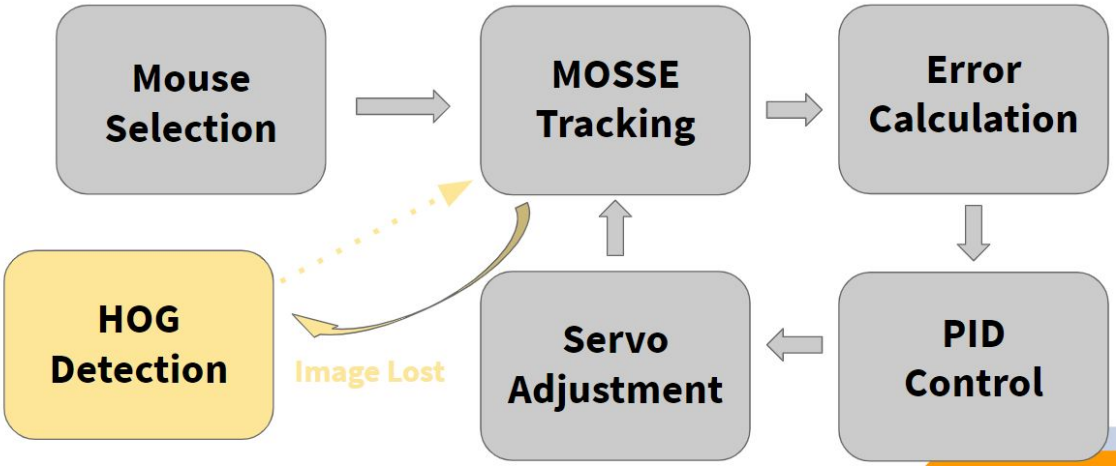
Program Structure



The main program is run in integrated.py. This program takes the mouse input from the user and is running the MOSSE tracking algorithm, calculating errors based on the outputs of the MOSSE algorithm. It then passes errors into the gpio_servo.py which contains the PID controller and servo PWM functions which control the actual servos to move. It also keeps track of the running sum which is multiplied by the integral coefficient as well as the previous value which is multiplied by the derivative coefficient and added to the total correction.

The hog_integrated.py is the program which was in development at the end of the project for obtaining the bounding boxes from the HOG detection algorithm and using these boxes to pass as input for the MOSSE tracking algorithm. The hog_bb.py function is used by the hog_integrated.py to obtain the actual bounding boxes and return them to the hog_integrated.py function. The hog_integrated function then has all of the same functionality as the integrated.py function described above.

Bugs and Areas of Improvement

1. MOSSE tracker does not work well when initialized with bounding boxes created by HOG. Tracker can not detect object within the box.
2. PID loop is not fully ideal. Currently the loop sacrifices angular velocity for smoother movement and avoids losing bounding boxes due to blurry images.
3. At startup the camera is occasionally prone to jumping too fast and far to reach the initial bounding box, causing MOSSE to lose the object it was supposed to track in the process.

Future Development

1. Experiment with increasing the size of HOG bounding boxes to make them potentially work better with MOSSE.
2. Manually implement MOSSE in order to better optimize it and improve its synergy with HOG.
3. Further tune the PID loop or use a more advanced control technique like a Kalman filter.
4. Consider a more robust design that includes stepper motors/driver, an aluminum or wooden gimbal, and perhaps a very affordable yet better suited microcontroller for this task, such as the NVIDIA Jetson Nano.