TWELVE ANCHOR POINTS DETECTION BY DIRECT POINT CALCULATION

*F. Khalid, **T. M. T. Sembok and ***K. Omar

**Department of Multimedia, Faculty of Computer Science and Information Technology*
Universiti Putra Malaysia, 43400 UPM Serdang, Selangor Darul Ehsan
***Department of Information Science, Faculty of Technology and Information Science*
****Department of Science and System Management, Faculty of Technology and Information*
University Kebangsaan Malaysia, 43600 UKM Bangi, Selangor Darul Ehsan

Email: fatimahk@fsktm.upm.edu.my, tmts@ftsm.ukm.my, ko@ftsm.ukm.my

ABSTRACT

Facial features can be categorized it into three approaches; Region Approaches, Anchor Point (Landmark) Approaches and Contour approaches. Generally, anchor points approach provide more accurate and consistent representation. For this reason, anchor points approach has been chose to utilize. Although, as the experiment data sets have become larger, algorithms have become more sophisticated even if the reported recognition rates are not as high as in some earlier works. This will cause a higher complexity and computer burden. Indirectly, it also will affect the time for real time face recognition systems. Here, it is proposed the approach of calculating the points directly from the text file to detect twelve anchor points (nose tip, mouth centre, right eye centre, left eye centre, upper nose and chin). In order to get the anchor points, points for the nose tip have to be detected first. Then the upper nose and face point is localized. Next step is chin point localization and followed by nose base points localization. Lastly, the outer and inner eyes corner is localized. An experiment has been carried out with 420 models taken from GavabDB in two positions with frontal view and variation of expressions and positions. Our results are compared with three researchers that is similar to and show that better result is obtained with a median error of the eight points is around 5.53 mm.

Keywords : 3D Face Recognition, 3D Face Detection

1.0 INTRODUCTION

The initial task in the entire face recognition system is to detect a face and locate the face area (if any) from a given facial scan. Definition 3.1 Face Detection. Given an arbitrary image, the goal of face detection is to determine whether or not there are any faces in the image and, if present, return the image location and extent of each face (Yang, 2002).

The detection of faces and facial features from an arbitrary is a critical stage. A robust and simple scheme is needed to detect the face as well as determine its precise placement to extract the relevant data from an input image. This is necessary to properly prepare the image description of the face for input to a recognition system. This detection scheme must operate flexibly the reliably regardless of lighting conditions, background clutter in the image, multiple faces in the image, as well as variations in face position, scale, pose and expression. In this research, variation of pose and expression will be highlighted.

Facial features can be classified into 3 categories, namely region, landmark (points), and contour. These features will be used as a representation of faces for recognition. Generally, landmarks provide more accurate and consistent representation for alignment purposes than region-based features and have lower complexity and computational burden than contour feature extraction (Lu, 2006). For this reason, landmark approach has been chose to utilize for this research.

2.0 RELATED WORK

Even though the experimental data sets have become larger, algorithms have become more sophisticated even if the reported recognition rates are not as high as in some earlier works. In this research, local geometric approach has been used. Studies example considering local geometric feature of 3D facial surfaces are fewer in number. This is due to the requirement of automatic segmentation of facial landmarks. The local shapes of facial landmarks/regions about fiducial points have been quantified. For instance, Gordon (1992) and Moreno et al. (2005) used Gaussian curvature values for segmenting the regions and lines of interest. Meanwhile, Xu et al. (2004) applied Guassian-Hermite moments technique for feature extraction. Chua et al. (2000) and Wang et al. (2002) employed 'point signatures' technique for registration where to find the transformation which optimally aligns the rigid regions of the two face surfaces. And Wang and Chua (2005) utilized 3D Gabor filters to extract expression-invariant features and also view-invariant features by rotation-invariant 3D spherical Gabor filter. This will cause a higher complexity and computer burden. The complexity needs a higher availability of computers with greater speeds and memory. Indirectly it also will affect the time for real time face recognition systems.

Based on the scenario above, it reveals that in order to calculate and understand the local curvature, the researchers need to really understand mathematical calculation. In addition, as data sets become larger, the algorithms have become more sophisticated even if the reported identification rates are not as high as in some earlier works (Bowyer et al. ,2005). Because of this reason, a technique has been built which does not need to understand the mathematical calculation which directly calculated the syntax of the 3D points with curvature information. The work presented in this paper used entirely 3D point approach of Kuang and Leng (2006) as a foundation with modifications and seven anchor points added. In their research, they detected five points which is right eye center, left eye center, upper nose, nose tip and mouth center. They only test their algorithm with five data sets in frontal view (their own data set) and it is not sufficient to prove their algorithm. Besides, the points detected are not suitable for variation of pose and expression. For example, middle mouth will affect the face while smiling and this point cannot be used for expression variation. For this reason, a set of quality points in different expression and position need to be explored.

It is hard to compare with other researchers due to different data set and difficulty to obtain a freely data set. Although, in this research we take three works that are similar to as our benchmark. Lu(2006) identified features through range and intensity images with pose and expression variation and proposed using shape index to identify the nose tip, and corners of the eyes and mouth. The average distance to the true feature locations was 10 mm resulting in relatively low precision. Colbry et al. (2005) used the same approached as Lu in different anchor points. They identify inside and outside eyes, nose tip, chin tip, and corner and middle mouth. Their result is approximately 90% below 20mm error for each anchor point. Boehnan and Russ (2005) proposed a method which focus on the 2D color information with registered range image to identify facial features. On a database of 1,500 images, they achieved a facial feature detection rate of 99.6%. Another interesting method is by using discriminating power of 3D descriptors to extract features (Moreno et al. (2003)) with 86 descriptors.

3.0 ANCHOR POINTS DETECTION

Due to the criteria of saliency, robustness and utility, the anchor points are added in this research. The selection is also for the variation of expression and position that obtained from the GavabDB database freely (Moreno and Angel (2004)). It contains facial surfaces that represent the faces by three-dimensional meshes (Xiao et al. (2004)). The data set is in WRL file where consist of 3D point coordinates and Indexed Face Set. Here we only used the 3D point coordinates data by discarding unnecessary data so that can be fitted with the system of feature point detection such as deleting the ear points, points missing while capturing and others. This unnecessary data will eventually cause wrong detection of the feature points and the editing process has been done using Geomagic Studio.

The important main contribution is extending a new algorithm which only involve text file that consist of 3D points. The primary advantage of this algorithm is that it doesn't involve any complicated mathematical formula. This is due to the fact that we only involve text file (3D-x,y,z) as compared to other researchers where they used 3D models or 2D images to support the detection stage. In addition, they have to use basic mathematical functions in order to calculate the surface to get the anchor points. In this algorithm, we have just to look at the xz and yz graphs plotted as illustrated in later section. Fig. 1 shows examples of the anchor points the prototype is designed to detect.

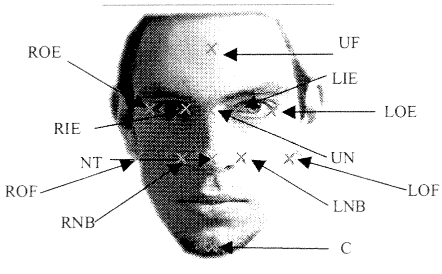


Fig. 1: Anchor point locations. ROE – Right Outer Eye; RIE – Right Inner Eye; LOE – Left Outer Eye; LIE – Left Inner Eye; UN – Upper nose point; NT – Nose Tip; RNB – Right Nose Base; LNB – Left Nose Base; ROF – Right Outer Face; LOF – Left Outer Face; C – Chin; UF – Upper Face

3.1 The Anchor Point Detection Architecture

The section describes the anchor point detection architecture of the proposed prototype system that aims to automate the anchor point's detection. The prototype consists of the following steps:-

a) Input data
b) Preprocessing
c) Face detection
d) Generate output

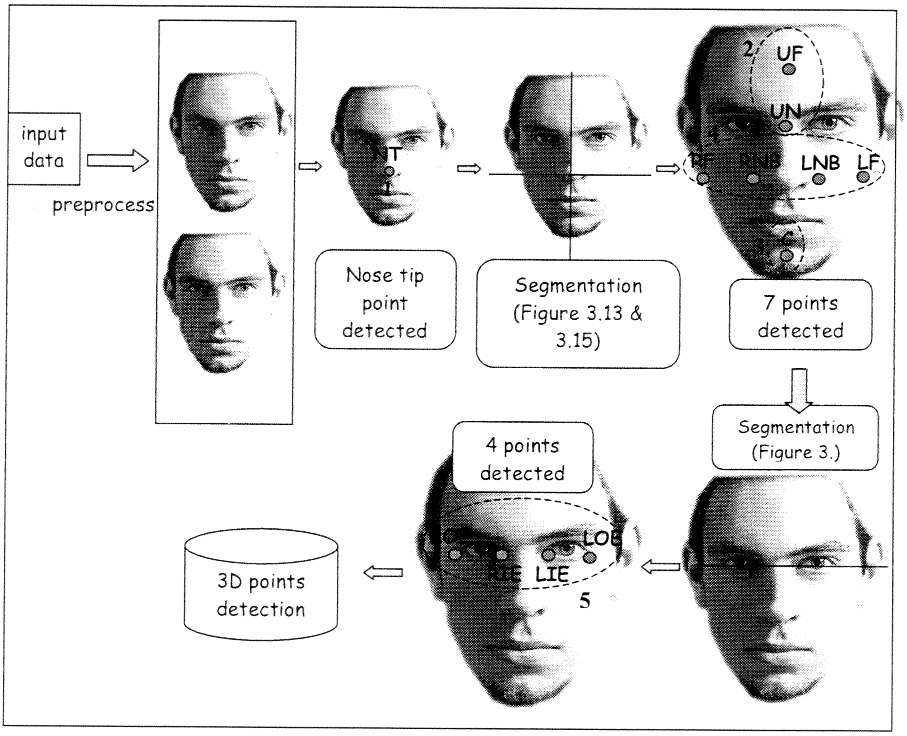


Fig. 2: The operational architecture of anchor point detection prototype

The system architecture in Fig. 2 shows the main entities used in this prototype. The architecture consists of five main processes that has been labelled in numbers: i) nose tip point localization, ii) upper nose and upper face points localization, iii) chin point localization, iv) nose base and outer face points localization and v) outer and inner eyes corner points localization.

3.2 The Anchor Point Detection Algorithm

Based on the architecture discussed in previous section, the algorithm will be explained detail in the following section.

i) Nose tip point localization

The first step in 3D face detection is to detect the nose tip which can be easily identified by finding the highest z value in the 3D face model. This rule can be used on face that is in frontal view. Furthermore, the nose tip is a distinctive point of the human face and also insensitive to the facial expression changes (Lu 2006). To determine the faces are in frontal view, the direction of Z axis should be on top of the nose as illustrated in Fig. 3.

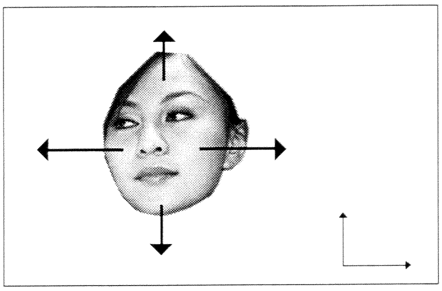


Fig. 3: Directional maximum of the nose tip. The nose tip will have the largest value along the Z-axis

ii) Upper nose and upper face point localization

Given the estimated nose tip, the next step is to obtain the eye and mouth region. This can be done by constraining the particular points by fixing the X axis as explained in Table 1. The illustration can be viewed in Fig. 4 where

Table 1: The upper nose and upper face anchor point detection algorithm

1. Nose tip estimated
2. Calculate the constrained eye and mouth region by fixing the X axis and taking the larger and lower Y axis within 0.5 neighborhoods.
3. From nose tip point, points are traversed north towards the forehead until the first lowest z value is encountered. This point is in the middle of inner left and right eye which called as Upper nose point.
4. Then the last point of this eye region is encountered as Upper face.

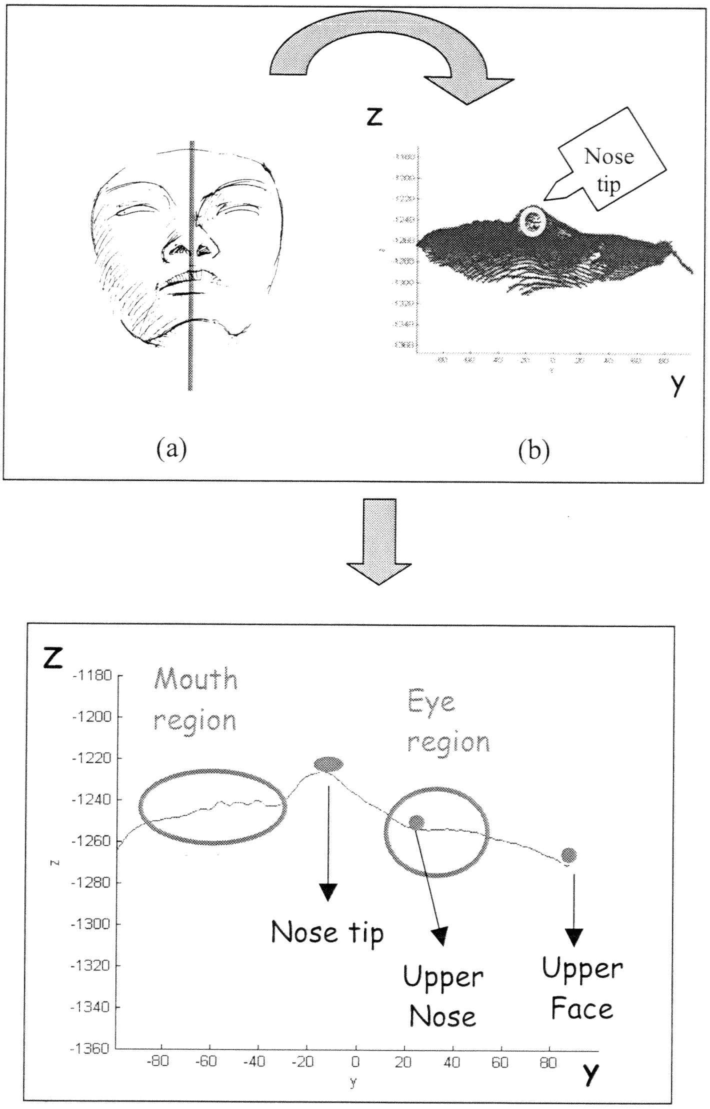


Fig. 4 (a): Full face in points representation with vertical line which show the location of eye and mouth area; (b) Plotting in YZ axis with full points; (c) Line to show the Nose Tip location

the vertical line is the line segmented and plotted on the Fig. 4(c). From the plotted YZ axis, mouth region and eye region can be determined. Thus, the last point of the traversed north is determined as upper face point. Besides that, upper nose point also obtained by encountering the lowest z point value.

iii) Chin point localization

From the vertical line segmentation as illustrated in Fig. 4 (a), mouth region is obtained from nose tip point until the lowest Y value. Then to obtain chin point, from the minimum of Y value, traversed along the Y axis until reached the first peak point called Chin point. For illustration, refer to Fig. 5.

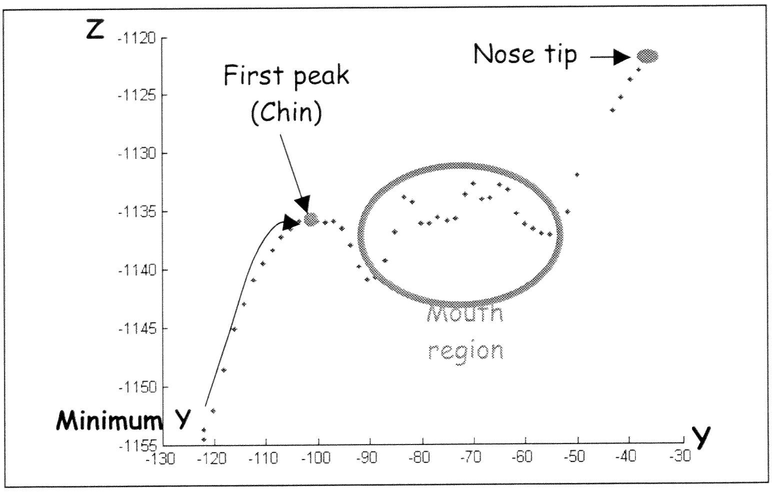


Fig. 5: YZ axis for mouth detection

iv) Nose base and outer face points localization

Given the estimated nose tip again, the search region for nose base point location is constrained. It can be viewed in Fig. 6 (right) and the algorithm explained in Table 2. These two points will be used for estimating the nose width and depth in feature extraction.

Table 2: The nose base and outer face anchor point detection algorithm

1. Nose tip estimated 2. Calculate the constrained left and right nose region by fixing the Y axis and taking the larger and lower X axis within 0.5 neighborhoods.

a. lower – Right nose region
b. larger – Left nose region
3. Sort ascending the left nose region points by X axis.
4. From nose tip point, points are traversed along Z axis points until get the point that is more than previous point or within 0.9 differences called left nose base point.
5. The last point of this left nose region is encountered as left outer face point.
6. For the right nose point detection, sort descending the right nose region points by X axis.
7. Same as left nose base point, traversed along Z axis points until get the point that is more than or within 0.9 differences with the previous point. This is encountered as the right nose base point.
8. The last point of this right nose region is encountered as right outer face.

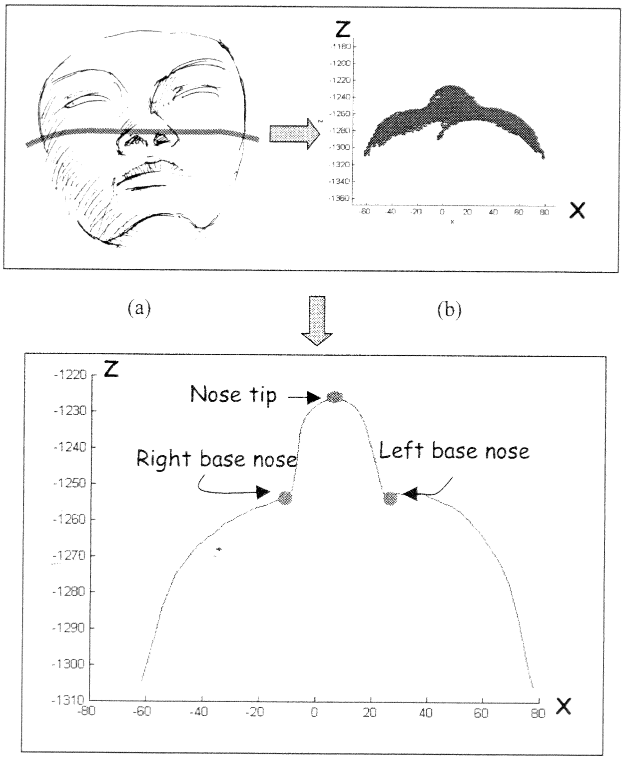
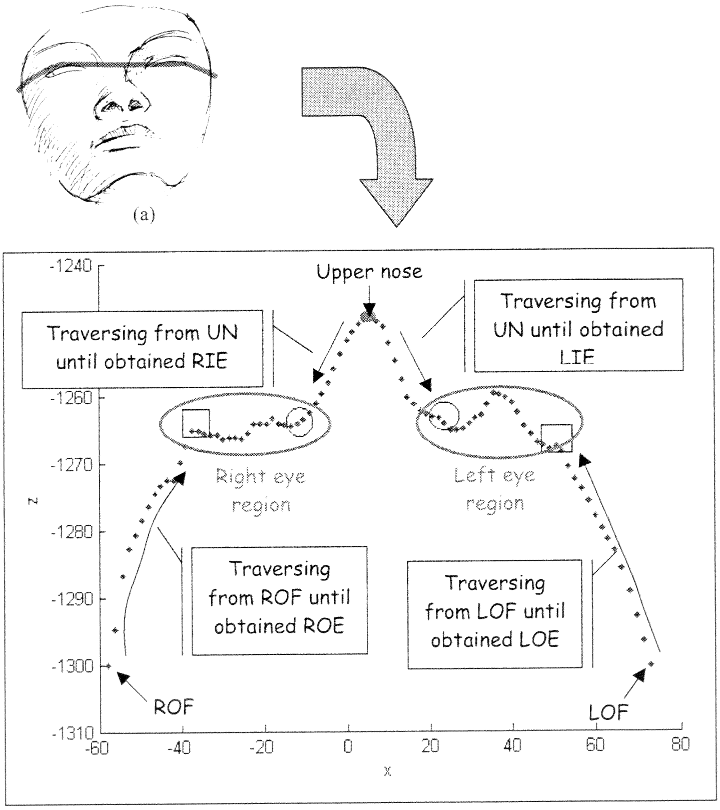


Figure 6 (a): Full face in points representation with horizontal line which show the location of nose area (b) Plotting in XZ axis with full points (c) Line to show the nose base location

v) Outer and inner eyes corner localization

Last is to identify both points for eyes corner. It starts from the upper nose point, the eye corners are then determined with the same concept in obtaining nose base points which detail algorithm in Table 3. The inner and outer eye corners are obtained in two steps. First is by traversing from upper nose point until obtained the point that is more than or within 0.9 differences with the previous point (explained in Fig. 7).



(a) (b)

Fig 7 (a): Full face in point representation with horizontal line which shows the location of eyes area (b) Eye point region by plotting in XZ axis. The rectangles are the location of outer eyes, while the circles are the location of inner eyes

Table 3: The eyes corner point detection algorithm

1. Upper nose point estimated
2. Calculate the left and right eyes region by fixing the Z axis and taking the larger and lower X axis within 0.5 neighbourhoods
   - Lower – right eye region
   - Higher – left eye region
3. Traversing from upper nose until get the point that is more than or within 0.9 differences with the previous point. This point is encountered as inner eye points (inner left and right eye point).
4. Then, traversing from outer points (left and right) until reaching first peak and this will be determined as outer eye points.

4.0 EXPERIMENT AND DISCUSSION

Based on the 3D points file, the feature points of each face are manually labelled. Using the manually labeled position as the ground truth, the localization displacement is computed as the Euclidean distance between the position of the automatically extracted feature point and the ground truth position. For easy notation, we introduce the following terms. NT : nose tip; UN : upper nose; LEC : left eye centre; REC : right eye centre; MC : mouth centre; and C : chin. Only eight anchor points result shown because another four get small displacement. Table 4 provides the statistics of the localization displacement on the GavabDB database which will explain detail in the next section. Figure 6 provides examples of the feature extraction results. The large displacement of eyes localization is often due to unsmooth point distributions.

Table 4 provides the statistics of the localization displacement on the GavabDB database which will explain detail in the next section. The mean, median and standard deviation distance for each of the anchor points are illustrated in Fig. 8. The median error of the eight points is around 5.53 mm and the mean for each point shows higher in eye and nose displacement. The large displacement of eyes localization is often due to unsmooth point distributions. Thus, the nose tip and upper nose point show small displacement.

Boehnen and Russ (2005) said that the range errors have no significant impact on performance until the error is approximately 20 mm. All each error anchor points fall below 20 mm and only few that fall over 20 mm range errors. Over 20 mm range errors are often due to noise in scan images. Nose tip range errors are below 14 mm and it shows that small displacement of this point localization.

**Table 4: Statistics of the distance (in 3D) between the automatically extracted and manually labelled feature points for the GavabDB-database
(For the point image used in the experiments, the point distances in x and y directions are both in mm.)**

	NT	UN	LOE	LIE	ROE	RIE	LNB	RNB
Mean (mm)	3.23	5.33	7.89	6.39	7.54	6.13	6.54	6.66
Median (mm)	3.02	4.68	7.23	5.76	7.12	5.26	5.19	5.95
Std (mm)	2.08	3.78	4.61	3.91	3.98	3.76	5.04	4.23

Fig. 8: Anchor point/ landmark extraction results

i) Comparative with Lu 2006, Colbry et al. 2005, Boehnen et al. 2005

Although three researches have been compared but the result appealed are different. This can be shown in Table 5. Since Colbry et al. (2005) and Boehnen et al. (2005) reported their results as median error only, this research reported the median error as well for comparing. Hence, this research result performs better among them except UND data set produced by Lu (2006) with 0.4 mm different.

ii) Comparative with Lu 2006

For detail comparison, this research produced a result with mean, standard deviation and median based on Lu (2006). Only five anchor points can be compared which are nose tip (NT), left eye (LE), right eye (RE), outer right eye (ORE), and outer left eye (OLE). The results can be revealed in Table 6 and shows that this current research obtained better result since the standard deviation can be decreased. It shows how tightly concentrated the distribution is. In other words, the spread of data is near to the mean value.

Table 5: The median error (20mm) of Lu, Colbry, and Boehnen

Median error (20 mm)				
Fatimah 2008	Lu 2006		Colbry et al. 2005	Boehnen et al. 2005
5.53 mm (8 points)	MSU-I 6.7 mm (7 points)	UND 5.1mm (7 points)	10 mm (5 points)	40% of mouth 20 mm error & 0.4% error due to hair

5.0 CONCLUSION

We have developed algorithm to detect face anchor points using 3D point file and knowledge of the structure of the face from the plotting graph. These algorithms produce good results especially for points nose tip, mouth centre, chin, and upper nose. However, in order to achieve higher levels of accuracy we need to test with presenting facial expressions and light rotations. This will be our future work by using the database we are using now.

REFERENCES

Boehnen, C. & Russ, T. (2005). A Fast Multi-Modal Approach to Facial Feature Detection. In Proceedings of the Seventh IEEE Workshop on Applications of Computer Vision (WACV/MOTION'05), 135-142.

Bowyer, K., Chang, W. K. & Flynn, P. (2006). A survey of approaches and challenges in 3D and multi-modal 3D + 2D face recognition. Computer Vision and Image Understanding ,(101): 1–15.

Chua , C-S., Han, F. & Ho, Y-K. (2000). 3D human face recognition using point signature. Proceeding IEEE International Conference on Automatic Face and Gesture Recognition, pp. 233–238.

Colbry, D., Stockman, G., & Jain, A. (2005). Detection of Anchor Points for 3D Face Verification. In Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), 3:118-118.

Gordon, G.G. (1992). Face Recognition based on depth and curvature features. Computer Vision and Pattern Recognition (CVPR), pp. 108-110.

Kuang, K. T, and Leng, H. K.(2006). 3drec : 3d Face Recognition System, Thesis, Faculty of Computer Science and Information Technology, UPM.

Li, S. & Jain, A. (2005). Handbook of Face Recognition. New York: Springer.

Lu, X. (2006). 3D Face Recognition Across Pose and Expression. PhD thesis, Michigan State University.

Moreno, B. & Angel, S. (2004). GavabDB : a 3D Face Database. URL: http://gavab.escet.urjc.es.

Moreno, B., Sanchez, A., Velez, J. F. and Diaz, F. J. (2005). Face Recognition Using 3D Local Geometrical Features: PCA vs. SVM. Image and Vision Computing, pp. 339-352.

Wang, Y. & Chua, C-S. (2005). Face recognition from 2D and 3D images using 3D Gabor filters. Image Vision Computing 23(11): 1018-1028.

Wang. Y., Chua., C.-S. & Ho., Y.-K. (2002). Facial feature detection and face recognition from 2D and 3D images. Pattern Recognition Letters 23: 1191–1202.

Xiao, J., Baker, S., Matthews, I., & Kanade, T. (2004). Real-time combined 2D + 3D active appearance models , CVPR 2 : 535 - 542.

Yang, M. H., Kriegman, D. J. & Ahuja, N. (2002). Detecting Faces in Images : A Survey. IEEE Transactions on Pattern Analysis and Machine Intelligence, 24: 34 -58