**Saket Kulkarni**
15-112 Term Project

# Interactive Fruit Ninja

**17th November 2023**

## Project Description

The Interactive Fruit Ninja game will be a spin on the classic game Fruit Ninja, where instead of just being able to use your cursor to break up the fruit, you can also turn on your webcam and use your hands. The game will use OpenCV to detect the player's hand positions and transpose them onto the visual design in CMU Graphics. From there, the player can embody their inner ninja and play fruit ninja in real life!

## Similar Projects

There have been many versions of Fruit Ninja online, including somewhere people use OpenCV to play the game. However, from the ones I have been seeing online, no one has directly built opencv into the game. They all use opencv to interact with their computer screen to play, without building it from scratch. Additionally, I intend to customize Fruit Ninja, creating procedurally generated fruits and infinite game mode, rather than only implementing the classic mode.

## Structural Plan

The structural plan will be up to change depending on how the project works but I am thinking of having multiple classes at the beginning. The first class will be the game class, which will handle score and calling the creation of a fruit. It will keep track of time and maintain all the elements of pausing, playing, and resetting. Then comes the board element, which I think will keep track of the cursor position and the fruit position. I think the fruit class should just contain the fruit itself, the images associated with the fruit and it will store the location that gets accessed by the board class. And finally, there will be an interaction class, that will be responsible for determining the players chosen input type, like accessing the correct camera adjusting for os, or just using the mouse. That way the board won't have to handle the accessing, it can just call the helper function from the interaction class. All these classes might have subclasses, like the fruits class may have apple, orange, watermelon, etc. subclasses but that is subject to the complexity of each fruit (complexity from how fast they fall, color after destroyed, etc.)

## Algorithmic Complexity

The hardest thing algorithmically is developing a model capable of identifying the hands of a person via their webcam and then laying that over the game itself. As described in the previous section, by abstracting every call element for user input of the game to a function, it should be easy to switch inputs however getting the inputs will be difficult. The current plan is using opencv which will allow me to get webcam access. Following that, I will use media pipe, an opensource software developed by Google which can allow me to identify hands from a given image. From there I will have to optimize it so that it can reliably and effectively get hand positions with minimal lag so the game is still fun to play.

## Timeline of Plan

The current timeline will consist of a week spent building up the entire game and its features. By the end of the week, I expect to have completed every single class, sans interaction, and have a game playable with just the cursor. From there, I want to spend the next week trying to implement hand tracking, optimization, and bug fixing.

## Version Control

For version control, I will be using Github. Here is the link to the Github Repository. The way this will work is I will have the main branch that should have the final working code. I will personally push to branches named after the class I plan on working on, so if I want to switch topics of coding for any reason, I can switch branches. Then, easily, I can create issues and merge into the main branch so that its clear to see and access older code in case I accidentally break something in the integration of everything.

## Module List

I plan on using 2 outside modules: OpenCV and MediaPipe. I plan on tech demoing this during my TP0 meeting with my TP Mentor, but that will happen after the submission of the assignment due to their availability. Thus, I am writing this down with the expectation that I will get cleared to use these modules, and if not, I will pivot to a different strategy.

**TP1 -** I Have No Update **TP2 -** I Have No Update **TP3 -** I Have No Updates