

MIND - CONTROLLED



FLAPPY BIRD

ANUSHA MADAPURA, ELLIOT BROTH, ALLYSON LE, AVISHI SHAH,

CAROLINE NGUYEN, AMY LI, LILY HIRASAWA, NIDHI BHOGI





Overview of our project

- Transforms *Flappy Bird* into an interactive BCI (brain computer interface) experience.
- Instead of using a keyboard/mouse, players control bird's jumps real-time using EEG signals.
- We collect and process brain activity to translate intentional mental state into game movements.



Why it matters

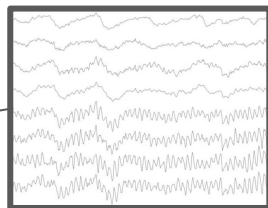
- Pathway to assistive technology + proves real-time BCI feasibility
- Demonstrates real-time neurofeedback - converting thought into action.
- Our work contributes to the product development for an emerging market for future possibilities for accessible gaming for people of all abilities.



How our system would look



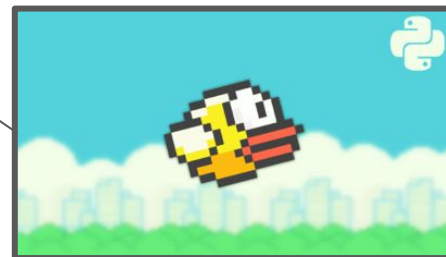
Emotiv - 5
channel
(hopefully)



Bandpass filters
+ artifact
removal

Classifier

Binary value
detection
(ML/threshold
based)



Python
(PyGame)

What is our target control signal?

Concentration & Relaxation

Features: Alpha/Beta Ratio

Pros: Simpler implementation & calibration, fast, widely proven

Cons: Extreme variability among users – recalibration

Blinks

Pros: Easiest to execute/straight forward, easiest to track

Cons: Simple/basic – many games also already have this implemented

Other metrics (Stress/Focus as a trigger)

Pros: More robust (multiple bands), potentially more personalized

Cons: Complex preprocessing and feature selection

No feature selection (Deep Learning)

Pros: Higher accuracy, learns optimal features

Cons: Computationally expensive (probably overkill) and higher latency (slower)

Lit Review- Liao et al.,

Main Idea: Participants played an EEG-based archery game where shots were controlled by focus level (FL) derived from EEG alpha rhythm (8-12 Hz)

1. Rejection of artifact signals
2. Extraction of the focusing feature on the power spectrum within the alpha band
3. Determination of the FL values
 - a. Setting that as the threshold for measuring focus levels

Application to the current project:

1. Methodologies are outlined in the experiment on a basic level
 2. Either change the control signal or stick with FL

Lit Review- More Advanced Ideas

Mind Drone Chasing Using EEG-based Brain Computer Interface (Al-Nuaimi et al.,)

- Control signal via looking at specific images on a screen
- Linking certain images to motor signals such as “up”, “down”, “left” and “right” caused by P300 response
- Linear Discriminant Analysis (LDA)

Towards Zero training for BCI (Krauledat et al.,)

- Control signal via motor imagery (thinking of moving particular body parts)
- Used Least Squares Regression

Timeline: Coding and Integrating



- Code up Flappy Bird w/ PyGame. Make sure it's working with spacebar

- Make sure entire pipeline is working with eye-blinks as control feature.

- Integrate the system with the selected model/feature

- Cont.

- Cont.

Nov/Dec

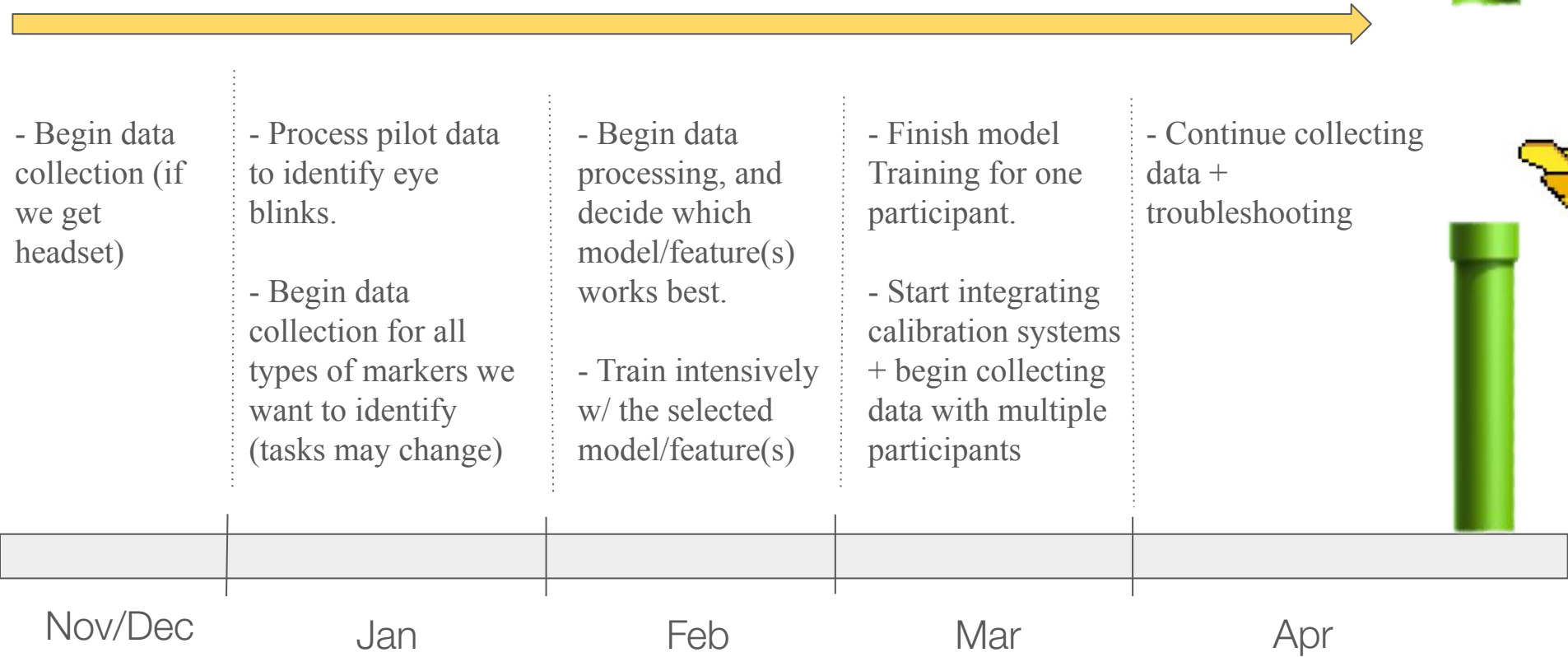
Jan

Feb

Mar

Apr

Timeline: Data Collection and Processing



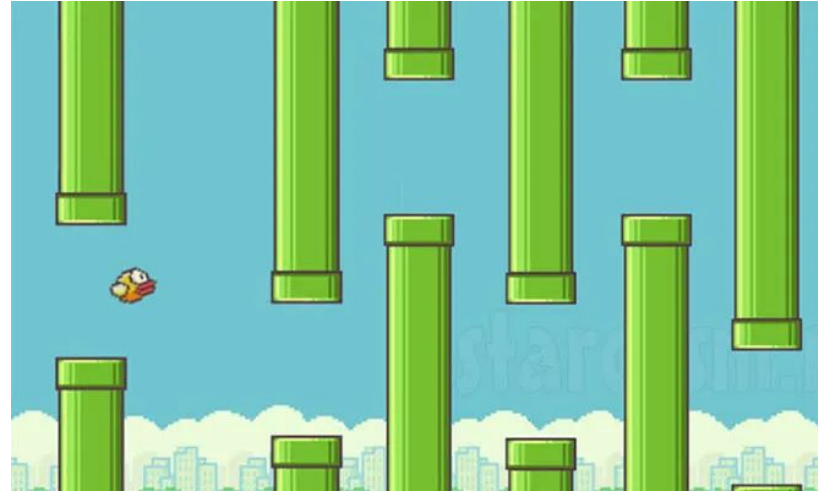
Technical Challenges We Anticipate

- Real-time responsiveness (latency)
 - Delay shouldn't be more than 100 - 150 ms
 - Solutions: make it threshold based
- Depending on feature we use - noisy data
 - If using EEGNet /some other “abstract” feature → heavily preprocess data
- Individual variability
 - Add calibration (each user's brain signals recorded prior to game)
- Human Focus/Fatigue
 - Can experience low attention over time → affects results

MVP

Total timeframe 3 and a half months - MVP in one month

- MVP: functional game calibrated to one person through eye blinks
 - Flappy bird UI built
 - EEG data collected and processed
 - Product calibrated to one group member
- Motor command: Eye blinks
 - Easily controllable command
 - Blink -> Up (Space)



Tasks/Teams

Coding

- Literature Review
- Converting EEG data to a binary input (0-1)
 - 0 -> no jump
 - 1 -> jump
- Real-time data handling
- Interface b/w EEG and game

Data Collection

- Literature review
- Collect EEG Data
- Preprocessing
- Coordinate w coding team to identify optimal features

Game

- Game design and development
- Build UI overlays - eg. being able to visualize concentration/relaxation threshold
- Optimize real-time responsiveness



**Playing
Flappy Bird**



**Becoming
Flappy Bird**