Edge Detection 5p

```python
def main(argv):

    pkm_file = open('parking_map_python.txt', 'r')
    pkm_lines = pkm_file.readlines()
    pkm_coordinates = []

    for line in pkm_lines:
        st_line = line.strip()
        sp_line = list(st_line.split(" "))
        pkm_coordinates.append(sp_line)


    test_images = [img for img in glob.glob("test_images_zao/*.jpg")]
    test_images.sort()
    print(pkm_coordinates)

    for img_path in test_images:
        whole_parking_img = cv2.imread(img_path)
        cv2.imshow("whole_parking_img", whole_parking_img)
        cv2.waitKey()

        for one_c in pkm_coordinates:
            one_space = four_point_transform(whole_parking_img, one_c)
            cv2.imshow("one_space", one_space)
            cv2.waitKey()
```

TASK – parking lot occupation detection

- continue with the template:

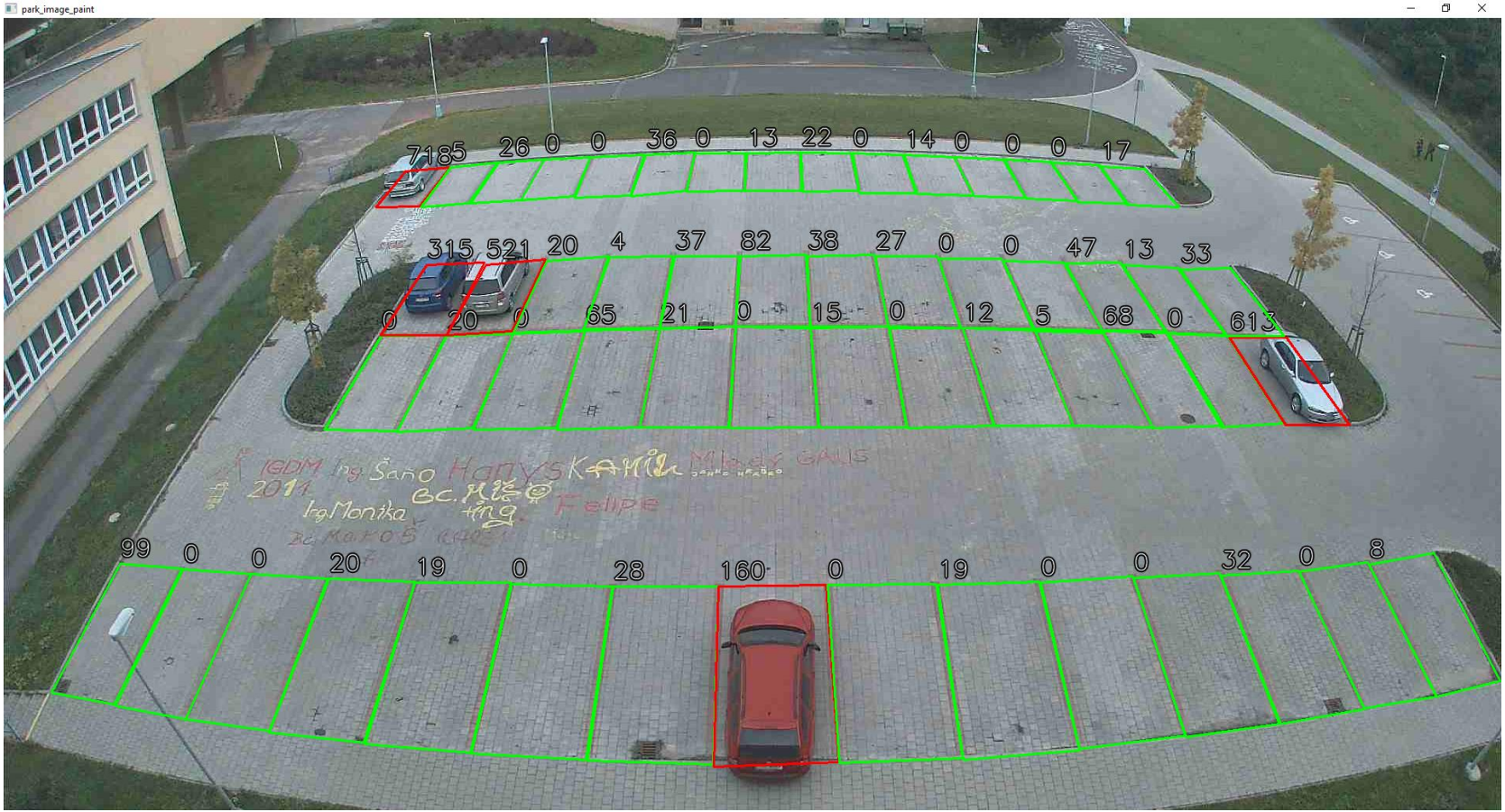
https://mrl.cs.vsb.cz/data/vyuka/zao/parking/parking_zao_template.zip

- experiments with different types of image filtering
- experiments with different types of edge detectors
- Hint: `cv.CountNonZero` after edge detection
- read .txt file with ground truth data of each parking place image (in "test_images_zao" folder) and calculate accuracy for each detector configuration:

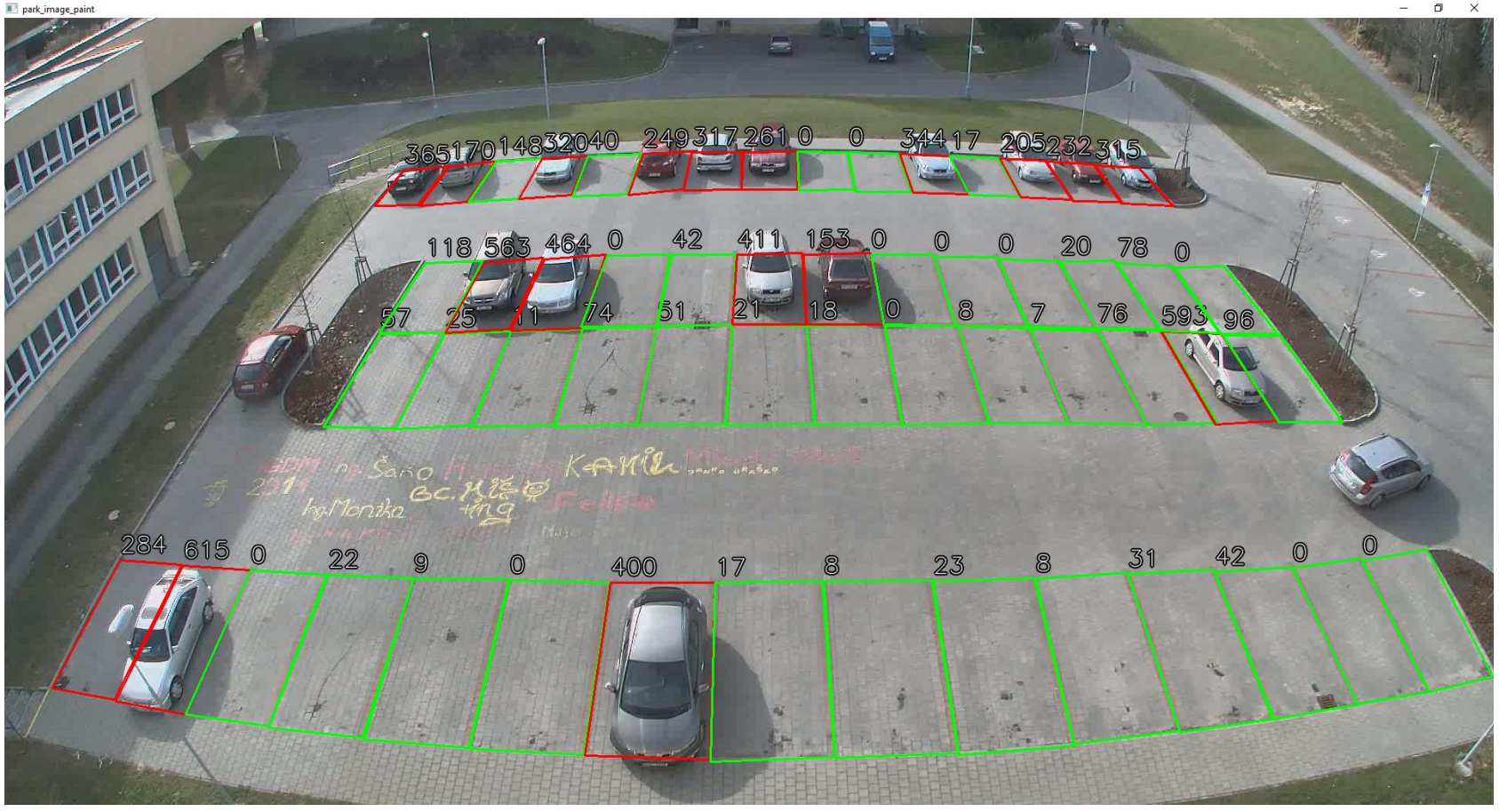
$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

where TP = True positive; FP = False positive; TN = True negative; FN = False negative

Edge Detection



Edge Detection



Edge Detection