Implementation and Evaluation of an Adaptive Neuro-Haptic Interface for Virtual Reality (ANHIVR)

Author: Gotam Sai Varshith (22BCE1605)

Abstract

Virtual Reality (VR) interaction design is characterized by a fundamental trade-off between user immersion and task precision. This paper details the implementation plan for an **Adaptive Neuro-Haptic Interface for VR (ANHIVR)**, a novel system designed to resolve this conflict. ANHIVR leverages the naturalness of bare-hand tracking for high immersion while dynamically mitigating its inherent imprecision. The system operates on a closed-loop principle, using a non-invasive Electroencephalography (EEG) headset to objectively detect a user's cognitive conflict by measuring the Feedback-Related Negativity (FRN) neural signal. When frustration or repeated errors are detected, the interface intelligently transitions to a high-precision, controller-assisted mode. The proposed implementation will be developed in the Unity 2023 engine, integrating a Leap Motion sensor for hand tracking and an Emotiv EEG headset. The system's efficacy will be validated through a within-subjects user study comparing its performance and user experience against traditional hand-tracking-only and controller-only interaction paradigms.

1. Introduction

1.1. Problem Statement

The primary goal of Virtual Reality is to create a sense of **presence**—the feeling of being genuinely inside a virtual space. Direct, unmediated interaction using one's own hands is a powerful mechanism for achieving this, offering a natural and intuitive user experience. However, as identified in the literature, current hand-tracking technology often struggles with the precision required for fine motor tasks, leading to tracking errors, failed interactions, and significant user frustration. Conversely, physical controllers provide the reliability, precision, and haptic feedback necessary for complex tasks but act as a constant physical and cognitive reminder that the user is holding a tool, thereby breaking the immersive illusion. This creates a dichotomy where designers must choose between optimizing for immersion or for usability, often sacrificing one for the other.

1.2. Key Insight from Literature

Recent advancements in Brain-Computer Interfaces (BCI) offer a pathway to resolve this dilemma. Research by Singh et al. (2017) demonstrated that cognitive conflict—the brain's response to an unexpected or negative outcome—can be measured objectively using EEG. Specifically, the **Feedback-Related Negativity (FRN)** is a neural signal that reliably appears approximately 250ms after a user recognizes an error. This signal acts as a biological marker for frustration or difficulty. The ability to detect this signal in real-time provides the scientific foundation for a system that can understand *when* a user is struggling, without needing explicit feedback.

1.3. Proposed Solution: ANHIVR

This work proposes the **Adaptive Neuro-Haptic Interface for VR (ANHIVR)**, a bio-adaptive system designed to dynamically balance immersion and usability. ANHIVR operates as a closed-loop system that constantly monitors the user's cognitive state. By default, the user interacts via immersive hand-tracking. The system's EEG processing core analyzes the user's brainwaves for FRN events that indicate struggle. If the level of cognitive conflict surpasses a predefined threshold, ANHIVR seamlessly transitions the interaction modality to a controller-assisted mode, offering higher precision to overcome the challenging task. Once the task is completed and the cognitive conflict subsides, the system reverts to hand-tracking, restoring maximum immersion. This approach aims to provide the best of both paradigms: the effortless immersion of hands-on interaction and the robust precision of controllers, delivered precisely when needed.

2. System Architecture and Technology Stack

The ANHIVR system is designed with a modular architecture to handle the flow of data from input sensors, through the processing core, and to the VR application.

The proposed closed-loop architecture for the ANHIVR system.

2.1. Hardware Components

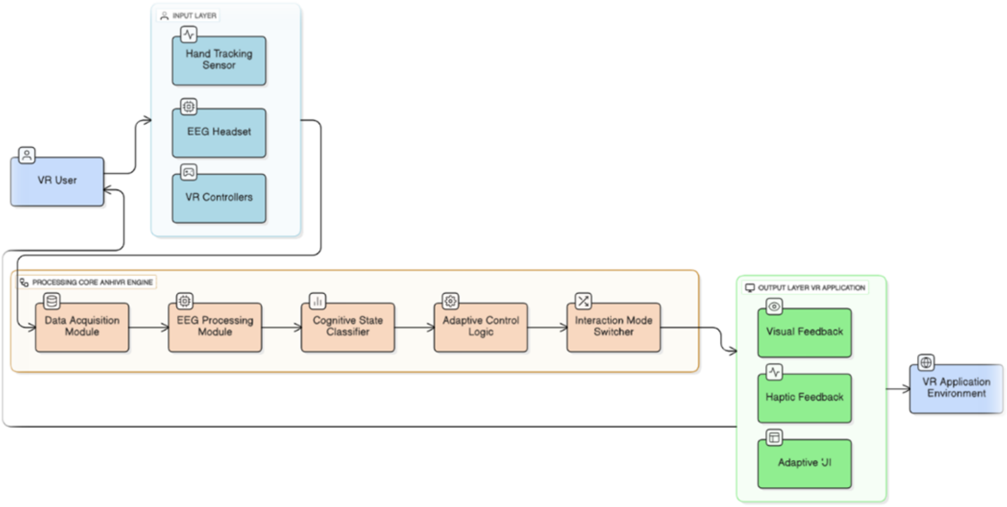
- **VR Headset: HTC Vive Pro 2**. Chosen for its high-resolution display (2448 × 2448 pixels per eye) which is crucial for tasks requiring visual acuity, and its robust SteamVR external tracking system that minimizes occlusion.
- **Hand Tracking: Leap Motion Controller 2**. This sensor is selected for its high-fidelity, low-latency articulated hand tracking and its well-supported Unity SDK, allowing for the capture of nuanced finger movements essential for natural gesture recognition.
- **EEG Headset: Emotiv EPOC+**. This 14-channel wireless EEG headset is chosen for its balance of signal quality and user comfort. Crucially, it provides raw EEG data access and a mature SDK (Emotiv Cortex SDK) compatible with external applications like Unity, which is a prerequisite for real-time FRN detection.
- **Haptic Controllers: Standard HTC Vive Controllers**. These are used to provide precise input and tactile haptic feedback during the controller-assisted mode, confirming successful actions like grabbing or placing objects.

2.2. Software Components

- **Game Engine: Unity 2023**. Unity is the chosen development platform due to its extensive support for VR hardware, a rich asset store for rapid environment creation, and its C# scripting environment, which is well-suited for implementing complex logic.
- **Programming Language: C#**. The object-oriented nature of C# allows for the clean, modular implementation of the ANHIVR engine's distinct components (Data Acquisition, EEG Processing, Adaptive Logic, etc.).
- **Key SDKs:** The implementation will integrate the **Unity VR SDK** (for headset integration), the **Leap Motion Unity SDK** (for hand data), and the **Emotiv Cortex SDK** (for streaming and processing EEG data).

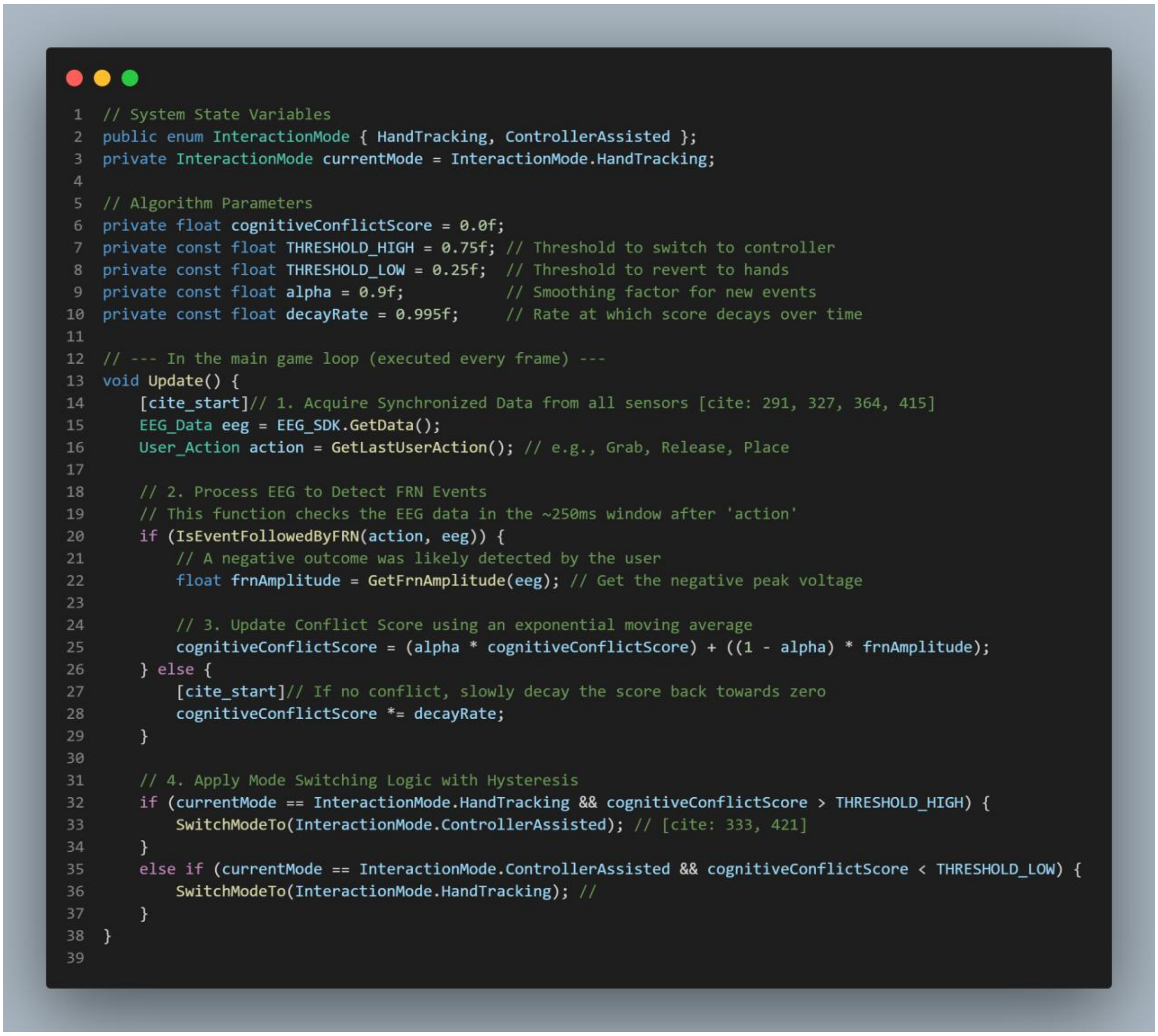
3. Core Algorithm Implementation

The heart of the ANHIVR system is the **Adaptive Control Logic Algorithm**. This algorithm runs continuously, maintaining a cognitiveConflictScore that quantifies the user's recent level of struggle. It functions as a state machine, using this score to decide when to transition between the HandTracking and ControllerAssisted modes.



3.1. Pseudocode for the ANHIVR Logic

C#



```
// System State Variables
public enum InteractionMode { HandTracking, ControllerAssisted };
private InteractionMode currentMode = InteractionMode.HandTracking;

// Algorithm Parameters
private float cognitiveConflictScore = 0.0f;
private const float THRESHOLD_HIGH = 0.75f; // Threshold to switch to controller
private const float THRESHOLD_LOW = 0.25f;  // Threshold to revert to hands
private const float alpha = 0.9f;           // Smoothing factor for new events
private const float decayRate = 0.995f;     // Rate at which score decays over time

// --- In the main game loop (executed every frame) ---
void Update() {
    [cite_start]// 1. Acquire Synchronized Data from all sensors [cite: 291, 327, 364, 415]
    EEG_Data eeg = EEG_SDK.GetData();
    User_Action action = GetLastUserAction(); // e.g., Grab, Release, Place

    // 2. Process EEG to Detect FRN Events
    // This function checks the EEG data in the ~250ms window after 'action'
    if (IsEventFollowedByFRN(action, eeg)) {
        // A negative outcome was likely detected by the user
        float frnAmplitude = GetFrnAmplitude(eeg); // Get the negative peak voltage

        // 3. Update Conflict Score using an exponential moving average
        cognitiveConflictScore = (alpha * cognitiveConflictScore) + ((1 - alpha) * frnAmplitude);
    } else {
        [cite_start]// If no conflict, slowly decay the score back towards zero
        cognitiveConflictScore *= decayRate;
    }

    // 4. Apply Mode Switching Logic with Hysteresis
    if (currentMode == InteractionMode.HandTracking && cognitiveConflictScore > THRESHOLD_HIGH) {
        SwitchModeTo(InteractionMode.ControllerAssisted); // [cite: 333, 421]
    }
    else if (currentMode == InteractionMode.ControllerAssisted && cognitiveConflictScore < THRESHOLD_LOW) {
        SwitchModeTo(InteractionMode.HandTracking); //
    }
}

```

3.2. Key Parameter Explanation

- **alpha (α)**: This is a smoothing factor used in the exponential moving average calculation. A value of 0.9 ensures that the cognitiveConflictScore is weighted towards its recent history, making it robust against single, spurious EEG artifacts while still being responsive to a consistent series of FRN events.
- **decayRate**: This factor ensures that if a user is performing well, their conflict score gradually reduces over time. This allows the system to "forgive" past struggles and revert to the more immersive mode when the user has regained composure.
- **THRESHOLD_HIGH & THRESHOLD_LOW**: Using two distinct thresholds introduces **hysteresis**. This prevents the system from rapidly oscillating between modes if the user's score hovers near a single point, which would be a jarring experience. The user must exhibit significant, sustained conflict to trigger the switch to controller mode, and then demonstrate sustained successful interaction to switch back.

4. VR Task Environment

To test the ANHIVR system effectively, a task was designed to require both broad navigation and fine-grained object manipulation.

- **Scene Setup:** The environment is a well-lit virtual workshop. A central workbench holds an incomplete engine block. Across the room is a parts bin containing various small components like screws, washers, and gears.
- **Task Flow:** The experimental task is "engine assembly". The user must:
    1. Navigate across the room to the parts bin.
    2. Identify and pick up specific small parts from the bin—a task that demands high precision.
    3. Carry the parts back to the workbench without dropping them.
    4. Precisely align and assemble the parts onto designated nodes on the engine block.
- **Interaction Switching:** The Interaction Mode Switcher module manages the transition smoothly. When the system switches to ControllerAssisted mode:
    1. A semi-transparent "ghost" of the Vive controller fades into view, superimposed over the user's hand model, providing a clear visual cue.
    2. Input mapping is updated. The 'pinch-to-grab' gesture is remapped to the controller's trigger, providing a discrete and reliable action.
    3. The physical controller's haptic motor provides a brief vibration to confirm the mode switch and subsequent successful actions.

5. Evaluation Plan

The effectiveness of the ANHIVR system will be assessed through a formal within-subjects user study. This design allows each participant to act as their own control, reducing inter-subject variability.

- **Participants:** 30 volunteers with prior VR experience will be recruited to minimize novelty effects that could confound the results.

- **Experimental Conditions:** Each participant will perform the engine assembly task under all three of the following conditions, with the order counterbalanced across participants to prevent learning effects:
    1. **Controller-Only:** The baseline for precision and performance.
    2. **Hand-Tracking-Only:** The baseline for immersion and naturalness.
    3. **Adaptive (ANHIVR):** The proposed system, which starts with hand-tracking and adapts based on cognitive conflict.
- **Performance Metrics:**
    - **Objective Metrics:** We will log task completion time, number of errors (e.g., dropped parts, incorrect placements), and the average FRN amplitude to objectively quantify performance and cognitive load.
    - **Subjective Metrics:** After each condition, participants will complete questionnaires to measure perceived experience : the **System Usability Scale (SUS)** for usability , the **NASA-TLX** for cognitive workload , and a standard immersion questionnaire to gauge their sense of presence.
- **Hypotheses:**
    - **H1:** The ANHIVR condition will result in a significantly lower error rate compared to the Hand-Tracking-Only condition.
    - **H2:** The ANHIVR condition will receive significantly higher usability scores (SUS) than the Hand-Tracking-Only condition.
    - **H3:** The ANHIVR condition will yield significantly higher immersion scores than the Controller-Only condition, with no significant difference from the Hand-Tracking-Only condition.

6. References:

- Chen, M.-X., Lai, Y.-H., Wu, T.-C., & Huang, Y.-M. (2024). A Survey on the Design of Virtual Reality Interaction Interfaces. *Sensors*, *24*(19), 6204. MDPI AG. Retrieved from
- Capece, N., D'Angelo, L., Erra, U., & Lamberti, F. (2023). *Evaluation of Virtual Reality Interaction Techniques: The Case of 3D Graph*. arXiv preprint arXiv:2302.05660.
- Khundam, C., Chaitanuwong, P., Kasetsuwan, J., & Thiemjarus, S. (2020). A Comparative Study of Interaction Time and Usability of Using Controllers and Hand Tracking in VR Training. *Informatics*, *8*(1),
- MDPI AG. Retrieved from 4. Selzer, M. N., & Castro, S. M. (2022). *Immersion Metrics for Virtual Reality*. arXiv preprint arXiv:2206.07748.
- Singh, A. K., Salvi, S., Gonzalez, D., & O'Donovan, J. (2017). *Measuring Cognitive Conflict in Virtual Reality with Feedback-Related Negativity*. arXiv preprint arXiv:1703.05462.