

## Motivation

We came in with the goal of using face tracking to control a cursor to make computer use more accessible for people with limited mobility.



## Define the problem

### Ability to use computer for those with mobility limitations

Those with mobility issues or lack of control of their hands may not be able to reliably control the mouse on their computer, so we wanted to create a way for users to control their cursors with just their faces.

We also see this functionality as adding a new way for all users to use their computers without having to risk wrist damage using un-ergonomic mice/trackpads.

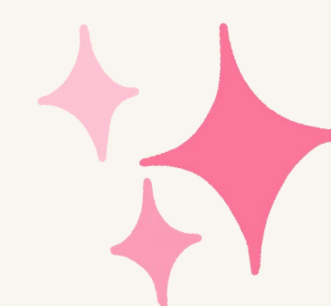
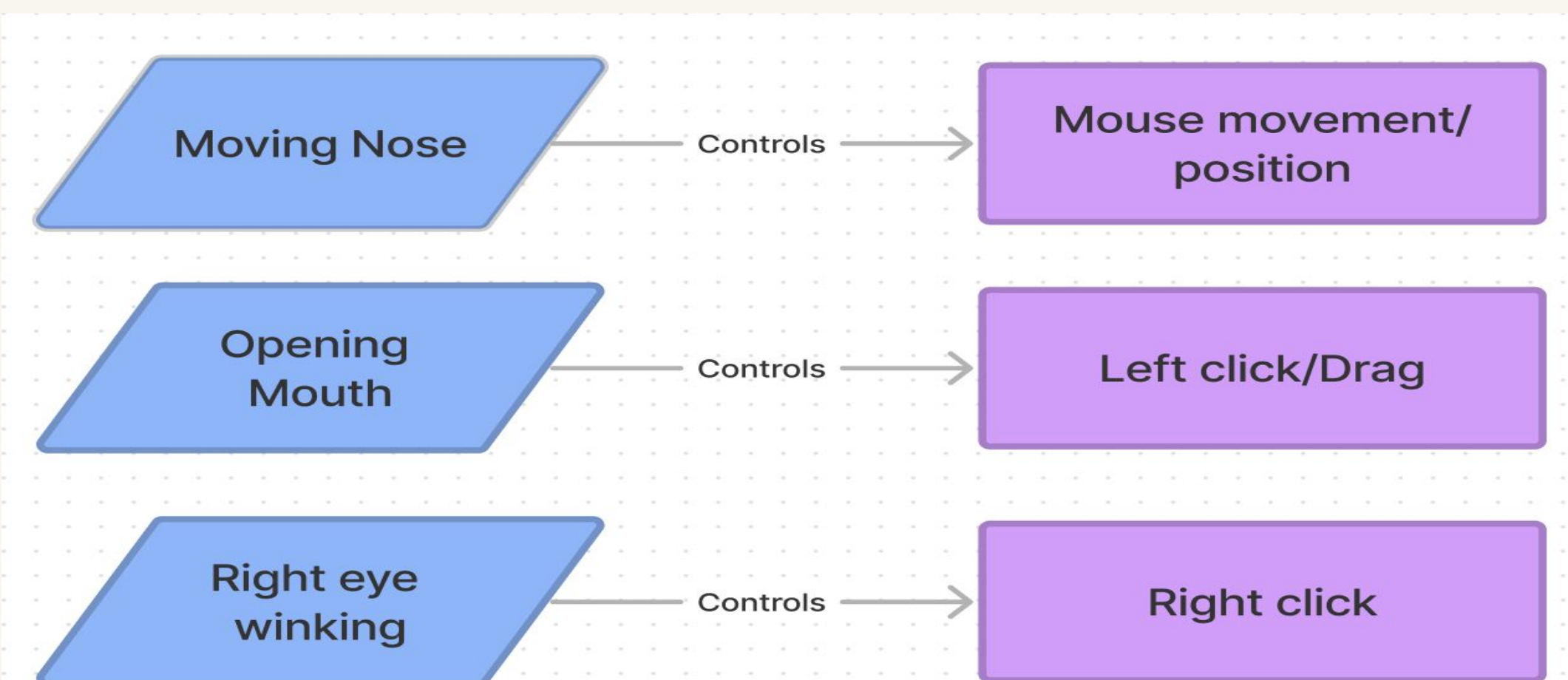
## Goal

### These were our goals for implementation:

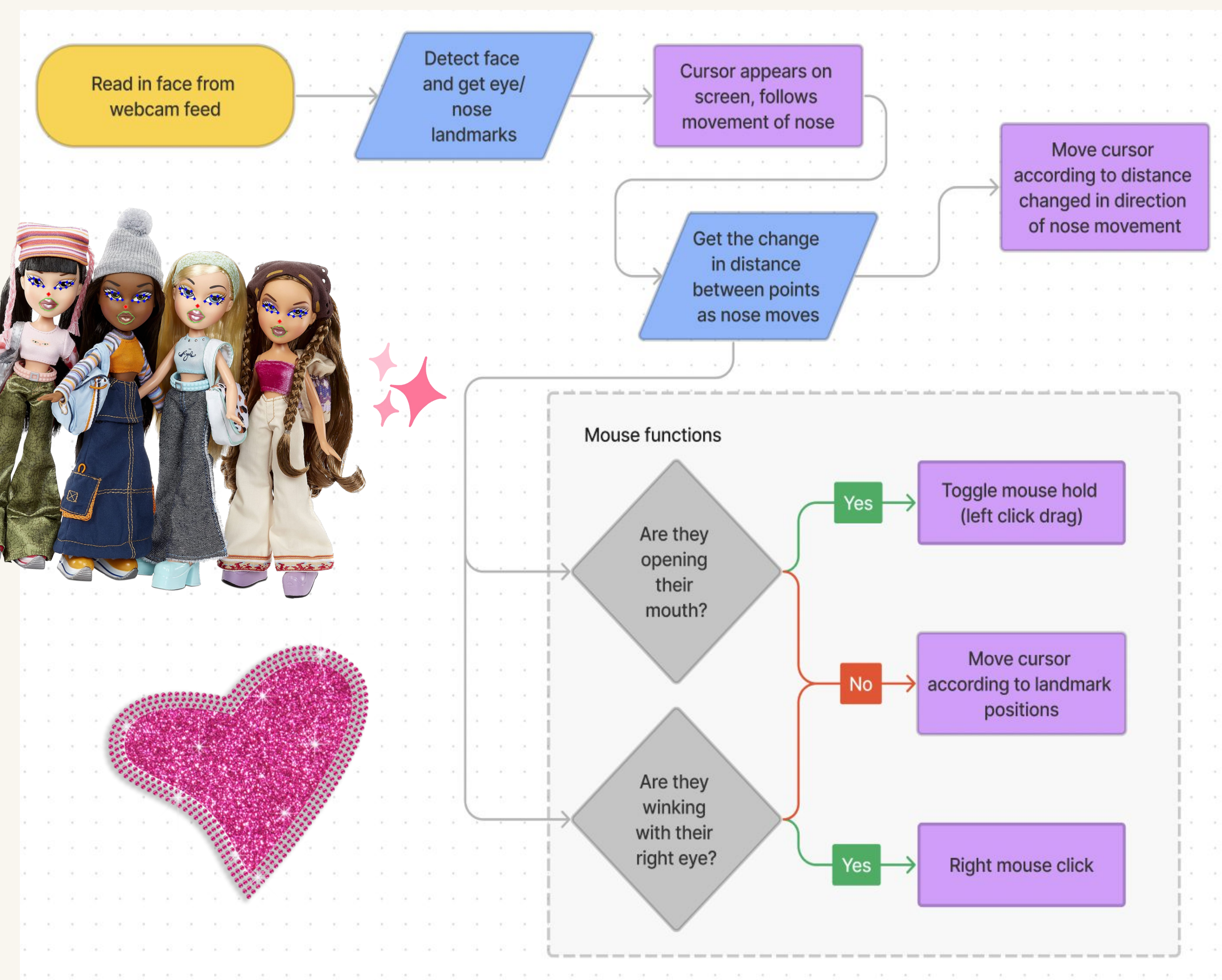
1. Detect facial landmark points via webcam
2. Connect cursor movement to face movement
3. Detect certain facial movements (like opening mouth) using distance between facial landmarks
4. Match facial movements to other mouse controls like clicking

## Implementation

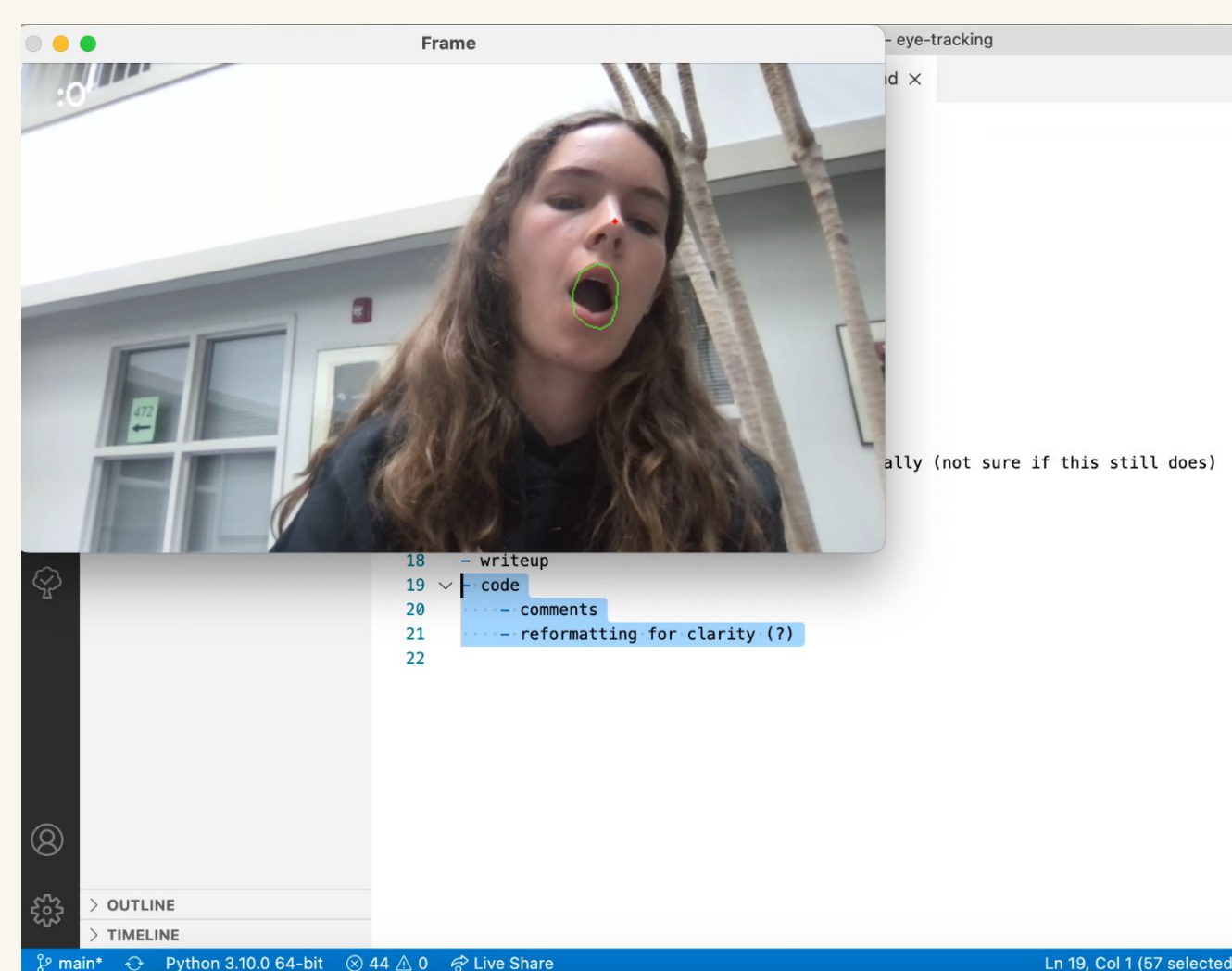
For face tracking, we decided on the following movements for the most robust cursor control:



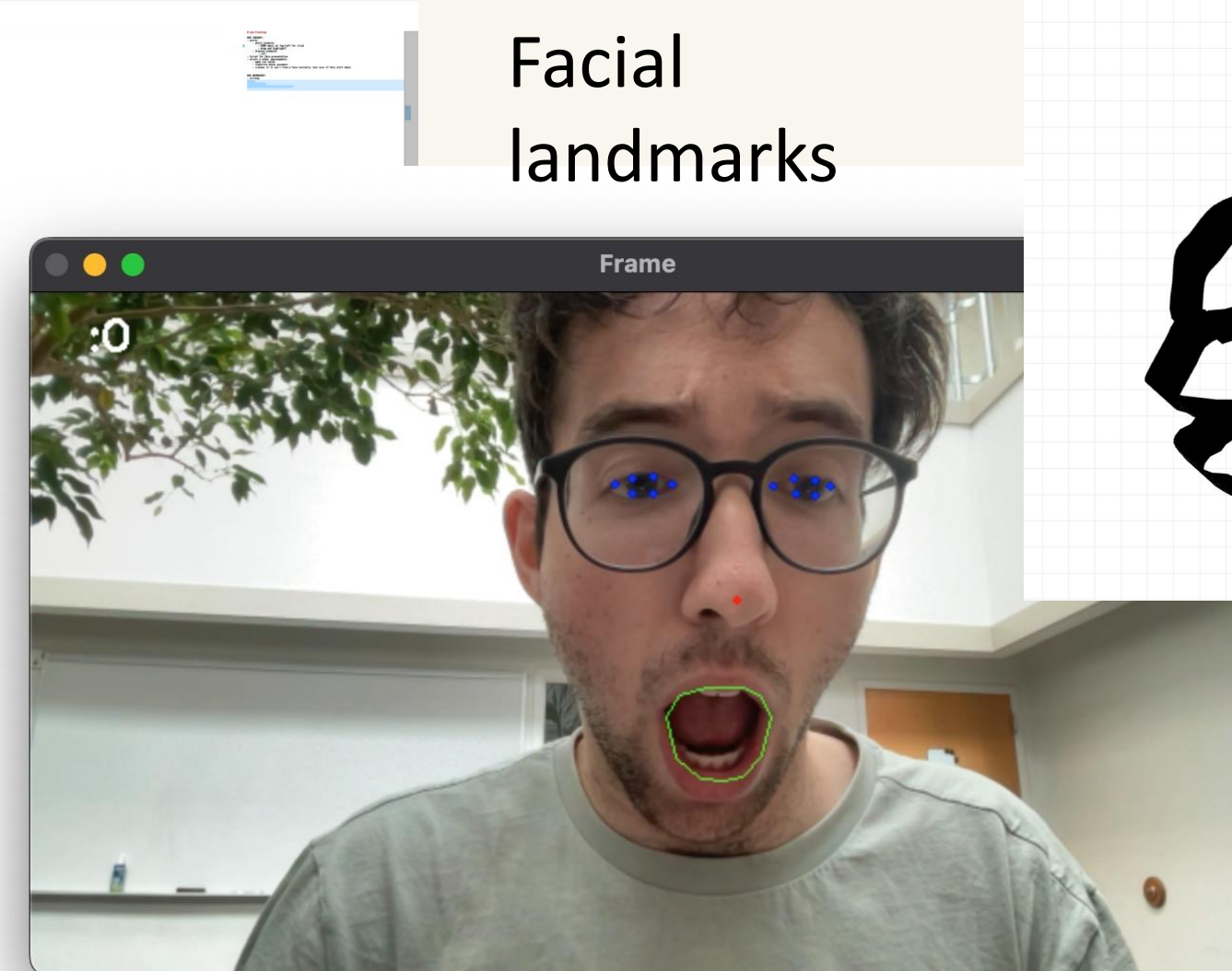
## Pipeline



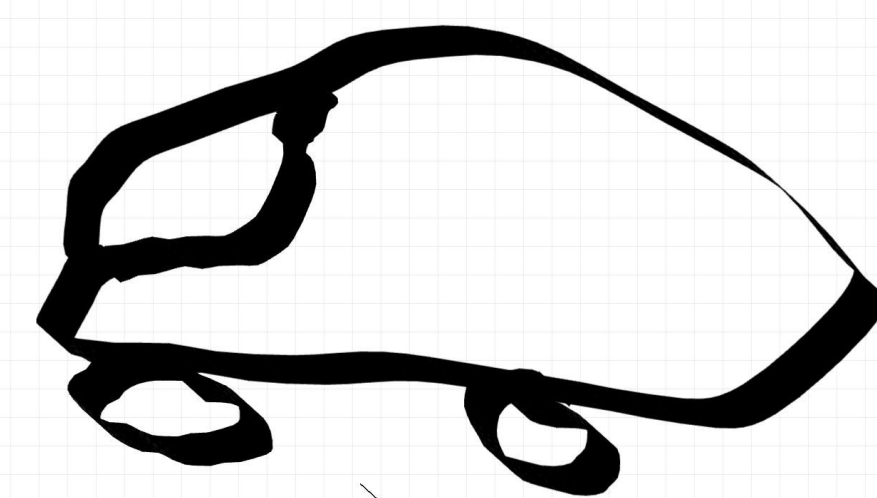
## Results (images/figures)



Open mouth + move head to click + drag



Facial landmarks



We drew a car using with just a face!

## References

- [1] Khalid, Irfan Alghani. "Face Landmark Detection Using Python." *Towards Data Science*, 6 Sept. 2021, <https://towardsdatascience.com/face-landmark-detection-using-python-1964cb620837>.
- [2] Mauck, Ross. "Detecting When a Human's Mouth Is Open." *GitHub*, 12 June 2019, [github.com/mauck/mouth-open](https://github.com/mauck/mouth-open).
- [3] Nielsen, Nicolai. *Face Detection and 3D Position Estimation in OpenCV*. *YouTube*, 24 Sept. 2020, <https://www.youtube.com/watch?v=-86rnJpIQ5o>.

## Acknowledgements

We would like to thank our TA, Qian Zhang, for helping guide us throughout this project, as well as the rest of the CS1430 staff and Professor Tompkin for all their support and hard work throughout the year!