VYTAUTAS MAGNUS UNIVERSITY
FACULTY OF INFORMATICS

Paulius Lapienis

COMPUTER SYSTEMS ADMINISTRATION
LAB WORK NO. 2

Kaunas, 2022

CONTENT

ABSTRACT

This work shows how changing the scale factor and minimum neighbours parameters affect the accuracy of Haar cascades face recognizer. The experiments performed follows this order:

1. Images/Videos are loaded using OpenCV
2. Images/Videos are converted to grayscale
3. Each Image/Video is loaded into the Haar cascades face recognizer
4. The recognizer is used to mark spots where a face is detected
5. The data is hand-checked and aggregated to a table
6. The data is visualized in a graph
7. Results/Conclusions are made

1 TASK: FACE RECOGNITION

1.1 Recognition in images

Images used for transformation were obtained from the internet. Each chosen image contained more 4 or more faces (4 and 6 respectively).

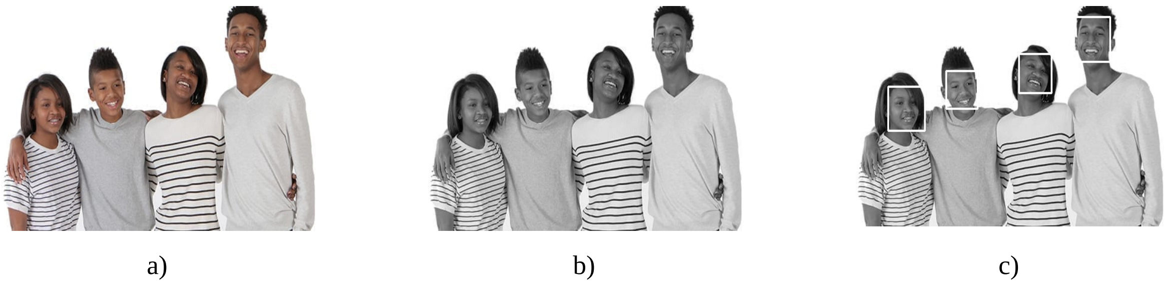
Images were then loaded using the OpenCV library and transformed to gray-scale using the cvtColor function.

cv2.cvtColor(img, cv2.COLOR_BGR2GRAY) (1)

These images were then used to detect images using the Haar Cascade classifier. To set the scale_factor an min_neighbours parameters values were randomly sampled from two ranges. For scale_factor 10 samples were picked from a range from 1.01 to 3.0 was chosen with a step size of 0.01. For min_neighbours parameter 5 samples were picked from a range of 1 to 10 with a step size of 1.

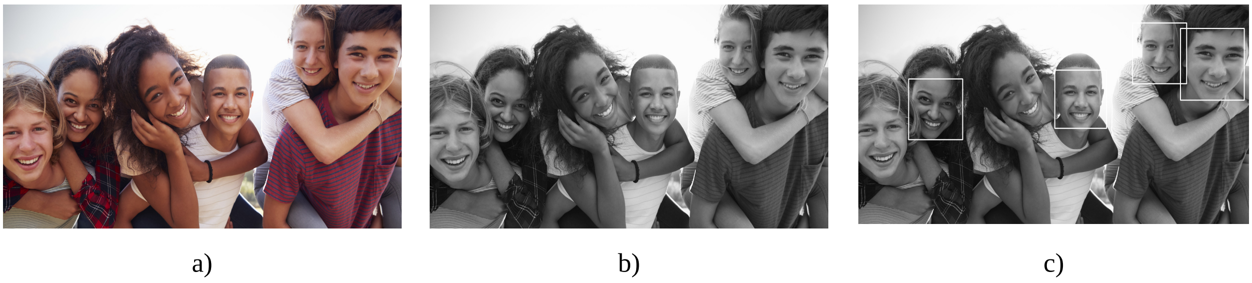
faces = face_cascade.detectMultiScale(input_image, scale_factor, min_neighbours) (2)

The results were then hand checked by opening each image, counting the faces in the image, then counting detected, incorrect and undetected faces. A check-sum and validation checks were then used to validate that the inputted results are correct. The original, best and worst results can be seen in the Figures 1 and 2 (a total from both images was used to determine the best and worst result). Figure 1 shows the results from the first image, Figure 2 shows the results from the second image.



a) b) c)

Figure 1. Image 1 used in experiments: a) original, b) worst, c) best



a) b) c)

Figure 1. Image 2 in experiments: a) original, b) worst, c) best

The results of the experiments can be seen in Tables 1, 2, 3 and 4.

Table 1. Results from the experiments (because no faces were recognized incorrectly the column is omitted)

Experiment no.	Minimum neighbors	Scale Factor	Faces (image 1, 2)	Correct (image 1, 2)	Unrecognized (image 1, 2)	Total faces	Total Correct	Total incorrect
1	3	1.30	(4, 6)	(4, 4)	(0, 2)	10	8	2
2	3	1.38	(4, 6)	(2, 3)	(2, 3)	10	5	5
3	3	1.56	(4, 6)	(3, 3)	(1, 3)	10	6	4
4	3	1.72	(4, 6)	(2, 2)	(2, 4)	10	4	6
5	3	1.81	(4, 6)	(0, 3)	(4, 3)	10	3	7
6	3	2.52	(4, 6)	(3, 4)	(1, 2)	10	7	3
7	3	2.53	(4, 6)	(3, 4)	(1, 2)	10	7	3
8	3	2.54	(4, 6)	(2, 3)	(2, 3)	10	5	5
9	3	2.85	(4, 6)	(2, 4)	(2, 2)	10	6	4
10	3	2.96	(4, 6)	(2, 1)	(2, 5)	10	3	7
11	5	1.30	(4, 6)	(3, 4)	(1, 2)	10	7	3
12	5	1.38	(4, 6)	(2, 3)	(2, 3)	10	5	5
13	5	1.56	(4, 6)	(2, 3)	(2, 3)	10	5	5
14	5	1.72	(4, 6)	(2, 1)	(2, 5)	10	3	7
15	5	1.81	(4, 6)	(0, 3)	(4, 3)	10	3	7
16	5	2.52	(4, 6)	(3, 3)	(1, 3)	10	6	4
17	5	2.53	(4, 6)	(3, 3)	(1, 3)	10	6	4
18	5	2.54	(4, 6)	(2, 3)	(2, 3)	10	5	5
19	5	2.85	(4, 6)	(1, 4)	(3, 2)	10	5	5
20	5	2.96	(4, 6)	(2, 0)	(2, 6)	10	2	8
21	6	1.30	(4, 6)	(3, 4)	(1, 2)	10	7	3
22	6	1.38	(4, 6)	(2, 3)	(2, 3)	10	5	5
23	6	1.56	(4, 6)	(2, 3)	(2, 3)	10	5	5
24	6	1.72	(4, 6)	(2, 0)	(2, 6)	10	2	8
25	6	1.81	(4, 6)	(0, 3)	(4, 3)	10	3	7
26	6	2.52	(4, 6)	(2, 3)	(2, 3)	10	5	5
27	6	2.53	(4, 6)	(2, 3)	(2, 3)	10	5	5
28	6	2.54	(4, 6)	(2, 3)	(2, 3)	10	5	5

Experiment no.	**Minimum neighbors**	**Scale Factor**	**Faces (image 1, 2)**	**Correct (image 1, 2)**	**Unrecognized (image 1, 2)**	**Total faces**	**Total Correct**	**Total incorrect**
29	6	2.85	(4, 6)	(1, 3)	(3, 3)	10	4	6
30	6	2.96	(4, 6)	(2, 0)	(2, 6)	10	2	8
31	8	1.30	(4, 6)	(2, 3)	(2, 3)	10	5	5
32	8	1.38	(4, 6)	(2, 3)	(2, 3)	10	5	5
33	8	1.56	(4, 6)	(1, 2)	(3, 4)	10	3	7
34	8	1.72	(4, 6)	(0, 0)	(4, 6)	10	0	10
35	8	1.81	(4, 6)	(0, 3)	(4, 3)	10	3	7
36	8	2.52	(4, 6)	(2, 3)	(2, 3)	10	5	5
37	8	2.53	(4, 6)	(1, 2)	(3, 4)	10	3	7
38	8	2.54	(4, 6)	(1, 2)	(3, 4)	10	3	7
39	8	2.85	(4, 6)	(1, 2)	(3, 4)	10	3	7
40	8	2.96	(4, 6)	(0, 0)	(4, 6)	10	0	10
41	9	1.30	(4, 6)	(2, 3)	(2, 3)	10	5	5
42	9	1.38	(4, 6)	(2, 2)	(2, 4)	10	4	6
43	9	1.56	(4, 6)	(1, 1)	(3, 5)	10	2	8
44	9	1.72	(4, 6)	(0, 0)	(4, 6)	10	0	10
45	9	1.81	(4, 6)	(0, 2)	(4, 4)	10	2	8
46	9	2.52	(4, 6)	(2, 3)	(2, 3)	10	5	5
47	9	2.53	(4, 6)	(1, 2)	(3, 4)	10	3	7
48	9	2.54	(4, 6)	(1, 2)	(3, 4)	10	3	7
49	9	2.85	(4, 6)	(1, 0)	(3, 6)	10	1	9
50	9	2.96	(4, 6)	(0, 0)	(4, 6)	10	0	10
Average:	6.2	2.12	(4, 6)	(1.62, 2.36)	(2.38, 3.64)	10	3.98	6.02
Standard deviation:	2.157096	0.6	(0, 0)	(1.01, 1.27)	(1.01, 1.27)	0	1.96	1.96

The tables are also visualized as figures as seen in Figure 2, 3 and figure 4.

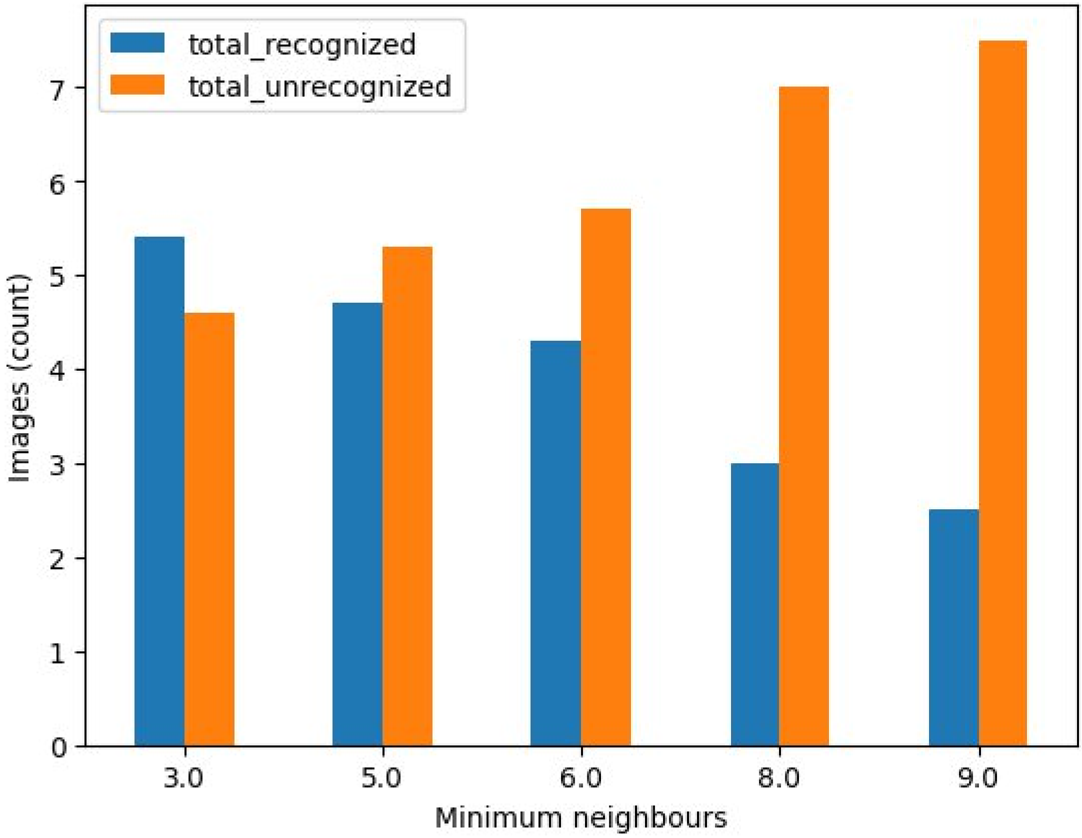


Figure 2. Recognition dependence on minimum neighbors. As we can see there is a clear correlation where less minimum neighbors lead to better results.

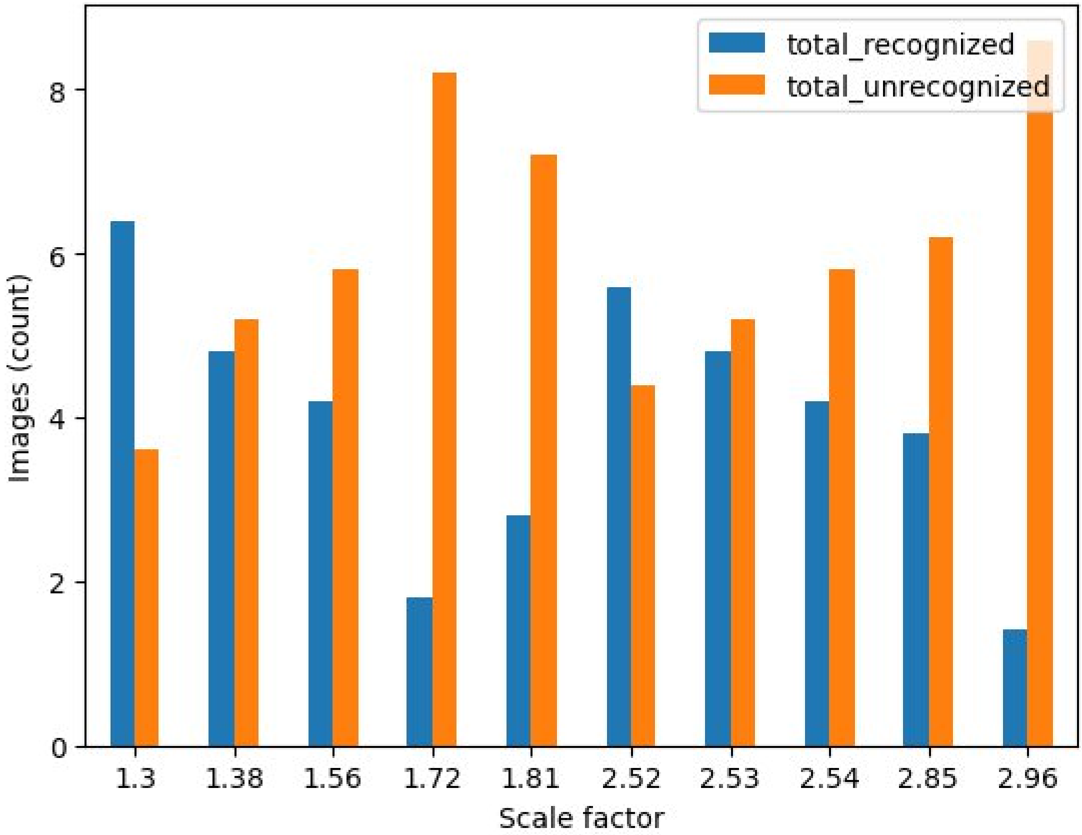


Figure 3. Recognition dependence on scale factor. No clear patterns can be seen, although the results seem to follow a sinusoidal pattern.

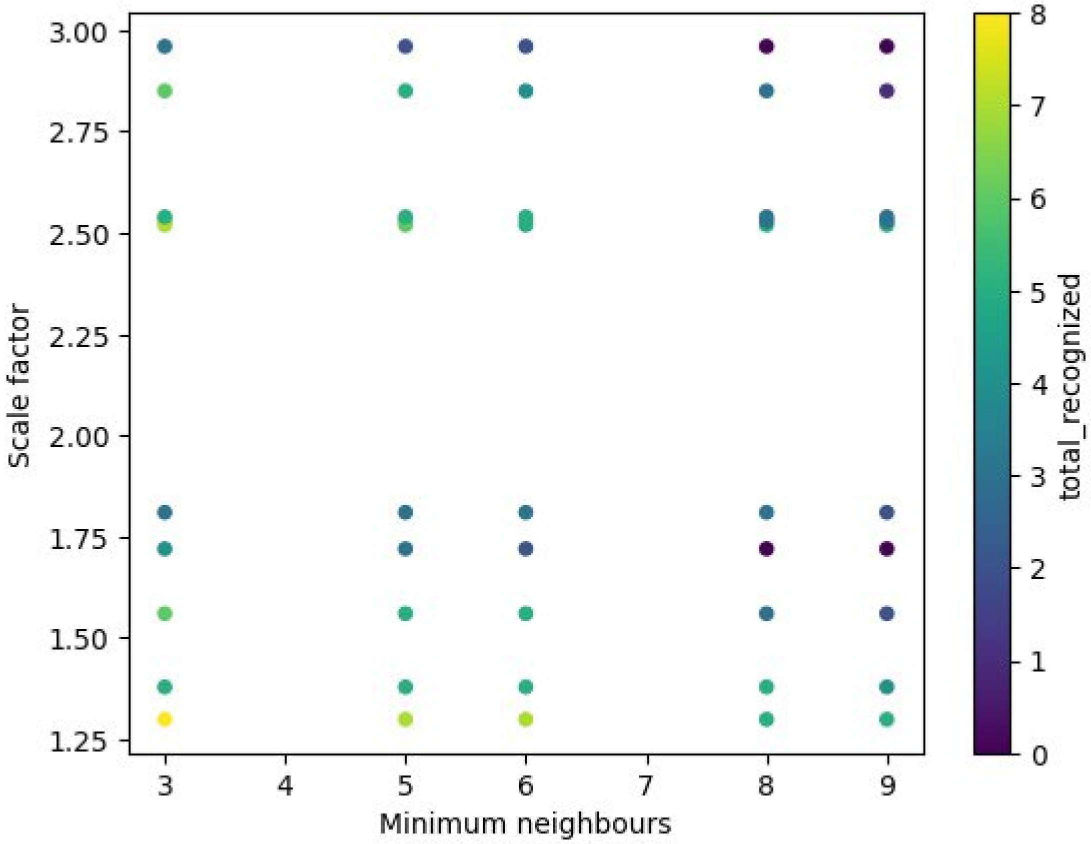


Figure 4. Scatter-plot showing the relationship between minimum neighbors, scale factor and total recognized faces. This scatter-plot seems to suggest that values in the lower left corner (less scale factor and minimum neighbors) leads to better results, although no clear correlation can be seen.

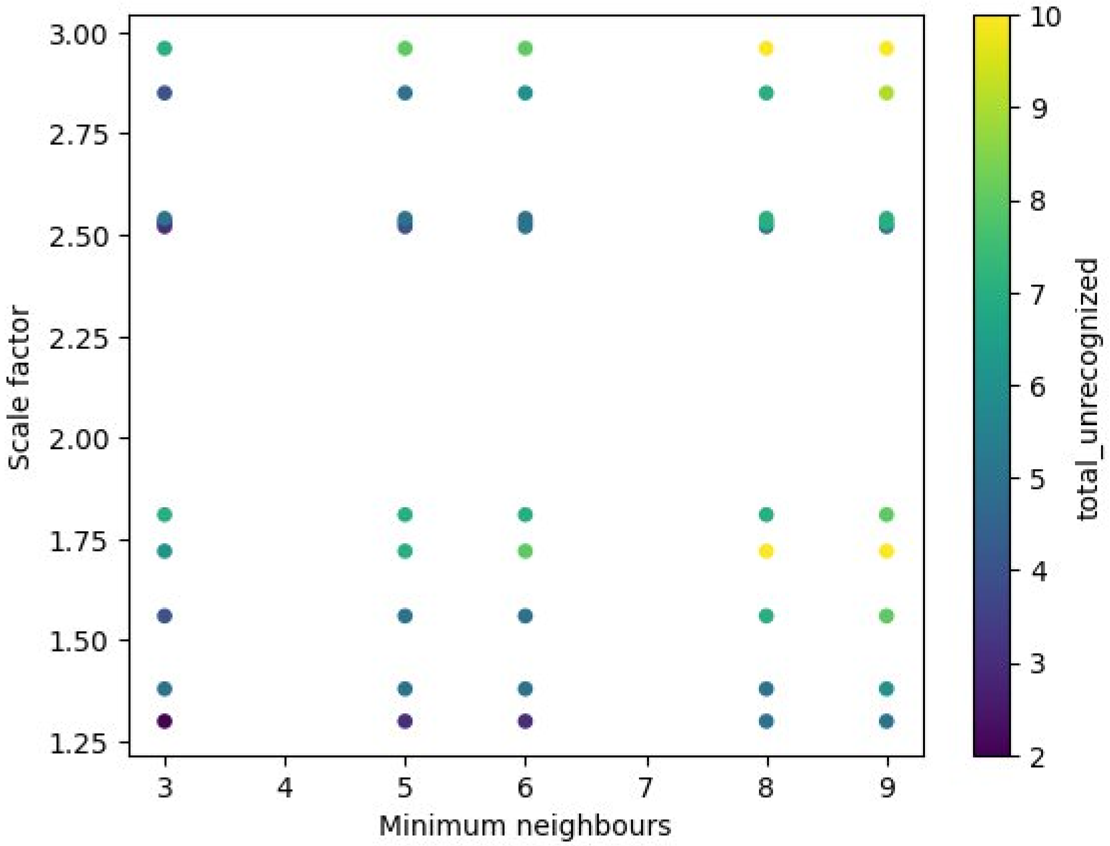


Figure 5. Scatter-plot showing the relationship between minimum neighbors, scale factor and total unrecognized faces. An inverse relationship to Figure 4 can be seen.

1.2 Recognition in video

The parameters from best and worst results from the previous task were then used in a video format from my laptops camera. The results followed closely to the ones described above where the parameters that performed best in recognition in images, also performed best in recognition in video and the other way around.

2 RESULTS AND CONSLUSIONS

Patterns can be seen emerging from the variation in the minimum neighbors and scale factor parameters for the Haar cascades face recognition classifier.

2.1 Results

1. The best performing scale factor and minimum neighbors were 1.30 and 3 respectively with 80% of faces being correctly classified.
2. No faces were detected incorrectly with any chosen scale factor or minimum neighbors parameter.
3. The worst performing scale factor and minimum neighbor parameters detected 0% of the faces present.
4. There is a clear correlation where less minimum neighbors lead to better results.
5. Image 2 never had 100% of faces recognized with any of the chosen parameters. The best result for it was 66.(6)%.
6. On average around 40% of faces were recognized, thus with unoptimized parameters these are the results we could expect while using this classifier.

2.2 Conclusions

1. With parameter optimization we can expect on average 40% better results.
2. Trying lower minimum neighbors and scale factor values are likely to lead to better results.
3. It is unreasonable to expect 100% accuracy from the classifier when classifying 4 or more faces at once.

3 REFERENCES

[1] opencv-python (n.d.) OpenCV Team. Retrieved from https://pypi.org/project/opencv-python/