Chapter 8

Application of λ_g^δ-closed sets in image processing

8.1 Introduction

A face recognition system is a computer application capable of identifying or verifying a person from a digital image or a video frame from a video source. In biometrics, face recognition plays a phenomenal role. Face recognition is broadly classified into face identification and face verification. Much research has been conducted on both face identification and face verification, with greater focus on the latter. Research on face identification has mostly focused on using under closed set protocol, which assumes that all images used in evaluation contain properties of images that are enrolled in the gallery. Real systems, however, where only a fraction of the image properties are enrolled in the gallery, cannot make this closed-set assumption. Instead, they must assume an open set of gallery images and be able to reject/ignore those that correspond to unknown properties. Face recognition has been widely applied in security system, credit-card verification, and criminal identifications, teleconference and so on.

In this work, we have developed an algorithm which can be used as a component of biometric analysis in various institutions. We adopt the scenario at an Educational institution, where the faculty members undergo biometric analysis(finger prints) to indicate their entry and exit into the institution. In our case, we are using the facial image of the faculty member instead of the finger print.

We broadly classify the algorithm into three parts. In the first part, the facial image of all the faculty members in the institution is captured and stored in a database. All these images are termed as training images. Next, we use SUSAN algorithm to exact the facial features of each person. The facial features are the feature values corresponding to a pair of eyes, nose and mouth of a person. After this process, we obtain a 4x2 matrix for each person. We reduce this matrix to a single value called the 'feature point' by calculating the mean of the feature values.

Second part begins when a faculty member presents himself before the capturing device to input his entry/exit. During this part, the current facial image of the person is captured, feature values of this person are extracted and feature point is computed in the foresaid manner. The image involved in this part is termed as a test image.

In the third part, the compare the feature points of the training images and the test image using three types of filters which are framed using the definition of Regular closed, δ-closed and λ_g^δ-closed sets. This comparison concludes if the faculty member is authenticated or not. Further the accuracy of these filters in the process of comparison are also determined. Finally the time consumed by each of these filters to compute and summarize the results is also presented. Among these protocols, λ_g^δ-closed based protocol gives better results when compared to the other two. The protocols are tested against accuracy and time consumption for attaining the results using MATLAB.

Using MATLAB, we have automated the procedure of finding the regular closed, δ-closed and λ_g^δ-closed sets.

8.2 Methodology

8.2.1 Facial Feature Extraction

In order to perceive and recognize human faces, we must extract the prominent features on the faces. Usually the features like eyes, nose and mouth together with their geometry distribution and the shape of face are used for this purpose. By applying human visual property in the recognition of faces we can identify face from very far distance, even if the details are vague. That means the symmetry characteristic is enough to be recognized. Human face consists of eyes, nose, mouth and chin etc., which are different in shape, size and structure. One can describe the organs with the shape and structure so as to recognize them. One common method is to extract the shape of the eyes, nose, mouth and chin, followed by distinguishing the faces by distance and scale of those organs. To automate this process of facial feature extraction, we use SUSAN algorithm.

Advantages of using SUSAN algorithm

SUSAN (Small univalue segment assimilating nucleus) operator was developed by Smith and Brady, Oxford University scholars. It uses gray-scale features of images which are applied in edge and corner detection. Simultaneously, we can remove the noises in the image using SUSAN.

SUSAN algorithm's computational speed is very fast. Compared with traditional algorithms, this method not only has strong ability to anti-noise characteristics, but also for a variety of shapes it can accurately detect the edge, and the measurement accuracy can basically reached the pixel level. Experimental results show that the method is suitable for the target edge detection of the ray image like aluminium wheels which have low contrast, large noise interference, and the detected object have irregular shape.

8.3 Working Strategy

The algorithm is broadly classified into two parts as follows:

8.3.1 Training

The facial feature obtained through SUSAN algorithm is in the form of a matrix. The following figure displays the extracted feature values of a person during training.

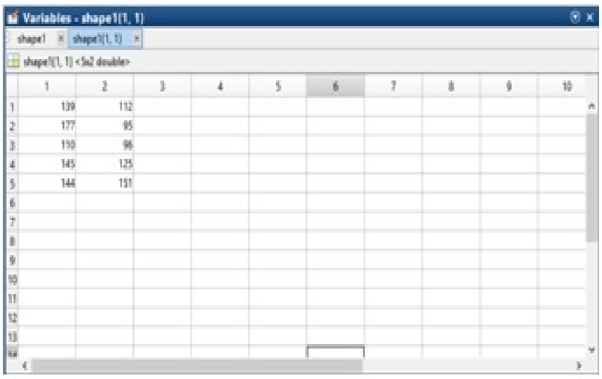
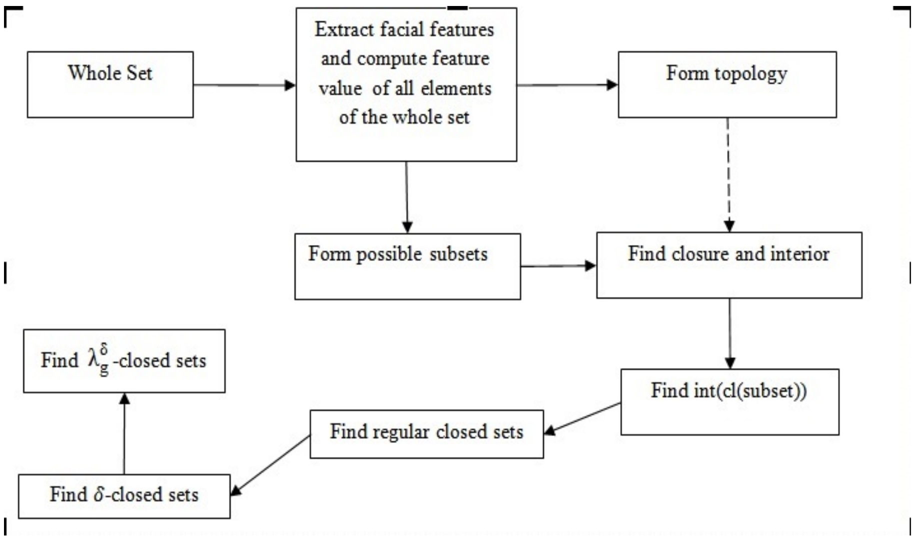


Figure 8.1: Extracted Feature Value of a person

From the extracted facial feature matrix, we extract the columns separately to form a new column vector which corresponds to the pixel value of each of the feature. We compute the mean of these feature values so that it contributes to a single value named as a feature point.

At the end of this procedure, we will have one feature point for each person. Further, these feature point takes the role of the elements of the set X, mentioned in the Example 1. This feature point contributes to the training as well as the test sessions.

Figure 8.2: Training Algorithm

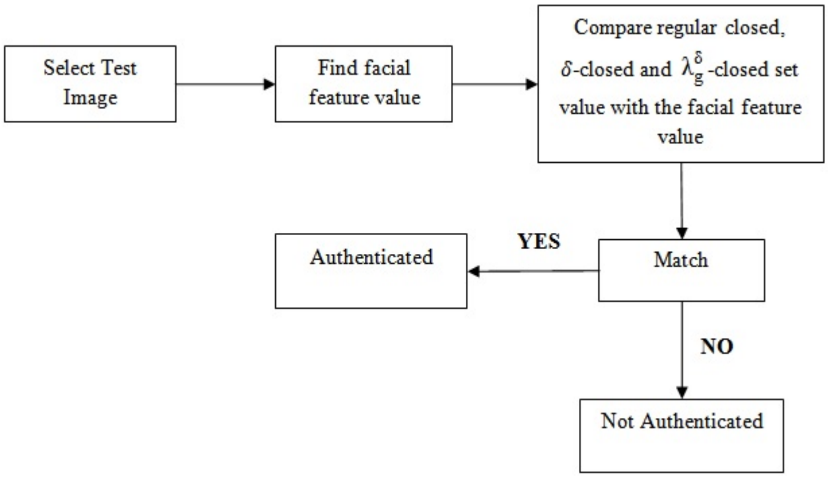


In this work, we have taken our whole set with the feature value of 10 images. The feature point remains unique for each image. Using this feature point, we check the nature of this value using the conditions of regular closed, δ-closed and λ_g^δ-closed sets.

8.3.2 Testing

In this part, the facial image of a faculty member is captured during his time of entry/exit using a capturing device. This image may or may not be a member of the database. We extract the facial features and compute the feature point in the foresaid manner.

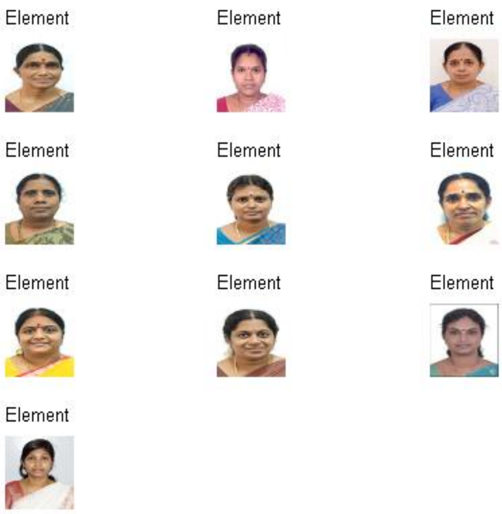
Figure 8.3: Testing Algorithm



8.4 Experimental results

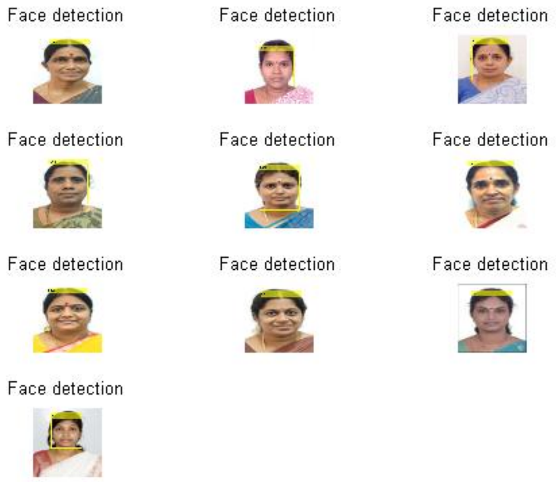
The following figure displays the images of the faculty members stored in the database.

Figure 8.4: Database Images



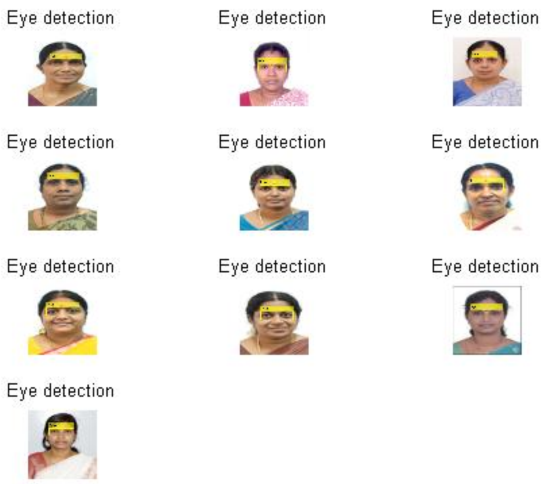
The following figure represents the training images with the faces being detected.

Figure 8.5: Face Detection



The following figure represents the training images with the eyes being detected.

Figure 8.6: Eye Detection

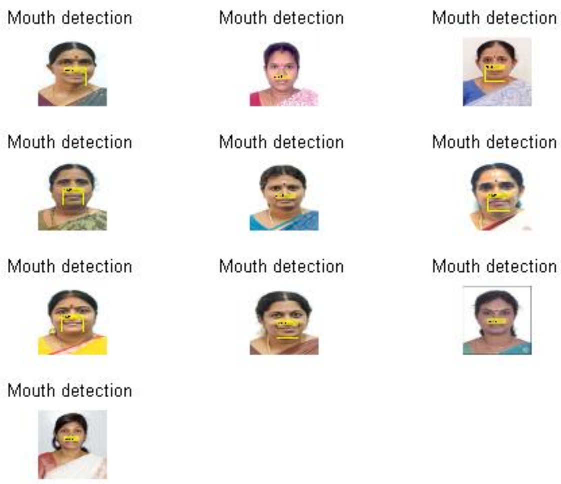


The following figure represents the training images with the nose being detected.

Figure 8.7: Nose Detection

The following figure represents the training images with the mouth being detected.

Figure 8.8: Mouth Detection



The following figure represents the training images with the facial features being extracted.

Figure 8.9: Facial Feature Extraction

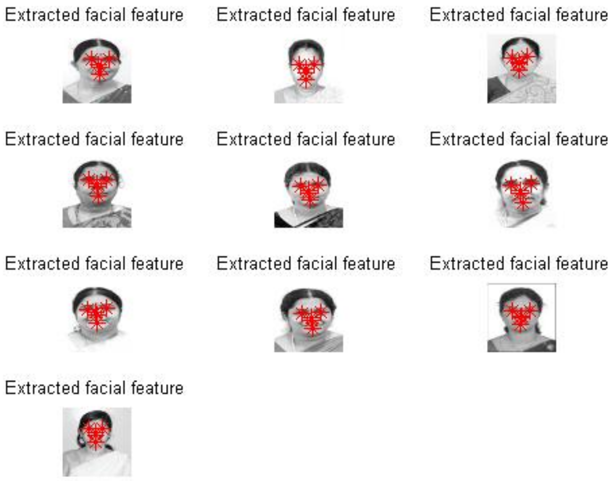


Figure 8.10: Test Image

Figure 8.11: Face Detection

Figure 8.12: Eyes Detection

Figure 8.13: Nose Detection

Figure 8.14: Mouth Detection

Figure 8.15: Facial Feature Extraction

8.5 Results

After comparison between the training and test images using the three protocols, their authenticity is displayed. If the protocol criteria matches, the test image is notified as authenticated under the corresponding protocol. If not, non-authenticity is stated under the corresponding protocol.

The accuracy of these results are analyzed using the true positive, true negative, false positive and false negative conditions when are usually used in image processing. The accuracy is calculated as follows:

$$\text{Accuracy}(\%) = \{(\text{True Positive} + \text{False Positive})/(\text{Total number of pages})\} \times 100$$

where True Positive = No. of images which are correctly analyzed as an member,
False Positive = No. of images which are correctly analyzed as a non-member.

Authentication Results

Figure 8.16:

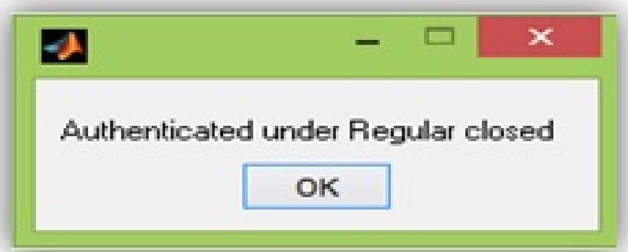


Figure 8.17:

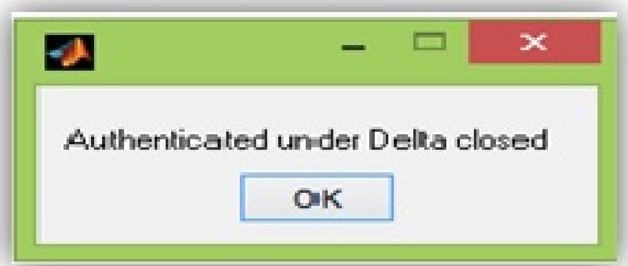
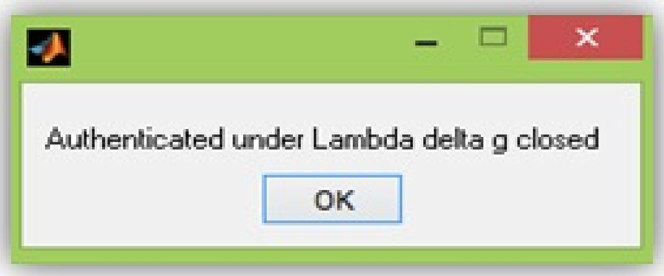
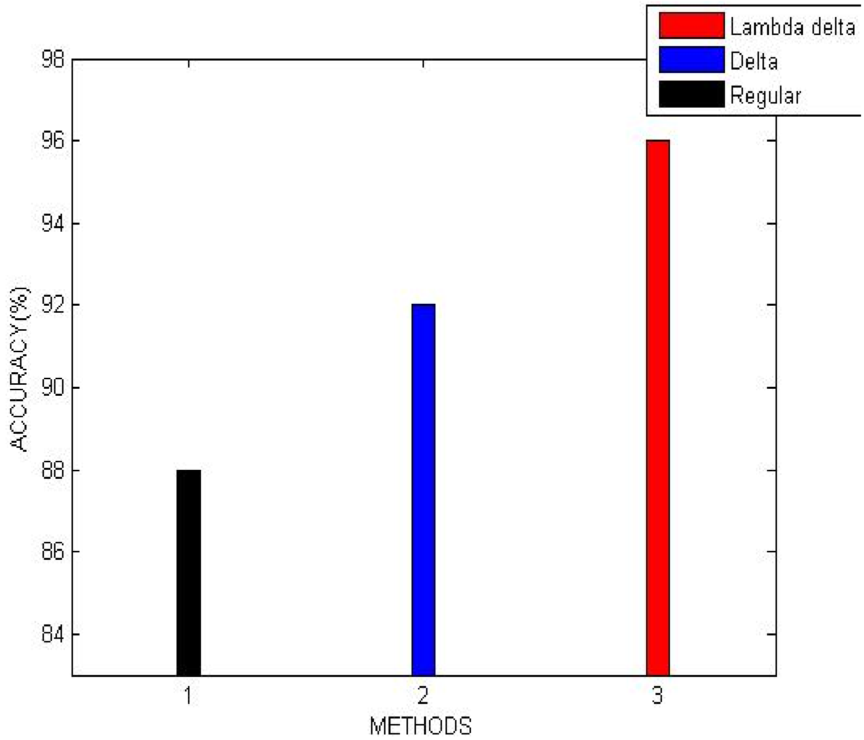


Figure 8.18:



Accuracy

Figure 8.19:

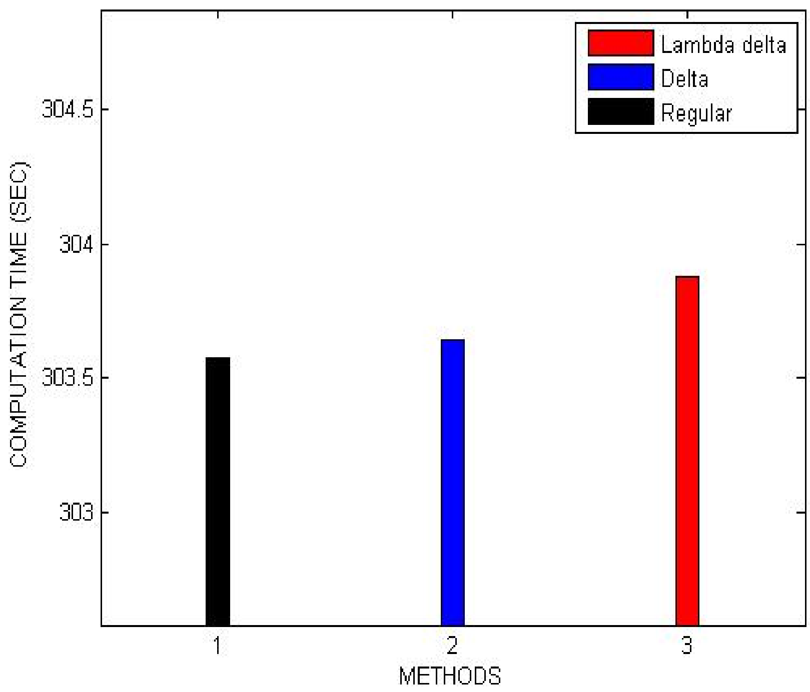


Under the accuracy conditions, λ_g^δ-closed set based protocol is superior than the other two protocols. The previous figure represents the comparative graph of the three protocols.

Since these three protocols involve the usage of various combinations of operators, it is essential to calculate the time consumed by MATLAB, to process these protocols. The time consumption is displayed in the following figure.

Consumption Time

Figure 8.20:



8.6 Conclusion

Although all three sets namely Regular closed, δ-closed and λ_g^δ-closed take almost same computation time and same results on authenticity, the accuracy for each of these sets are different. Interestingly λ_g^δ-closed sets are more accurate than Regular closed and δ-closed sets.