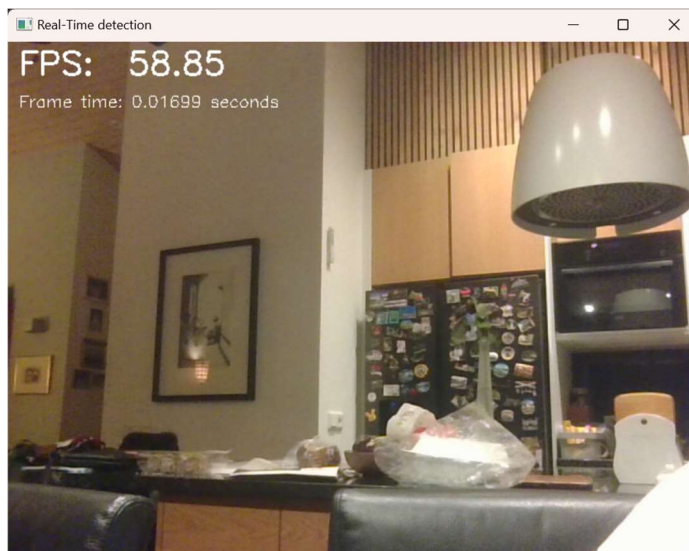## Assignment 1: OpenCV and point operations

After following the steps regarding how the code should work, I could answer the following questions:

- The processing time for one vide frame or image was calculated using Time_per_frame = 1 / FPS, which from the image below, was 0.01699 seconds


The average FPS when only displaying the image, is 74.72 FPS.
- When adding the brightest spot detection, the average FPS increased, which is now around 88.91 FPS, however, the screenshot I got only displays 39 FPS, which results in a higher processing time. I would assume that given the higher average

FPS, that the time should be lower.
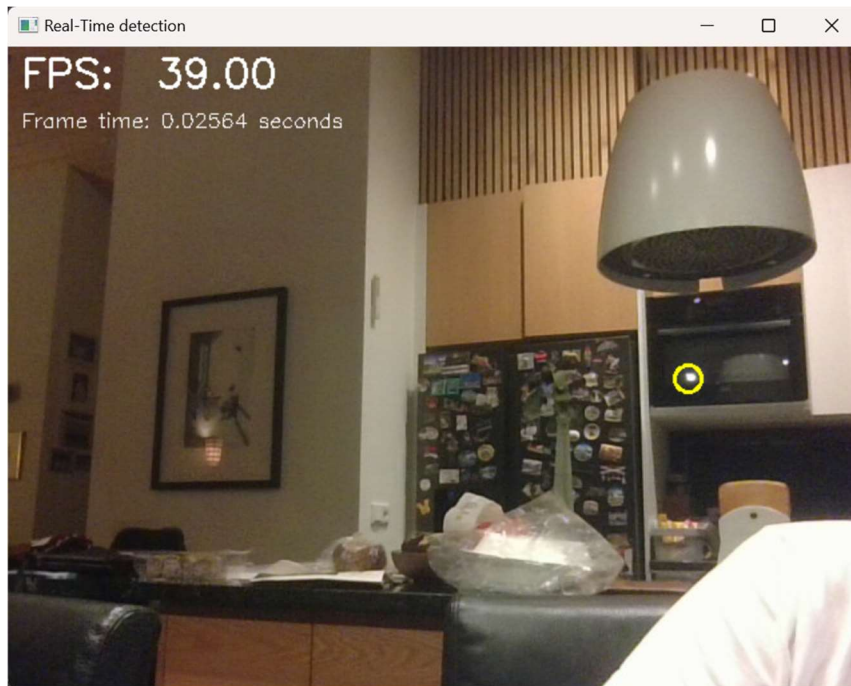
- If I do not display the image, and only print out the average FPS after 200 frames, I get that the average FPS is around 53.24 FPS, which is lower than before, so no, the processing time is not identical if I do not display the image.
- When comparing the for-loop implementation to the built-in function, I can see visually that they are marking the same (or similar) spots, as can be seen in the image below, (Here I am marking the brightest spot (both manually and with OpenCV, and the "reddest" spot as well)
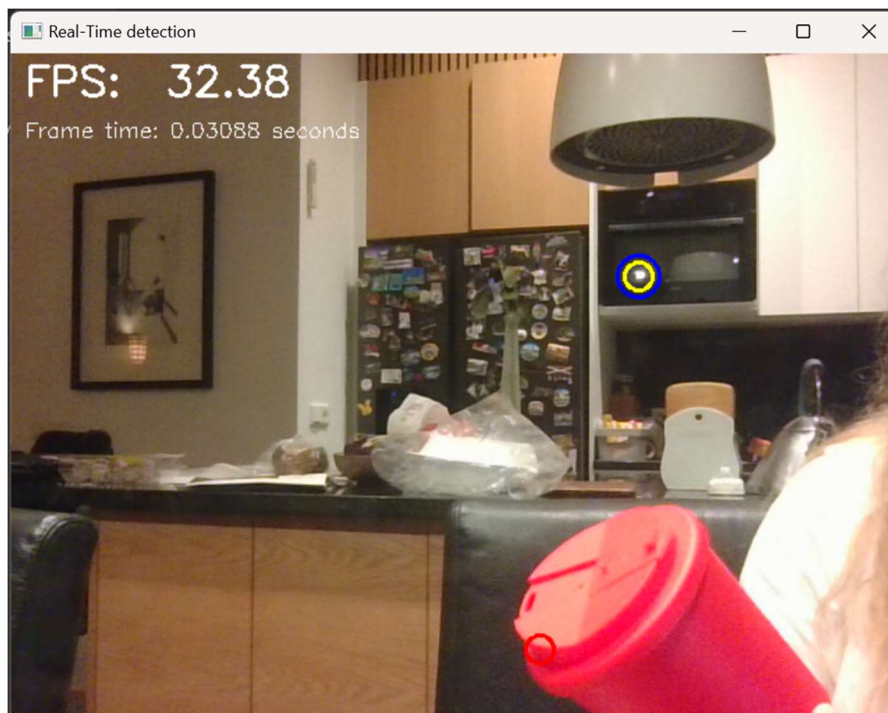
The average FPS is about 31.48 FPS now.

I timed the difference between the implementations and from the image below we can see that the built in function is quicker

- When moving my hand in front of the camera I can see visually that there is a little bit of a latency. It is however very small, and I am not sure how I am supposed to be estimating it correctly.
- The latency is more visible when capturing images from a mobile phone. This is likely due to the phone needing to transmit data over wi-fi, as well as the IP camera might be using higher resolution which can increase the latency.
  In addition to higher latency, the average FPS was lower. It was around 18.87 FPS.