

# Process Log

**Project Title (working title):** Match up add-on

**Medium:** p5.js

## Motivation:

In today's internet culture, short-video platforms are everywhere. People spend a lot of time scrolling through videos, and the videos they choose to watch do reflect their interests and preferences. However, this kind of interest matching is often too direct. This project starts from a different question. Instead of asking what people watch, I ask: What if we look at how people swipe?

I am interested in the idea that video content still matters, but rather than using the content itself, I focus on the physical response it triggers, how fast someone scrolls, how often they change direction, whether they use one finger or multiple fingers, and how they exit the app.

By recording swipe gestures and translating them into abstract patterns, this project explores a more indirect and gentle way to reflect personal interests, and potentially connect people who respond to content in similar ways.

## Version 1:

At first, swipe trails were drawn with random colors. While visually interesting, but I think this randomness had no meaning.

I then redesigned the color system so that colors respond to gesture types:

- one finger vs. multiple fingers
- up, down, left, right
- zoom in vs. zoom out

Now, the colors are not decoration. They are the system's way of "reading" the gesture. I tried turn it from visual effect to behavioral meaning.

## Version 2:

When swipe traces stayed on screen, the interaction started to feel like drawing. But swiping is not drawing.

In real phone use, gestures are temporary. They happen, then disappear as new content replaces them.

So I changed the behavior:

- swipe traces appear only while swiping
- they disappear immediately after release
- but the system still records them in the background
- 

## **Version 3:**

To create a clear end point, I used a very familiar gesture: swiping up from the bottom to exit an app.

Only when this gesture is detected does the system change state:

- all recorded swipe traces are revealed
- they form a single static pattern

Once the static pattern is created, it is converted into a low-resolution fingerprint (a data array). This allows different sessions or different users to be compared.

## **Version 4:**

Early versions I used strict similarity calculations, which often produced very low or negative results. This felt cold and judgmental.

So I adjusted the similarity logic to be softer and more forgiving. The result is not a strict score, but a suggestion like a sense of closeness rather than a verdict.

## **Final version:**

As the interaction became more complex, the visual phone frame felt unnecessary. Since the project now runs directly on the phone browser and relies on real touch input. I removed it completely. Further, to place the gesture back into its real context, I introduced a vertical video feed with TikTok-style card transitions.