Name Tenzin Konchok

Email tkoncok@calpoly.edu

Program or Major Computer Science

Project Title: Emotion-Tracking and Visualization Dashboard: A Generalized Affective Computing Framework Demonstrated with Chess

Umbrella:
Our project aims to create a unique chess system where a virtual reality (VR) player competes against a physical robotic arm, with emotional data influencing how the game unfolds. The project integrates four independent yet connected components: a VR chess environment, a robotic arm that executes moves on a physical chessboard, an emotional reading system (e.g braincap or EEG-based tool) that detects the player's mental/emotional state, and a front-end middleware that coordinates data flow and provides a spectator dashboard. For example, when the VR player chooses a move in the VR game the robotic arm will physically pick up the corresponding chess piece on the real board and place it in the correct square, allowing both the virtual and physical games to stay synchronized. The robot will be able to make precise movements capable of seeing the VR player's chess movements without the player physically touching the real game board. It will also be able to move pieces it has captured off the board. The front-end interface will also provide live video gameplay to spectators and will provide data analytic visuals. It will ensure the robotic and VR player data connections (via MQTT) are current and reliable.

Project Description: My responsibility is specifically focused on developing the web based affective computing framework and visualization dashboard, while the VR and robotics components are handled by other team members.

The emotion-tracking framework will prioritize the following input modalities, in order of importance:

1. Facial expression recognition (via computer vision)

2. Heart rate monitoring (via connected device or API)

3. EEG signals (if time and cost allow)

These signals will be processed into quantifiable emotional states such as stress, focus, or frustration.

The web app/mobile dashboard will present emotional data in real time, mapped to contextual events from the game. The design will highlight correlations between player behavior and activity milestones, helping spectators and participants see how emotions evolve during gameplay. While chess is the initial test case, the goal is for the architecture to be general enough to connect with any event driven activity. Therefore, as a secondary demonstration, I will apply the dashboard to a simple video reaction test. The system will record and visualize facial-expression and heart-rate data while a user watches a short video clip. This illustrates the framework's applicability to non-game, event-driven activities.

Deliverables

- A modular data pipeline for real-time emotion and behavior signal capture.
- Facial-expression recognition system capable of detecting at least three core emotions with ≥70% accuracy under controlled conditions
- Heart-rate signal acquisition and processing module
- A web and/or mobile dashboard that visualizes emotion states over time and aligns them with activity events.
- Integration hooks for the middleware, enabling real-time synchronization between activity events and emotion data.
- Demonstration with chess as the primary case study and one simple secondary test that would be a video reaction analysis.
- Final documentation, system architecture diagram, dashboard mockup, demo video, and project report

Key Challenges

- Emotion recognition accuracy (lighting, headset, noisy signals, fusion complexity).
- Real-time performance (low latency required).
- Event synchronization (continuous emotion signals vs. discrete game events).
- Generalizability (extending beyond chess without overcomplicating).
- Visualization design (clear, intuitive dashboards).
- Hardware integration (different sensor APIs, reliability issues).
- Validation/testing (ground truth for emotions is subjective).

Testing

To evaluate my system, I will focus heavily on **user feedback** in addition to technical checks. The testing phase will aim to answer questions such as:

- Does the dashboard display emotional states in a way that spectators can easily understand?
- Do players and observers feel that the system accurately reflects the player's emotions and reactions during the game?
- Are spectators more engaged in the match when emotional data is shown alongside moves?
- Can the emotional data help players reflect on their performance after the game?
- For generalizability: how well the system works for the video reaction analysis?

Tools to Be Used

Known tools: Python (signal processing), JavaScript/React (web), Git/GitHub (version control).
To learn: TensorFlow/PyTorch (CV-based emotion recognition), React Native or Flutter (mobile app), D3.js or charting libraries (visualization), WebSockets (real-time streaming).

Timeline

First Quarter:

Week	Tasks	Milestone
1–5	Define project scope, complete mockup wireframes and system architecture diagram, finalize proposal and GitHub setup.	Proposal submitted, full design documentation and repo ready.
6	Begin implementation of facial-expression recognition prototype using open datasets and local webcam input	Basic detection model functioning on sample data.

7	Develop dashboard front-end skeleton (React + mock JSON pipeline) displaying live test data.	UI prototype visualizing simulated emotion/time data.
8	Integrate facial-expression module with dashboard for real-time visualization, refine emotion categories at least 3 emotions.	Live dashboard demo with camera feed and emotion overlay.
9	Add simulated heart-rate module (mock API or recorded dataset) and merge with emotion timeline, test synchronization logic.	Multi-modal mock pipeline working end-to-end.
10	Polish prototype, record demo video, and submit mid-project report including system diagram,	Q1 Milestone: “Emotion Dashboard MVP” ready with simulated data and documentation.

Second Quarter

Week	Tasks	Milestone
1–2	Integrate dashboard with real chess middleware and event stream.	Chess demo working.
3–4	Add multi-modal inputs (combining computer vison + heart rate).	Multi-modal input complete.
5–6	Enhance dashboard features.	Full dashboard features.
7–8	testing with real players; evaluate accuracy, latency, usability.	testing feedbacks or test results
9–10	Polish system, finalize docs, prepare final showcase.	Final demo-ready generalizable system.

My Availability:

M W F : 11am -1pm, 3pm - 6pm

T Th : 9am - 12pm, 4pm-6pm

Meeting Time

Wednesday 3PM Zoom meeting

Comments/Questions: