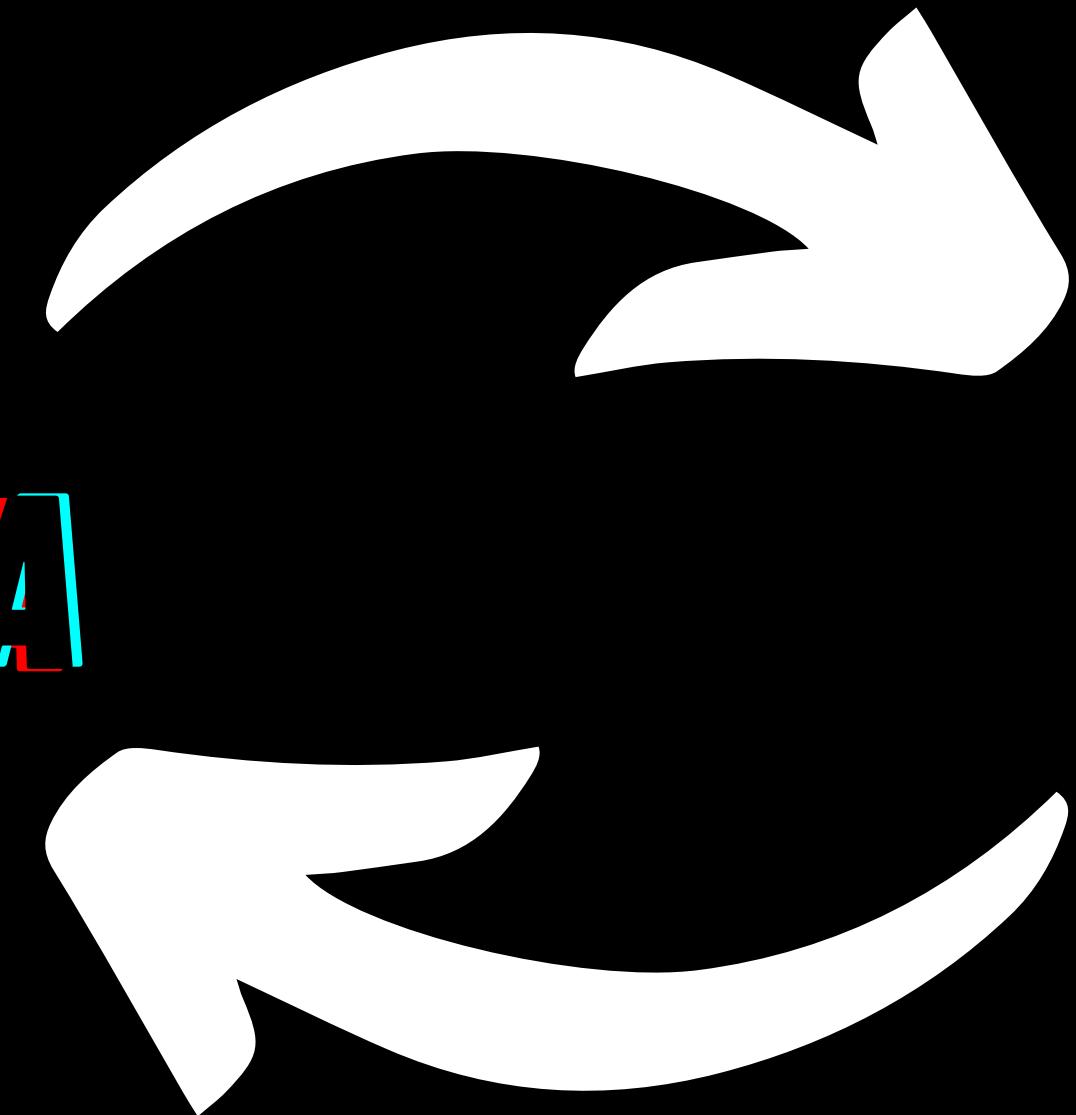


CLOSED LOOP CONTROL

GETTING CONTROL OVER THE MOTION OF THESE ROBOTS !



PYTHON



CAMERA

SLM

ROBOT PVS

STEPS

1. INITIALISING SLM
AND CAMERA

2. LOADING OPENCV

3. IDENTIFYING PVS
AND ROBOTS USING
CONTOURS

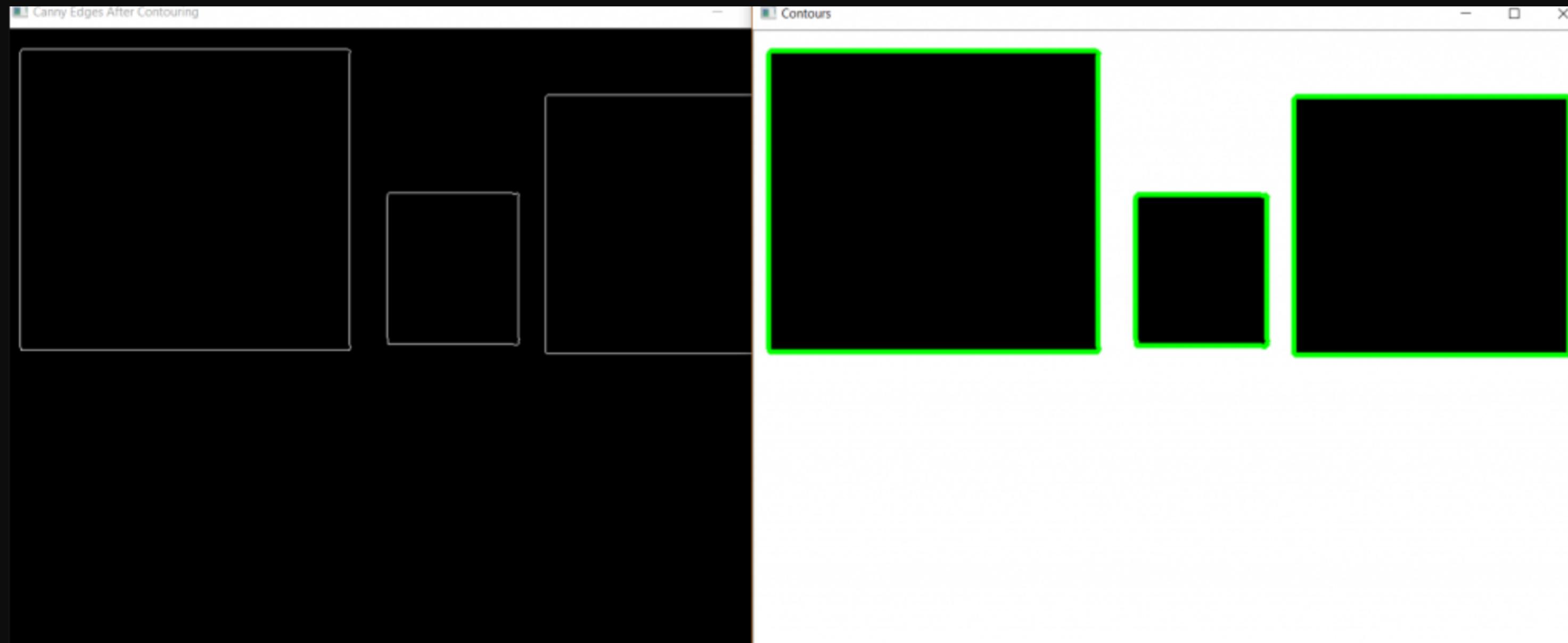
4. CREATING AN SLM
IMAGE

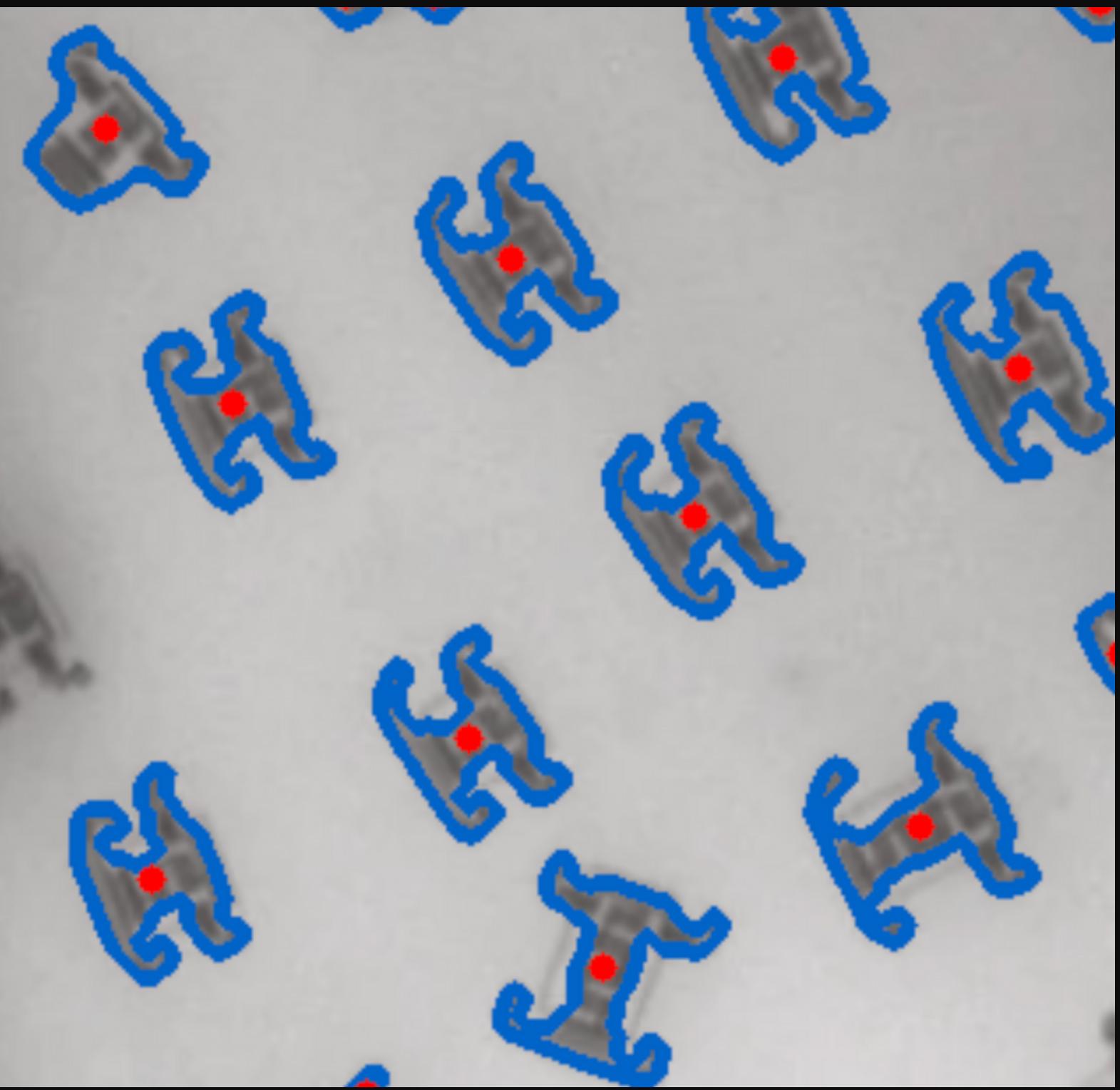
5. FIX ALIGNMENT
ISSUES

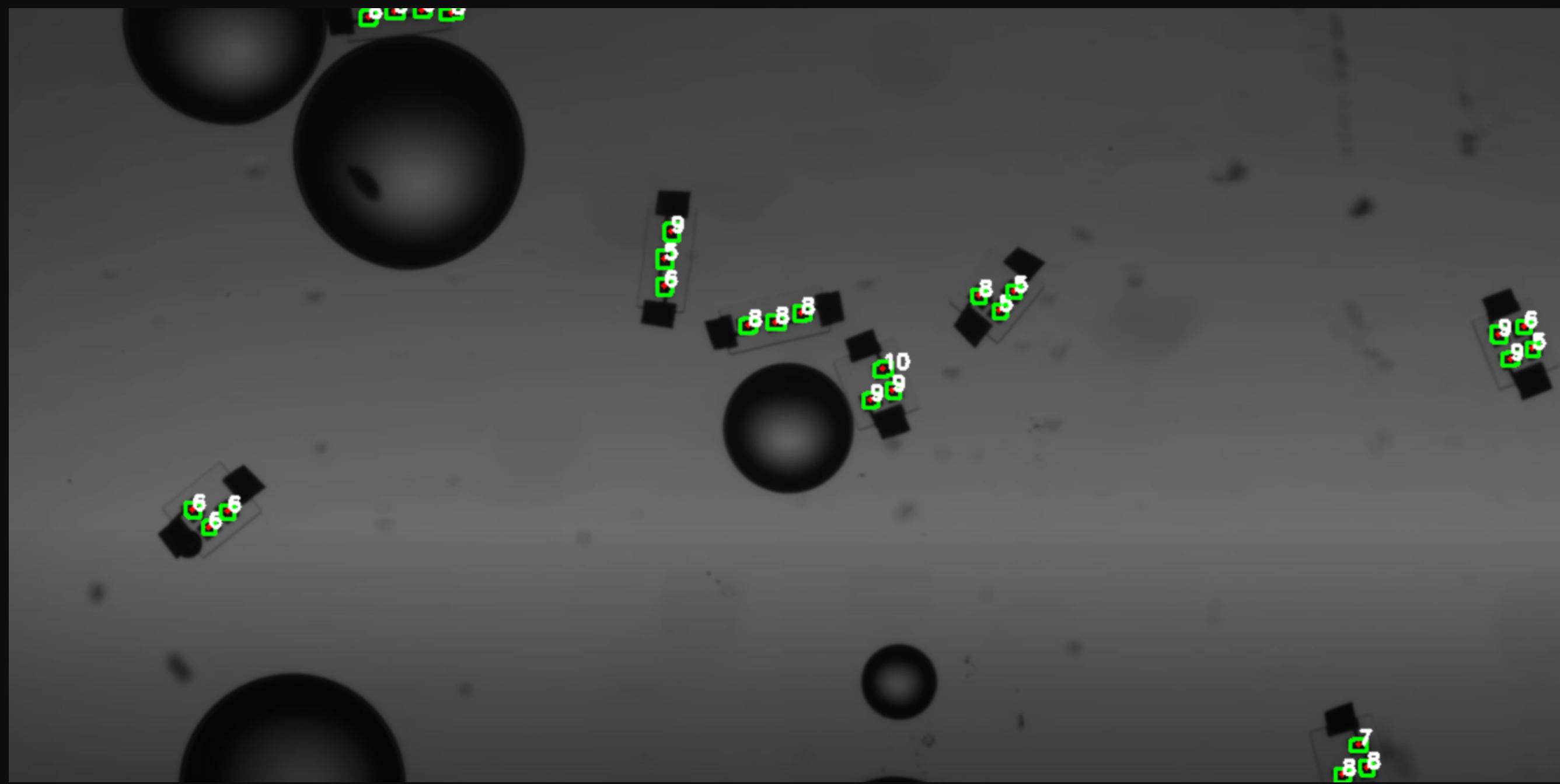
6. MAKE OPTICAL
SYSTEM MORE STABLE

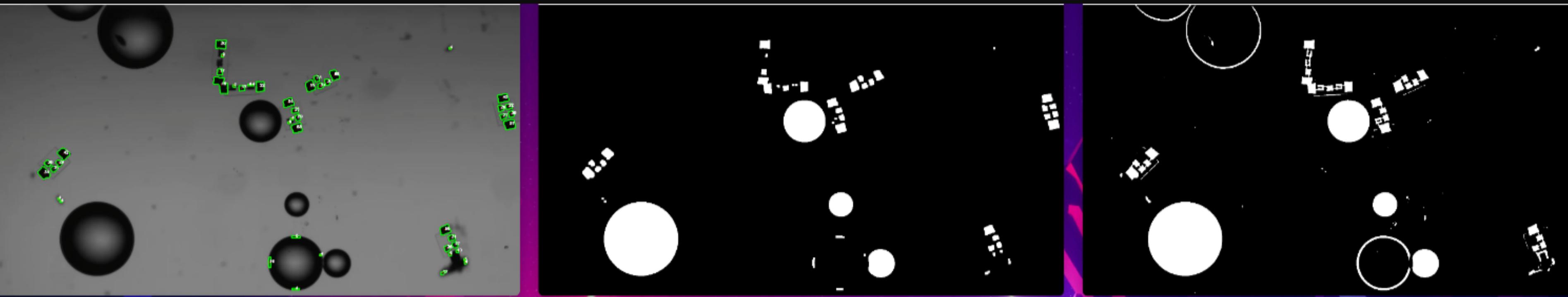
7. EXPERIMENT WITH A
ROBOT

8. ADD MORE
FUNCTIONALITIES SO IT
BECOMES A VIDEO
GAME









image

6: Morph Open

3: Filled Rectangular Contours

2: Threshold

1: Binary Image

2: Threshold

3: Binary Image

STEPS

1. INITIALISING SLM
AND CAMERA

2. LOADING OPENCV

3. IDENTIFYING PVS
AND ROBOTS USING
CONTOURS

4. CREATING AN SLM
IMAGE

5. FIX ALIGNMENT
ISSUES

6. MAKE OPTICAL
SYSTEM MORE STABLE

7. EXPERIMENT WITH A
ROBOT

8. ADD MORE
FUNCTIONALITIES SO IT
BECOMES A VIDEO
GAME



STEPS

1. INITIALISING SLM
AND CAMERA

2. LOADING OPENCV

3. IDENTIFYING PVS
AND ROBOTS USING
CONTOURS

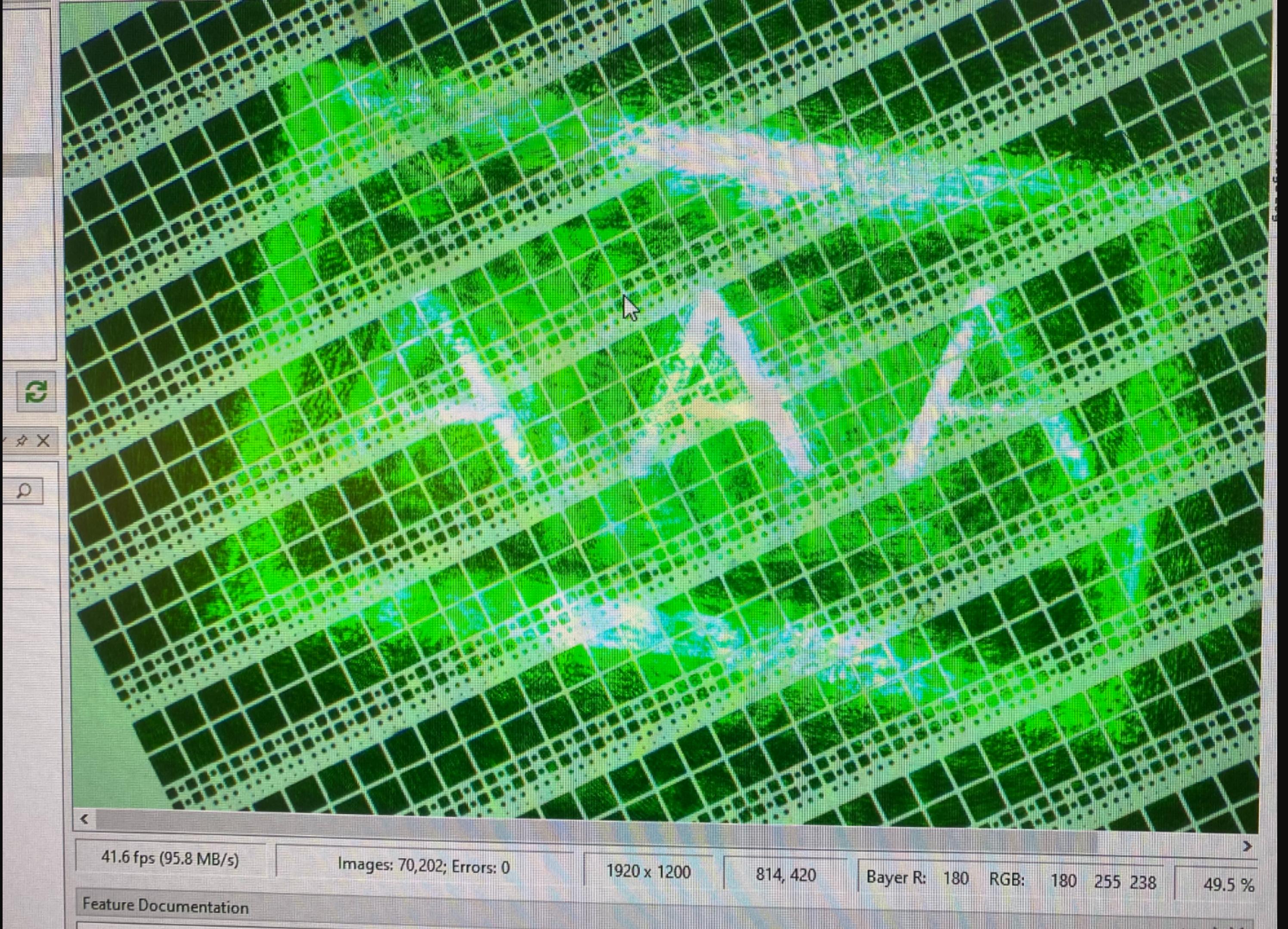
4. CREATING AN SLM
IMAGE

5. FIX ALIGNMENT
ISSUES

6. MAKE OPTICAL
SYSTEM MORE STABLE

7. EXPERIMENT WITH A
ROBOT

8. ADD MORE
FUNCTIONALITIES SO IT
BECOMES A VIDEO
GAME



41.6 fps (95.8 MB/s)

Images: 70,202; Errors: 0

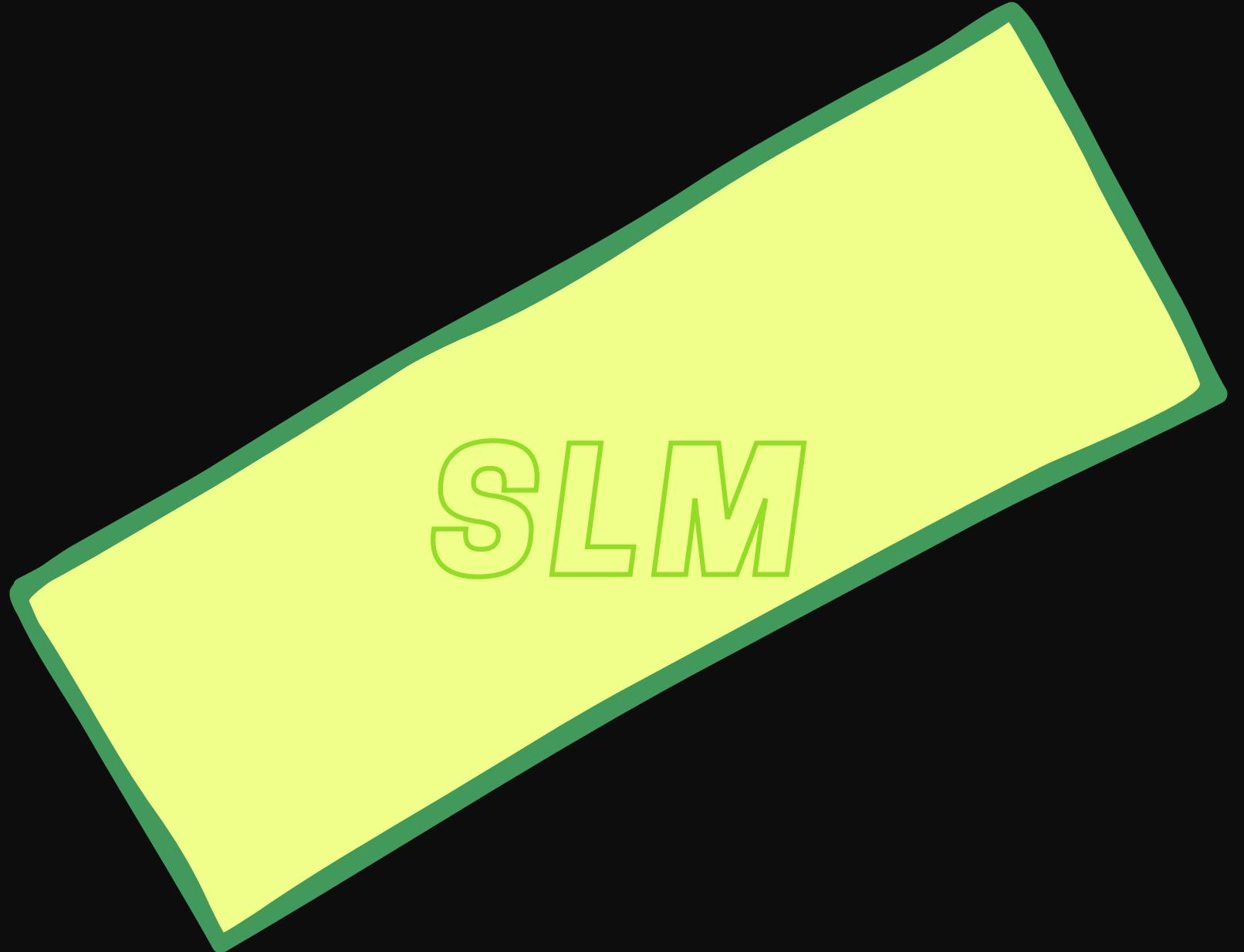
1920 x 1200

814, 420

Bayer R: 180 RGB: 180 255 238

49.5 %

Feature Documentation



STEPS

1. INITIALISING SLM
AND CAMERA

2. LOADING OPENCV

3. IDENTIFYING PVS
AND ROBOTS USING
CONTOURS

4. CREATING AN SLM
IMAGE

5. FIX ALIGNMENT
ISSUES

6. MAKE OPTICAL
SYSTEM MORE STABLE

7. EXPERIMENT WITH A
ROBOT

8. ADD MORE
FUNCTIONALITIES SO IT
BECOMES A VIDEO
GAME

