CS 763/764: Computer Vision

Answers B

Question 1. Take new snapshots of each of your faces, and add these to the database.
Solution. The faces corresponding to Low, Middle and High (sorted in terms of roll numbers), have been added to the **/replicate/database_image**. The images have been cropped such that only the face and some part of hair is visible in the images.

Question 2. It may be difficult to spot a masked intruder. To make this task easier, we will allow you to calibrate. Take a picture **maskLow** of**Low** with mask on. Using this image and the database of images, obtain a reference retrieval score with which you can be confident that you are able to report a true positive, and minimize false positives. Here's an idea for calibration. Suppose we normalize the match score from 0 to 1, with 1 being a certainty. If we set the reference score as 0.01, given **maskLow**, we will say "yes, the intruder is in the database" because the matcher will almost surely return a score higher than 0.01 with some random person in the database. False positives value is very large. If we set the reference score as 0.99, then because of the mask, even if the intruder **Low** (one of the group members) is present in the database, there may not be a match. Pick your score but stick to it. Store all freshly captured images in **replicate/capturedImages**.
Solution. The calibration is done by setting the retrieval score corresponding to the **maskLow.jpg** and **Low.jpg**. The obtained reference retrieval score is **0.298**, where a score of 1.0 represents a certainty. With respect to this reference score, if we find a score higher than the reference corresponding to some **maskMiddle?.jpg**, then there is a high chance that the intruder might be present in the database.

Question 3. Now comes the interesting part to answer the question: How much masking is allowed? We work with a new second image of **Middle** which is going to be variable. The goal is to optimize on the masking. The second set of images, named **maskMiddle?.png**, should capture image of the same group member with a mask" defined as follows. (Ideally we want to use bigger and bigger mouth-nose-forehead masks but since we don't want to keep cutting the mask, we will use a paper to simulate this).
Solution. We used different extent of masking for the three **maskMiddle?.jpg** intruder images. We observed that the image with maximum masking (**maskMiddle1.jpg**), gave a score below the reference value set using **maskLow**, we know this because we created a set of possible intruder images which had a score above the reference value, and then crosschecked if **Middle** was present in that list or not. If it was not in the list, it meant the score corresponding to the intruder was less than the reference. We then tested with **maskMiddle2.jpg**, which also gave a score less than the reference for the aforementioned reasons. Finally, we used the image with the lowest masking (maskMiddle3.jpg), which gave a score greater than the reference, which shows that among the list of possible intruders having scores higher than the reference, we have our intruder. The diameter of the hole in the third case was approximately **7.5cm**.

Question 4. Try the experiment with Low (i.e., a somewhat contrived experiment, since you have already calibrated with Low), if you get unsatifactory results working with **Middle**.
Solution. We did experiment with **Low.jpg** because during our initial experimentation the **maskLow.jpg** had a very large hole size. Thus the reference was set way above to find any potential intruders. Thus in the second experiment we changed the **Low.jpg** and **maskLow.jpg** pair and obtained satisfactory results.