assumes imperfection as the default (see Appendix H for the full robustness argument).

Online EM. OLIVE optimizes its parameters (evidence reliability and VLM virtual prompt) in an EM loop during task execution, triggered whenever the user takes an action, or has a new fixation.

E-step. For each item i, log-odds accumulate additively over all three evidence channels:

$$\ell_i \;=\; \log\frac{\pi}{1-\pi} + \Lambda_{\text{vis}}(i) + \Lambda_{\text{act}}(\hat{y}_i^{\text{act}}) + \Lambda_{\text{physio}}(\hat{y}_i^{\text{physio}}), \quad (5)$$

where Λ_{physio} contributes only when a fixation has been observed for item i.

Since targets are rare ($\pi < 0.5$), distractor evidence dominates, collapsing all posteriors toward zero and leaving the operator without any guidance despite their behavioral and neural signals (*zero collapse*). To prevent this, after computing $\{\ell_i\}$ a global bias b^* is found by binary search such that

$$\frac{1}{N}\sum_{i=1}^{N}\sigma(\ell_i + b^*) = \pi, \quad (6)$$

and the posterior is set to $\mu_i = \sigma(\ell_i + b^*)$. This *prevalence anchoring* step pins the mean posterior to π at every round regardless of evidence balance; by the intermediate value theorem a unique b^* always exists, guaranteeing $\bar{\mu} = \pi$ and $\max_i \mu_i \geq \pi > 0$ at every round: the agent cannot stop issuing guidance no matter how one-sided the evidence (Proposition 1; proof in Appendix G).

M-step A: source reliability. Action channel parameters are updated by soft-weighted maximum likelihood with pseudo-count regularization (strength λ):

$$\hat{\alpha} = \frac{\sum_i \mu_i \hat{y}_i^{\text{act}} + \lambda\alpha^{(0)}}{\sum_i \mu_i + \lambda}, \qquad \hat{\beta} = \frac{\sum_i (1-\mu_i)\hat{y}_i^{\text{act}} + \lambda\beta^{(0)}}{\sum_i (1-\mu_i) + \lambda}. \quad (7)$$

All seen items contribute to denominators, even those the user did not act on (prior values and initialisation in Appendix H).

With sparse data, raw M-step estimates can invert channel polarity, turning guidance adversarial: pointing the operator toward wrong items is worse than no guidance at all. Raw estimates are therefore projected onto the ordered half-space $(\alpha \geq \beta, \nu_+ \geq \nu_-)$ via a shrinkage operator: letting $c = \frac{1}{2}[\text{logit}(\hat{\alpha}) + \text{logit}(\hat{\beta})]$ and $\Delta = \frac{n_{\text{eff}}}{n_{\text{eff}}+\delta}\max(\text{logit}(\hat{\alpha}) - \text{logit}(\hat{\beta}), 0)$,

$$\alpha = \sigma\Big(c + \tfrac{\Delta}{2}\Big), \quad \beta = \sigma\Big(c - \tfrac{\Delta}{2}\Big). \quad (8)$$

The physiological class-conditional means are updated by quality-weighted soft MLE:

$$\hat{\nu}_+ = \frac{\sum_i \mu_i q_i \hat{y}_i^{\text{physio}}}{\sum_i \mu_i q_i}, \qquad \hat{\nu}_- = \frac{\sum_i (1-\mu_i) q_i \hat{y}_i^{\text{physio}}}{\sum_i (1-\mu_i) q_i}, \quad (9)$$

then projected to maintain $\nu_+ \geq \nu_-$ via the same shrinkage operator (8).

Since $\Delta \geq 0$ by construction, $\alpha \geq \beta$ (and $\nu_+ \geq \nu_-$) is guaranteed at every round; when the raw M-step would invert polarity, $\Delta = 0$ maps both parameters to the most conservative non-inverted estimate, so misidentified reliability never flips a channel against the operator (Proposition 2; proof in Appendix G).

Together, Propositions 1 and 2 guarantee that OLIVE cannot collapse under any data sequence: the mean posterior is always anchored to π, and all annotator parameters always push posteriors in the semantically correct direction.

M-step B: prompt updates. θ_{prompt} is updated by a gradient step minimizing the quality-weighted soft binary cross-entropy between VLM scores and current posteriors:

$$\mathcal{L}_{\text{VLM}}(\theta_{\text{prompt}}) = -\sum_i \sum_c \gamma_{ic}\Big[\mu_i \log f_{\text{vis}}(c;\theta_{\text{prompt}}) + (1-\mu_i)\log\big(1-f_{\text{vis}}(c;\theta_{\text{prompt}})\big)\Big], \quad (10)$$

where c indexes sampled crops and γ_{ic} is the crop quality weight. The VLM backbone is frozen; only θ_{prompt} receives gradients.

4 User Study 1: Convergence and Robustness

US1 evaluates OLIVE's preference estimator across three evidence configurations (E, I, IE) and three model-type baselines (olive-base, tpt, tda), testing whether multi-modal fusion yields a Pareto-superior operating point that is robust to both user-quality moderators and cognitive overload. We test:

- **H1.1 Pareto frontier:** OLIVE-IE will occupy the upper-right corner of the convergence rate × speed space, outperforming all single-evidence variants and algorithmic baselines on both

dimensions simultaneously. Multi-modal fusion provides complementary evidence streams (EEG enables early coarse estimation while behavioral signals sharpen beliefs), so together they should achieve both higher convergence rates and lower latency than any single channel.

- **H1.2 Robustness to evidence quality:** OLIVE-IE's convergence will be significantly less sensitive to user-level evidence quality (shot accuracy and EEG classifier AUC) than single-modality variants and baselines. When one channel is degraded, the other compensates; single-modality systems have no such fallback.
- **H1.3 Overload robustness:** OLIVE-IE will show no significant relationship between cognitive overload and any convergence metric, while single-modality variants and baselines will show significant overload-related degradation. Overload degrades the two channels through independent mechanisms (overwhelm disrupts attentional EEG signals; high workload reduces shooting precision), so multi-modal fusion remains stable when either channel alone is compromised.

4.1 Study Design

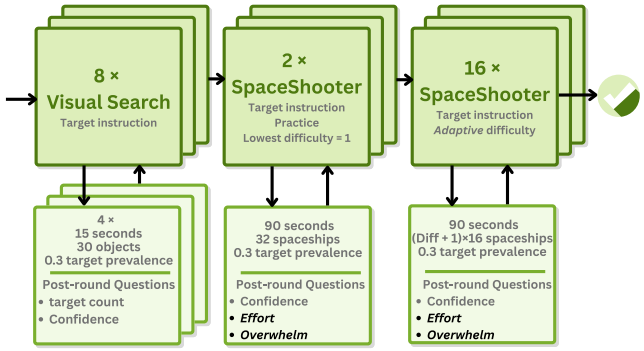


Figure 3: US1 session structure. Visual Search calibration, two practice rounds, then 14 scored SpaceShooter blocks (90 s each); 16 blocks total.

US1 uses a within-subject offline replay design to characterize OLIVE across evidence configurations and algorithmic baselines before any live deployment. Three evidence configurations (full details in Appendix C): **E** (shots only; EEG-free deployment reference), **I** (EEG fixation labels only; no shots), **IE** (both channels; primary condition; E and I are ablations). Three algorithmic baselines receive the same evidence streams as the corresponding OLIVE variant: **olive-base** (M-step A disabled; reliability weights frozen), **tpt** (VLM prompt via entropy minimisation [60]), **tda** (feature-cache retrieval, no backpropagation [17]).

The primary outcomes are **belief convergence** (OLIVE has formed a stable, correct internal ranking) and **guidance convergence**: the tighter, user-facing bar: all four of the agent's highlighted items must be true targets, sustained. A model that ranks correctly only *occasionally* still misdirects the operator; guidance is only trustworthy when the agent is *consistently* right. At the 30% target base rate, chance guidance precision is roughly 0.3; clearing the bar means an operator can follow the cues with confidence. Formal thresholds and rationale are in Appendix E. US1 asks whether

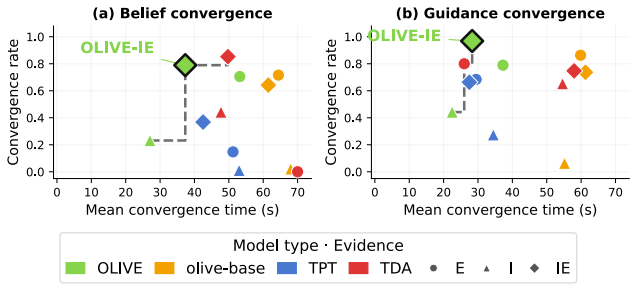


Figure 4: OLIVE-IE occupies the upper-right corner of the convergence rate × speed space for both belief and guidance convergence ($N = 57$; 5 seeds; thresholds in Appendix E). No other variant or baseline simultaneously matches its rate and speed. Best viewed in color.

OLIVE achieves this bar, how quickly, and whether it holds under degraded evidence and cognitive overload.

4.2 Participants

$N = 57$ participants (ages 19 to 35; $\bar{x} = 23.9$, $\sigma = 3.4$ years) completed Study 1. Participants were recruited from the local university community. All procedures were approved by an Institutional Review Board[1], and all participants provided written informed consent prior to participation. Participants reported normal or corrected-to-normal vision. All completed the visual search calibration phase, providing participant-specific physio decoders.

4.3 Results and Discussion

4.3.1 H1.1: OLIVE Sits at the Pareto Frontier. OLIVE-IE achieves the highest guidance convergence rate of all models (96.8%, $n = 92/95$ blocks; 28.3 s), dominating the closest rate competitor (olive-base-E: 86.3%, 59.9 s) and the fastest baseline (tpt-E: 29.5 s but 68.4% rate). TDA-E is notable: it achieves 80.0% guidance rate at 26.0 s yet registers *zero* belief convergence across all blocks; its top-4 rankings are correct but its probability scores never reach the AUC > 0.90 threshold, a model that ranks well without forming a calibrated belief. For belief convergence, OLIVE-IE (78.9%, 37.3 s) is the fastest model on the high-rate frontier; tda-IE reaches a marginally higher rate (85.3%) but at substantially longer latency (49.7 s). No variant or baseline simultaneously matches OLIVE-IE's rate and speed on both metrics, supporting **H1.1**.

OLIVE-IE locks onto the correct target set within the first third of every block (28.3 s of 90 s), giving users trustworthy agent support for the rest of the round, and no user's shooting skill or EEG quality changes that.

4.3.2 H1.2: Sensitivity to Evidence Quality. OLIVE (E and IE variants) shows no sensitivity to shot accuracy (all $|r| < 0.32$, $p > 0.20$), while olive-base-E degrades significantly with lower accuracy ($r = 0.564$, $p = .015$). OLIVE-IE's guidance convergence rate additionally improves with higher EEG quality ($r = 0.717$, $p < .001$), a positive dependency confirming the implicit channel's contribution, not a fragility.

[1]IRB protocol details omitted for blind review.

Table 1: Guidance convergence rate by EEG quality group (median AUC split, cut = 0.77). †$p < .10$, one-sided Fisher exact.

Model	Low EEG	High EEG	Δ	p
OLIVE-I	36%	53%	−17 pp	$0.068^{\dagger}$
OLIVE-IE	94%	100%	−6 pp	0.14

Splitting participants by median decoder AUC (cut = 0.77; Table 1; 95 blocks per variant) shows OLIVE-I drops from 53% to 36% guidance convergence with low EEG (Fisher $p = 0.068$), while OLIVE-IE holds at 94%–100% ($p = 0.14$, n.s.): M-step A suppresses Λ_{physio} when $\hat{\nu}_+ - \hat{\nu}_-$ collapses, falling back to behavioral evidence before corrupted posteriors accumulate. This supports **H1.2**.

4.3.3 H1.3: User Overload's Effect on Model Performance. Using Spearman correlations and mixed-effects models ($N = 57$ participants): OLIVE-I shows a significant overload–convergence relationship ($\rho = 0.539$, $p = .021$), with overwhelm degrading the attentional signals driving EEG evidence, slowing convergence. olive-base-E shows the complementary failure: workload significantly predicts lower belief convergence rate ($r = -0.564$, $p = .015$), degrading the explicit channel. OLIVE-IE shows no significant relationship with any overload metric (all $|\rho| < 0.25$, $p > 0.11$), supporting **H1.3**.

OV tasks demand the most from a user at precisely the moments when individual judgment is most fragile, and this is exactly when OLIVE-IE is most indispensable. In ATC, medical triage, and border surveillance, operator failure concentrates at moments of peak cognitive load, exactly when single-modality systems also fail. OLIVE-IE is immune because the channels fail through independent mechanisms; participants reflected this naturally, shifting toward agent reliance as blocks progressed (*"I often start with shooting by myself and then follow the agent later"*: P34), trusting the model for harder decisions as convergence arrived.

4.3.4 Task Characterization. The adaptive difficulty system confirmed that the task reliably induces real cognitive load (block-level overwhelm $r = 0.15$ with difficulty; full performance and trajectory data in Appendix R).

5 User Study 2: Live Agent Performance

US2 asks whether OLIVE's convergence behavior is preserved in live deployment and whether it translates to measurable operational benefit. We test:

- **H2.1 Convergence in live deployment:** OLIVE will achieve comparable or superior guidance convergence rates and speeds in live deployment (US2) relative to offline simulation (US1). Active operator participation generates richer, more varied behavioral evidence than the fixed offline replays in US1, which may strengthen rather than disrupt the convergence process.
- **H2.2 EEG-augmented throughput advantage:** OLIVE-IE will produce significant within-session target throughput improvement independently of operator shooting skill; OLIVE-E will benefit lower-skill operators disproportionately. Fixation-locked EEG responses occur regardless of shooting precision, making the implicit channel's contribution skill-independent, whereas shot-pattern feedback is weaker for imprecise shooters.

5.1 Study Design

US2 uses a between-subjects design with OLIVE deployed live; conditions are assigned once at session start to prevent belief contamination across conditions. The VS calibration and adaptive difficulty protocol are identical to US1; per-condition difficulty trajectories are in Appendix K. Participants were *single-blind* to condition assignment. Unlike US1, US2 uses **TimeScaledRespawn**: a destroyed ship immediately respawns, keeping the battlefield populated throughout each 120 s block (15 SS blocks: 2 acclimation + 13 adaptive-difficulty).

Conditions and assignment. Control and Oracle bracket the performance range (unaided baseline and ground-truth ceiling); IE is the primary OLIVE condition (Pareto-dominant in US1); E is retained as the EEG-free deployment baseline. Assignment is stratified by US1 starting difficulty to prevent skill confounding (Appendix I).

Primary outcome metric. We measure **target throughput**: confirmed target kills per 120 s block, computed from per-block shot-event logs. Fractional coverage (kills / targets present) is unsuitable in TimeScaledRespawn because destroyed targets immediately respawn, growing the denominator with performance; throughput uses fixed time as the denominator and avoids this bias.

Statistical approach. Condition group sizes are unequal by design: stratified randomization balances ability tiers, not raw counts. All between-condition comparisons use Welch's t-test (no equal-variance or equal-size assumption); within-condition improvement uses one-sample t-tests against zero; LMEs with by-participant random intercepts handle the imbalance through their likelihood framework (full specifications in the appendix).

5.2 Agent Support Policies: Visual Guidance Cues.

The agent withholds all guidance when OLIVE's beliefs are still in flux; actively misleading users with uncertain cues is worse than no guidance at all. Once beliefs concentrate sufficiently, three cue modalities activate: **out-of-view directional lines** (inspired by [29, 30]) pointing toward high-belief targets outside the user's gaze, **outline highlights** on in-view suspects, and **aim assist** that engages when the user's crosshair approaches a top-ranked target. Cue intensity is graded continuously by belief strength, requiring no task-specific calibration. Full algorithmic details (suppression threshold, quantile mapping, smoothing) are in Appendix T.

5.3 Participants

$N = 29$ participants who had completed US1 returned for US2 (Control: n=7; Oracle: n=6; OLIVE-E: n=8; OLIVE-IE: n=8; ages 19 to 35; $\bar{x} = 24.1$, $\sigma = 3.7$ years), 18 male, 11 female, with a minimum 4-day washout between sessions.

5.4 Results and Discussion

5.4.1 H2.1: OLIVE Guidance Convergence Is Maintained in Live Deployment. A key question is whether active user participation alters



Figure 5: Agent cues. *Left:* out-of-view directional lines toward high-belief targets. *Right:* aim-assist highlight on OLIVE's top-ranked item.

the agent's convergence behavior relative to the offline simulation in US1. Table 2 compares per-block guidance and belief convergence rates between US1 (offline simulation) and US2 (live deployment); full trajectory plots are in Appendix F.

Both IE and E achieve **100% guidance convergence** in the live task, matching or exceeding US1 benchmarks of 96.8% (IE) and 79.0% (E). Belief convergence rates also improve (IE: 81.7% vs. 78.9%; E: 85.7% vs. 70.5%), consistent with denser behavioral evidence from live operator decisions. Convergence *times* are longer than US1 (IE guidance: 74.6 vs. 28.3 s), expected given the more demanding live environment (continuous fleet turnover, escalating difficulty). These results support **H2.1**: active deployment does not disrupt convergence rates despite the harder conditions.

Table 2: OLIVE convergence: US2 live deployment vs. US1 offline simulation. Times are mean ± SE (seconds).

Cond.	Type	US2 Rate	US2 Time	US1 Rate	US1 Time
IE	Guidance	**100**%	74.6 ± 5.5	96.8%	28.3 ± 2.1
IE	Belief	81.7%	69.0 ± 6.2	78.9%	37.3 ± 2.3
E	Guidance	**100**%	74.2 ± 6.8	79.0%	37.3 ± 3.5
E	Belief	**85.7**%	60.8 ± 6.5	70.5%	53.2 ± 3.0

5.4.2 *H2.2: OLIVE-IE Produces Reliable Within-Session Throughput Improvement.*

Within-session improvement. Table 3 reports mean target throughput in the first three and last three blocks of each session. OLIVE-IE is the only condition with a statistically significant within-session improvement ($\Delta = +0.034$ kills/s; one-sample $t = 4.62$, $p = 0.004$), and every one of its 8 participants improved. OLIVE-E and Control show marginal positive trends ($p = 0.083$ and $p = 0.062$); Oracle shows no reliable improvement ($p = 0.277$), consistent with automation complacency: Oracle participants received continuous ground-truth guidance throughout, reducing the pressure to self-develop target-discrimination skill across the session.

OLIVE-IE improves uniformly across skill levels. Table 4 shows the correlation between each participant's baseline shooting precision (hit rate) and their within-session throughput improvement. OLIVE-IE's improvement is essentially orthogonal to baseline skill ($r = +0.15$, $p = 0.75$): low-precision and high-precision participants improve equally, consistent with the agent compensating for whatever the operator cannot cover. OLIVE-E shows a strong negative association ($r = -0.90$, $p = 0.10$): its improvement is concentrated among lower-skill participants for whom shot-pattern guidance is most beneficial. Control improvement is positively skill-dependent ($r = +0.34$), as expected without agent assistance.

Table 3: Target throughput (kills/s) early vs. late in session (±SE). †$p < .10$, ∗∗ $p < .01$; one-sample t-test against zero.

Condition	n	Early	Late	Δ	p
Control	7	0.101 ± 0.017	0.123 ± 0.019	+0.022 ± 0.010	.062†
OLIVE-E	8	0.085 ± 0.009	0.113 ± 0.020	+0.028 ± 0.011	.083†
OLIVE-IE	8	0.076 ± 0.011	0.110 ± 0.008	**+0.034 ± 0.007**	.004**
Oracle	6	0.053 ± 0.015	0.071 ± 0.010	+0.018 ± 0.015	.277

Table 4: Skill moderation: correlation between baseline hit rate and Δ throughput. Negative r = lower-skill participants improve more.

Condition	n	r	slope	p
Control	7	+0.34	+0.164	.41
OLIVE-E	8	−0.90	−0.298	.10
OLIVE-IE	8	**+0.15**	+0.037	.75
Oracle	6	−0.21	−0.192	.69

These results support **H2.2**: OLIVE-IE generates consistent, statistically significant within-session throughput improvement that is independent of operator shooting skill. OLIVE-E lifts lower-skill participants; OLIVE-IE lifts everyone. Together they provide complementary evidence that OLIVE's guidance translates to operational benefit across the operator skill distribution. Post-block subjective ratings corroborate this pattern: gaze-reliance scores increase monotonically from Control to OLIVE-E to OLIVE-IE to Oracle, with Oracle reaching statistical separation from Control on both trust ($t = 2.69$, $p = 0.043$) and gaze reliance ($t = 2.97$, $p = 0.039$); full ratings are in Appendix L.

OLIVE-IE is the only condition where every participant improves, regardless of how well they shoot. US3 tests whether OLIVE maintains this partnership when the target definition itself changes without warning.

6 User Study 3: Adapting to Silent Target Changes

US1 and US2 established that OLIVE converges reliably and that convergence translates to operational benefit under sustained load. US3 asks a harder question: what happens when the target changes *mid-block*, without warning and without any system reset? Real OV scenarios routinely demand this kind of re-orientation (an air-traffic controller re-assigned to a different sector, a medic whose patient profile changes), and any viable agent must handle it without collapsing or requiring manual recalibration.

US3 asks whether OLIVE can detect and respond to an abrupt, unannounced change in target identity, a qualitatively harder problem than steady-state convergence. We test:

- **H3.1 Faster model reconvergence:** OLIVE-IE will achieve shorter post-switch guidance and belief reconvergence times, and higher reconvergence rates, than OLIVE-E. Fixation-locked EEG responses during the operator's perceptual reorientation begin shifting model priors before any confirming shot fires, providing an earlier reconvergence signal than behavioral feedback alone.

- **H3.2 Reliable within-session readaptation:** OLIVE-IE will show reliable within-session improvement in new-target throughput across repeated switch exposures. Faster model reconvergence (H3.1) means reliable guidance arrives sooner after each switch; this advantage accumulates over the session through evidence-driven co-adaptation, without any explicit system reset.
- **H3.3 EEG augmentation advantage:** OLIVE-IE will show a significantly larger within-session throughput gain than OLIVE-E. If the advantage is EEG-driven, behavioral-only learning (OLIVE-E) should show a smaller gain, directly isolating the implicit channel's operational contribution.

6.1 Study Design

US3 shares the between-subjects design (Control, Oracle, E, IE) and VS calibration protocol with US2; per-condition difficulty trajectories are in Appendix M. At a pre-specified time within each block, the enemy signature changes without visible system feedback: the current target type becomes friendly and a previously ignored type becomes the new target. Participants were *not informed* that a switch would occur and had to infer it from score feedback alone. OLIVE receives *no explicit reset signal*: readaptation must emerge from shifts in the evidence streams. Session structure and switch-timing balance are detailed in Appendix D.

Primary outcome metric. We measure **new-target throughput**: confirmed hits on the new target category per second in the post-switch window (second_target_shot / $(200 - t_{\text{switch}})$ s), computed per block. Because switch times are balanced across blocks (three blocks at each of 75, 90, and 105 s), participant-level means are unconfounded by the switch timing distribution. Absolute post-switch hit counts cannot serve as the primary metric: an earlier switch yields a proportionally longer post-switch window and therefore mechanically more hits, independent of how well the participant re-oriented. The same statistical approach as US2 applies here: unequal condition sizes (C=7, O=5, E=8, IE=8) are handled by Welch's t-tests for between-condition comparisons and by the LME likelihood framework (see US2 Study Design, Statistical approach).

6.2 Participants

$N = 28$ participants who had completed US2 returned for US3 (Control: n=7; Oracle: n=5; OLIVE-E: n=8; OLIVE-IE: n=8; ages 19 to 30; $\bar{x} = 23.6$, $\sigma = 3.0$ years), 15 male, 13 female, with a minimum 4-day washout between sessions. Participants were naive to the existence of a target switch prior to the study.

6.3 Results and Discussion

6.3.1 H3.1: OLIVE-IE Reconverges Faster After the Switch. As a validity check, near-universal switch detection was observed across all conditions (Control: 100%; E: 98.6%; IE: 100%; Oracle: 100%), confirming that any performance differences reflect re-orientation ability, not unawareness.

Table 5 reports formal post-switch reconvergence using the same criteria as US1 and US2 (Appendix E). Both conditions achieve 100% **guidance reconvergence** (top-4 ranking stability ≥ 0.75 sustained ≥ 10 s); OLIVE-IE does so 11.8 s faster on average (55.0 ± 4.5 s vs. 66.8 ± 4.3 s; $t = -1.88$, $p = 0.061^{\dagger}$). For **belief reconvergence** (additionally requiring belief spread ≥ 0.02), OLIVE-IE succeeds in more blocks (55% vs. 39%) and with a shorter mean time (25.7±4.4 s vs. 29.7 ± 6.1 s), though the time difference is not significant at the current sample size. EEG provides a *leading* signal: fixation-locked responses during the participant's own perceptual reorientation begin shifting the model's priors before any confirming shot on the new target arrives. This supports **H3.1** directionally.

Table 5: Post-switch formal reconvergence (OLIVE-E: n=8; OLIVE-IE: n=8). Times are mean ± SE (seconds). $\dagger p < .10$.

	Guidance		Belief	
Cond.	Rate	Time (s)	Rate	Time (s)
OLIVE-E	100%	66.8 ± 4.3	39%	29.7 ± 6.1
OLIVE-IE	100%	**55.0 ± 4.5**	**55%**	**25.7 ± 4.4**
IE vs. E	—	$t = -1.88,\ p = .061^{\dagger}$	—	$t = -0.52,\ p = .603$

Faster model reconvergence has a direct behavioral consequence: as OLIVE-IE's guidance becomes reliable sooner after each switch, the operator–agent team builds a cumulative re-orientation advantage across repeated exposures.

6.3.2 H3.2: OLIVE-IE Produces Reliable Within-Session Readaptation. Table 6 reports new-target throughput in the first three and last three blocks of each session. **OLIVE-IE is the only condition with a statistically significant within-session improvement** ($\Delta = +0.084$ kills/s; one-sample $t = 11.49$, $p = 0.002^{**}$), and all eight participants improved. Oracle also improves significantly ($p = 0.046^{*}$), consistent with participants learning to follow the agent's immediate re-orientation cue more efficiently across repeated switches. OLIVE-E and Control show positive but non-significant trends.

Table 6: New-target throughput (kills/s in post-switch window) early vs. late session. $\dagger p < .10$, $* p < .05$, $** p < .01$.

Condition	n	Early	Late	Δ	p
Control	7	0.223 ± 0.021	0.253 ± 0.020	+0.022 ± 0.015	.214
OLIVE-E	8	0.227 ± 0.020	0.258 ± 0.014	+0.031 ± 0.019	.149
OLIVE-IE	8	0.185 ± 0.020	0.269 ± 0.030	**+0.084 ± 0.011**	$.002^{**}$
Oracle	5	0.338 ± 0.032	0.416 ± 0.018	+0.078 ± 0.027	$.046^{*}$

Skill moderation. Table 7 shows that OLIVE-IE's improvement is essentially independent of baseline shooting skill ($r = +0.33$, $p = 0.59$, ns) — all eight participants improved by similar margins regardless of hit rate. This replicates the US2 finding: EEG-augmented guidance lifts all operators uniformly, not just high-performers.

Table 7: Skill moderation in US3: correlation between baseline hit rate and Δ new-target throughput.

Condition	n	r	slope	p
Control	7	−0.09	−0.122	.88
OLIVE-E	8	+0.27	+0.366	.56
OLIVE-IE	8	**+0.33**	+0.187	.59
Oracle	5	+0.47	+0.715	.43

6.3.3 H3.3: EEG Integration Produces a Larger Within-Session Gain. Using the within-session delta as the outcome, OLIVE-IE improves significantly more than both Control ($\Delta = +0.062$, $t = 3.35$, $p = 0.011^*$) and OLIVE-E ($\Delta = +0.053$, $t = 2.45$, $p = 0.036^*$). OLIVE-E does not separate from Control ($p = 0.716$), confirming that the H3.2 advantage is specifically EEG-driven: the behavioral channel alone is insufficient to produce reliable co-adaptation across repeated target switches. This supports **H3.3**.

Subjective workload and adaptation ratings are reported in Appendix N.

7 Additional Results

Figure 18 (Appendix S.7) shows the scalp distribution of the grand-average target–distractor ERP difference at successive 100 ms windows across all three user studies' SpaceShooter sessions, confirming a robust posterior P300 (300–500 ms at Pz/POz) consistent with the decoder's discriminative signal.

8 Discussion and Future Work

OLIVE learns user intent from physiology and behavior, guides performance under load, and readapts to silent target shifts, all within the task's own time budget. We discuss what this means beyond the immediate setting.

Effective human–AI teaming requires the agent to earn trust ***through demonstrated alignment with user intent, not through pre-assigned configuration.*** Traditional automation fixes roles at design time; OLIVE inverts this by building trust incrementally, as gaze-reliance ratings increased from Control to OLIVE-E to OLIVE-IE across the session (Appendix L). Preserving human agency at the decision boundary also appears critical: Oracle participants, despite receiving ground-truth guidance, showed no reliable within-session improvement ($p = 0.277$, US2) and the highest frustration scores ($M = 5.33$ vs. $M = 3.2$–3.5 for C/E/IE, US3), consistent with Lee and See's [40] prediction that trust calibrated to demonstrated reliability outperforms trust granted by design.

OLIVE's reliability weights provide a principled path toward ***task-tunable precision–coverage trade-offs.*** OLIVE's per-source reliability weights can be initialized conservatively or updated asymmetrically, shifting from coverage-maximizing to precision-first behavior without structural changes. The E and IE conditions already demonstrate two points on this curve: OLIVE-E's improvement concentrated among lower-skill operators ($r = -0.90$, US2; $r = +0.27$, US3), while OLIVE-IE lifted all operators uniformly ($r = +0.15$, $r = +0.33$), making the evidence-fusion posture an explicit, auditable design parameter rather than an implicit property of training data, as Parasuraman and Riley [45] argue is essential for deployable human–automation systems.

The gap from lab to field is ***a design problem about what constitutes sufficient evidence, not necessarily a signal-processing one.*** SpaceShooter provides three conditions OLIVE depends on: (1) a known target prevalence π for anchoring posteriors, (2) fixation-to-item alignment for EEG epoch extraction, and (3) a confirmation signal (trigger pull) for positive labels. Real OV domains vary on all three (radiology lacks explicit confirmation, ATC has coarser fixation mapping, surveillance may have unknown prevalence), and deployment to each requires validating these conditions. Portable EEG simulation (Appendix S.9) suggests the signal-processing bar is clearable with consumer hardware, and prior deployments confirm that conditions (2) and (3) are achievable in related domains: Aricò et al. [3] deployed passive EEG for cognitive-state monitoring in live ATC controllers, and Pope et al.'s [51] biocybernetic loop operated without explicit confirmation feedback. The harder question is condition (1): domains with unknown or shifting target prevalence require an adaptive anchoring strategy beyond OLIVE's fixed π.

Several extensions would deepen the evidence base and broaden applicability. Multi-fixation fusion (aggregating repeated fixations on the same item) captures inferential value the current architecture discards: US1 used only the first fixation, yet OLIVE-IE still achieved 96.8% guidance convergence, suggesting meaningful signal remains in subsequent fixations. Extending OLIVE's binary posterior to a multi-class belief would support settings demanding distinct responses per category (engage vs. avoid, escalate vs. monitor); US3's silent target-switch paradigm, where OLIVE adapted between two enemy classes purely from evidence, suggests the inference loop can already distinguish class transitions without structural change. OLIVE's per-source reliability weighting is agnostic to channel count: adding dwell time, voice annotation, pupil dilation, or multi-user aggregation requires only a compatible soft annotator. Response-locked ERN (Appendix S.8) is one concrete next channel: frontocentral EEG at shot execution shows a trending target–distractor difference ($p = 0.096$) under active XR conditions, and incorporating it as a third annotator would let the EM framework down-weight misfired actions and update target-class posteriors from the error signal itself.

Limitations. The primary limitation is task scope: SpaceShooter provides ground-truth target labels, EEG-compatible free-viewing fixations, and explicit confirmation feedback that may not hold uniformly in real OV domains. Generalisation to radiology or ATC requires validating that those domains' fixation structure and feedback rate are sufficient for OLIVE's EM updates. The physiological decoder is proprietary; the damage-control analysis (Table 1) shows OLIVE's self-gating compensates for below-median decoder quality, but full replication requires an independently trained decoder. US3's EEG reconvergence advantage is directional at current sample sizes (n=5–8 per condition); a powered replication is needed before H3.1 is treated as confirmed.

9 Conclusion

We presented **OLIVE**, an online latent-inference framework that fuses fixation-locked EEG with behavioral evidence to adapt a VLM within each task round. Three user studies show reliable convergence across the full range of user skill, substantial operational gains from a live agent acting on those beliefs, and real-time readaptation to silent, unannounced target switches. The core lesson: in attention-limited tasks, the AI should earn authority through demonstrated alignment with operator intent, rather than be given it through configuration. We hope this opens a productive design space for passive neural co-pilots wherever human attention is the binding constraint.

References

[1] Daniel Afergan, Evan M. Peck, Erin T. Solovey, Andrew Jenkins, Samuel W. Hincks, Eli T. Brown, Remco Chang, and Robert J. K. Jacob. 2014. Dynamic Difficulty Using Brain Metrics of Workload. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '14)*. ACM, 3797–3806. doi:10.1145/2556288.2557230

[2] Saleema Amershi, Dan Weld, Mihaela Vorvoreanu, Adam Fourney, Besmira Nushi, Penny Collisson, Jina Suh, Shamsi Iqbal, Paul N. Bennett, Kori Inkpen, Jaime Teevan, Ruth Kikin-Gil, and Eric Horvitz. 2019. Guidelines for Human-AI Interaction. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (CHI '19)*. ACM, 1–13. doi:10.1145/3290605.3300233

[3] Pietro Aricò, Gianluca Borghini, Gianluca Di Flumeri, Alfredo Colosimo, Stefano Pozzi, and Fabio Babiloni. 2016. A passive brain–computer interface application for the mental workload assessment on professional air traffic controllers during realistic air traffic control tasks. In *Progress in Brain Research*. Vol. 228. Elsevier, 295–328. doi:10.1016/bs.pbr.2016.04.021

[4] Jackson Beatty. 1982. Task-evoked pupillary responses, processing load, and the structure of processing resources. *Psychological bulletin* 91, 2 (1982), 276–292. doi:10.1037/0033-2909.91.2.276

[5] Dan Bennett, Oussama Metatla, Anne Roudaut, and Elisa D. Mekler. 2023. How does HCI Understand Human Agency and Autonomy?. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*. ACM, 1–18. doi:10.1145/3544548.3580651

[6] Paul Bergmann, Michael Fauser, David Sattlegger, and Carsten Steger. 2019. MVTec AD — A Comprehensive Real-World Dataset for Unsupervised Anomaly Detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 9592–9600. doi:10.1109/cvpr.2019.00982

[7] Carrie J. Cai, Emily Reif, Narayan Hegde, Jason Hipp, Been Kim, Daniel Smilkov, Martin Wattenberg, Fernanda Viégas, Greg S. Corrado, Martin C. Stumpe, and Michael Terry. 2019. Human-Centered Tools for Coping with Imperfect Algorithms During Medical Decision-Making. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (CHI '19)*. ACM, 1–14. doi:10.1145/3290605.3300234

[8] Varun Chandola, Arindam Banerjee, and Vipin Kumar. 2009. Anomaly Detection: A Survey. *Comput. Surveys* 41, 3 (2009), 1–58. doi:10.1145/1541880.1541882

[9] A. P. Dawid and A. M. Skene. 1979. Maximum Likelihood Estimation of Observer Error-Rates Using the EM Algorithm. *Journal of the Royal Statistical Society: Series C (Applied Statistics)* 28, 1 (1979), 20–28. doi:10.2307/2346806

[10] Floris Den Hengst, Eoin Martino Grua, Ali El Hassouni, and Mark Hoogendoorn. 2020. Reinforcement learning for personalization: A systematic literature review. *Data Science* 3, 2 (2020), 107–147. doi:10.3233/DS-200028

[11] Olaf Dimigen and Benedikt V. Ehinger. 2021. Regression-Based Analysis of Combined EEG and Eye-Tracking Data: Theory and Applications. *Journal of Vision* 21, 1 (2021), 3. doi:10.1167/jov.21.1.3

[12] Olaf Dimigen, Werner Sommer, Annette Hohlfeld, Arthur M Jacobs, and Reinhold Kliegl. 2011. Coregistration of eye movements and EEG in natural reading: analyses and review. *Journal of experimental psychology: General* 140, 4 (2011), 552–572. doi:10.1037/a0023885

[13] Kunio Doi. 2007. Computer-Aided Diagnosis in Medical Imaging: Historical Review, Current Status and Future Potential. *Computerized Medical Imaging and Graphics* 31, 4–5 (2007), 198–211. doi:10.1016/j.compmedimag.2007.02.002

[14] Martin Eimer. 1996. The N2pc component as an indicator of attentional selectivity. *Electroencephalography and clinical neurophysiology* 99, 3 (1996), 225–234. doi:10.1016/0013-4694(96)95711-9

[15] Mica R. Endsley. 1995. Toward a Theory of Situation Awareness in Dynamic Systems. *Human Factors: The Journal of the Human Factors and Ergonomics Society* 37, 1 (1995), 32–64. doi:10.1518/001872095779049543

[16] Ronald Fagin, Ravi Kumar, and D. Sivakumar. 2003. Comparing Top k Lists. *SIAM Journal on Discrete Mathematics* 17, 1 (2003), 134–160. doi:10.1137/S0895480102412856

[17] Matteo Farina, Gianni Franchi, Giovanni Iacca, Massimiliano Mancini, and Elisa Ricci. 2024. Frustratingly easy test-time adaptation of vision-language models. *Advances in Neural Information Processing Systems* 37 (2024), 129062–129093. doi:10.48550/arXiv.2405.18330

[18] Aline Xavier Fidêncio, Felix Grün, Christian Klaes, and Ioannis Iossifidis. 2025. Error-related Potential driven Reinforcement Learning for adaptive Brain-Computer Interfaces. *arXiv preprint* (2025). arXiv:2502.18594 [cs.LG] doi:10.48550/arXiv.2502.18594

[19] James Fogarty, Andrew J. Ko, Htet Htet Aung, Elspeth Golden, Karen P. Tang, and Scott E. Hudson. 2005. Examining Task Engagement in Sensor-Based Statistical Models of Human Interruptibility. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '05)*. ACM, 331–340. doi:10.1145/1054972.1055018

[20] Kilian Freitag, Yiannis Karayiannidis, Jan Zbinden, and Rita Laezza. 2025. Fine-tuning Myoelectric Control through Reinforcement Learning in a Game Environment. *IEEE Transactions on Biomedical Engineering* (2025), 1–12. doi:10.1109/TBME.2025.3578855

[21] João Gama, Indrė Žliobaitė, Albert Bifet, Mykola Pechenizkiy, and Abdelhamid Bouchachia. 2014. A Survey on Concept Drift Adaptation. *Comput. Surveys* 46, 4 (2014), 1–37. doi:10.1145/2523813

[22] W. J. Gehring, Y. Liu, J. M. Orr, and J. Carp. 2012. The error-related negativity (ERN/Ne). In *Oxford Handbook of Event-Related Potential Components*, S. J. Luck and E. S. Kappenman (Eds.). Oxford University Press, 231–291.

[23] Laurent George and Anatole Lécuyer. 2010. An overview of research on "passive" brain-computer interfaces for implicit human-computer interaction. In *International Conference on Applied Bionics and Biomechanics ICABB 2010 - Workshop W1 "Brain-Computer Interfacing and Virtual Reality"*.

[24] Benton Girdler, William Caldbeck, and Jihye Bae. 2022. Neural Decoders Using Reinforcement Learning in Brain Machine Interfaces: A Technical Review. *Frontiers in Systems Neuroscience* 16 (2022), 836778. doi:10.3389/fnsys.2022.836778

[25] Jan E Golenia, Michael A Wenzel, Mario Bogojeski, and Benjamin Blankertz. 2018. Implicit relevance feedback from electroencephalography and eye tracking in image search. *Journal of neural engineering* 15, 2 (2018), 026002. doi:10.1088/1741-2552/aa9999

[26] Google LLC. 2015. gRPC: A High-Performance, Open-Source Universal RPC Framework. https://grpc.io.

[27] Emma C Gordon and Anil K Seth. 2024. Ethical considerations for the use of brain–computer interfaces for cognitive enhancement. *PLOS Biology* 22, 10 (2024), e3002899. doi:10.1371/journal.pbio.3002899

[28] Colin M Gray and Shruthi Sai Chivukula. 2019. Ethical mediation in UX practice. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. 1–11. doi:10.1145/3290605.3300410

[29] Uwe Gruenefeld, Abdallah El Ali, Wilko Heuten, and Susanne Boll. 2018. Beyond Halo and Wedge: Visualizing Out-of-View Objects on Head-Mounted Virtual and Augmented Reality Devices. In *Proceedings of the 20th International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI '18)*. ACM, 40:1–40:11. doi:10.1145/3229434.3229438

[30] Sean Gustafson, Patrick Baudisch, Carl Gutwin, and Pourang Irani. 2008. Wedge: Clutter-Free Visualization of Off-Screen Locations. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '08)*. ACM, 787–796. doi:10.1145/1357054.1357179

[31] Leanne Hirshfield, Rebecca Gulotta, Sayer Hirshfield, Samuel Hincks, Michael Russell, Robert Ward, Tim Williams, and Robert Jacob. 2011. This is Your Brain on Interfaces: Enhancing Usability Testing with Functional Near-Infrared Spectroscopy. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '11)*. ACM, 373–382. doi:10.1145/1978942.1978996

[32] Eric Horvitz. 1999. Principles of Mixed-Initiative User Interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '99)*. ACM, 159–166. doi:10.1145/302979.303030

[33] Scott Hudson, James Fogarty, Christopher Atkeson, Daniel Avrahami, Jodi Forlizzi, Sara Kiesler, Johnny Lee, and Jie Yang. 2003. Predicting Human Interruptibility with Sensors: A Wizard of Oz Feasibility Study. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '03)*. ACM, 257–264. doi:10.1145/642611.642657

[34] Wacharawan Intayoad, Chayapol Kamyod, and Punnarumol Temdee. 2020. Reinforcement Learning Based on Contextual Bandits for Personalized Online Learning Recommendation Systems. *Wireless Personal Communications* 115, 4 (2020), 2917–2932. doi:10.1007/s11277-020-07199-0

[35] Shamsi T. Iqbal and Brian P. Bailey. 2008. Effects of Intelligent Notification Management on Users and Their Tasks. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '08)*. ACM, 93–102. doi:10.1145/1357054.1357070

[36] I. Iturrate, L. Montesano, and J. Minguez. 2010. Robot reinforcement learning using EEG-based reward signals. In *2010 IEEE International Conference on Robotics and Automation*. 4822–4829. doi:10.1109/ROBOT.2010.5509734

[37] Yusuke Iwasawa and Yutaka Matsuo. 2021. Test-time classifier adjustment module for model-agnostic domain generalization. *Advances in Neural Information Processing Systems* 34 (2021), 2427–2440. doi:10.5555/3540261.3540447

[38] Rafal Kocielnik, Saleema Amershi, and Paul N. Bennett. 2019. Will You Accept an Imperfect AI? Exploring Designs for Adjusting End-User Expectations of AI Systems. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (CHI '19)*. ACM, 1–14. doi:10.1145/3290605.3300641

[39] James L. Kuchar and L. C. Yang. 2000. A Review of Conflict Detection and Resolution Modeling Methods. *IEEE Transactions on Intelligent Transportation Systems* 1, 4 (2000), 179–189. doi:10.1109/6979.898217

[40] John D Lee and Katrina A See. 2004. Trust in automation: designing for appropriate reliance. *Human Factors* 46, 1 (2004), 50–80. doi:10.1518/hfes.46.1.50.30392

[41] Steven J Luck and Steven A Hillyard. 1994. Electrophysiological correlates of feature analysis during visual search. *Psychophysiology* 31, 3 (1994), 291–308. doi:10.1111/j.1469-8986.1994.tb02218.x

[42] Tiffany Luong and Christian Holz. 2022. Characterizing Physiological Responses to Fear, Frustration, and Insight in Virtual Reality. *IEEE Transactions on Visualization and Computer Graphics* 28, 11 (2022), 3917–3927. doi:10.1109/TVCG.2022.3203113

[43] Norman H. Mackworth. 1948. The Breakdown of Vigilance During Prolonged Visual Search. *Quarterly Journal of Experimental Psychology* 1, 1 (1948), 6–21. doi:10.1080/17470214808416738

[44] Sarah Nogueira, Konstantinos Sechidis, and Gavin Brown. 2018. On the Stability of Feature Selection Algorithms. *Journal of Machine Learning Research* 18, 174 (2018), 1–54. doi:10.5555/3122009.3242031

[45] Raja Parasuraman and Victor Riley. 1997. Humans and Automation: Use, Misuse, Disuse, Abuse. *Human Factors: The Journal of the Human Factors and Ergonomics Society* 39, 2 (1997), 230–253. doi:10.1518/001872097778543886

[46] Jongchan Park, Mingyu Park, and Donghwan Lee. 2025. *Pretraining a Shared Q-Network for Data-Efficient Offline Reinforcement Learning.* arXiv:2505.05701 [cs.AI] doi:10.48550/arXiv.2505.05701

[47] Evan M. Peck, Beste F. Yuksel, Alvitta Ottley, Robert J. K. Jacob, and Remco Chang. 2013. Using fNIRS Brain Sensing to Evaluate Information Visualization Interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '13).* ACM, 473–482. doi:10.1145/2470654.2470723

[48] F. Perrin, J. Pernier, O. Bertrand, and J. F. Echallier. 1989. Spherical splines for scalp potential and current density mapping. *Electroencephalography and Clinical Neurophysiology* 72, 2 (1989), 184–187. doi:10.1016/0013-4694(89)90180-6

[49] Eric A. Pohlmeyer, Babak Mahmoudi, Shijia Geng, Noeline W. Prins, and Justin C. Sanchez. 2014. Using Reinforcement Learning to Provide Stable Brain-Machine Interface Control Despite Neural Input Reorganization. *PLOS ONE* 9, 1 (2014), e87253. doi:10.1371/journal.pone.0087253

[50] John Polich. 2007. Updating P300: an integrative theory of P3a and P3b. *Clinical neurophysiology* 118, 10 (2007), 2128–2148. doi:10.1016/j.clinph.2007.04.019

[51] Alan T Pope, Edward H Bogart, and Debbie S Bartolome. 1995. Biocybernetic system evaluates indices of operator engagement in automated task. *Biological Psychology* 40, 1-2 (1995), 187–195. doi:10.1016/0301-0511(95)05116-3

[52] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning Transferable Visual Models From Natural Language Supervision. In *Proceedings of the 38th International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 139),* Marina Meila and Tong Zhang (Eds.). PMLR, 8748–8763.

[53] Pranav Rajpurkar, Jeremy Irvin, Robyn L. Ball, Kaylie Zhu, Brandon Yang, Hershel Mehta, Tony Duan, Daisy Ding, Aaditya Bagul, Curtis P. Langlotz, Bhavik N. Patel, Kristen W. Yeom, Katie Shpanskaya, Francis G. Blankenberg, Josie Seekins, Timothy J. Amrhein, David A. Mong, Safwan S. Halabi, Evan J. Zucker, Andrew Y. Ng, and Matthew P. Lungren. 2018. Deep Learning for Chest Radiograph Diagnosis: A Retrospective Comparison of the CheXNeXt Algorithm to Practicing Radiologists. *PLOS Medicine* 15, 11 (2018), e1002686. doi:10.1371/journal.pmed.1002686

[54] Vikas C. Raykar, Shipeng Yu, Linda H. Zhao, Gerardo Hermosillo Valadez, Charles Florin, Luca Bogoni, and Linda Moy. 2010. Learning From Crowds. *Journal of Machine Learning Research* 11 (2010), 1297–1322. https://www.jmlr.org/papers/volume11/raykar10a/raykar10a.pdf

[55] Sameer Saproo, Victor Shih, David C. Jangraw, and Paul Sajda. 2016. Neural mechanisms underlying catastrophic failure in human-machine interaction during aerial navigation. *Journal of Neural Engineering* 13, 6 (2016), 066005. doi:10.1088/1741-2560/13/6/066005

[56] Emile Savalle, Léa Pillette, Kyungho Won, Ferran Argelaguet, Anatole Lécuyer, and Marc J-M Macé. 2024. Towards Electrophysiological Measurement of Presence in Virtual Reality Through Auditory Oddball Stimuli. *Journal of Neural Engineering* 21, 4 (2024), 046015. doi:10.1088/1741-2552/ad5cc2

[57] Charles Schaff, Matthew R Walter, and Stefanie Tellex Rao. 2020. Residual policy learning for shared autonomy. In *Proceedings of Robotics: Science and Systems (RSS).*

[58] Thomas B. Sheridan. 1992. *Telerobotics, automation, and human supervisory control.* MIT Press, Cambridge, MA.

[59] Sergei L. Shishkin, Yuri O. Nuzhdin, Evgeny P. Svirin, Alexander G. Trofimov, Anastasia A. Fedorova, Bogdan L. Kozyrskiy, and Boris M. Velichkovsky. 2016. EEG Negativity in Fixations Used for Gaze-Based Control: Toward Converting Intentions into Actions with an Eye–Brain–Computer Interface. *Frontiers in Neuroscience* 10 (11 2016). doi:10.3389/fnins.2016.00528

[60] Manli Shu, Weili Nie, De-An Huang, Zhiding Yu, Tom Goldstein, Anima Anandkumar, and Chaowei Xiao. 2022. Test-time prompt tuning for zero-shot generalization in vision-language models. *Advances in Neural Information Processing Systems* 35 (2022), 14274–14289. doi:10.48550/arXiv.2209.07511

[61] Erin Treacy Solovey, Audrey Girouard, Krysta Chauncey, Leanne M. Hirshfield, Angelo Sassaroli, Feng Zheng, Sergio Fantini, and Robert J. K. Jacob. 2009. Using fNIRS Brain Sensing in Realistic HCI Settings: Experiments and Guidelines. In *Proceedings of the 22nd Annual ACM Symposium on User Interface Software and Technology (UIST '09).* ACM, 157–166. doi:10.1145/1622176.1622207

[62] Erin Treacy Solovey, Paul Schermerhorn, Matthias Scheutz, Angelo Sassaroli, Sergio Fantini, and Robert J. K. Jacob. 2012. Brainput: Enhancing Interactive Systems with Streaming fNIRS Brain Input. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '12).* ACM, 2193–2202. doi:10.1145/2207676.2208372

[63] Michael N Tombu, Christopher L Asplund, Paul E Dux, Douglass Godwin, Justin W Martin, and René Marois. 2011. A unified attentional bottleneck in the human brain. *Proceedings of the National Academy of Sciences* 108, 33 (2011), 13426–13431. doi:10.1073/pnas.1103583108

[64] Lucas Tonin, Robert Leeb, Matilde Tavella, Stefanos Perdikis, and José del R Millán. 2010. The role of shared-control in BCI-based telepresence. In *2010 IEEE International Conference on Systems, Man and Cybernetics.* IEEE, 1462–1466.

[65] Pauline Van der Wel and Henk Van Steenbergen. 2018. Pupil dilation as an index of effort in cognitive control tasks: A review. *Psychonomic bulletin & review* 25, 6 (2018), 2005–2015. doi:10.3758/s13423-018-1432-y

[66] Mathias Vukelić, Michael Bui, Anna Vorreuther, and Katharina Lingelbach. 2023. Combining brain-computer interfaces with deep reinforcement learning for robot training: a feasibility study in a simulation environment. *Frontiers in Neuroergonomics* 4 (2023), 1274730. doi:10.3389/fnrgo.2023.1274730

[67] Liang Wang, Zhe Huang, Ziyu Zhou, Devon McKeon, Giles Blaney, Michael C. Hughes, and Robert J. K. Jacob. 2021. Taming fNIRS-based BCI Input for Better Calibration and Broader Use. In *Proceedings of the 34th Annual ACM Symposium on User Interface Software and Technology (UIST '21).* ACM, 1126–1140. doi:10.1145/3472749.3474743

[68] Joel S. Warm, Raja Parasuraman, and Gerald Matthews. 2008. Vigilance Requires Hard Mental Work and Is Stressful. *Human Factors* 50, 3 (2008), 433–441. doi:10.1518/001872008X312152

[69] Andrew B. Watson. 2017. QUEST+: A General Multidimensional Bayesian Adaptive Psychometric Method. *Journal of Vision* 17, 3 (2017), 10. doi:10.1167/17.3.10

[70] David Whitney and Dennis M Levi. 2011. Visual crowding: A fundamental limit on conscious perception and object recognition. *Trends in cognitive sciences* 15, 4 (2011), 160–168. doi:10.1016/j.tics.2011.02.005

[71] Jeremy M Wolfe. 2021. Guided Search 6.0: An updated model of visual search. *Psychonomic bulletin & review* 28, 4 (2021), 1060–1092. doi:10.3758/s13423-020-01859-9

[72] Z. Xu et al. 2023. PhysioLabXR: A Python Platform for Real-Time, Multi-Modal Brain–Computer Interfaces and Extended Reality Experiments. *Journal of Open Source Software* 8, 89 (2023), 5854. doi:10.21105/joss.05854

[73] Beste F. Yuksel, Kurt B. Oleson, Lane Harrison, Evan M. Peck, Daniel Afergan, Remco Chang, and Robert J. K. Jacob. 2016. Learn Piano with BACh: An Adaptive Learning Interface that Adjusts Task Difficulty Based on Brain State. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16).* ACM, 5372–5384. doi:10.1145/2858036.2858388

[74] Thorsten O Zander and Christian Kothe. 2011. Towards passive brain–computer interfaces: applying brain–computer interface technology to human–machine systems in general. *Journal of neural engineering* 8, 2 (2011), 025005. doi:10.1088/1741-2560/8/2/025005

[75] Han Zheng, Xufang Luo, Pengfei Wei, Xuan Song, Dongsheng Li, and Jing Jiang. 2023. Adaptive Policy Learning for Offline-to-Online Reinforcement Learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 37. 11372–11380. doi:10.1609/aaai.v37i9.26345

[76] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. 2022. Learning to Prompt for Vision-Language Models. *International Journal of Computer Vision* 130, 9 (2022), 2337–2348. doi:10.1007/s11263-022-01653-1

[77] Manuela Züger, Sebastian C. Müller, André N. Meyer, and Thomas Fritz. 2018. Sensing Interruptibility in the Office: A Field Study on the Use of Biometric and Computer Interaction Sensors. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18).* ACM, 1–14. doi:10.1145/3173574.3174165

A SpaceShooter Evidence Streams

OLIVE ingests three evidence streams from SpaceShooter.

Visual. Two cameras capture the scene at 1 FPS (448 × 448 pixels): a head-worn *user camera* and an *agent camera* that orbits the user and continuously steers toward the least-recently-observed sector of the upper hemisphere, providing visual coverage of items outside the participant's current gaze (algorithm in Appendix P). Each spaceship instance is object-tracked; bounding-box crops are quality-weighted and cached as visual evidence for OLIVE's VLM scorer.

Explicit. When the user's shot lands on a ship, OLIVE receives an action label marking that ship as confirmed enemy. Non-shots are treated as unlabeled (absence of action does not imply the item is a non-target; the user may simply not have reached it in time).

Implicit. Each fixation triggers extraction of a fixation-locked Event-Related Potential (ERP) from the participant's EEG. The per-participant physiological decoder (trained offline on Visual Search data; see Appendix S) returns a soft probability $\hat{y}_i^{\text{physio}} \in (0, 1)$ that the fixated item is a target, used as the implicit evidence channel.

B Experimental Setup

The experimental setup used a **Varjo XR-3** mixed-reality headset (eye tracking at 200 Hz, 115° horizontal field of view) running SpaceShooter in stereoscopic XR, paired with an **Advanced Brain Monitoring (ABM) B-Alert X24** wireless EEG system (256 Hz, 20 electrodes, standard 10–20 layout). Figure 6 shows a screenshot captured during a live SpaceShooter session, with the four simultaneous data streams visible to the experimenter. The main XR viewport occupies most of the screen; the three inset panels document the physiological and physical recording environment. This layout was used to monitor data quality and participant state throughout each session without interrupting task performance.

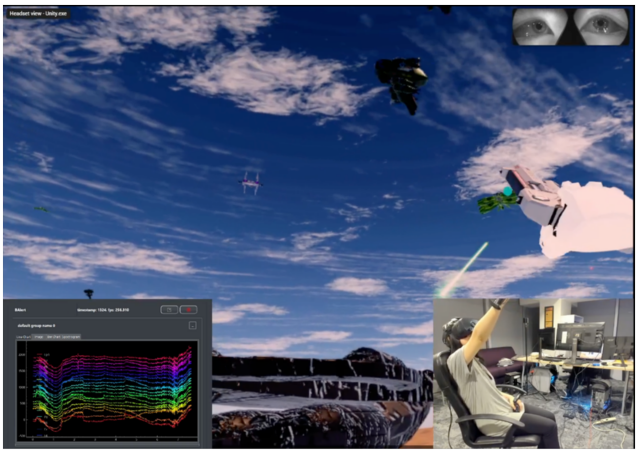


Figure 6: Annotated screenshot from a live SpaceShooter session. Main area: first-person XR view through the VR headset; the participant's virtual hand and controller are visible on the right alongside enemy ships. Top right: eye-tracking camera feeds for both eyes, used to detect fixation onsets and align EEG epochs to specific on-screen items. Bottom left: real-time EEG monitor displaying raw traces from all 20 electrodes, confirming continuous recording. Bottom right: lab webcam of the real participant seated with the EEG headset and VR headset.

C US1 Evidence Configurations and Baselines

Evidence configurations. **E (Explicit only)** receives shot events as positive labels and no EEG signal; it represents the EEG-free deployment scenario, requiring no neural hardware, and is the reference against which EEG's added value is measured in US2 and US3. **I (Implicit only)** receives EEG-derived fixation labels only, with no shot events; it tests whether neural signals alone are sufficient without any behavioral confirmation. **IE (Implicit + Explicit)** receives both channels and is the primary condition; E and I are ablations that decompose its advantage.

Baselines. **olive-base** is OLIVE with M-step A disabled: evidence reliability weights (α, β, ν_+, ν_-) are frozen at pilot-calibrated values rather than estimated online, ablating the benefit of learned reliability.

tpt adapts the VLM prompt via entropy minimisation on N=7 augmented views per fixation-locked crop, keeping the top selection_p=0.1 fraction of views by self-consistency before computing the entropy loss [60].

tda maintains positive and negative CLIP feature caches updated from confirmed targets (shots / high-EEG soft labels) and distractors (low-EEG soft labels); posteriors are computed by cosine-similarity retrieval against the two caches with no backpropagation [17].

All baselines receive the same evidence streams as the corresponding OLIVE evidence variant and share the same visual encoder (frozen CLIP ViT-B/16).

D US2 and US3 Session Structure

Figures 7 and 8 show the session flow for US2 and US3 respectively. Both follow the same structure as US1 (Visual Search calibration then SpaceShooter blocks) but differ in two key ways: US2 introduces TimeScaledRespawn (ships respawn when shot, 120 s blocks) and live OLIVE deployment; US3 adds a mid-block silent target switch at a randomised time ($\in \{75, 90, 105\}$ s) unknown to participants.

US2 conditions. All four conditions share the same task; they differ only in what assistance the agent provides. **Control**: no agent assistance: the participant acts alone and receives no guidance cues. **Oracle**: the agent receives ground-truth target labels directly from the game engine and uses them to guide the participant, establishing an empirical performance ceiling. **E** and **IE** use OLIVE (EEG-free and EEG-augmented respectively); see §3.4 and Appendix C.

US2 TimeScaledRespawn rationale. Because a ship respawns immediately when shot, the battlefield stays populated and the agent must maintain useful guidance throughout each block rather than converging once to a static target set. This is essential for testing whether OLIVE's guidance holds under ongoing pressure, not just during an initial convergence window.

US3 switch-timing design. Switch times were drawn from $\{75, 90, 105\}$ s, with each value appearing exactly three times across the nine data-collection blocks in a randomised order per participant. This balanced design controls for the structural confound that earlier switches yield longer post-switch windows (≈125 s, 110 s, and 95 s respectively): across a full session, each participant has the same aggregate distribution of post-switch time, ensuring participant-level means are unconfounded by switch timing. Participants were required to infer the switch entirely from evolving in-game score feedback (a change in which ship type yields favourable outcomes) without any system alert or instruction to re-orient, replicating real-world settings where task priorities shift through environmental cues rather than explicit notifications. The session comprised 9 SS blocks (1 acclimation + 8 data blocks, plus one additional exploratory block per participant to maintain the 3×3 timing balance).

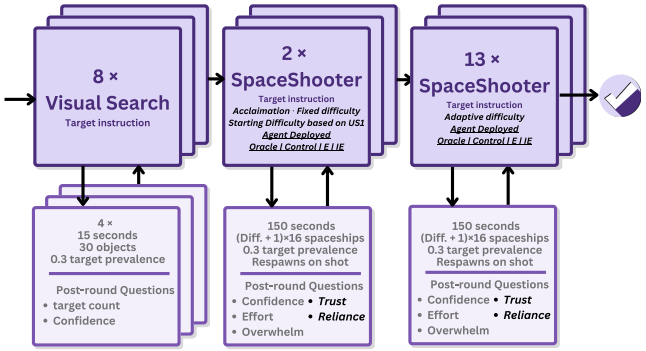


Figure 7: US2 session structure. Participants are assigned to one of four between-subjects conditions at session start. OLIVE runs live throughout all SpaceShooter rounds (TimeScaledRespawn variant).

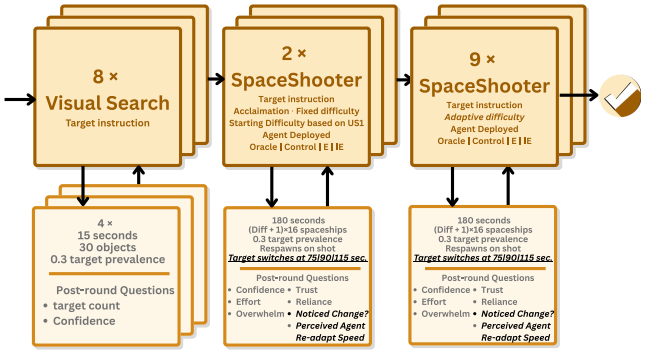


Figure 8: US3 session structure. Identical to US2 except a silent target switch occurs at a pre-randomised time (unknown to participants).

E Convergence Metrics

An unconverged model does not merely withhold useful guidance: it actively misleads: its highest-ranked items are plausible-looking distractors, and a user following the cues will make worse decisions than one ignoring them entirely. We therefore define convergence not as internal accuracy alone, but as the moment when the agent's guidance becomes reliably worth acting on.

Both metrics share a *ranking stability* component. Let T_t denote the set of four items with the highest posterior beliefs at second t. We define

$$S_t = \frac{|T_t \cap T_{t-1}|}{4}, \tag{11}$$

the fraction of the top-4 set unchanged from the previous second, an intersection-based overlap measure analogous to those used for comparing top-k lists [16] and quantifying feature selection stability [44]. When S_t is high, the agent consistently surfaces the same items; when S_t is low, its suggestions are still in flux and following them would send the user in different directions moment to moment.

Belief convergence: the model has formed a stable, correct view. We operationalize this as: (i) the posterior-based ranking achieves AUC > 0.90 against ground-truth target labels, (ii) $S_t \geq 0.75$ (at least three of the four highlighted items unchanged each second), and (iii) both conditions hold for at least 10 consecutive seconds, long enough for the user to receive and act on the agent's support.

Guidance convergence: the model's suggestions are trustworthy. Stability alone is insufficient: a model can be stably wrong. Guidance convergence marks the point at which the items the agent surfaces are actually the true targets. We operationalize this as: Precision@4 ≥ 1.0 (all four highlighted items are true targets) *and* $S_t \geq 0.75$, sustained for ≥ 10 consecutive seconds. We report time-to-guidance-convergence as a primary outcome in US1 and US2.

Both thresholds are intentionally strict: requiring *all four* highlighted items to be correct and *sustained* for 10 seconds rules out transient agreement: the agent must be consistently right. At a 30% target base rate, chance Precision@4 ≈ 0.3; a system that clears this bar within the first half of a task block is genuinely useful, allowing users to redirect their attention with confidence.

F Convergence: Offline vs. Live Deployment

Figure 9 compares OLIVE's per-block convergence trajectories between US1 (offline simulation replay) and US2 (live deployment) for the IE and E conditions. Both conditions reach 100% guidance convergence in live deployment, exceeding their US1 offline benchmarks (IE: 96.8%; E: 79.0%), consistent with denser and more varied behavioral evidence in the live task.

G Formal Propositions and Proofs

Proposition 1 (Prevalence anchoring prevents zero collapse). *Let $\pi \in (0,1)$ and $N \geq 1$. After anchoring* (6), $\frac{1}{N}\sum_i \mu_i = \pi$, *so* $\max_i \mu_i \geq \pi > 0$ *at every EM round.*

Proof sketch. $g(b) = \frac{1}{N}\sum_i \sigma(\ell_i + b)$ is continuous, strictly increasing, with $g \to 0$ as $b \to -\infty$ and $g \to 1$ as $b \to +\infty$. By the intermediate value theorem a unique b^* with $g(b^*) = \pi$ exists, so after anchoring $\bar{\mu} = \pi$ and $\max_i \mu_i \geq \pi > 0$. □

Proposition 2 (Ordered projection prevents polarity inversion). *The projection* (8) *ensures $\alpha \geq \beta$ (and $v_+ \geq v_-$) at every EM round, regardless of the raw M-step estimates.*

Proof sketch. Since $\Delta = \frac{n_{\text{eff}}}{n_{\text{eff}}+\delta}\max(\cdot, 0) \geq 0$, we have $c + \Delta/2 \geq c - \Delta/2$ and therefore $\alpha \geq \beta$ always. When the raw M-step would invert polarity ($\hat{\alpha} \leq \hat{\beta}$), $\Delta = 0$ and both are mapped to $\sigma(c)$, the most conservative non-inverted estimate. □

H OLIVE Parameterization: Robustness Rationale

OLIVE's four learned scalars each absorb a specific real-world failure mode rather than propagating it as corrupted evidence. For the action channel, $\alpha = P(\hat{y}^{\text{act}}{=}1 \mid z{=}1)$ and $\beta = P(\hat{y}^{\text{act}}{=}1 \mid z{=}0)$ are estimated from observed shots: if a participant occasionally fires on a non-target, OLIVE raises $\hat{\beta}$ and automatically down-weights the action channel's log-likelihood ratio, absorbing imprecision rather than propagating it. For the physiological channel, (v_+, v_-) are the class-conditional EEG means estimated online per session: motion artifacts and facial-muscle activity compress the inferred

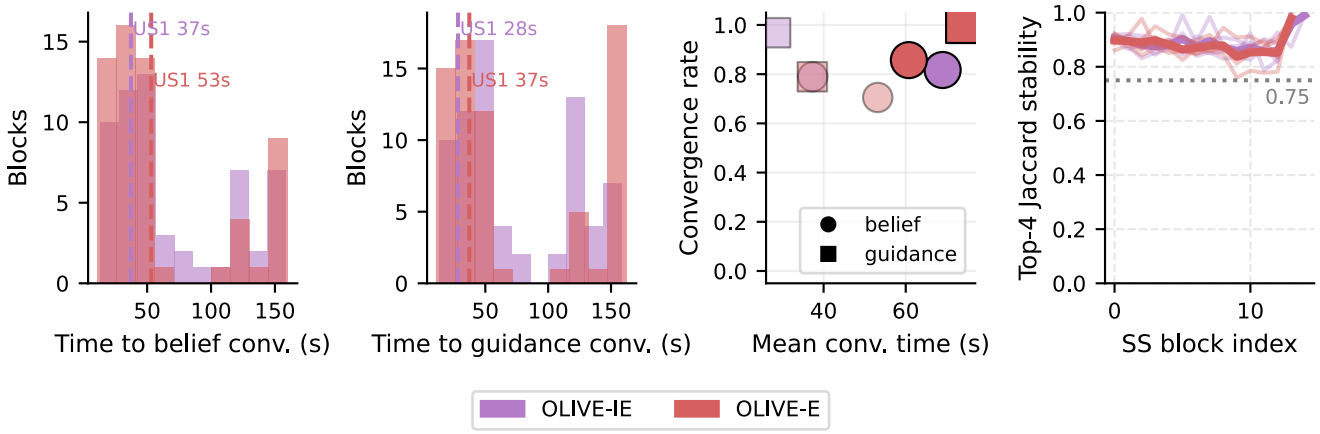


Figure 9: OLIVE convergence: US2 live deployment vs. US1 offline simulation (IE and E). Top row: time-to-convergence curves for belief spread (score std) and top-4 ranking stability over SS block sequence. Bottom row: convergence rate comparison across all condition × type combinations.

separation ($v_+ - v_-$), attenuating the channel's influence on the posterior rather than corrupting it. OLIVE does not require well-behaved users or clean sensors; it treats imperfection as the default and continuously calibrates how much to trust each channel from the evidence it receives.

Initialisation. Action channel priors are set from pilot data: $\alpha^{(0)} = 0.85$, $\beta^{(0)} = 0.15$ (pilot specificity = 0.85). The physiological quality weight q_i is initialised to the offline-trained AUC of the participant's physiological decoder from the Visual Search calibration phase.

I US2 Condition Assignment Procedure

To prevent shooting skill from confounding the between-subjects comparison, each participant's US2 starting difficulty was computed as the score-weighted mean difficulty over their last six qualifying US1 blocks (blocks meeting thresholds on shooting precision, throughput, and mothership health). Participants were then binned into four ability tiers ($d < 2$, $2 \le d < 3$, $3 \le d < 4$, $d \ge 4$) and conditions were balanced within each tier before global equalization, ensuring no condition disproportionately received higher- or lower-ability participants.

J US3 Statistical Model Details

The primary metric is new-target throughput: second_target_shot/$(200 - t_{\text{switch}})$ kills/s per block, averaged per participant. Switch times are balanced (three blocks each at 75, 90, 105 s), so participant-level means are unconfounded by timing. The within-session delta (late three blocks minus early three blocks) is the primary inferential quantity; one-sample t-tests are used to test improvement against zero, and Welch t-tests for between-condition comparisons (see Statistical approach in US2 Study Design).

For reference, a block-level LME on raw `second_target_shot` with condition, difficulty, and switch time as fixed effects and by-participant random intercepts (Control as reference) yields $\hat{\beta} = 9.51$ ($p = 0.168$) for OLIVE-E and $\hat{\beta} = 11.50$ ($p = 0.122$) for OLIVE-IE; these effects are directionally consistent but strongly attenuated by the switch-timing covariate ($\hat{\beta} = -0.228$, $p < 0.001$), confirming that normalization by post-switch window duration is essential. Oracle remains significant in the LME ($\hat{\beta} = 23.30$, $p = 0.007$).

K US2 Adaptive Difficulty Trajectories by Condition

Figure 10 shows mean difficulty level per block for each condition. All conditions start near difficulty 3–4 and converge to difficulty 5 by mid-session, confirming that the stratified assignment (based on US1 max-difficulty) successfully balanced ability across conditions.

Pairwise Welch t-tests on per-participant maximum difficulty reached show no significant overall difference (one-way ANOVA: $F = 1.50$, $p = 0.239$). However, OLIVE-E and Oracle more reliably saturated the difficulty ceiling: both conditions had every participant reach difficulty 6 (mean 6.00 ± 0.00), compared to Control (mean 5.38 ± 0.86; $t = 1.93$, $p = 0.095$, marginal). This is not a confound but the expected consequence of agent assistance: participants who receive reliable target guidance maintain high enough performance to keep the adaptive controller escalating. OLIVE-IE showed no significant ceiling advantage over Control ($p = 0.790$), consistent with its within-session improvement story: the EEG channel's benefit manifests progressively over blocks rather than immediately saturating the ceiling.

L US2 Subjective Trust and Reliance Ratings

After each SpaceShooter block, participants rated three items on a 1–7 scale: *how much did you trust the agent* (`q_trust`), *how much did you rely on it to look at targets* (`q_rely_look`), and *how much did you rely on it when shooting* (`q_rely_shoot`). Table 8 reports per-condition means averaged across blocks and participants.

All three measures increase monotonically from Control to OLIVE-E to OLIVE-IE to Oracle. Oracle reaches statistical separation from Control on agent trust ($t = 2.69$, $p = 0.043$) and gaze reliance

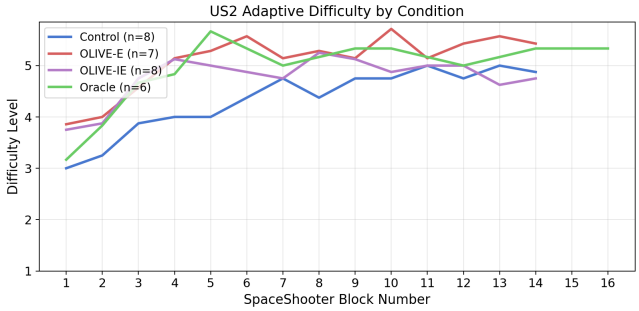


Figure 10: US2 mean adaptive difficulty by condition (grand average per condition; stratified assignment ensures comparable starting levels). All conditions converge to difficulty 5 by block 6; OLIVE-E and Oracle sustain slightly higher difficulty in late blocks.

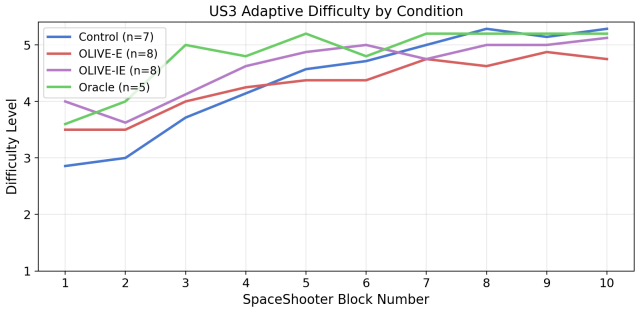


Figure 11: US3 mean adaptive difficulty by condition. All conditions converge to difficulty 4–5 by block 4, confirming that between-condition throughput differences in H3.1–H3.2 are not confounded by systematically different task demands at the time of the silent switch.

Table 8: US2 post-block trust and reliance ratings (1–7 scale, mean).

Condition	n	Trust	Rely (look)	Rely (shoot)
OLIVE-E	8	3.93	3.19	4.54
OLIVE-IE	8	3.70	3.57	3.97
Oracle	6	**4.83**	**4.67**	**5.04**

($t = 2.97$, $p = 0.039$), consistent with its performance ceiling. OLIVE-IE's higher gaze-reliance score relative to OLIVE-E (3.57 vs. 3.19) suggests that EEG-informed guidance produces more directionally specific cues that operators find worth following with their eyes, a soft corroboration of the throughput advantage reported in Section 5.

M US3 Adaptive Difficulty Trajectories by Condition

Figure 11 shows mean difficulty level per block for each condition. All four conditions follow similar trajectories, reaching difficulty 4–5 by mid-session and sustaining it through the nine data-collection blocks.

A one-way ANOVA on per-participant maximum difficulty reached confirms no significant between-condition differences ($F = 0.20$, $p = 0.896$), and no pairwise comparison reaches even marginal significance (all $p > 0.40$, $|\Delta| \leq 0.38$). This is a strong null result: the difficulty confound concern can be dismissed entirely for US3. All conditions faced comparably challenging enemy compositions when the silent switches occurred, so the H3.1 and H3.2 throughput differences reflect genuine adaptation performance, not task demand asymmetries.

N US3 Subjective Measures

Wingman adaptation speed. OLIVE-E and OLIVE-IE participants rated how quickly their wingman adapted to the silent switch on a 1–7 scale (post-block item; Oracle excluded as its adaptation was automatic). IE participants rated wingman adaptation speed notably higher than E (OLIVE-IE: $M = 5.1$ vs. OLIVE-E: $M = 4.4$), consistent with the faster objective reconvergence (H3.1).

NASA-TLX.. Figure 12 shows the post-experiment NASA-TLX profile by condition ($n = 7$ C, 8 E, 8 IE, 5 O; Welch t-tests). Two dimensions show statistically notable differences. **Temporal demand** was significantly higher in both OLIVE-E ($M = 5.2$) and Oracle ($M = 5.33$) than in Control ($M = 2.8$; E vs. C: $t = 2.56$, $p = 0.035$; Oracle vs. C: $t = 2.55$, $p = 0.046$), while OLIVE-IE ($M = 4.0$) fell intermediate, suggesting that EEG augmentation partially relieves the temporal pressure of monitoring for the switch, as the wingman re-orients faster and reduces the window during which the operator must bear that burden alone. **Self-rated performance** was significantly higher for OLIVE-IE ($M = 5.75$) than Control ($M = 4.0$; $t = 2.57$, $p = 0.048$), consistent with the within-session readaptation improvement (H3.2). Frustration was low and comparable across C, E, and IE ($M = 3.2$–3.5), but notably elevated for Oracle ($M = 5.33$; Oracle vs. C: $t = 2.02$, $p = 0.118$), suggesting that automatic adaptation without operator agency may be perceived as disorienting rather than helpful.

O Simulation-based error bound for the 16-block QUEST+-style calibration

A potential concern is that the individualized difficulty calibration in Study 1 uses only 16 SpaceShooter blocks in total (2 fixed practice blocks followed by 14 adaptive blocks). To quantify how accurately this procedure can recover a participant-specific operating point, we conducted a Monte Carlo simulation using the *same* QUEST+-style Bayesian difficulty selection rule as in our implementation.

Generative model. We simulated $N = 2500$ virtual participants with latent ability $\theta \sim \mathcal{U}(1, 6)$ and slope $\beta \in \{0.4, 0.6, 0.8, 1.0, 1.2\}$ (uniformly sampled). For a given difficulty level d, the expected block score (0–100) was

$$\mathbb{E}[\text{Score}(d)] \;=\; 100 \cdot \sigma\!\left(\frac{\theta - d}{\beta}\right), \tag{12}$$

where $\sigma(\cdot)$ is the logistic sigmoid. We then added logistic noise on the score axis (scale 14), clamped the score to $[0, 100]$, and discretized it into 5 outcome categories: $[0, 20]$, $(20, 40]$, $(40, 60]$, $(60, 80]$, $(80, 100]$.

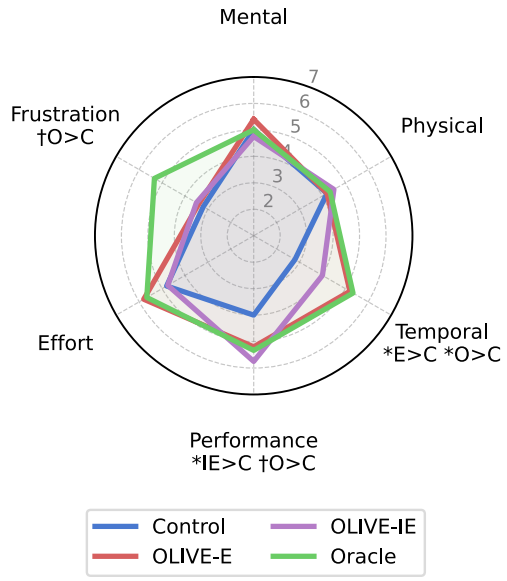


Figure 12: NASA-TLX profile by condition (post-experiment; 1–7 scale). Axis labels annotated with pairwise significance: $\dagger p < 0.10$, $* p < 0.05$.

Calibration procedure and target difficulty. For each simulated participant, we ran exactly 16 blocks: two fixed practice blocks at d=1 and d=2, followed by 14 adaptive blocks over an internal 0.5-step grid $d \in \{1.0, 1.5, \ldots, 6.0\}$. The adaptive policy chose the next difficulty to maximize expected information gain (equivalently, minimize expected posterior entropy) over the latent parameters. We define the participant's *target difficulty* $d^\star$ as the difficulty at which the expected score equals 70 (our "challenging-but-playable" operating point), i.e.,

$$d^\star \triangleq \{d : \mathbb{E}[\text{Score}(d)] = 70\} = \theta - \beta \cdot \text{logit}(0.70). \quad (13)$$

After block 16, we estimated $\hat{d}$ as the posterior mean of $d^\star$.

Results. Figure 13 shows the cumulative distribution of absolute estimation error $|\hat{d} - d^\star|$. Across simulations, the median absolute error was 0.20 difficulty levels (75th percentile 0.34, 90th percentile 0.50), with 90.0% of participants within ±0.5 difficulty and 99.5% within ±1.0. Figure 14 plots $\hat{d}$ against the ground-truth $d^\star$, showing close agreement and negligible bias (mean signed error 0.006). Together, these results indicate that, under a plausible score-generation model, the proposed 16-block calibration is sufficient to place most participants within roughly half a difficulty level of their individualized operating point, which is the intended purpose of Study 1 (initializing Study 2 at a participant-appropriate difficulty band).

P Autonomous Scene Camera: Kinematics and Coverage Algorithm

OLIVE's visual evidence stream includes crops from an *autonomous scene camera* ("agent camera") that orbits the user's position and steers toward under-observed regions of the upper hemisphere,

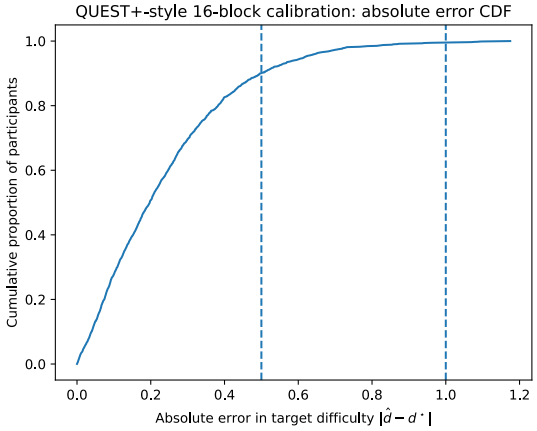


Figure 13: CDF of absolute error $|\hat{d} - d^\star|$ for the 16-block QUEST+-style calibration. Dashed lines mark ±0.5 and ±1.0 difficulty levels.

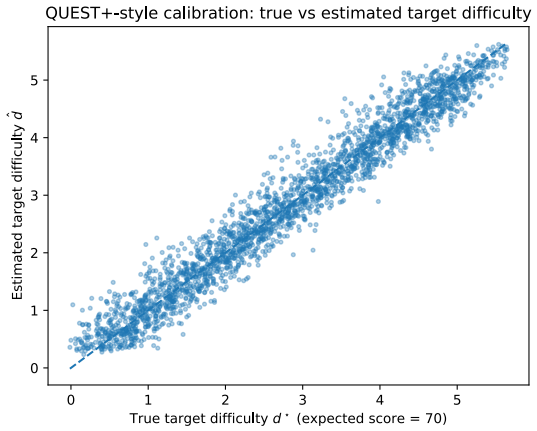


Figure 14: True vs. estimated target difficulty ($d^\star$ vs. $\hat{d}$) across simulated participants. The dashed line indicates y=x.

providing visual coverage of ships outside the participant's current gaze direction.

Coverage grid. The upper hemisphere is tessellated into $N = 512$ cells using a quasi-uniform grid (azimuth × elevation, 32 × 16 cells). Each cell tracks its *last-observed timestamp*; when either the user's gaze-aligned camera or the wingman camera renders a ship in a cell, the cell's timestamp is refreshed. Cells with elapsed time > `memoryDuration` = 3 s are considered stale.

Steering policy. At each physics update, the controller identifies the stale cell with the maximum angular distance from the current camera direction (breaking ties by largest elapsed time) and sets it as the steering target. Camera position is updated via position interpolation toward the target cell's representative point at $v_{\max} = 1.2$ m/s, and orientation via Slerp toward the target direction at $\omega_{\max} = 30$ deg/s.

Pitch oscillation. To sweep both low- and high-elevation sectors, the wingman camera's pitch oscillates between 0 deg and $\theta_{\max} = 90$ deg at a frequency of $f = 0.3$ Hz, driven by a sinusoidal modulation superimposed on the steering direction. This ensures that even when the steering target is at a fixed azimuth, the camera periodically covers the full elevation range.

Rendering and cropping. The wingman camera renders at 512 × 512 pixels. Per-ship crops are extracted at each frame by projecting the ship's bounding box into the camera's image plane; crops are padded to a fixed 224 × 224 window and passed to the OLIVE visual encoder (frozen ViT-B/16 backbone).

Q SpaceShooter Block Score

The block score $B \in [0, 100]$ is a weighted geometric composite of two normalized rates: enemy clearance (s_E) and friendly safety (s_F):

$$B = 100 \cdot s_E^{\alpha} \cdot s_F^{\beta}, \quad \alpha = 1.2,\ \beta = 2.0, \tag{14}$$

where

$$s_E = \text{enemies shot/enemies spawned},$$
$$s_F = 1 - \text{friendlies shot/friendlies spawned},$$

both clamped to $[0, 1]$. The higher exponent on s_F ($\beta = 2.0 > \alpha = 1.2$) makes friendly fire disproportionately costly: a single unnecessary shot deflates the score more than a missed enemy.

US2 (TimeScaledRespawn). Because US2 blocks run for 120 s versus the 90 s US1 reference, raw shot counts are scaled by $90/120 = 0.75$ before computing s_E and s_F, normalizing engagement opportunity across block lengths.

US3 (mid-block target switch). The block is split into two phases at the switch time $\tau \in \{75, 90, 105\}$ s. Each phase produces its own (s_E, s_F) pair (with phase-specific time-scaling to the 90 s reference), and the two phases are merged via a duration-weighted geometric combination:

$$S_E = s_{E,\text{pre}}^{w} \cdot s_{E,\text{post}}^{1-w}, \quad S_F = s_{F,\text{pre}}^{w} \cdot s_{F,\text{post}}^{1-w}, \quad w = \tau/T, \tag{15}$$

where T is the total block duration. The final score is $B = 100 \cdot S_E^{\alpha} \cdot S_F^{\beta}$. This formulation ensures that neither phase dominates the score when switch timing is unbalanced, and that friendly-fire penalties are propagated within each phase independently.

R US1 Task Characterization Details

R.1 Objective Performance by Difficulty

Table 9 summarizes SpaceShooter performance metrics by adaptive difficulty level ($N = 57$ participants). Hit rate remains stable across difficulty levels (~0.67–0.71), indicating that participants maintain shot accuracy even as the scene becomes more complex. Mothership health decreases monotonically from 85.7 at difficulty 1 to 52.6 at difficulty 5 ($r = -0.90$), reflecting the increased number of enemies that leak past the user. Target kill rate drops from 0.73–0.75 at levels 1–3 to 0.61 at level 5, as the higher target count (~24 at level 4–5 vs. 9 at level 1) outpaces the user's engagement capacity.

Table 9: SpaceShooter objective performance by adaptive difficulty level. Hit rate = fraction of shots on enemy targets; FF rate = fraction on friendly ships; target kill rate = fraction of true targets destroyed; MS health = mothership HP remaining (0–100).

Diff.	Hit Rate	FF Rate	Tgt Kill	MS Health
1	0.67 ± 0.21	0.33 ± 0.21	0.73 ± 0.19	85.7 ± 3.0
2	0.67 ± 0.14	0.33 ± 0.14	0.74 ± 0.17	77.9 ± 3.9
3	0.71 ± 0.11	0.29 ± 0.11	0.75 ± 0.13	69.9 ± 4.4
4	0.71 ± 0.15	0.29 ± 0.15	0.69 ± 0.17	60.4 ± 5.5
5	0.71 ± 0.11	0.29 ± 0.11	0.61 ± 0.11	52.6 ± 2.6

R.2 Adaptive Difficulty Trajectories

All participants begin at difficulty 1 and the system ramps adaptively based on performance. Individual trajectories show substantial variation: high-performing participants (P13, P25) reach difficulty 5 by blocks 12 and 14 respectively, while others (P8, P12) plateau at difficulty 2. The median maximum difficulty reached is 3 (IQR: 3–4). Most participants stabilize within 4–6 blocks, after which difficulty fluctuates by ±1 level. The non-monotonic trajectories (e.g., P18 peaks at difficulty 4 then regresses to 2) confirm that overwhelmed periods occur and the system responds by reducing load. Figure 15 shows individual trajectories alongside the grand average.

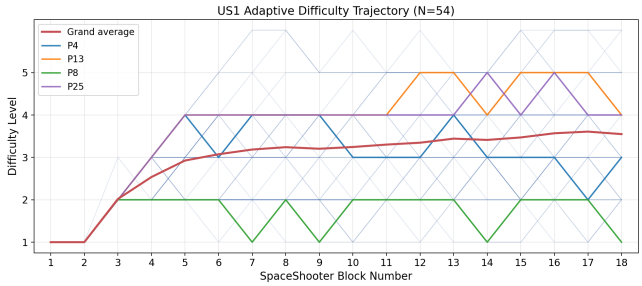


Figure 15: Adaptive difficulty trajectories per participant ($N = 54$). Grand average in red; four representative participants highlighted. High performers (P13, P25) reach difficulty 5; P8 plateaus at 2. Non-monotonic trajectories confirm the adaptive controller steps down under overload.

R.3 Subjective Ratings by Difficulty

Table 10 shows self-reported workload, confidence, and overwhelm by difficulty level. Overwhelm increases with difficulty ($r = 0.15$), consistent with the task's design intent. Confidence also increases ($r = 0.24$), reflecting a selection effect: participants who reach higher difficulty levels tend to be more skilled rather than difficulty causing higher confidence. Mothership health correlates with overwhelm ($r = -0.28$) and workload ($r = -0.21$), confirming that subjective experience tracks objective performance. Hit rate correlates with confidence ($r = 0.34$), indicating that shot accuracy is a salient cue for self-assessed performance. Figure 16 shows these relationships.

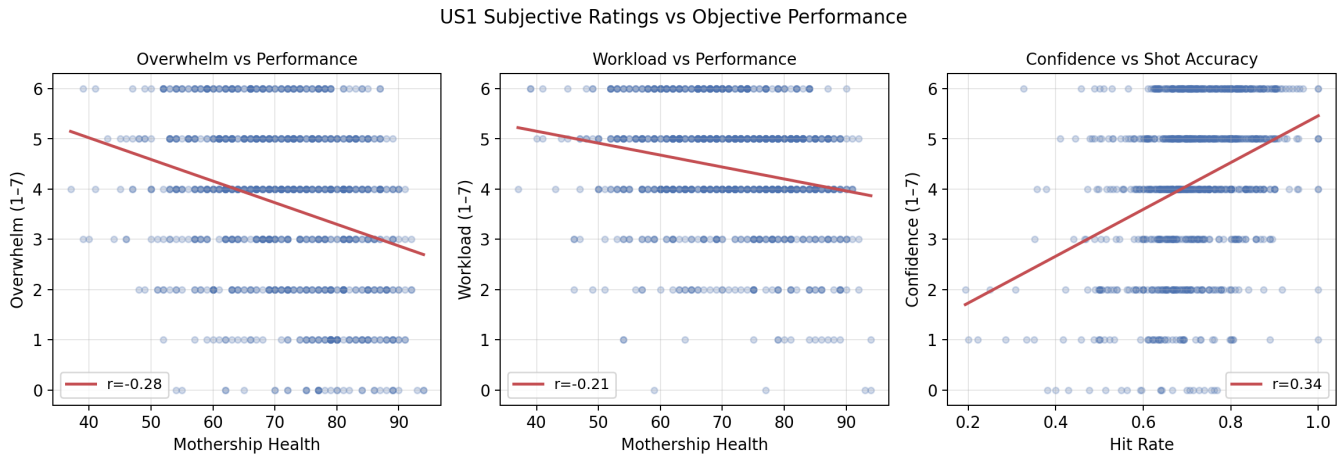


Figure 16: Subjective ratings vs. objective performance in US1 ($N = 54$, 952 blocks). Overwhelm and workload decrease with mothership health ($r = -0.28$, $r = -0.21$); confidence increases with hit rate ($r = 0.34$). Each dot is one block.

Table 10: Subjective ratings (1–7 Likert scale) by adaptive difficulty level.

Diff.	Workload	Confidence	Overwhelm
1	3.97 ± 1.45	3.20 ± 1.56	3.05 ± 1.88
2	4.36 ± 1.12	3.71 ± 1.65	3.11 ± 2.03
3	4.61 ± 1.46	4.24 ± 1.61	3.64 ± 1.51
4	3.87 ± 1.69	4.25 ± 1.37	3.72 ± 1.89
5	4.14 ± 2.19	4.86 ± 1.07	4.29 ± 2.29

S Physiological Decoder: Training, Signal Characterization, and Design Decisions

OLIVE requires, as implicit evidence, a per-fixation soft label $e_i \in (0, 1)$: a calibrated probability that the currently fixated item is a target, derived from the neural response at fixation onset. This value is produced by a *physiological decoder*, a model that maps fixation-locked EEG epochs to calibrated target-presence probabilities. OLIVE is decoder-agnostic by design (§3.4): any module that returns a calibrated scalar per fixation can serve as the implicit evidence source. This appendix documents the training pipeline and design decisions for the decoder used in our experiments.

Decoder architecture note. The physio decoder used in this work was developed as part of a collaborative research project whose full architecture is being prepared for separate publication; its implementation details are therefore beyond the scope of this work. What we describe below: calibration data format, preprocessing, feature extraction, and training protocol, is fully sufficient to replicate the training pipeline with any binary classifier that accepts the described feature vectors.

S.1 Calibration Data: The Visual Search Phase

Before each experimental session, participants complete a *visual search calibration phase* in which they view static arrays of targets and distractors and count the number of targets within a fixed time budget. No per-item responses are collected during viewing; participants only report the count afterward. This means the full gaze record during viewing is available as free-viewing fixation data, providing naturally labeled fixation epochs without disrupting the observation behavior the decoder is designed to characterize.

For each fixation on an item, we extract a *fixation-related potential* (FRP): an EEG epoch time-locked to fixation onset, spanning $[-200, 800]$ ms, with a $[-200, 0]$ ms pre-fixation baseline. The label is the ground-truth identity of the fixated item (target = 1, non-target = 0). Fixations shorter than 100 ms are excluded. This yields a set of labeled (epoch, label) pairs that constitute the per-participant calibration dataset.

S.2 EEG Preprocessing

Raw EEG is recorded at 256 Hz from 20 electrodes (B-Alert X24, standard 10–20 layout) and synchronized with gaze events via Lab Streaming Layer (LSL) with < 5 ms skew. The continuous signal is bandpass-filtered (0.1–40 Hz, zero-phase FIR) and notch-filtered at 60 Hz (harmonics at 120 Hz), then re-referenced to the common average. Independent component analysis (ICA) is applied to attenuate ocular artifacts, with saccade onset times used as informed priors for artifact component identification. Epochs are baseline-corrected to $[-200, 0]$ ms. Trials are rejected if peak-to-peak amplitude exceeds 150 μV in any channel, or if the gaze sample at fixation onset falls off the item.

S.3 Feature Extraction

The primary discriminative signals are the N2pc and P300 fixation-related components (see §2). Features are extracted from posterior and parietal electrodes (Pz, P3, P4, POz, O1, O2, Oz):

- **N2pc window (180–300 ms):** lateralization index computed as the mean amplitude difference between ipsilateral and contralateral electrode pairs relative to the stimulus position of the fixated item; this indexes covert attentional selection.
- **P300 window (300–600 ms):** mean amplitude at midline parietal and occipital sites (Pz, Oz); this indexes target-category confirmation and context updating.

These windows were chosen based on the FRP literature [11, 12, 14, 50] and confirmed by the grand-average waveforms in Figure 17.

S.4 Decoder Training

The physio decoder is trained per participant on the calibration dataset. It accepts the feature vector described above and returns a calibrated probability $e_i \in (0, 1)$ that the fixated item is a target. Because the specific decoder architecture is proprietary, only the inputs, outputs, and training protocol are described here.

Training uses the labeled calibration epochs as a supervised binary classification problem. Calibration of the output probability is verified by checking that the mean predicted score across held-out non-target fixations matches the empirical non-target base rate; if necessary, Platt scaling is applied to recalibrate the output. The resulting e_i values serve as soft labels for OLIVE's EM updates.

A quality score $w_i \in [0, 1]$ accompanies each prediction, derived from a measure of decoder confidence on individual epochs (e.g., margin from the decision boundary or entropy of the output distribution). High-quality fixations (those where the decoder is confident and the epoch is artifact-free) receive higher w_i, down-weighting noisy or ambiguous fixation events in the OLIVE EM update.

S.5 Why Pupil Was Excluded

Task-evoked pupil responses (TEPR) are a natural complement to EEG: non-invasive, easy to record, and providing an independent index of cognitive effort and uncertainty [4, 65]. We initially planned to include TEPR as a third evidence stream in OLIVE.

In practice, the pupil signal proved too contaminated by luminance-driven responses in SpaceShooter to be useful. The task involves rapid, large luminance transients (explosions, beam effects, and animated backgrounds) that produce strong light-reflex-driven pupil constriction and dilation unrelated to cognitive evaluation of individual items. Disentangling TEPR from luminance responses requires trial-by-trial luminance control or matched stimuli, which conflicts with the ecological validity of the game environment. The coarser 500–2000 ms time window of TEPR also makes it difficult to attribute responses to specific fixations in fast-paced multi-target scenes where fixations are short and many items are processed in rapid succession.

We therefore excluded pupil from the current implementation and rely solely on fixation-locked EEG as the implicit channel. This is a practical scope decision, not a claim that TEPR is uninformative: in environments with controlled luminance profiles (e.g., clinical imaging workstations, document review interfaces), TEPR could be incorporated as an additional reliability-weighted annotator in OLIVE's EM framework with no change to the core algorithm.

S.6 FRP Signal Characterization

Figure 17 shows the grand-average FRP waveforms locked to first fixation onset, averaged across all participants ($N = 57$) and split by item category (target vs. non-target). Waveforms are plotted across all 20 EEG channels in topomap layout over the epoch window $[-200, 800]$ ms, with a $[-200, 0]$ ms pre-fixation baseline.

A clear P300 deflection (300–600 ms) is visible at Pz and occipital sites for target items relative to non-targets, consistent with prior fixation-related EEG studies [11, 12, 25, 59]. The orange shading marks time points with significant target vs. non-target differences (uncorrected Welch's t-test, $p < 0.05$); the bottom strip shows Cohen's d. Peak discrimination occurs at posterior sites (Pz, P3, P4, POz, O1, O2), matching the expected P300 topography. These waveforms confirm that the calibration phase captures robust, spatiotemporally consistent neural discrimination between target and non-target fixations, providing the signal that the physio decoder is trained to classify and that OLIVE uses as implicit evidence.

S.7 Topomap Timeline of the Fixation-Related P300

Figure 18 shows the spatial distribution of the target–distractor ERP difference at successive 100 ms windows from fixation onset to 800 ms. The posterior positivity emerging at 300–500 ms at parietal and parieto-occipital sites (Pz, POz, P3, P4) is consistent with the P300 target-detection response [50]: a late, broadly distributed positivity that reflects category confirmation when the fixated item matches the current target class. The early (0–200 ms) windows show minimal difference, as expected before attentional selection and category evaluation occur. The sustained positivity at POz through 600 ms is particularly notable: it aligns with the channel importance ranking (Figure 20), where POz ranks as the single most discriminative electrode for the physio decoder, lending neural validity to the algorithmic importance estimate.

S.8 Error-Related Negativity During Friendly-Fire Events

SpaceShooter generates a naturalistic error paradigm: every friendly-fire shot constitutes a response error, an action directed at the wrong target class, compounded by an explicit score penalty, making these events both objectively incorrect and subjectively salient. We extracted response-locked EEG epochs from continuous recordings ($N = 57$ US1 participants with available session data) centred on each shot event, labelled as errors (DTN= 1, friendly fire) or correct responses (DTN= 2, enemy destroyed), using the shot timestamps recorded in the same LSL stream as the EEG. Preprocessing: continuous 0.5–30 Hz bandpass filter applied before epoching, then epoch extraction ($[-200, +600]$ ms), baseline correction ($[-200, 0]$ ms), amplitude rejection ($> 150\,\mu V$ per channel), and per-participant FastICA using Fp1/Fp2 as ocular proxy channels.

Figure 19 shows the grand-average ERN and CRN waveforms at frontocentral sites (Fz+Cz mean; the B-Alert X24 lacks FCz) and the difference topomaps at the ERN peak and Pe window. A frontocentral positivity for error relative to correct trials is visible in the 40–140 ms window, though the effect does not reach conventional significance ($M = +1.05\,\mu V$, $SE = 0.60$, $t(37) = 1.71$, $p = 0.096$; cluster permutation: no significant cluster). The modest amplitude and marginal significance are consistent with the ecological validity of the paradigm: friendly-fire events in SpaceShooter are heterogeneous (some intentional, some accidental), the B-Alert headset records under active VR conditions with elevated muscle and motion artifact, and the ERN is known to be attenuated for high-workload, time-pressured tasks where error monitoring competes with ongoing perceptual demands [22]. Nonetheless, the scalp topography at the ERN peak (frontocentral) and Pe window

(centroparietal) is consistent with canonical ERN/Pe distributions, suggesting that the error-monitoring signal is present even if it does not reach significance in this sample.

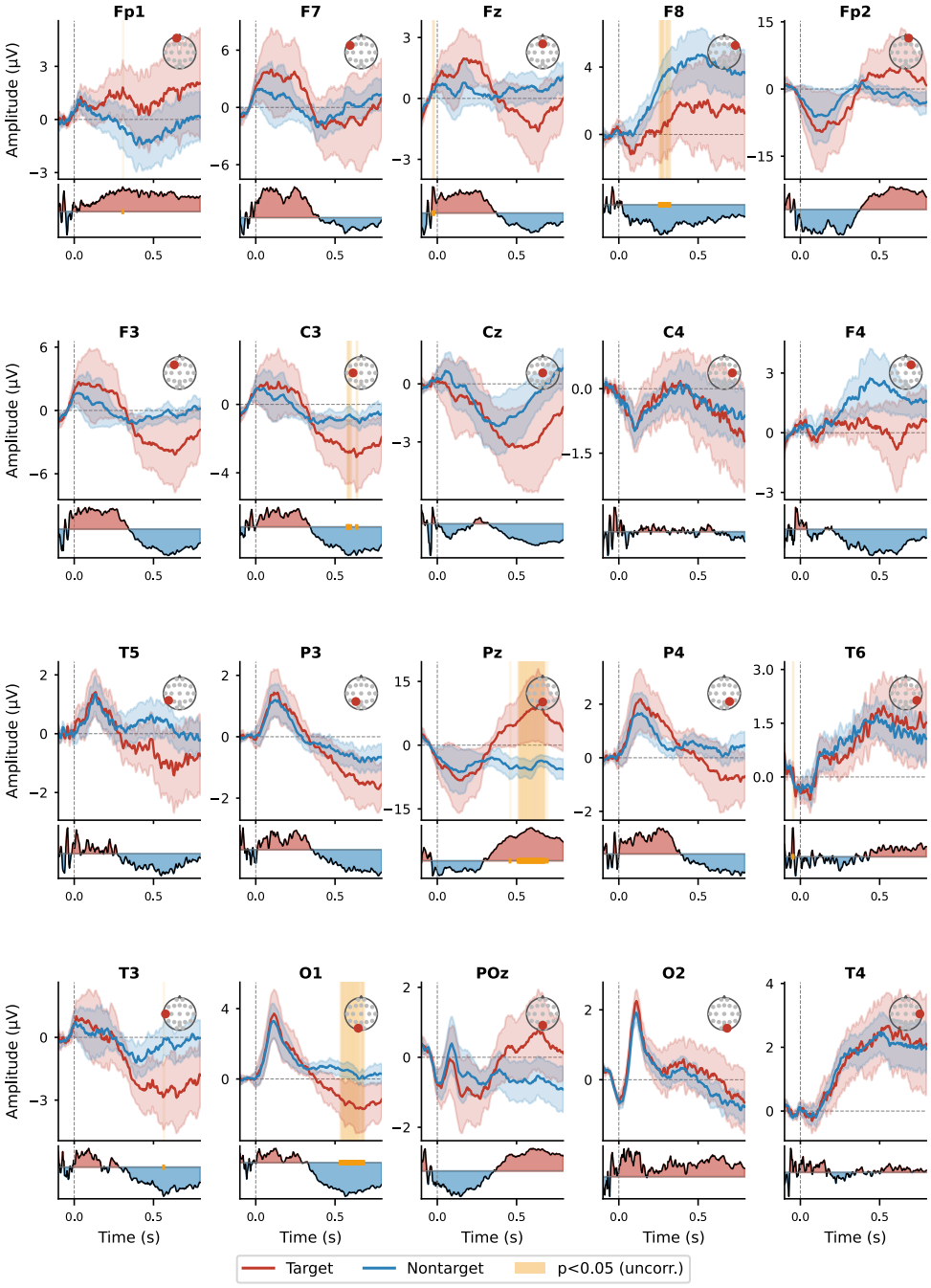


Figure 17: Grand-average fixation-related potentials (FRPs) for target and non-target items across all 20 EEG channels in topomap layout ($N = 57$ participants; shading = ±1 SEM; orange shading = $p < 0.05$ uncorrected Welch's t-test; bottom strip = Cohen's d). The posterior P300 (300–600 ms) is the primary discriminative component exploited by the physio decoder.

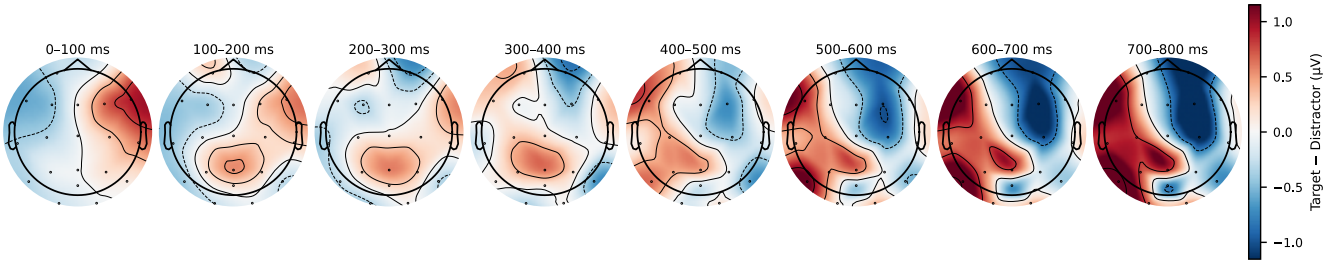


Figure 18: Grand-average target–distractor ERP difference topomaps across US1, US2, and US3 SpaceShooter sessions ($N = 57$ participants; 2,060 target and 3,946 distractor fixation epochs after preprocessing: amplitude rejection $> 100\,\mu$V per channel, per-participant FastICA with Fp1/Fp2 ocular proxies; baseline $[-100, 0]$ ms). Each panel shows the scalp distribution of the grand-average amplitude difference (target − distractor, µV) averaged within the indicated 100 ms window. The posterior positivity emerging at 300–500 ms (Pz, POz) is consistent with the P300 orienting response, validating that the decoder captures genuine neural evidence of target detection. The laterization is likely from the users taking aim using the right-hand right after fixation.

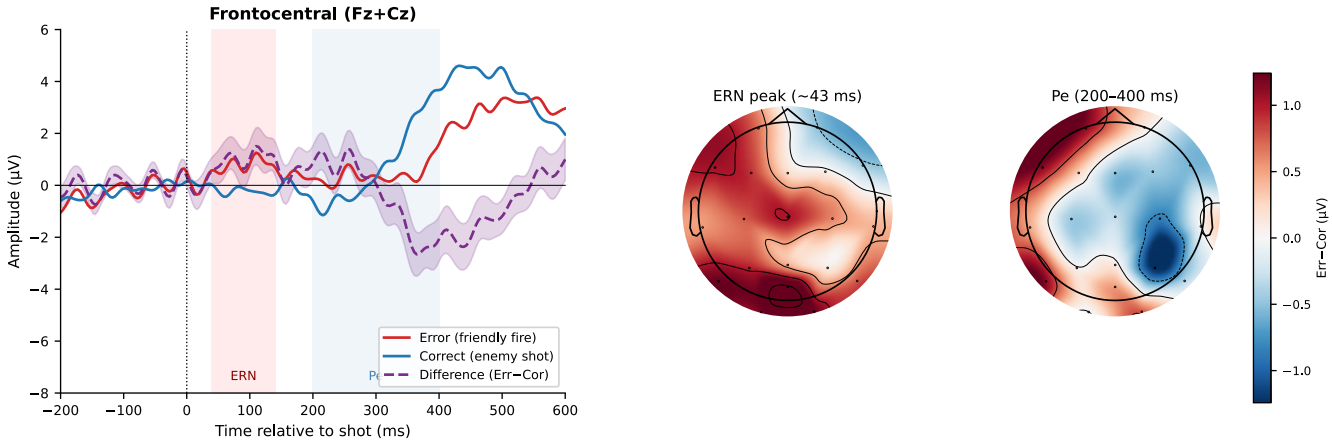


Figure 19: Error-Related Negativity (ERN) from SpaceShooter friendly-fire events. Left: Grand-average frontocentral waveforms (Fz+Cz) for error (red) and correct (blue) trials; dashed purple = difference wave (Err−Cor), shading = ±1 SEM across participants. Center/right: Scalp topomaps of the error−correct difference at the ERN peak and Pe window. ($N = 57$ participants; preprocessing: 0.5–30 Hz continuous bandpass, $[-200, 0]$ ms baseline, amplitude rejection $> 150\,\mu$V, FastICA; $p = 0.096$, one-sample t-test on frontocentral ERN amplitude.)

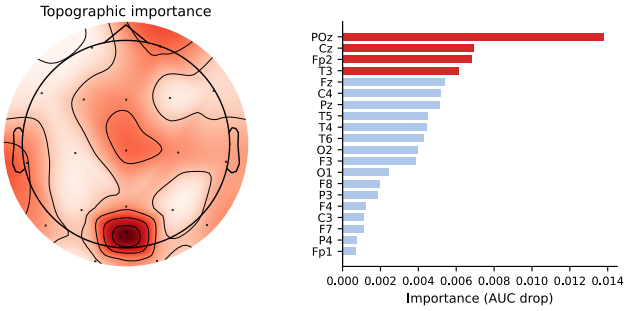


Figure 20: Group-average EEG channel importance from permutation ablation (N=57 participants; 459 ± 174 fixation epochs per participant, val-set runs 33–50). *Left:* topographic importance map. *Right:* channels ranked by mean AUC drop when zeroed; top-4 (POz, Cz, Fp2, T3) are highlighted in red. Drops are modest due to the joint multi-channel tokenizer architecture.

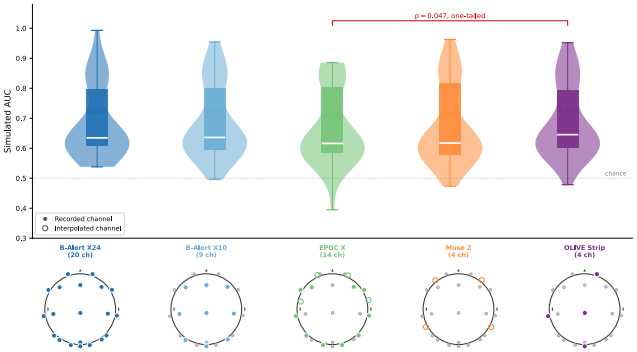


Figure 21: Simulated decoder AUC by portable EEG form factor (N=57). Violins show participant distribution; boxes show median and IQR; means annotated numerically. Topomap insets: filled circles = recorded channels; open circles = interpolated via spherical spline. The red bracket indicates the significant OLIVE Strip vs. Emotiv EPOC X comparison ($t(40) = 1.72$, $p = 0.047$, one-tailed; pre-specified directional hypothesis).

S.9 EEG Channel Importance and Portable Form-Factor Simulation

Channel importance via permutation ablation. To identify which of the 20 B-Alert X24 electrodes contribute most to the physiological decoder's target-discrimination performance, we conducted a permutation ablation analysis on all N=57 US1 participants using the val-run data (runs 33–50 per participant). For each of the 20 channels, we zeroed out that channel's signal across all fixation epochs and re-computed the model's AUC; importance is defined as $\max(0, \text{AUC}_{\text{baseline}} - \text{AUC}_{\text{channel zeroed}})$. The decoder architecture uses a `TemporalEmbedChannelConv` tokenizer that jointly encodes all channels via a 2-D convolution before attention-based temporal pooling, meaning no single channel has an independent representational role. Consequently, individual ablation drops are modest (group average peak drop < 0.02 AUC), yet consistent across participants. Figure 20 shows the group-average importance map and channel ranking. The four most important channels are **POz, Cz, Fp2, and T3** (T7 in standard 10-20 notation): a midline centro-parietal cluster (POz, Cz) and two frontal-pole/temporal sites (Fp2, T3), consistent with the posterior P300 and frontal N2pc topographies that drive FRP discrimination (§S).

Portable form-factor simulation. We simulated decoder performance under four commercially available EEG form factors: the ABM B-Alert X10 (9-channel subset of X24), Emotiv EPOC X (14 channels), Muse 2 (4 channels), and an OLIVE-optimized strip (the 4 most important channels identified above: POz, Cz, Fp2, T3). For each device, a two-pass MNE spherical spline interpolation [48] reconstructed the full 20-channel B-Alert X24 montage: channels present in both the device and B-Alert X24 were taken directly from the recording, while channels exclusive to the device (Emotiv EPOC X: AF3, AF4, FC5, FC6; Muse 2: all 4 channels) were first estimated from B-Alert data via a forward pass, then the full B-Alert montage was reconstructed from the combined device set via a backward pass. The reconstructed signals were passed through the per-participant decoder and the resulting soft labels were used to refit beta distributions and compute the simulated OLIVE AUC, targeting the same +10% boost above the empirical val-set AUC.

Results and comparison. Figure 21 shows the simulated AUC distribution across all 41 participants for each form factor. Using one-tailed paired t-tests (directional hypotheses pre-specified from channel importance), the OLIVE strip significantly outperforms Emotiv EPOC X ($t(40) = 1.72$, $p = 0.047$, mean difference +0.02 AUC). No other pairwise comparison reached significance ($p > 0.05$ two-tailed for all remaining pairs; $n = 41$). Group means ranged from 0.664 (EPOC X) to 0.685 (OLIVE strip), with the full 20-channel headset at 0.682. A counterintuitive finding is that Muse 2 (0.670; 4 interpolated channels) performed comparably to EPOC X (0.664; 14 channels, 10 recorded). This follows directly from the channel importance analysis: EPOC X covers only one of the four most important channels (T7/T3), with the remaining 13 channels spanning temporal, frontal, and occipital regions that carry lower discriminative weight for this decoder. By contrast, the OLIVE strip covers all four top channels and achieves the highest simulated AUC despite having the same electrode count as Muse 2, illustrating that *channel placement relative to the model learned feature map* matters more than electrode count per se.

Limitations. These simulations have two important caveats. First, the reported AUC values are *simulated priors for OLIVE's Bayesian update*, not measured decoder performance on an actual portable device. They reflect a beta-distribution model fitted to val-set decoder outputs, boosted to approximate a plausible performance improvement; the true AUC of a decoder retrained or adapted from a specific device's native recording could differ substantially. Second, the spherical spline reconstruction is most reliable when the interpolated channels are geometrically surrounded by recorded channels; for Muse 2, where all four channels fall outside the B-Alert X24 layout, the backward reconstruction of midline and parietal

channels relies entirely on extrapolation, introducing approximation error that is not present in a real headset recording. These results should therefore be interpreted as a simulation-based plausibility check for OLIVE's prior construction rather than an empirical evaluation of portable EEG hardware.

T Agent Cue Policy: Algorithmic Details

OLIVE's per-item posterior scores are converted to guidance cues via a **quantile suppression policy**. Raw scores are first smoothed with an exponential moving average ($\alpha = 0.25$) to suppress transient noise. The system then computes P50 and P90 of all smoothed scores across active ships. If the spread $P90 - P50 < 0.01$ (beliefs insufficiently differentiated from the uniform prior), all cues are suppressed entirely: an uncertain model actively misleads rather than helps. Otherwise:

- Items above P50 receive a *soft cue* with opacity proportional to $(s - P_{50})/(P_{90} - P_{50})$.
- The top three items above P90 receive a *hard cue* (full-opacity highlight).

Three modalities render the two tiers:

- **Out-of-view directional line** (inspired by [29, 30]; preliminary studies showed operators preferred lines over wedges in this fast-paced task): a viewport-edge line pointing toward off-screen targets, opacity scaled by soft-cue strength.
- **Outline highlight**: a colored border on in-view ships carrying a hard cue, providing visual pop-out without occluding scene content.
- **Aim assist**: when the user's crosshair is within ±20° of a highlighted ship, the border changes color; pressing the trigger at that moment engages the assist and destroys the target.

This policy requires no task-specific calibration: thresholds are computed relative to the current belief distribution and adapt automatically as OLIVE's posteriors evolve.

U System Implementation

Figure 22 shows the full runtime architecture. The system comprises four communicating processes: a **Unity frontend** running on a dedicated VR PC and three Python backend services — the **Vision Service**, the **OLIVE Service**, and the **Physio Service** — each hosted on a separate GPU server. All inter-process communication uses gRPC [26], enabling low-latency, language-agnostic remote procedure calls between Unity (C#) and the Python backends. EEG and eye-tracking streams are acquired using PhysioLabXR [72], a real-time multi-modal BCI platform that exposes LSL (Lab Streaming Layer) streams consumed by the Physio Service.

Unity frontend. The Unity application (C#) renders the XR environment, captures two camera streams at 1 FPS (448 × 448 pixels each) — a head-worn user camera and a scene-orbiting agent camera — and streams item-marker events via LSL. On each frame it calls the Vision Service via gRPC to retrieve current item beliefs and render guidance cues (directional lines, highlights, aim assist). When the player fires, Unity sends a *ShotEvent* RPC to the Vision Service carrying the shot item's identity, which is forwarded as an explicit positive label to the OLIVE Service.

Vision Service (bridge). The Vision Service is the architectural hub, implemented as an asyncio gRPC server listening on five ports (5550–5554). It performs object tracking: cameras stream raw frames, and the service extracts per-item bounding-box crops using rendering metadata provided by Unity. Crops are cached and quality-weighted ($\gamma_{ic} \in [0, 1]$, lower for blurry or partly occluded instances) before being dispatched to the OLIVE Service for inference and training. The *First-Fixation Watch* monitors the Unity.ReNa.ItemMarkers LSL stream (sampled at the LSL rate) to detect the first sufficiently long gaze on each item per block; on detection it issues a `PhysioService.OnFixate` RPC to the Physio Service and, on reply, forwards the returned soft EEG score to the OLIVE Service as implicit evidence. The *Prediction API* exposes current OLIVE beliefs to Unity via a polling endpoint used for cue rendering. At block start, the Vision Service initialises its local state (crop buffer, block index, difficulty level) and calls the OLIVE Service's reset endpoint to warm-start the virtual prompt and reliability weights from the previous block.

OLIVE Service. The OLIVE Service implements the virtual-prompt CLIP adaptation loop, listening on port 50051. It exposes four primary RPCs: `RunInference` (compute per-item cosine logits from cached crops), `TrainOnBatch` (one gradient step on the virtual prompt given soft labels), `SetHyperparams` (update learning rate, prompt length, loss type, and adapter mode at runtime), and `GetFeatures` (return frozen CLIP embeddings for external use). Training is triggered asynchronously by the Vision Service whenever a batch of labeled crops accumulates; inference runs on demand for cue rendering. The server uses a thread-pool executor with the model and AdamW optimizer resident on the GPU throughout the session, enabling sub-10 ms inference latency for typical block-level item counts (15–30 items).

Physio Service. The Physio Service handles all EEG processing, listening on port 50052. It subscribes to the EEG and eye-tracking LSL streams via PhysioLabXR's buffering utilities [72], applying an online preprocessing pipeline (bandpass filtering, baseline correction, and per-participant ICA artefact suppression). On each `OnFixate` call, it aligns a $[-200, +800]$ ms EEG epoch to the fixation onset, extracts the fixation-related potential (FRP) features used during Visual Search calibration, and passes them through the participant's pre-trained physiological decoder to return a soft target probability $\hat{y}^{\text{physio}} \in (0, 1)$ within a single RPC round-trip. The decoder weights are loaded at block initialisation via the `SetExperimentMeta` RPC and remain constant throughout the session.

V SpaceShooter Game Design

This appendix details the design of the XR SpaceShooter environment introduced in Section 3.2. Beyond OV, the environment is engineered in a way that preserves tight experiment control and ensures intuitive and smooth user interactions.

V.1 User and Environment Setup

Participants are positioned at a fixed egocentric viewpoint centered on a stationary mothership, which serves as the protected objective. Spaceships spawn within a fixed 3D box region surrounding the

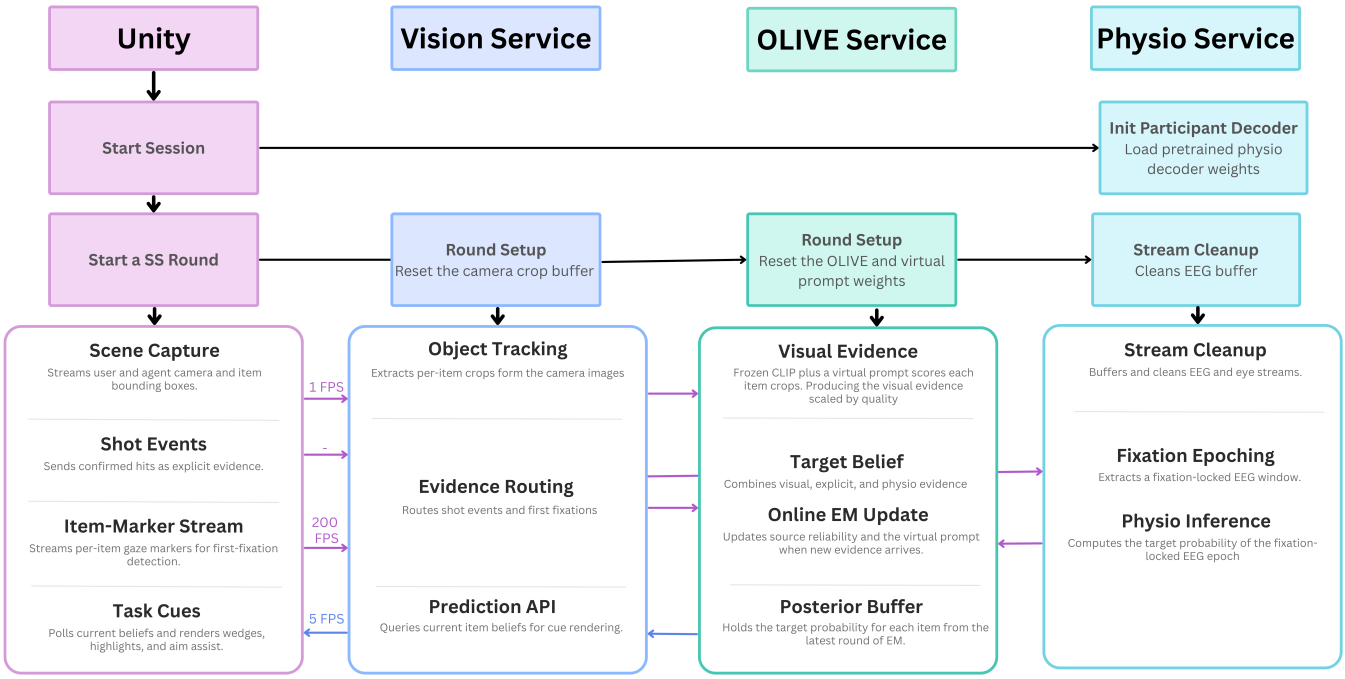


Figure 22: Runtime system architecture. Unity communicates with the Vision Service via gRPC. The Vision Service acts as a bridge, routing visual crops and shot events to the OLIVE Service and forwarding fixation events to the Physio Service. Initialization calls (dashed) set up per-participant state before each block.

mothership, with a minimum height constraint ensuring all ships remain visually separable from the mothership body.

Fixed viewpoint and bounded spawn volume. Users do not translate in space and interact via head orientation and shooting. Ships spawn within a bounded region rather than arbitrarily in space to ensure consistent density and visibility across blocks. The minimum height constraint prevents occlusion and guarantees that visual features remain observable. For the user, this creates a stable perceptual field where attention allocation—not navigation—determines performance. This aligns with OV tasks where operators monitor a fixed display under controlled spatial constraints.

Perceptual environment simplification. The skybox is set to a uniform blue with sparse clouds, and shadows on spaceships are disabled. These choices reduce background clutter and avoid contrast artifacts (e.g., dark undersides), ensuring that visual discrimination depends on object features rather than lighting conditions. This improves signal quality for both the user and the VLM, and ensures that difficulty arises from task structure rather than rendering noise.

V.2 Spaceship Dynamics

All spaceships (targets and non-targets) share identical motion dynamics, differing only in semantic role. Each ship follows a structured multi-phase trajectory:

(1) **Cruise phase:** ships move at lower speed toward a distant waypoint away from the mothership.
(2) **Attack phase:** ships transition into a faster "bombing" trajectory toward the mothership, descending as they approach and firing a laser.
(3) **Escape phase:** ships ascend and exit the engagement zone, after which they may re-enter cruising behavior.

Cruise vs. attack speed asymmetry. Cruising speed is intentionally lower than attack speed. This ensures that ships are harder to intercept during the high-urgency attack phase, forcing the user to anticipate targets earlier rather than reacting at the last moment. For the user, this creates a temporal trade-off: acting early with uncertainty in accuracy due to target being further away versus acting late with reduced success probability due to faster target movement. This directly enforces the OV property of limited action windows.

Structured attack behavior. All ships descend toward the mothership and fire a laser upon entering the attack phase. Each laser deals exactly 1 damage, and the mothership begins with 100 health. These values are chosen to create a smooth and interpretable degradation signal in stages rather than an abrupt failure, which clearly paints a mental picture of the difficulty of the task for users. For the experiment, it yields a stable performance metric tied directly to unhandled threats.

Uniform motion across classes. Targets and non-targets share identical trajectories and behaviors. This prevents motion cues from trivially revealing class identity. As a result, classification must rely on learned semantic features and accumulated evidence,

supporting the OV requirement that relevance is inferred rather than directly observable.

V.3 Target Structure and Feedback

Discrete target classes. Each block contains five spaceship types, with exactly one designated as the target class at any given time. This number is chosen to balance task complexity: fewer types would make classification trivial, while more would dilute learning signal within short blocks. Similarly, across blocks, the target class also randomizes. For OLIVE, it ensures that adaptation corresponds to learning a concept, not memorizing individual items.

Silent target switch behavior. During target switches (US3), previously valid targets immediately disengage and fly away from the mothership, while the new target class begins attacking. This behavioral transition is designed not to be explicitly announced, while perceptually non-obvious. Hence, for the user, prior knowledge becomes actively misleading so they have to re-identify the target, while for OLIVE, this introduces conflicting evidence that must be resolved during training, directly learning contextual target shift.

Post-destruction feedback. Destroyed ships produce an explosion followed by a spinning collapse with colored smoke: red for targets and green for non-targets. This delayed but unambiguous feedback reinforces correct or incorrect actions. Users can then rapidly learn the current correct target class, leading to behaviors that would influence OLIVE convergence rate.

V.4 Gun Interaction Mechanics

Bullet assistance and curvature. Projectiles include a mild homing effect: bullets curve toward nearby ships, with curvature increasing as distance from the gun increases. This design reduces motor noise and ensures that misses reflect decision errors rather than aiming difficulty. For the user, shooting feels responsive and forgiving. For the experiment, it preserves the interpretation of actions as meaningful evidence.

Action rate limit. The weapon enforces a 1-second cooldown, visualized via an energy bar (blue = ready, red = depleted) and an audio charging cue. This limit is chosen to cap the maximum number of actions per block, discouraging user from mindless spraying without calculated pre-aiming that truly reflects decision. In addition, for the user, this forces prioritization: not all perceived targets can be acted upon. This enforces a limited action bandwidth, a key OV constraint.

Handed interaction. The weapon is attached to the user's dominant virtual hand at the start of each block. This ensures consistent ergonomics and minimizes variability in motor behavior across participants.

V.5 Assistance Interface (AI Agent)

Highlight-based assistance. When the agent believes an item is a target, it renders a red outline around the spaceship. This representation is intentionally minimal: it provides guidance without adding much visual clutter. For the user, this acts as an attentional cue rather than an automated action, preserving agency in the hands of user.

Field-of-view augmentation. A secondary "wingman" agent scans regions outside the user's field of view and can highlight targets detected in those regions. If a highlighted target lies outside the user's view, an edge-of-screen line indicator points toward its direction. This design extends the user's effective perceptual field, which directly addresses competing concurrency by enabling the system to monitor items the user cannot currently attend to.

Shared perception model. The wingman highlights both items in the user's view and items in its own scanned region. This ensures that assistance is not limited by the user's gaze, enabling earlier detection of relevant items and faster convergence of wingman model as it obtains a broader range of visual signals.

Shooting as confirmatory action. Interaction is designed as a confirmatory targeting mechanism rather than continuous shooting. When the user aims at a spaceship, its outline turns green; pressing the trigger causes immediate destruction. This removes fine motor requirements (e.g., projectile accuracy) and ensures that shooting reflects deliberate intent. For non-highlighted spaceships, the shooting remains unchanged.

V.6 User Interface and Feedback

In-task information display. A smartwatch interface on the user's left hand displays key information such as mothership health and remaining time. This placement ensures that status information is accessible but does not interfere with primary task engagement.

Pre- and post-block interfaces. Before each block, users are shown the target class as an interactive 3D object that can be inspected. This ensures that the initial task definition is unambiguous. Post-block interfaces collect subjective responses and provide structured transitions between blocks.