DeepTeeth Implementation and Result Analysis

Ruchir Jain

March 24, 2022

1. Introduction

This report discusses the implementation of a new teeth-based authentication system for mobile devices called DeepTeeth [1]. DeepTeeth offers a stands apart from other such systems as it doesn't involve expensive scanning machines such as X-ray machines. This allows it to be used extensively by anyone with a standard mobile device and a functioning camera unit. It is also resistant towards spoof attacks which are common in face-based identification systems.
For the purpose of developing the algorithm, an image database was created. 10 images of teeth were taken per subject in various lighting conditions and in 2 sessions. $n * n$ matching was done on the dataset using popular feature extractors and their matching scores were calculated. Using these scores, the genuine/imposter histogram was plotted for each feature extractor. By varying the threshold, ROC curve was plotted. Based on the ROC curve, EER and CRR were calculated as well.
Ablation study has also been performed by changing the various parameters while calculating the matching score. The results of these study were used to find the best suited parameters for each method used.

2. System Pipeline

The general recognition pipeline includes RoI extraction, image enhancement, feature extraction and matching. The matching scores are then used to plot genuine/imposter curve. The ROC is plotted by varying the thresholds and calculating FAR and FRR for each threshold. The EER is calculated by taking the point on the ROC curve where the FRR and FAR are equal (difference is minimum while dealing with discrete values). The maximum accuracy is also calculated while varying the thresholds. CRR is calculated using the EER threshold.

2.1 ROI Extraction and Image Enhancement

The images were taken using the front-camera of mobile devices. Manual cropping was performed on every image and the new image was saved with the same name. For enhancing the images, the images were first converted to grayscale and histogram equalization was performed on each image before feature extraction. Histogram equalization enhances the contrast in the images and this makes them more suitable for feature extraction. Figure 1 shows one of the images created by following the above method.

2.2 Feature Extraction and Matching Score Calculation

For the purpose of comparison, various feature extractors have been used to detect descriptors in the given images. Some of these include SIFT, ORB and AKAZE. These methods are used to get the local image features for each image. These feature extractor vectors are compared for 2 images and the matches are computed using Brute Force algorithm, which compares each descriptor of 1st image with those of the second image. The extracted features using SIFT can be seen in Figure 2.
To filter the matches, Lowe's ratio test is performed to try to eliminate ambiguous matches. Ambiguous matches include those matches where the distance between two nearest neighbours is close to 1. The distance ratio is calculated for 2 nearest matches and the match is considered as good if it is below some threshold ratio. The well discriminated matches were taken to compute the matching score.
Two other matching scores have also been calculated. SSIM (structural similarity index measure) has been used for measuring the similarity between two images. Unlike the other feature extractors, SSIM looks for similarities within pixels; if the pixels in the two images line up and or have similar pixel density values. It incorporates perceived

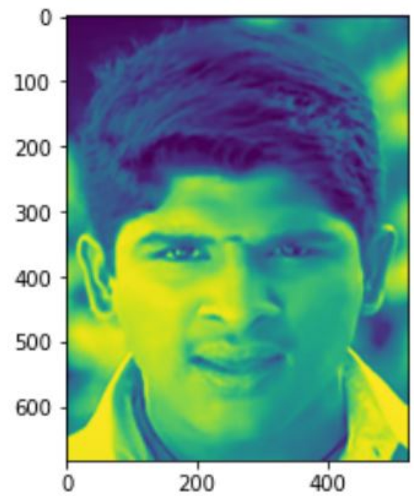

Figure 1: A teeth image created by doing image enhancement

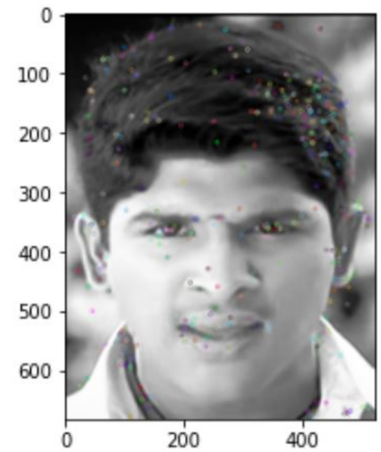

Figure 2: Extracted keypoints in a teeth image using SIFT

change in structural information, along with luminance and contrast. ArcFace is also used to compute the feature vector for images. After computing this, cosine distance is calculated for the two feature vectors corresponding to the two images.

2.3 Plotting Genuine-Imposter Histogram

The images in the database have been divided based on the subject and sessions for the same subject. A match is considered as genuine match if it is for the same subject, i.e., different session image of the same person. If the two subjects are different, the match is considered as imposter match.
After performing the n*n matching, the number of imposter matches are significantly higher than the genuine matches. Hence, probability has been plotted instead of frequency. Figure 3 depicts the histogram of scores obtained using SIFT for a subset of the dataset (one-sixth of the images taken and n*n matching has been performed).

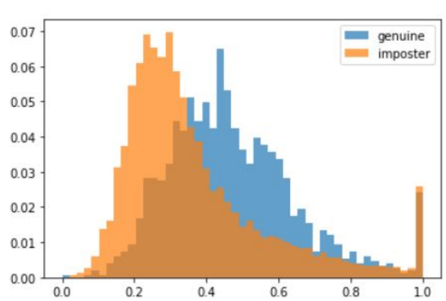

Figure 3: Histogram using SIFT

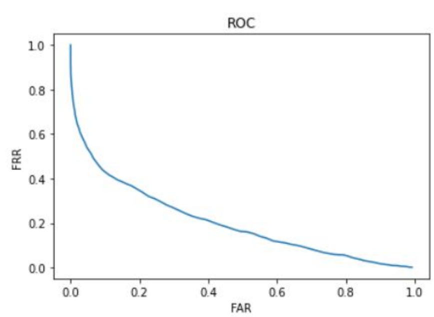

Figure 4: ROC using SIFT

2.4 Plotting ROC curve and Calculating EER

By varying the threshold, the False Acceptance Rate (FAR) and False Rejection Rate (FRR) have been computed using the following formulae:

$$FAR = \frac{FP}{FP + TN} = \frac{Number\,of\,incorrect\,matches\,recognized}{Total\,number\,of\,imposter\,attempts} \tag{1}$$

$$FRR = \frac{FN}{FN + TP} = \frac{Number\,of\,correct\,matches\,not\,recognized}{Total\,number\,of\,genuine\,attempts} \tag{2}$$

```
Threshold is: 0.25 FAR: 0.4128905345984726 FRR: 0.2040268456375839      DIFF: 0.2088636889608887
Threshold is: 0.255 FAR: 0.39305716269382085 FRR: 0.21543624161073827      DIFF: 0.17762092108308258
Threshold is: 0.26 FAR: 0.37428835917611664 FRR: 0.2214765100671141      DIFF: 0.15281184910900253
Threshold is: 0.265 FAR: 0.35484841471881506 FRR: 0.2302013422818792      DIFF: 0.12464707243693587
Threshold is: 0.27 FAR: 0.33617218236519325 FRR: 0.24161073825503357      DIFF: 0.09456144411015968
Threshold is: 0.275 FAR: 0.3167322379078917 FRR: 0.2557046979865772      DIFF: 0.06102753992131449
Threshold is: 0.28 FAR: 0.29708400833140475 FRR: 0.2697986577181208      DIFF: 0.027285350613283965
Threshold is: 0.285 FAR: 0.27917148808146264 FRR: 0.28120805369127516      DIFF: 0.002036565609812524
Threshold is: 0.29 FAR: 0.2611201110853969 FRR: 0.2966442953020134      DIFF: 0.03552418421661652
Threshold is: 0.295 FAR: 0.24385558898403148 FRR: 0.3100671140939597      DIFF: 0.06621152510992823
Threshold is: 0.3 FAR: 0.22645221013654246 FRR: 0.3201342281879195      DIFF: 0.09368201805137702
Threshold is: 0.305 FAR: 0.20904883128905347 FRR: 0.338255033557047      DIFF: 0.12920620226799354
```

Figure 5: EER calculation

By taking the corresponding FRR on y axis and FAR on x axis, the ROC curve is plotted. Figure 4 shows the ROC curve for the corresponding histogram and FAR-FRR values in Figure 3.
The accuracy is also calculated while varying the threshold using the FAR and FRR values. The following formula was used for the same. The maximum value is taken as the accuracy of the system.

$$Accuracy = (1 - \frac{(FRR + FAR)}{2}) * 100 \tag{3}$$

EER is calculated while varying the threshold. The point on the ROC curve where the difference in FAR and FRR value is minimum (both should be equal theoretically) is taken as EER. The corresponding values of FAR and FRR are taken and the average of them is computed. EER is taken as this average value. The threshold corresponding to this point is taken as the EER threshold. Figure 5 shows a snapshot of the values computed and the minimum FAR and FRR difference is highlighted. The corresponding threshold value is taken as the EER threshold.

2.5 CRR Calculation

The value for EER threshold is taken and the corresponding confusion matrix is created using the genuine and imposter scores obtained earlier. CRR is calculated using the following formula:

$$CRR = \frac{TP + TN}{TP + TN + FP + FN} = \frac{Number\, of\, matches\, correctly\, recognized}{Total\, number\, of\, matches} \tag{4}$$

3. Results and Analysis

Upon following the above system pipeline for a number of feature extractors, the results have been depicted in the following table. None of the parameters have been tuned while calculating the results. The ROC curves of the

Feature-Extractor \Method used	EER Threshold	EER	CRR	Accuracy
SIFT	0.380	0.327053922703078	66.8993288590604	67.57926405924555
ORB	0.385	0.4364218941140319	56.5391742028264	59.62380956275566
BRISK	0.445	0.4524092441581048	54.86515219337511	59.75773653822766
AKAZE	0.405	0.3123473244391314	69.31566056764155	69.71037960634642
SSIM	0.285	0.28018977088636887	72.07606263982103	73.59407544549872
ArcFace	0.980	0.35900000000000000	68.47999999999999	65.50000000000001

Table 1: Results using various methods for matching score calculation (without tuning parameters)

above methods have been combined and shown together in a single plot, given in Figure 10. Histograms for a few of the methods have also been depicted in the following Figures.

As seen from area under the curve and accuracy, SSIM and AKAZE perform well even under no parameter tuning. As seen in the histogram in BRISK and ORB, there are a lot of 1.0 scores in both imposter and genuine. This is because the Lowe's ratio has been taken as 0.9, which is very high. Typically, the ratio is kept at 0.7. High ratio means more matches are considered as good. The reason why ratio is kept at 0.9 is that lowering this ratio has lead to very low values of matching scores. Without parameter tuning, the feature extractors fail to give features, to the point that sometimes, no features are returned by them. This is due to the fact that many of the images are

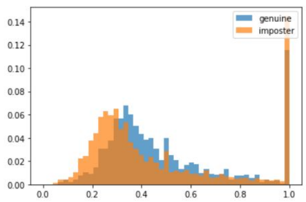

Figure 6: ORB

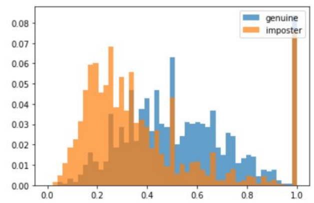

Figure 7: AKAZE

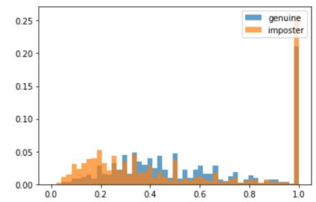

Figure 8: BRISK

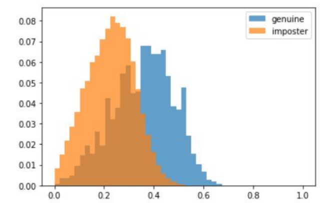

Figure 9: SSIM

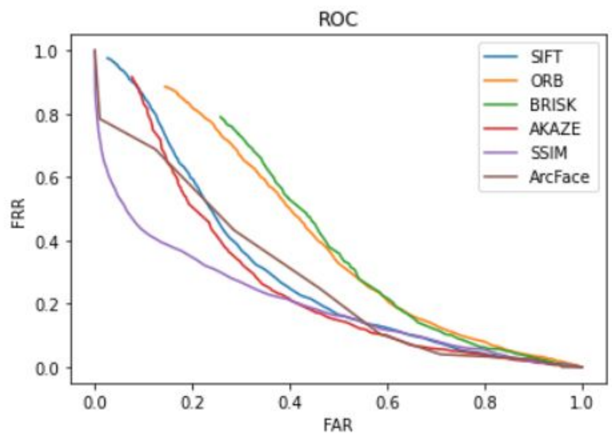

Figure 10: ROC curves combined

blurred and have not been removed from the dataset. If no features are returned, the only possible solution is to give 0 as matching score which is not a correct thing to do while calculating accuracy and comparing methods.
ArcFace doesn't give a smooth curve as it has been executed only for a subset of the dataset as it is very slow and executing it over the entire dataset would take a lot of time. The results aren't better than other feature extractors as it uses pre-trained weights and the parameters are set for facial feature extraction and not focused on teeth images.
Structural Similarity index (SSIM) works well due to the fact that the images have been taken at good camera angles and this leads to good structural information. If the images were taken in different lighting and contrast, along with multiple camera angles, SSIM would fail as it is not based on descriptors but on actual image data itself.

4. Ablation Study

In this section, various parameters and tasks have been changed to see the effect on the accuracy of the system.

4.1 Image Enhancement

ROI Extraction Manual cropping was performed on the dataset to extract region of interest in each image. This didn't lead to any noticable trend of change in the accuracies by the various methods used. This is mainly because the images in the dataset already have region of interest extracted to some extent due to the manner in which the images were captured. The subjects had to align their teeths in a rectangular region and this automatically cropped the image to extract the teeth part only.

Histogram Equalization Contrast in the images was boosted by the means of histogram equalization. This lead to an increase in the accuracies of most methods, as seen in the following Table 2. This also lead to lesser cases where no descriptors were present in the image, which explains the rise in accuracy. The number of descriptors extracted increased for most images.

Grayscale conversion The images were converted to grayscale. This didn't affect the accuracy much, but it substantially decreased the execution time. It can be explained in a way that now the amount of data to process was cut by a third due to 3 channel RGB image becoming single channel.

Feature-Extractor \Method used	Accuracy before Histo-Eq	Accuracy after
SIFT	64.37285611374211	67.57926405924555
ORB	55.10092348532676	59.62380956275566
BRISK	58.33419938319401	59.75773653822766
AKAZE	64.25644914395425	69.71037960634642
SSIM	73.8058086659257	73.59407544549872
ArcFace	63.75410260734991	65.50000000000001

Table 2: Results using Histogram equalization

4.2 SIFT

There are 3 main parameters that affected the performance of SIFT: $contrastThreshold$, $edgeThreshold$ and initial guassian blur ($sigma$). These values were varied in the following manner as shown in the Table 3 below ($nOctaveLayers$ was fixed at 3). The accuracy using combination given in the table was computed and the parameter values giving the highest values were selected.
This came out to be $CT = 0.033$, $Sigma = 2.4$ and $EdgeThr = 9.5$. The corresponding accuracy for this combination was 73.93991232382614, which is much higher than the initial method without parameter tuning.

Parameter	Default Value	Range in Analysis	Interval in Analysis
CT	0.03	0.01-0.05	0.001
Sigma	1.6	0.5-2.5	0.1
Edge Thr	10	5-15	0.5

Table 3: Varying parameters in SIFT

4.3 AKAZE

Similar to SIFT in above subsection, there are 3 main parameters that affect the performance of AKAZE: $threshold$, $nOctaves$ and $nOctaveLayers$. These values were varied in the following manner as shown in Table 4 below. The accuracy was computed and the parameters corresponding to maximum accuracy were stored.
This came out to be $threshold = 0.0001$, $nOctaves = 4$ and $nOctaveLayers = 3$. The corresponding accuracy for this combination was 77.54518112268782, which is significantly higher than the original accuracy.

Parameter	Default Value	Range in Analysis	Interval in Analysis
Threshold	0.0010	0-0.0015	0.0001
nOctaves	4	1-10	1
nOctaveLayers	4	1-10	1

Table 4: Varying parameters in AKAZE

4.4 Feature matching using FLANN

Instead of using the Brute Force matcher for feature matching, FLANN based matcher was used. This didn't affect the accuracies in a consistent manner. The accuracies rose for SIFT and AKAZE. However, the accuracies fell for ORB and BRISK.

References

[1] Geetika Arora, Rohit K Bharadwaj, and Kamlesh Tiwari. Deepteeth: A teeth-photo based human authentication system for mobile and hand-held devices. *arXiv preprint arXiv:2107.13217*, 2021.