Systematic

March report

1. **Optimizing the model for various tracks**

After making our test track, we observed that the road detection didn't work as expected, so we needed to add the new track to our dataset to improve performance. The desired result is still far from perfect, but a larger dataset should solve the issue.

2. **Adding a failsafe detection**

We wanted a backup detection method in case the model-based one couldn't keep up with the task, so we used a common OpenCV approach. We detect the white lines in a specified region and form the road using that information. This algorithm is inspired by this article.

In the end, we didn't implement the full logic for the failsafe detection because we wanted to focus more on the model-based approach, and the performance of this algorithm was somewhat poor. We could improve this type of detection by using the actual shape of the road between two white lines and inside the specified mask. We can achieve this by simply getting the biggest "blob" whose color is within a specified HSV threshold and then finding its center. This way, we can also use the current PID values since this approach matches our current model-based one.

3. **Adding traffic sign logic**

We also wanted to implement some logic for traffic signs. We started with the "STOP" sign because our model can easily recognize it. We have a counter that starts at 200 and increments while the loop is running. If a STOP sign is detected, the car stops, waits 3 seconds, and then continues. The counter resets to 0 when a STOP sign is detected and triggers the car's stopping response only when its value is above 200. We still need a better dataset to ensure we can recognize all traffic signs. One problem we encountered with the STOP sign was that it was detected from too far away, but this can be easily solved by measuring the size of the generated bounding box.