COMPX301 Vision Project

Report by Jedd Lupoy (1536884) and Jeffrey Luo (1535901)

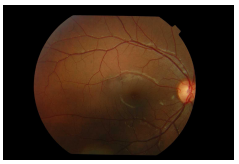
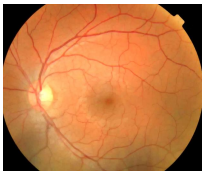
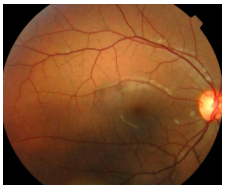
Introduction

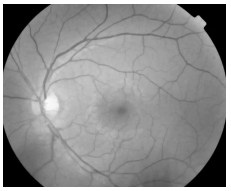
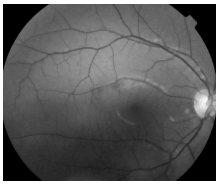
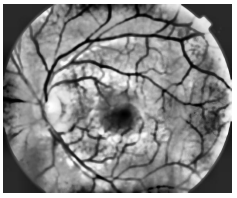
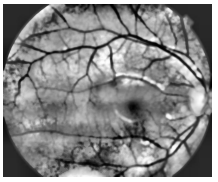
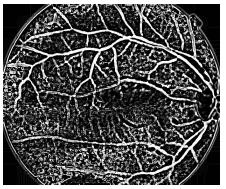
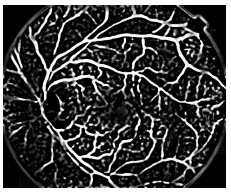
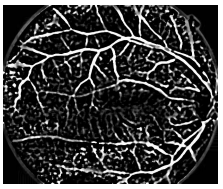
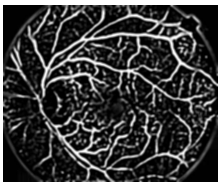
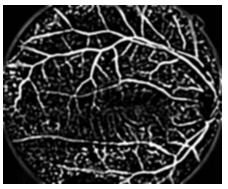
Retinal images are useful for person identification because the patterns of veins on the image are quite unique to an individual. Thus, the aim of this project is to produce a Java program, using the OpenCV library, to match two retinal images and decide if they come from the same person or not. If they are the same person and match, the program prints 1, otherwise 0. The images were retrieved from the RIDB dataset and each image was named the following way, IM00000[image#]_[person#].JPG. Hence, the solution ensures that pairs of images from the same person match and pairs of images from different people do not match. The following report documents the computer vision pipeline and image comparison testing.

Documentation

Image Pipeline

Due to images being of different quality (varying in colour, contrast and noise etc.), the images cannot be readily compared without applying filters. The pipeline was implemented so that it applies a series of filters to the image. As veins distinguish the individual, the aim of the pipeline was to reduce the image to retain as many veins as possible in the form of black lines with as little noise as possible. The computer can then readily compare the veins/images in a binarised way, black - 1, and white - 0. The table below shows a typical image and a darker image being processed, IM000002_2 and IM000003_7, respectively.

Filters applied in the order listed	**2_2 Image Pipeline Processing**	**3_7 Image Pipeline Processing**
No filter (original)		
Add resizing, cropping and opening Resizing speeds up the image processing. Cropping puts more focus on the retinal scan, removing unnecessary black background. Opening filter reduces some of the noise/artefacts from resizing.		
Add sharpening Increased the image's brightness as well as highlighting the retinal veins to prepare for auto contrasting.		

Add grayscale Grayscale colour is required for the auto contrast processing.		
Add gaussian blur and CLAHE autocontrast Gaussian blur to make the veins better defined for auto contrast. The CLAHE auto contrast significantly highlights all the veins in the image.		
Add laplacian Laplacian edge detection removes the unnecessary background information and only shows the retinal veins.		
Add median blur This blur minimises the noise that was picked up from laplacian edge detection.		
Add gaussian blur Another run of gaussian blur to soften the edges of the veins before thresholding.		
Add thresholding Further removing any more noise from the image and inverts the colour of the image.		
Add closing Closing filter will give a better visualisation of the veins as they get more connected together and not separated.		

Initial Filters Used

- Equalise Hist Filter: This is the non-adaptive version of the CLAHE auto-contrast filter. The resulting image showed less key vein lines and were thinner than using the actual CLAHE auto contrast filter.
- Canny Filter: This is an alternative edge detection filter. The resulting image made the vein lines seem more grainy and choppy than using the laplacian edge detection filter.

Comparing Images

The main method used for comparing two images was template matching. This involves splitting image 1 into a 4x4 grid of smaller image squares, each square being a template. The warp affine OpenCV method was used to auto-crop the template to match better. Each template was subsequently matched with image

2. The TM_CCOEFF_NORMED property was used to give the max value of how much the template is contained in image 2. If the value is over 0.45, then they match. It was decided that at least three template squares needed to match in order for the images to be from the same person, outputting 1, otherwise 0.

<table>
<tr><th>Image Comparison</th><th>Image 1</th><th>Image 2</th></tr>
<tr><td rowspan="2">Images 1_2 and 2_2 demonstrate typical lighting conditions for retinal vein matching. Both images are from the same person and it shows from the similar-looking vein lines. From the console, the debug messages announced it found ten matches in different sections of the images and therefore concluded that these images are indeed from the same person.</td><td> 
1_2</td><td> 
2_2</td></tr>
<tr><td colspan="2">jl615@lab-rg06-09:~/Documents/COMPX301/A4$ java RetinalMatch IM000001_2.jpg IM000002_2.jpg
Processing complete!
Processing complete!
0.4843665361404419
Matched!
0.8248766660690308
Matched!
0.7587177753448486
Matched!
0.6428731083869934
Matched!
0.6349618434906006
Matched!
0.638404655168152
Matched!
0.55064827203750 61
Matched!
0.77407526969900 67
Matched!
0.7326955795288086
Matched!
0.5588515400886536
Matched!
OFFICIAL MATCH!</td></tr>
<tr><td rowspan="2">Images 1_7 and 3_7 demonstrate what happens when attempting to match retinal veins in non-ideal lighting conditions (i.e. dark sections of image making it hard to see the veins). Another layer of difficulty is that the position in which 1_7 was scanned is different compared to 3_7, but the comparison is smart enough to match sections of the 3_7 image to different positions of 1_7 and attempts to identify the best overall match possible. From the console, the debug messages announced it found three matches and concluded that these images are from the same person.</td><td> 
1_7</td><td> 
3_7</td></tr>
<tr><td colspan="2">jl615@lab-rg06-09:~/Documents/COMPX301/A4$ java RetinalMatch IM000001_7.jpg IM000003_7.jpg
Processing complete!
Processing complete!
0.4699815809726715
Matched!
0.6669208407402039
Matched!
0.4992300271987915
Matched!
OFFICIAL MATCH!</td></tr>
<tr><td rowspan="2">Images 1_1 and 2_2 are examples of non-matching retinal veins. These two images have typical lighting conditions which made processed images clear that they should not match. After testing, the debug messages do not indicate any match in any section of the images and therefore these images are not from the same person.</td><td> 
1_1</td><td> 
2_2</td></tr>
<tr><td colspan="2">jl615@lab-rg06-09:~/Documents/COMPX301/A4$ java RetinalMatch IM000001_1.jpg IM000002_2.
Processing complete!
Processing complete!</td></tr>
</table>

Conclusion

Overall, the condition for the two images to be from the same person or fully match is that at least three cut-out/template squares of one image each (smartly) match the other image with at least 45% similarity. This decision was made because the condition could be met when comparing two images from the same person in the worst-case scenario, i.e., with image 1_7 and 3_7. The scenario included comparing images in non-ideal lighting conditions and were scanned at different positions of the retina. Pairs of images that were not from the same person were tested and all did not meet the condition to fully match. Thus, the program was quite accurate. However, the algorithm can still be improved in the future to provide clearer vein lines and better image comparison where the 45% threshold should be increased.