ROBUST NOSE DETECTION IN 3D FACIAL DATA USING LOCAL CHARACTERISTICS

Chenghua Xu, Yunhong Wang, Tieniu Tan

National Lab of Pattern Recognition,
Institute of Automation, CAS, Beijing, P.R.China
E-mails: {chxu,wangyh,tnt}@nlpr.ia.ac.cn

Long Quan

Department of Computer Science,
HKUST, Kowloon, Hong Kong
E-mails: quan@cs.ust.hk

ABSTRACT

The problem of detecting the feature points arises in many fields of science and engineering. In this paper, we focus on the 3D face range data and propose a robust scheme to solve a specific problem, i.e. locating the nose tip and nose ridge using the local statistic features and included angle curve. This work is very significant to 3D face modelling, recognition and registration. The key features of our method are the fully automated processing, the ability to deal with noisy and incomplete input data, the immunity to the rotation and translation and the adaptability to the different resolution. The experimental results in different databases fully show the robust and feasibility of the proposed method.

keyword: nose detection, local statistic features, included angle curve, SVM

1. INTRODUCTION

With the development of the 3D capture, 3D data is widely applied in many fields, such as face modelling, animation and recognition. Almost all the applications depend on the robust feature point location. In 3D facial data, the nose holds the most distinct feature. Many other features, such as eye corners and mouth corners, are positioned according to the nose. This paper focuses on this specific task and develops a robust method to position the nose.

Most existing methods [1, 2, 3, 4] for the nose detection are usually based on the assumption that the nose tip is the highest point in the range data. This assumption can largely reduce the complexity of detection. But due to the noise and rotation of the subject, the assumption does not always hold. Gordon [5, 6] used the curvature information to detect the nose. Their methods are suitable to the clean 3D data and would not work in the case that there are holes around the nose. The data from laser scanners usually contains much noise and holes around eyes, opening mouth, hair, clothes and so on. To our best knowledge, there seems no robust work on nose detection by now.

Here, we propose a new algorithm to locate the nose. The nose tip is first positioned based on local statistic features, and then nose ridge is marked using included angle curve. Our proposed method is immune to the rotation and translation, holes and outliers, and suitable to multi-resolution data.

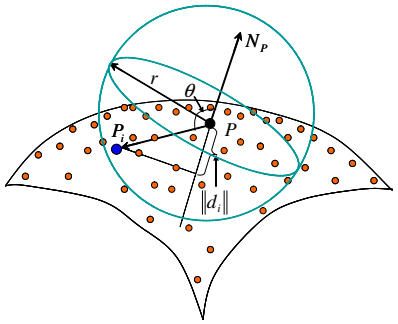

Fig. 1. Effective energy of the neighbors

The remainder of the paper is organized as follows. Section 2 describes the details of nose tip detection. The nose ridge detection using included angle curve is presented in Section 3. Section 4 shows the experimental results in different databases, and Section 5 concludes the whole paper.

2. NOSE TIP DETECTION

To locate the nose tip reliably, we must find its unique features. Here, we consider two properties: the nose tip is the local highest point and it has the special shape, like the peaked cap.

2.1. Local highest point

To each point, we can find its neighboring points within a given sphere with a radius r centered at this point. As shown in Fig.1, for the point P, its 3-D coordinates and the unit normal N_P is known. The point P_i is one of its neighbors. We can define the *Effective Energy* (EE) d_i for each neighboring point with the inner product of the vectors $P_i - P$ and N_P:

$$d_i = (P_i - P) \cdot N_P = \|P_i - P\| \cos\theta \qquad (1)$$

If P is the local protuberant point, i.e. the highest point along N_P, θ is always bigger than 90 degree, thus d_i always being negative. This condition is very weak and many

points lying in other areas, such as cheek, chin and so on, also meet this condition. But it may effectively reduce the searching space.

2.2. Local statistic features

Intuitively, the nose has the special shape, and we hope to describe it with scalar features to avoid the influence of transformation. Many statistics, such as mean, variance and moments, are available. To each point, we can obtain its statistic features from its neighboring area, and then the classic algorithm, Support Vector Machine(SVM) [7], is applied for learning and classifying.

Here, we only consider a 2D feature vector consisting of mean and variance of the effective energy of neighbors respectively:

$$\mu = \frac{1}{n}\sum_{i=1}^{n} d_i \tag{2}$$

$$\sigma^2 = \frac{1}{n}\sum_{i=1}^{n}(d_i - \mu)^2 \tag{3}$$

Intuitively, since the nose tip is the distinct protuberance in the local area, the mean is smaller and the variance is larger. Due to the similar shape of human nose, the mean and variance of the points lying around the nose tip are surely in a certain scope. We can use SVM [7] to learn the boundary between the nose tip and other areas.

All the points are classified into two categories: nose-tip point and non-nose-tip point. Each training vector is described with (x_i, y_i), $i = 1, 2, \cdots, l$, where $x_i \in R^2$ is the 2D feature vector (μ_i, σ_i^2), and $y_i \in \{-1, 1\}$ denotes the two classes. We can find an optimal hyperplane to distinguish these two classes by minimizing the following energy function:

$$Q(a) = \sum_{i=1}^{l} a_i - \frac{1}{2}\sum_{i,j=1}^{l} a_i a_j y_i y_j K(x_i \cdot x_j) \tag{4}$$

where $K(x_i \cdot x_j) = exp(-\frac{|x_i - x_j|^2}{\sigma^2})$ is the kernel of the SVM classifier.

For a testing vector, x, we can use the following discriminant for classifying:

$$f(x) = sgn(\sum_{i=1}^{l} a_i^* y_i K(x_i, x) + b^*) \tag{5}$$

where a^* and b^* are obtained by minimizing Eq.4.

Due to the complexity of the facial surface, especially the deformable cloth, some points in other areas can also be viewed as the nose tip. Fortunately, they are too sparse to be ignored.

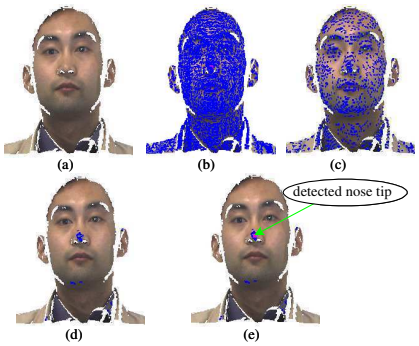

Fig. 2. Process for detecting nose tip. (a)The original point cloud; (b,c)The candidates after the first and second filer; (d)The candidates after SVM selection; (e)Nose tip.

2.3. Strategy for efficient implementation

Our process for detecting the nose tip includes three steps: candidate filter, SVM selection and nose tip decision.

For a given point, finding its neighboring points in 3D space has a large computational load, usually $O(n^2)$. We can use an alternative method to accelerate this process.

We first triangularize the point cloud by Delaunay algorithm. For a point P, it is easy to find its direct neighbors, which connect P with one edge. Then, we can determine that P is one candidate of the nose tip only if the effective energy of any neighbor is negative. In Fig.2b, the blue points are the candidates.

In Fig.2b, there are still too many candidates. To each candidate, we can use its indirect neighboring points (neighbor's neighbors) to filter the candidates further. Fig.2c shows the filtered results. Thus, we can obtain the fewer candidates by repetitive filtering. The times of the filtering depend on the density of the point cloud. Finally, we remain no more than 100 candidate points.

To each candidate, we can find its neighboring points within a given sphere. The statistic feature vector, $x = (\mu, \sigma 2)$, is calculated using Eq.2 and 3. We can further select the candidates of the nose tip by SVM selection Eq.5. The result is shown in Fig.2d.

In fact, our classifier is fairly weak, and can not recognize the nose tip strictly. However, By SVM selection, there are more dense candidates nearby the nose tip. We can consider that the position having the most dense candidates is the nose tip as shown in Fig.2e.

3. NOSE RIDGE DETECTION

After the position of the nose tip is estimated, we use included angle curve to determine the nose ridge.

For the detected nose tip P, we place two spheres with radius R and r, centered at P as shown in Fig.3. The intersection of two spheres and the object surface is a 3D space torus. We define a reference plane G by the normal vector

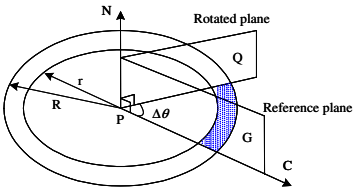

Fig. 3. Partition of the torus.

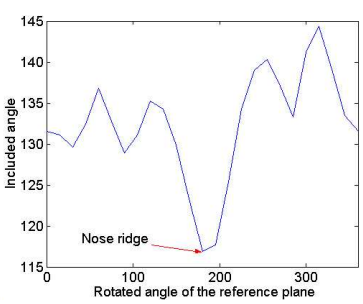

Fig. 4. Included angle curve of the nose tip

N and a reference vector C that is orthogonal to N. A rotated plane Q rotates along the axis N beginning from G with a fixed step $\Delta\theta$ and partitions the torus into $360/\Delta\theta$ parts. We obtain the barycenter B_i for ith part by meaning points within it and then calculate the included angle between the vector $B_i - P$ and N. Since P is the local highest point, all its included angles are larger than 90 degree. The area near the nose ridge has the smallest included angle as shown in Fig.4, which can be used to decide the approximate direction of the nose ridge.

This algorithm is similar to point signatures [8]. The main difference is that our method does not acquire a plane fit to the local points. Here, the radius R and r is the principle parameter for the successful implementation. Appropriate value can avoid the influence of the nosewing and makes the nose ridge standing out. From our experiments in different databases, the results are satisfying in the case that $R = 25mm$ and $r = 20mm$.

4. EXPERIMENTS

To test the performance of the proposed algorithm, we implement it with different databases. All the experiments are executed in the PC with PIV 1.3GHz processor, 128M RAM and the display card Nvidia Getforce2 MX 100/200.

4.1. Databases

We use three different databases to test our proposed algorithm. The first database (3D Pose and Expresion Face Models, 3DPEF) is sampled using Minolta VIVID 910 working on Fast Mode. This data set contains 30 persons (6 women), and each person includes five poses and five expressions. This set is mainly used to train SVM classifier

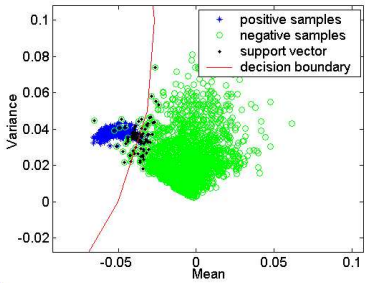

Fig. 5. The training result of SVM

and test the robust of our method. The second database (MPI) [9] is from MPI in Saarbrücken. We can only obtain five 3D head models from them. These models are used to test the performance in the high-dense data. The third data set (PART) is from 3D Part Database [10] of IRIS Laboratory at the University of Tennessee, Knoxville. It is used to test the performance under the different resolution. In advance, all the point clouds are triangularized and the normals of all the points are also calculated.

4.2. Training for SVM

During the training process, we select the samples of two persons (one man and one woman) in 3DPEF as the training set. Due to pose and expression variations, there exist some holes around the nose. To ensure that the positive samples can describe the correct shape of the nose tip, we first fill the holes manually. We select the points around the nose tip in the training point clouds and record their mean and variance as the positive samples. The negative samples can be obtained at any other area from the original point cloud. Finally, we obtain 325 positive samples and 3178 negative samples to train SVM classifier. We use RBF fuction ($c = 100, \sigma^2 = 14$) as the kernel. Fig.5 shows the training result.

From Fig.5, we can see that the positive samples congregate tightly, which shows that the human nose has the similar statistic shape. The negative samples distribute in a big space. We also see that the decision boundary does not strictly distinguish the two classes. But it can effectively select the candidates of the nose tip.

4.3. Detecting results

We first test our algorithm in our database. The total number of test samples is 280, not including the training samples for SVM. To each test sample, we detect the nose tip in the way of Section 2.3. Under the complex condition of pose and expression variations, the influence of clothes, and holes and noise, only two samples are falsely detected. Fig.6 shows some results.

In Fig.6, the first row is on pose variation, the second is on expression variation, and (d,f,g) are of two girls. In Fig.6(c,d), the nose tip is not the highest point due to up

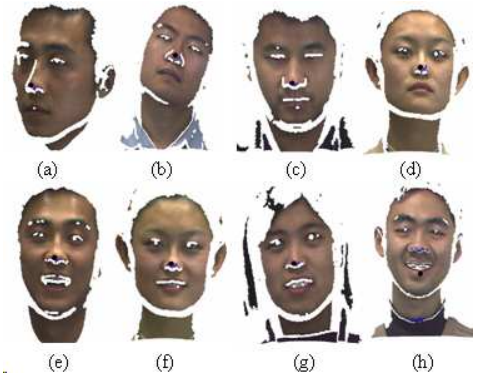

Fig. 6. Results in 3DPEF. The black point in each image is the detected nose tip.

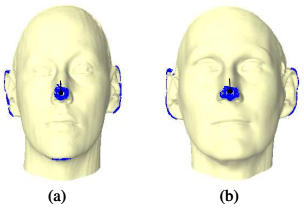

Fig. 7. Results in MPI set. The black line connecting to the black point is the detected nose ridge.

or down motion. Our method can detect the nose tip successfully. In Fig.6(b,c,g), the data contains parts of clothes, and especially in Fig.6g, there is much noise due to the disturbance of long hair. Fortunately, our method can conquer them effectively.

Both falsely detected samples are affected by clothes. When clothes are deformed into a certain shape, they have the similar statistic feature to the nose, which induces the incorrect detection. Fig.6h shows one example, in which the neckline also collects more dense candidates.

Intuitively, the jaw and the protrudent throat of the man have the similar statistic feature to the nose. Our proposed scheme can distinguish the nose tip from them effectively.

MPI set [9] has the high quality point clouds, which have dense points (more than 70,000 vertices, 150,000 triangles) and smooth surface. Our proposed method can correctly locate the nose tip and nose ridge as shown in Fig.7. However, due to too many local highest points, it is difficult to reduce the candidates, thus resulting in a large computational load. In fact, we can first reduce the vertices of the model by mesh optimization, and then locate the nose.

We also test our algorithm with PART set [10], which provides three complete 3D head models of a status as shown in Fig.8. In these models, the searching space is the whole head surface, and they are not vertical. The nose tip can be located successfully in the different resolution. Unfortunately, in Fig.8a, the points are too sparse, the nose ridge is not marked correctly.

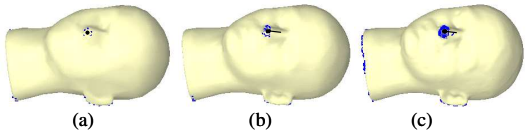

Fig. 8. Results in PART set. (a)502 vertices; (b)2502 vertices; (c)40071 vertices

In our proposed algorithm, nose tip detection occupies most computational cost, strongly depending on the number of points. It takes only 0.4s to locate the nose tip in the point cloud with three thousand points.

5. CONCLUSIONS

In this paper, we propose a new scheme to locate the nose tip and nose ridge in 3D facial data using the local statistic features and included angle curve. The proposed method is fully automated, robust to noisy and incomplete input data, immune to the rotation and translation and suitable to the different resolution. The experimental results fully show the excellent performance of our method. In the future, we will locate the other features based on the position of the nose, and it is also necessary to consider other statistic schemes for detection.

Acknowledgements: The authors gratefully thank Mr. Zhiping Jiao for helping us sample 3DPEF and Prof. H.H. Buelthoff for providing MPI head models. Also thanks to Mr. Jia-li Cui for discussion on SVM. This work is supported by research funds from NSFC (No. 60121302 and 60332010), the Outstanding Overseas Chinese Scholars Fund of CAS (No.2001-2-8).

6. REFERENCES

[1] Y.C. Lee, D. Terzopoulos, and K. Waters, "Constructing Physics-based Facial Models of Indibiduals", Proc. Graphics Interface, pp.1-8, 1993.
[2] C. Hesher, A. Srivastava, and G. Erlebacher, "A Novel Technique for Face Recognition Using Range Imaging", Inter. Multiconference in Computer Science, 2002.
[3] Y. Lee, K. Park, J. Shim, and T. Yi, "3D Face Recognition Using Statistical Multiple Features for the Local Depth Information", Proc. 16th Inter. Conf. Vision Interface, 2003.
[4] C. Beumier and M. Acheroy, "Automatic 3D Face Authentication", IVC, Vol.18, pp.315-321, 2000.
[5] G.G. Gordon, "Face Recognition based on Depth Maps and Surface Curvature", Proc. SPIE Geometric Methods in Computer Vision, Vol.1570, 1991.
[6] T.K. Kim, S.C. Kee, and S.R. Kim, "Real-time Normalization and Feature Extraction of 3D Face Data Using Curvature Characteristics", IEEE Inter. workshop on Robot and Human Interactive Communication, pp.74-79, 2001.
[7] C. Burges, "A tutorial on Support Vector Machines for Pattern Recognition", Data Mining and Knowledge Discovery, 2(2):121-167, 1998.
[8] C.S. Chua and R. Jarvis, "Point Signatures:A New Representation for 3D Object Recognition", IJCV, 25(1):63-85, 1997.
[9] http://faces.kyb.tuebingen.mpg.de.
[10] http://iristown.engr.utk.edu/%7Epage/database.