Biometrics: assignment 2

Ingmar Malfait r0693397

Question 1:

Q1: Now that we have some results, is the given similarity function a good metric to quantify the distance between two fingerprints? Is it reliable enough to incriminate a suspect? What are its limitations?

A1: This method is quite limited, not only does it not take into account any transformation nor destruction. It will also consider the background pixels to be data and match these as well. Also the images themselves can have different contrasts and brightness and this will also not be normalised thus creating a distorted view of reality. Scars, dirt, wettnes and any other form of alteration of the fingerprint will not be taken into account. This L0 approach is too limited to be used as real evidence. A better approach would be to use features such as minutiae, pores or features found in orientation fields.

Question 2:

Q2: Are all the keypoint matches accurate? Are they expected to be? Explain why.

A2: The keypoints for these two random fingerprints are of different people so what is expected is that not many keypoints will match. However many keypoints (100 matches) do match as can be seen on the visualisation this is an undesired result. Secondly, us humans can also see quite quickly that the points that are not straight horizontal lines are not the same keypoints on both images even if they were of the same person. These keypoints are not accurate. This is because local similarity metrics only really take into account local metrics and thus matching becomes inaccurate when also global features such as transformation, position, rotation etc… have to be taken into account. In short, it does not align and is thus not accurate because it doesn't take into account position and transformation.

Question 3:

Q3: Choose a global feature similarity function, (e.g. you can start from euclidean distance between the reduced sets of KeyPoints and count the values above a threshold).

A3: Use the scipy.spatial.distance.euclidean function as described in the documentation and count the values for which the result is higher than a choses threshold.

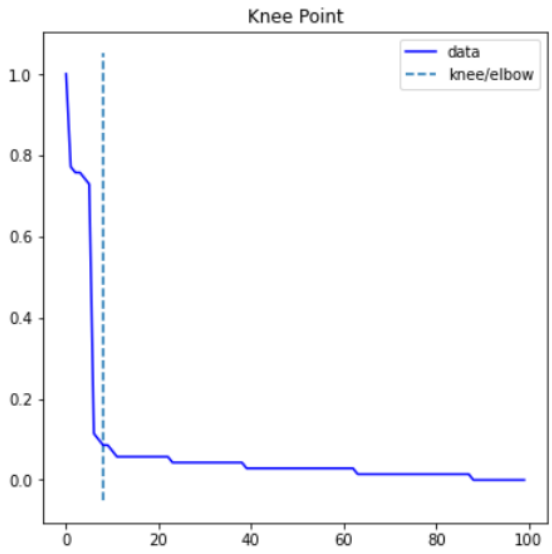
```
for match in matches:
dist = distance.euclidean(reg_kp1[match.queryIdx].pt, kp2[match.trainIdx].pt)
  if dist < threshold:
            amount += 1
```

Question 4:

Q4: Visualise the scores and determine a score threshold to discriminate the matching fingerprints. Explain how you determine the threshold.

A4: The goal of a threshold to have one or a limited amount of fingerprints to stand out. A threshold that is too small such as 0.1 will cause sample 56 and 49 to have the same value of 2 and the lower ranking fingerprints to all have 0. A value that is too large such as 100 will cause the same fingerprints as well as 48 to have similar values. Testing reveals that a threshold **value of 4** will give a distinctly higher result for 56 while still having different values for the lower ranking fingerprints. Even with a good value of 4 56,49,48,18,76 and 22 still have similar scores as compared to the lower ranking fingerprints. Something fishy is going on. The knee is at 8 and knee threshold at 0.0857… As you can see on the image it is not quite clear yet who is the perpetrator.

rank	id	score
1	56	72
2	22	56
3	76	55
4	48	55
5	49	54
6	18	53
7	40	10
8	68	9
9	13	8
10	66	8



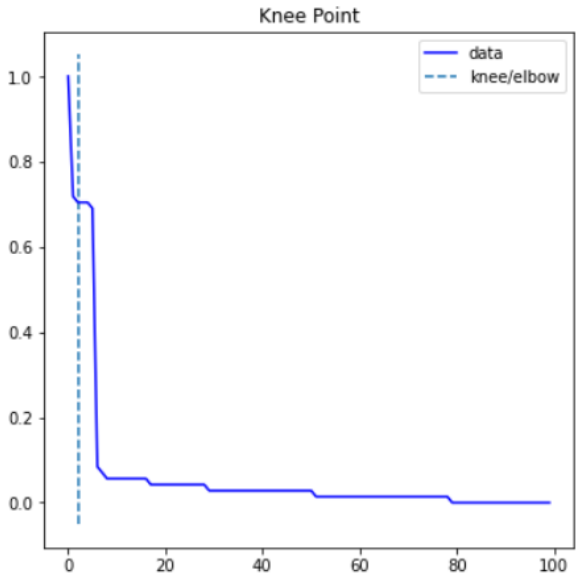
Question 5:

Q5: Choose a hybrid feature similarity function that makes use of both the geometric distance and the feature distance of the keypoints. Using this hybrid function, visualize and assess the matches.

A5: Instead of sorting for match.distance another threshold is used and measured in the same way as in question 4. This new threshold does change the shape of the knee point but only after about 6-8 images, to not squize the bottom bunch or top bunch of the scores a threshold of 80 is used. The knee is now at 2 and at a threshold of 0.704 but looking to the image the cutoff does not seem justified as the little platform at 0.7 still has about 5 more samples. Again, very fishy results.

```
if dist < threshold_eu and match.distance < threshold_fe:
            amount += 1
```

rank	id	score
1	56	71
2	22	51
3	76	50
4	48	50
5	49	50
6	18	49
7	40	6
8	68	5
9	13	4
10	66	4



Question 6:

Q6: Check out iris_perpetrator.png. Where do you see difficulties? What kind of similarity measures do you expect to work best?

A6: Many of the same issues arise for the Iris scan, the background will still be analysed, contrast needs to be normalised so it doesn't influence the metrics, transformation and rotation need to be applied etc… One more unique issue might be that the pupil also needs to be considered background. Besides this also the Iris lines are not that visible and might be hard to recognize for the ORB implementation. Some filtering is needed. For this particular image also not all zones seem to be present as one of the eyelids covers a large part of the top of the iris.

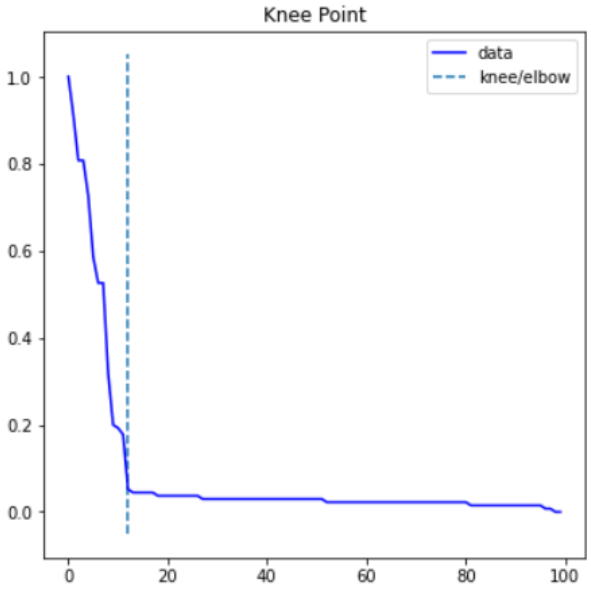
Question 7:

Q7: Construct a similarity table with the iris images. Use local OR global matches in order to get scores. Use the scores you get to determine the perpetrator.

A7: Global metrics were used and again no clear perpetrator was found, however the image labels are different this time around. 36,63,91,93,25,10,88 and 3 all have quite high scores. We assume that the IDs are the same for both finger and iris scans. If this is the case not only is there no clear perpetrator there is also no clear corresponding id.

The knee is also not great at 12 with a threshold of 0.0519.

rank	id	score
1	36	135
2	63	123
3	91	109
4	93	109
5	25	98
6	10	79
7	88	71
8	3	71
9	34	43
10	56	27

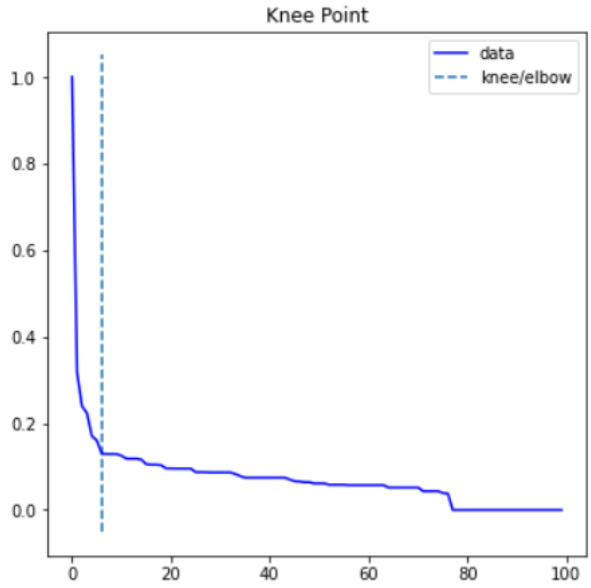


Question 8:

Q8: Fuse your iris and fingerprint biometric system on the score level to solve the murder case! Do you feel confident in your prediction? How do you fuse the scores? Why?

A8: Fusion can be on many different levels: Sensor, Feature, Score and Decision Fusion. For this example only the latter two are options as the sensor data is already acquired and feature fusion for two very different metrics is not very useful. So scores will be fused. Firstly an average can be taken but this doesn't result in a clear perpetrator either. To not have one metric overrule the other firstly both need to be normalised. afterwards the formula of (score_1*score_2)/(score_1+score_2) can be used this way the denominator makes sure that even if one metric is much more powerful that it won't overrule the other. When using this a clear perpetrator comes forward, id nr 56. This id can also be found in the previous tables and thus we can confidently say that they are the perpetrator.

rank	id	score
1	56	0.167
2	63	0.0531
3	25	0.0399
4	34	0.373
5	76	0.285
6	88	0.267
7	13	0.217
8	48	0.215
9	49	0.215
10	18	0.215



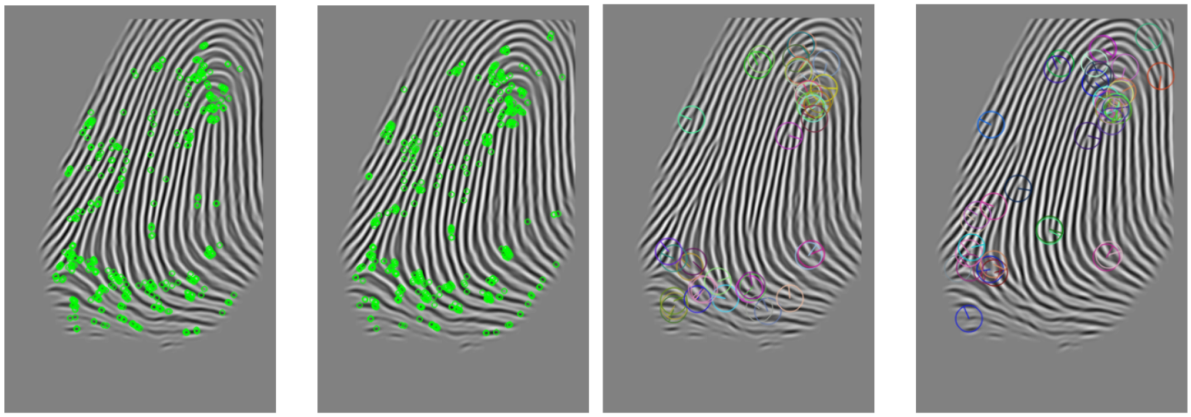
Part V: A

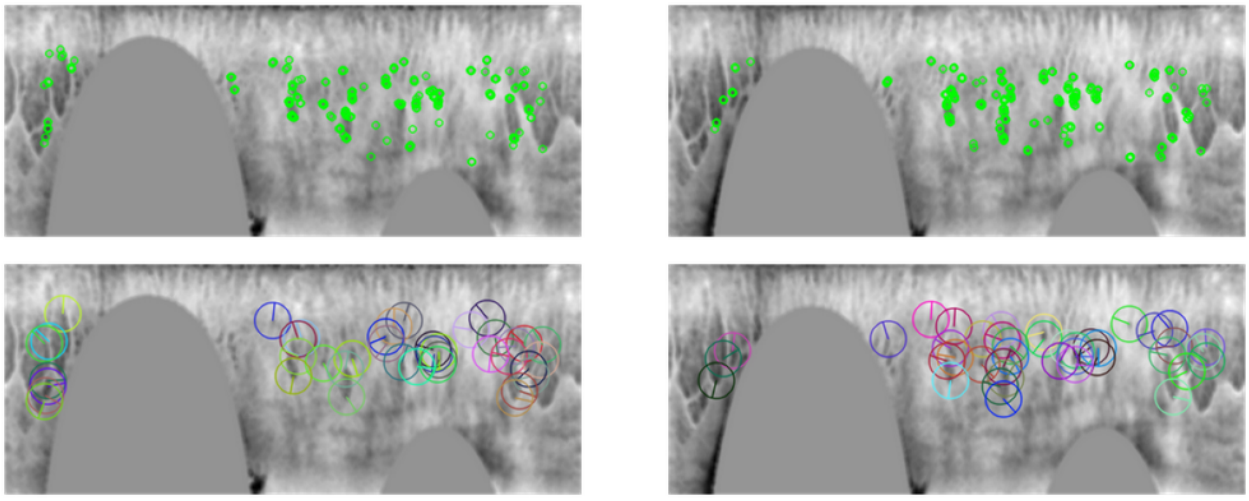
Q: [FING and IRIS] OpenCV provides different KeyPoint detectors and descriptors (ORB, SIFT, SURF, BRIEF, ...). Briefly test, visually, which of these seem to extract relatively reliable interesting points from the fingerprints or iris dataset (you can skip the ones that require a lisence). Use at least 3 detectors. (1pt)

SIFT and SURF both require licences and downgrading did not work after many attempts thats why the following three detectors were used:

ORB (baseline)

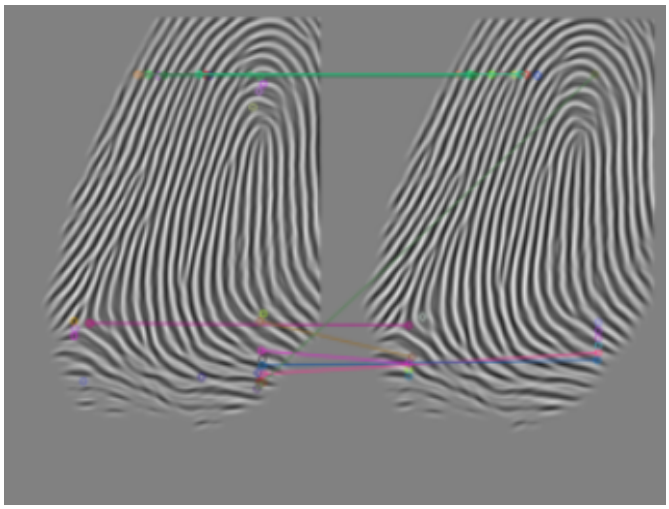
Baseline of comparison is ORB, here are the ORB keypoints (all kps) and descriptor results (top 40 kps)

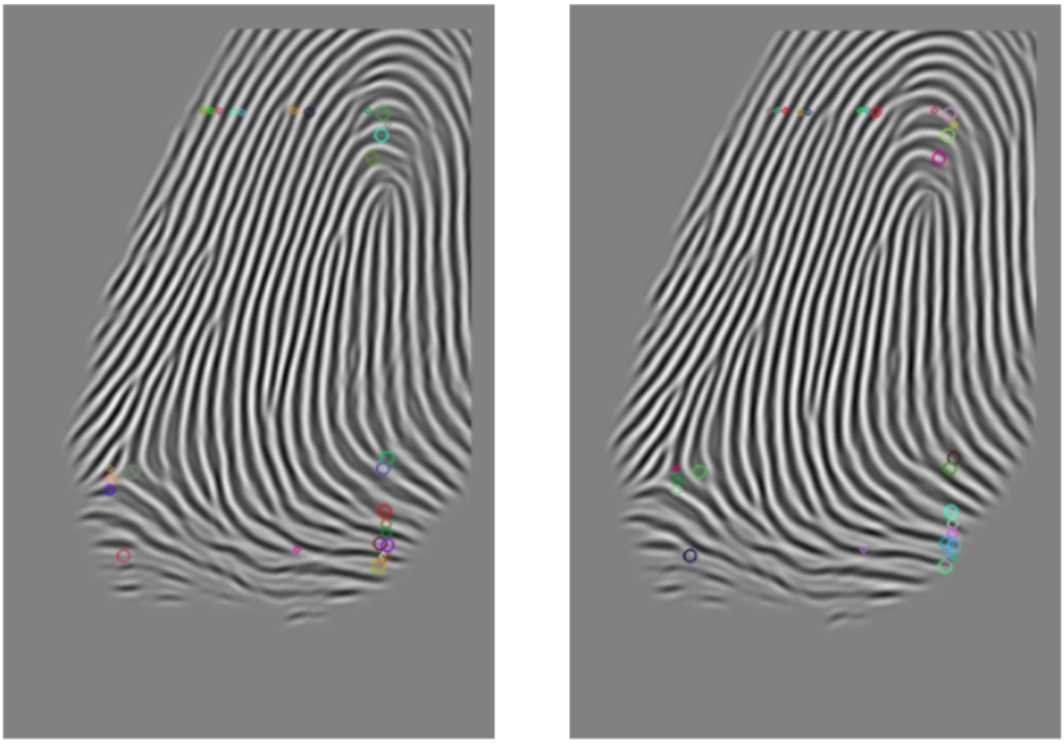


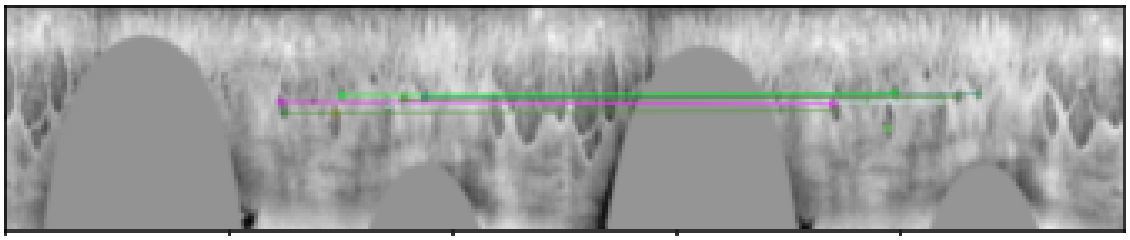


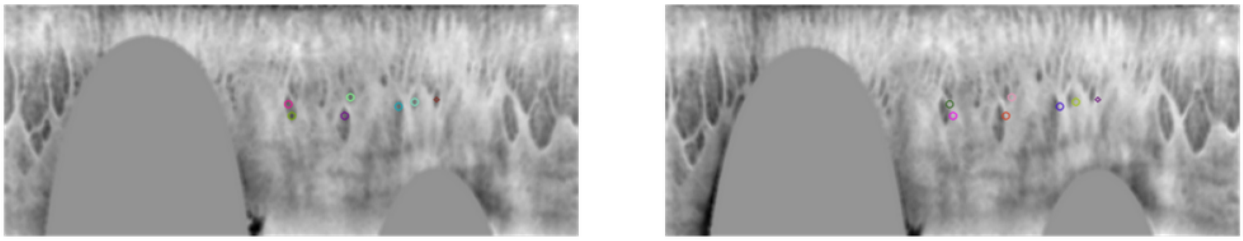
BRIEF and STAR

This detector descriptor couple Detects fewer key points in general as can be seen on the image below, the descriptors have a much smaller amplitude compared to ORB. Connecting these with BF matcher also shows promising results with (fairly) accurate connections.



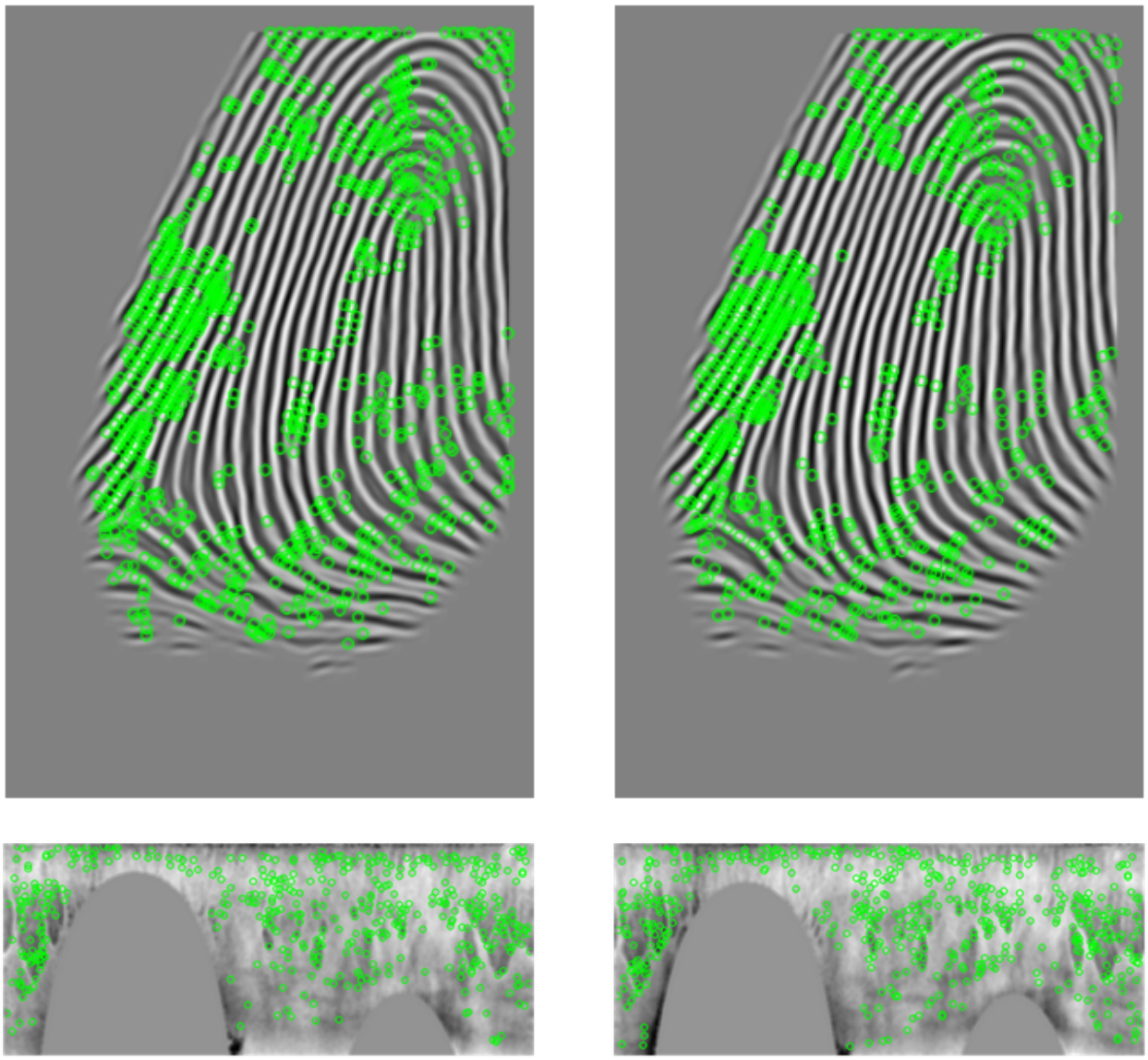






FAST

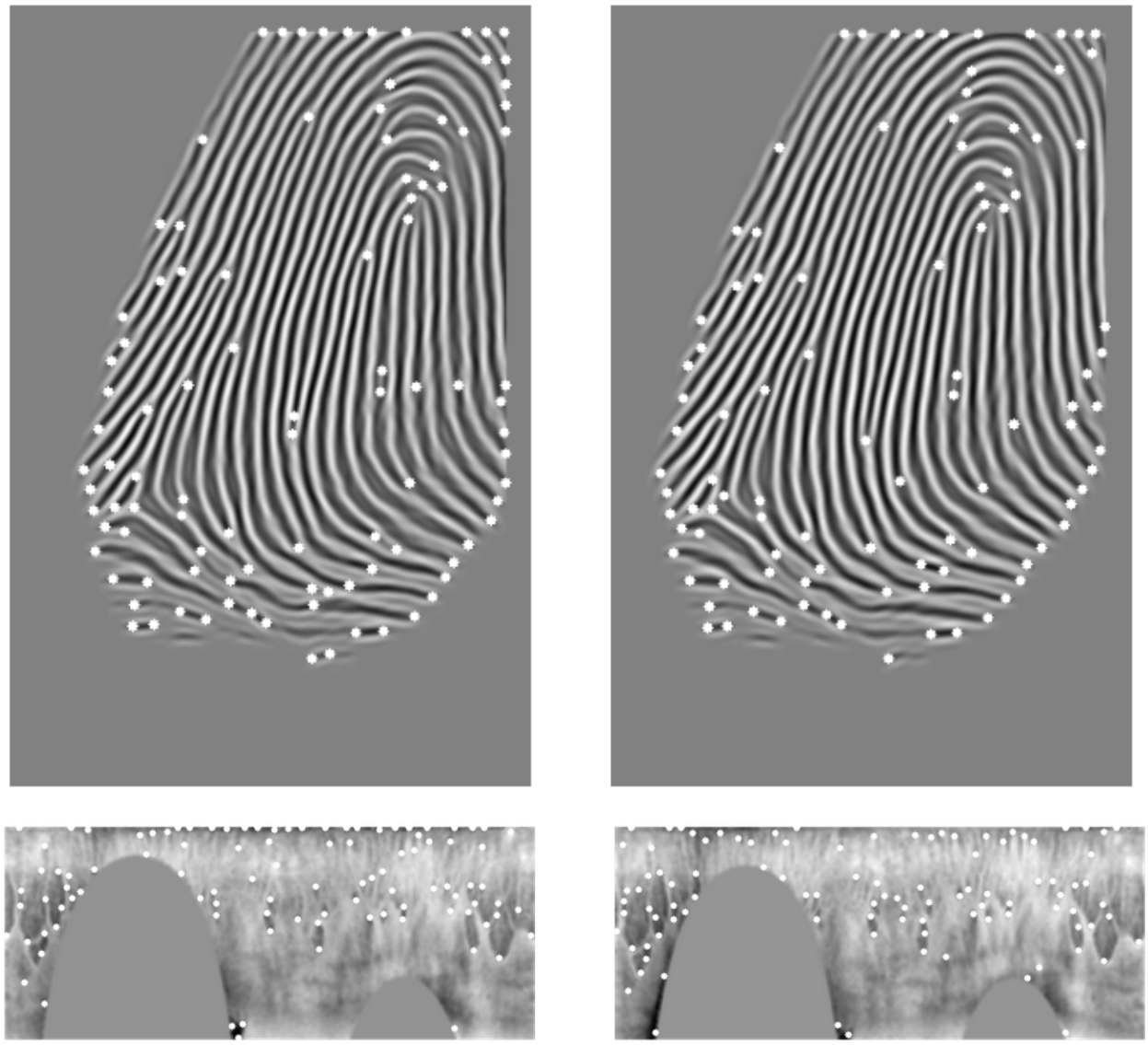
Fast seems to detect many the same keypoints as ORB but has difficulty with edges, this is hardly surprising as ORB uses both Oriented FAST and rotated BRIEF. Without BRIEF weak key points such as around the artificial edges from cutting off the top of the finger print appear. This appears to be less of an issue when looking at the keypoints of the iris scan.



Shi-Tomasi

Shi-Tomasi is good at corner detection and not much more, Minutiae such as branches and valley ends are detected quite well but not much more is detected besides some artificial minutiae at the edges of the fingerprint that should not be taken into account.

Iris characteristics show many of the same problems, corners are detected with many around the edges of the images.



Part V: B

Q: [IRIS] Play around with the gabor filter bank, what is the effect of the parameters? Can you setup a databank that results in better features? (1pt)

A:
Up to a certain point a sharper filtered image might help recognize some features however an image that is too sharp might also see details that do not matter to the feature recognition and slow down the software unnecessarily.
The parameters that matter most are as follows:

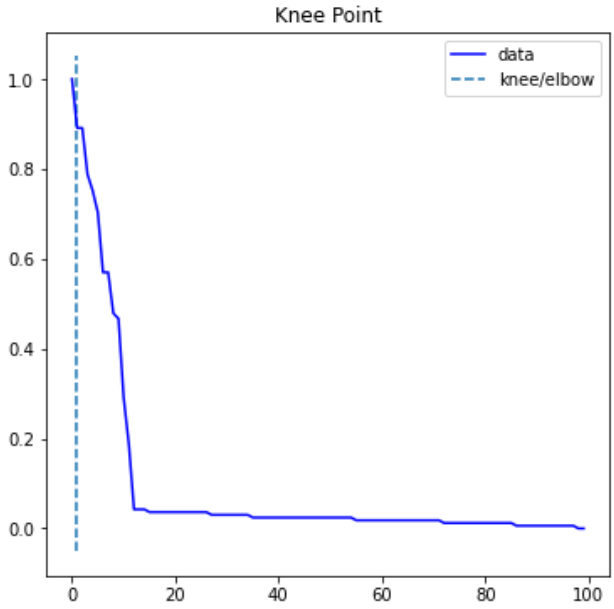
- sigma = standard deviation of the gaussian part of the gabor filter. A smaller sigma will result in a sharper image but it might have less clear patterns (like valleys)

- theta_range = the different orientations for the cosine part of the gabor functions. A range with smaller steps will result in more orientations and feature lines that are smoother. But steps that are too small might result is performance issues.
- lambda = the wavelength and thus 1/frequency of the sinusoidal part of the gabor function. A smaller wavelength will result in a sharper image but might also slow down the software.

Some changes are made to the parameters and the perpetrator image is looked at. Changing the sigma to a smaller value increases the sharpness of the image. The same is true for a smaller step of the theta_range. A lower wavelength also does the same.

Adjusting these parameters to be sigma = 0.1, theta_range = np.arange(0,np.pi/16,np.pi/16) and lambd=6, makes for a better performing database. A more distinct first value and a better knee of 1 and a threshold of 0.891.

rank	id	score
1	36	166
2	91	148
3	93	148
4	63	131
5	25	125
6	10	117
7	3	95
8	88	95
9	34	80
10	62	78

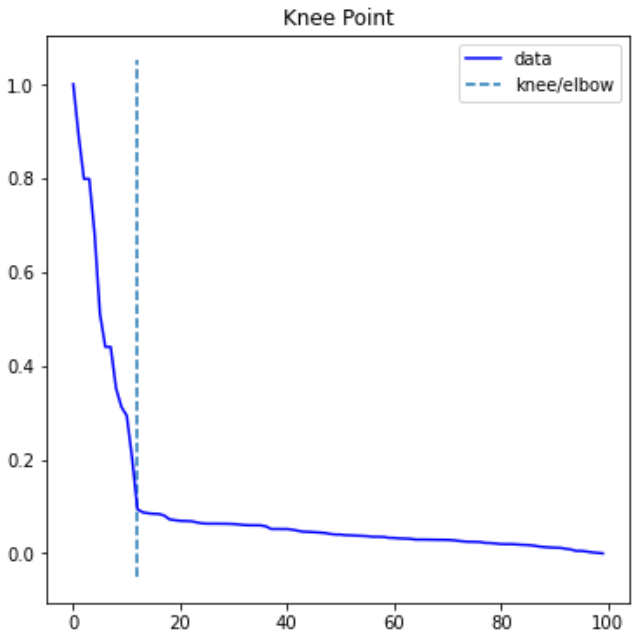


Part V: D

Q: D. [IRIS] Compare local hybrid and global features for the iris data (1pt).

In question seven the global iris results were discussed. Local results have a similar id ranking but have a totally different score range. as can be seen below. This is because without transformation many key points might very well still indicate they are the same but this is only slight as they do not match perfectly. When transformation is taken into acconu this changes and values skyrocket from 0.0673 all the way to 135.

rank	id	score
1	36	0.0673
2	63	0.0614
3	91	0.0563
4	93	0.0563
5	25	0.0498
6	10	0.0406
7	3	0.0368
8	88	0.0368
9	56	0.0320
10	34	0.0299



Part V: G

Q: [FING or IRIS] Attempt to improve the segmentation, evaluate. (3pt)

A: A third party segmentation library was used (https://github.com/CarlosCujcuj/Fingerprint-Segmentation). Due to time constraints not much could be tested but results do show similar segmentation quality with also color for connected ridges. This can be useful in feature extracting as these different intensities might also give rise to easy recognizable corner detection.