Facial Expression Recognition from 2.5D Partial Face Data by Using Face Plane

Theekapun Charoenpong[1], **Shogo Tokai**[2], and **Hiroyuki Hase**[3], Non-members

ABSTRACT

Due to a problem of current research occurring when recognizing facial expressions from a 2.5D partial face data set taken from any viewpoint ranging from -45° to +45°, we propose a novel algorithm for recognizing facial expressions from a 2.5D partial face data set. A 2.5D partial data set is captured from any viewpoint between -45° and +45°. For facial expression recognition, a 3D virtual expression face is first reconstructed from a 2.5D partial face data set. A facial expression is then represented in terms of the change of crossing points on a face plane. Next, two schemes are used to analyse the crossing points for recognition. In the first scheme, the distribution of crossing points was used. In the second scheme, the displacement vectors of crossing points by facial expression change were used. The classification is carried out by using a support vector machine (SVM) for the both recognition schemes. The experiments were done for four facial expressions (neutral, anger, surprise and smiling) of 22 persons. The results showed the feasibility of the proposed method.

Keywords: Facial Expression Recognition, Partial Face Data, Face Plane, Crossing Point, Displacement Vector

1. INTRODUCTION

As facial expressions provide important information such as affective stage, cognitive activity, temperament and personality, truthfulness, and psychopathology, the several researches have been studied on facial perception over the last three decades [1], [2], but these researches performed effectively for recognizing the facial expressions from only full frontal facial image. However, in the case of many applications concerning robots, the partial face data from only one viewpoint is available. Therefore, it is important to develop a new innovation technology to meet the requirements of robots that use a partial face data taken from only one viewpoint. In this paper, we propose a new method for recognizing the facial expressions using a 2.5-dimensional (2.5D) partial face data set. A 2.5D data set is a simplified three-dimensional (3D) (x, y, z) surface representation that contains at most one depth value (z-direction) for every point in the (x, y) plane, associated with a registered texture image. Each scan can only provide a single viewpoint of object, instead of the full 3D view [3]. However, the usage of a 2.5D image has the advantage over a two-dimensional (2D) image when the face pose is varied. The following facial expression recognition methods have been proposed so far.

M. Pantic and L.J.M Rothkantz [4] proposed a facial expression recognition method by defining their face model as a point-based model composed of the two 2D facial views of the frontal and side view face images. Multiple feature detectors were applied redundantly in order to localize contours of prominent facial features prior to their model. This method needs more than one viewpoint of face image including frontal face image. H., Michael et al. [5] use a generic 3D head model and frontal view image for the synthesis of facial expression, and then the facial expression was classified by local binary pattern techniques. I.A. Essa [6] adopts the optical flow processing for perception and measurement of facial motion. The 3D mesh model of face is fitted by 2D frontal face image. Muscle activation is extracted using a physics-based model of facial deformation, and then the deformation is estimated from the optical flow. C. Li and A. Barreto [7] proposed a framework of 3D face recognition involving an initial expression assessment of unknown face. The smiling and neutral expressions were recognized. The most distinctive features associated with the smile are the bulging of the cheek muscle and the uplift of the corner of the mouth, and then the histograms of the range value (z-coordinates) of cheeks from smiling and neutral face are used for recognition by means of a Linear Discriminant Analysis (LDA) and a Support Vector Machine (SVM). J. Wang et al. [8] used the rotated 3D frontal face model under the viewpoint varying between +40° and -40° for each rotation on pitch and yaw to recognize facial expressions. Based on the principal curvature information estimated on the 3D triangle mesh model, a surface labeling approach is applied to classify the 3D primitive surface features into twelve basic categories. The statistic histograms of the surface labels of all these regions are combined to construct the specific facial expression feature. Linear Discriminant Analysis (LDA) is used for classification. In this method,

Manuscript received on July 7, 2009 ; revised on November 17, 2009.

[1] The author is with The Biomedical Engineering Programme, Faculty of Engineering, Srinakharinwiroj University, 26120, THAILAND , E-mail: theekapun@swu.ac.th

[2,3] The authors are with The Graduate School of Engineering, University of Fukui, 3-9-1 Bunkyo, Fukui, 910-8507, JAPAN, E-mail: tokai@u-fukui.ac.jp and haseh@u-fukui.ac.jp

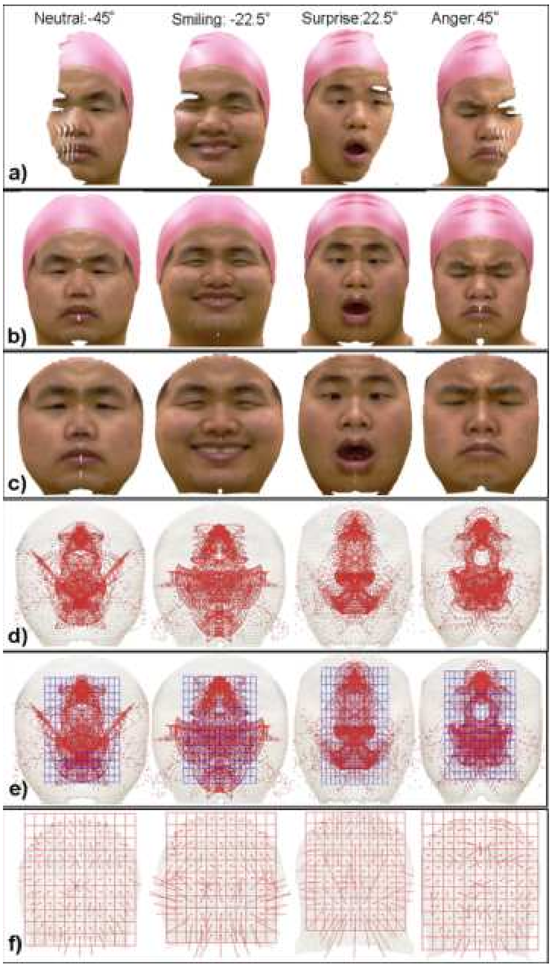

Fig.1: *a) Input 2.5D partial face images are captured in different viewpoints starting from -45° (Neutral), -22.5° (Smiling), 22.5° (Surprise) and 45° (Anger). b) 3D virtual expression faces are reconstructed. c) Only facial surface is extracted to be used in the face plane algorithm. d) Crossing points on a face plane. e) The crossing points are divided into M regions. f) A displacement vector set of each facial expression consisting of M average vectors is used to analyze the facial expression.*

the full frontal faces are needed but, in real application, only data of partial face can be taken in the case of capturing the face from side view.

In the current research, the frontal face image is important for the satisfactory results. However, the orientation of the face is also a factor to make the recognition result fail. Therefore, in this paper, we propose a novel method for the 3D face expression recognition from a 2.5D partial face data set. An input 2.5D partial face data set is captured from any viewpoint ranging from -45° to +45° around the y-axis (see Fig. 1(a)). The 3D virtual expression face is then reconstructed according to a 3D face reconstruction algorithm (see Fig. 1(b)). The facial skin is used for the facial expression recognition by using a face plane algorithm (see Fig. 1(c,d)). A facial expression is represented in terms of a crossing point distribution and displacement vectors on the face plane (see Fig. 1(e,f)). The recognition accuracies of the

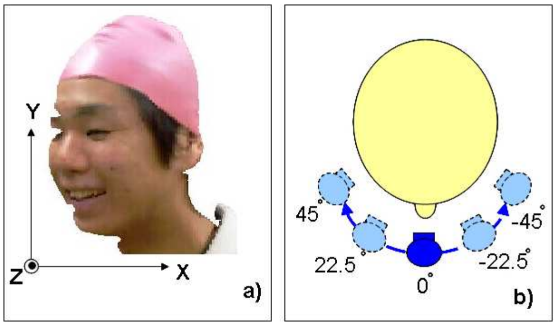

Fig.2: *: a) The black color surface as hair unmeasured by laser light is covered by a cap. b) Viewpoints for capturing the 2.5D image from the top view of human head. Five viewpoints are captured from -45°, -22.5°, 0°, 22.5° and 45°.*

crossing point distribution and the displacement vectors will be shown in the experiments. Our method was developed for recognizing four facial expressions (neutral, anger, surprise and smiling) by mean of a support vector machines. The advantage of our proposed method over the existing facial expression recognition systems is that the proposed method uses only one partial face data set from any viewpoint between 45° and +45° for facial expression recognition.

In the following sections, our facial expression recognition methods are described in Section 2. The experimental results and discussion are shown in Section 3.

2. FACIAL EXPRESSION RECOGNITION

The process of the facial expression recognition is illustrated in Fig. 1. The recognition method which consists of 3D face reconstruction, the face plane computation and the crossing point analysis schemes are described in the following subsections.

2.1 3D Face Reconstruction

As the input is a 2.5D partial face data set, a 3D virtual face is reconstructed by the method in ref. [9].

To reconstruct a 3D face, the 2.5D partial face data set is split into N cross sections along the y-axis. The data on each cross section was fitted by an ellipse fitting technique. The nose tip and nose ridge are then detected by correcting the face vector according to a nose ridge detection algorithm. A symmetry plane passing the nose tip, the nose ridge and the point on the center line of the face is used for reconstructing the 3D face according to a symmetry plane correction algorithm (see Figs. 4 and 5).

The 2.5D image acquisition, nose ridge detection and symmetry plane correction are explained in this subsection.

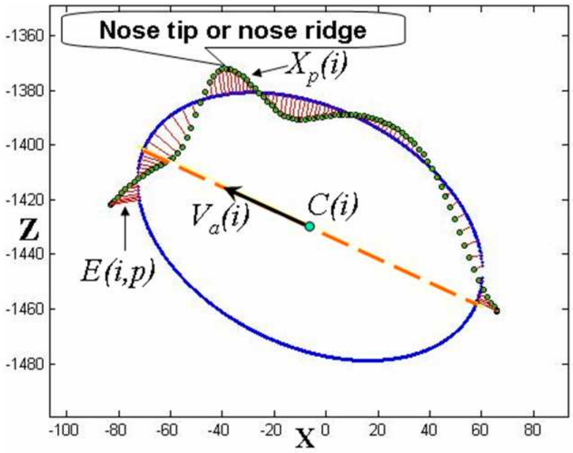

Fig.3: *The head cross section data passing the nose is fitted by an ellipse fitting technique. $X_p(i)$ is a point on the facial surface. The lines from the points on the facial surface to the ellipse are the ellipse fitting error E(i,p) of a point $X_p(i)$.*

2.1.1 2.5D Image Acquisition

For our 3D facial expression reconstruction algorithm, a single 2.5D head data set is used. The input data is captured by using a VIVID700 scanner and represented by point set data. The 2.5D data consists of range data and color data. The range image size is 200×200 pixels, the color image has 400×400 pixels in the same scanning region. The object stands in front of the scanner at a distance of about 1.5 m. Unfortunately, the VIVID700 scanner is not sensitive to black color such as the kind found typically in Asian hair because it uses a red laser. This problem can be solved by covering the hair with a cap (Fig. 2(a)). Only facial surface will be used in the recognition algorithm. The input data was captured from several viewpoints around the y-axis varying from -45° to +45° with step 22.5° (see Fig. 2(b)) for four facial expressions (neutral, anger, surprise and smiling).

2.1.2 Nose Ridge Detection

A) Ellipse Fitting Technique

In this subsection, three important parameters which can be obtained by the ellipse fitting technique will be described. The head data from the range image is split into N cross sections parallel with the xz-plane. The data on each cross section is fitted by using an ellipse fitting technique [10]. The three important parameters illustrated in Fig. 3 can be obtained as follows.

i) The Average Center of the Ellipse (C_{Ae}): This parameter is the average value computed from the centers of the ellipses $C(i)$ in all cross sections.

ii) The Semi-major Axis Vectors $V_a(i)$ of the i^{th} Cross Section: There are two opposite directions of the vector along the semi-major axis of the ellipse in each cross section pointing away from the center. In

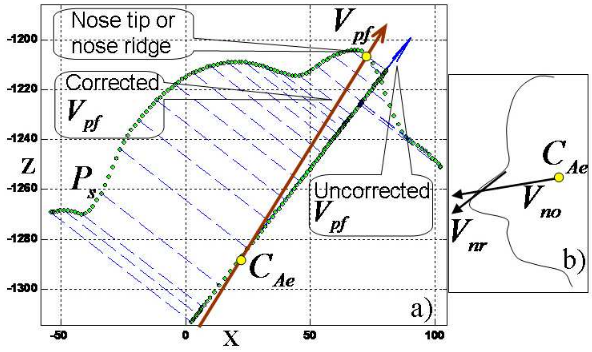

Fig.4: *: a) The projected face vector V_{pf} points from the average center point C_{Ae} to the nose ridge point. P_s is a points set on the facial surface. The dashed line illustrates the projection of Ps onto V_{pf}. b) The two vectors in a face side view are the nose ridge vector V_{nr} pointing along the nose ridge points, and the nose vector V_{no} pointing away from the average center of the ellipse C_{Ae}.*

each cross section, the semi-major axis vector pointing to the facial surface is selected.

iii) The Average Ellipse Fitting Error $E_{Ae}(i)$: The ellipse fitting error E(i,p) at a point $X_p(i) = (x_p, z_p)$ in the i^{th} cross section is defined as shown in Fig. 3, where p is a point in the cross section data. $E(i,p)$ is the orthogonal distance between the point $X_p(i)$ and the ellipse. $E_{Ae}(i)$ is the average of all $E(i,p)$ for the i^{th} cross section.

B) Nose Ridge Detection Algorithm

The nose ridge is detected from the cross sections in the nose region. According to a pilot test, there are fifteen cross sections in the nose region counting from the cross section passing the nose tip. Therefore, the nose region can be determined by detecting the nose tip. As an empirical observation, the cross section including the nose tip always has the maximum average ellipse fitting error.

[Nose ridge detection algorithm]

project the face vector onto each cross section of the nose region; // (projected face vector V_{pf})
set an initial value for the detected point;
for each cross section in the nose region {
  do{
    project the point set on facial surface (P_s) onto the projected face vector; // (see Fig. 4(a))
    find the projected point which shows the farthest distance from the average center C_{Ae};
    correct the projected face vector to the point which shows the farthest distance;
  }*while (the detected point is not the same with the previous point)*
}

[End of algorithm]

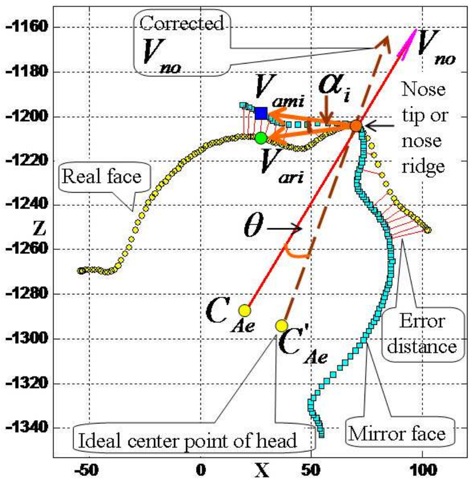

Fig.5: *C'_{Ae} is the new center instead of after rotating V_{no}. The rotation angle $\theta/2$ corresponds to α_i for correcting the nose vector.*

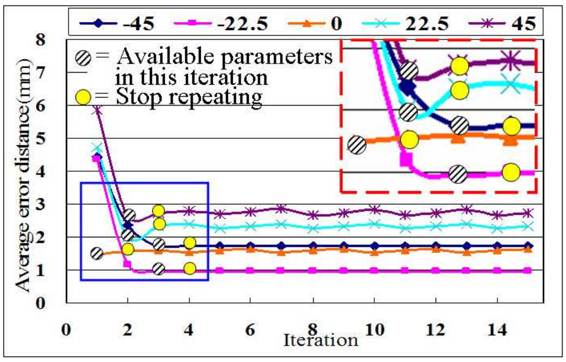

Fig.6: *The convergence of the average error distance over 15 iterations in the symmetry plane correction algorithm. If the average error distance increases compared to the previous iteration, the algorithm will stop and the parameter computed in the previous iteration will be used in the next part of algorithm. The graph in the red dash frame is an enlargement of the blue frame.*

This process is applied to all of the 15 cross sections in nose region. The 15 nose ridge points are determined tentatively because the average center point C_{Ae} is also determined temporarily. The nose ridge vector (V_{nr}) can be obtained as the eigenvector with the highest eigenvalue computed from the 15 nose ridge points by using a principle components analysis technique (PCA). The vector pointing from the average center point C_{Ae} to the center of 15 nose ridge points is then defined as nose vector (V_{no}) (Fig. 4(b)). In subsection 2.1.3., the correction of V_{no} and C_{Ae} will be discussed.

2.1.3 Symmetry Plane Correction

Due to the bilateral property of the face [11], the symmetry plane, which is called "midsagittal plane" in face anatomy, is defined as the vertical plane along the center line of the face through the nose tip and the nose ridge. The symmetry plane is defined by using two vectors: the nose ridge vector and the nose vector. By reflecting the real face in the symmetry plane, a mirror face can be computed. The virtual face is then reconstructed from the real face and its "mirror face". However some virtual faces result in faces that are either too fat or too thin. These problems occur due to the incorrect location of the center point of the head. In this subsection, a correction algorithm that determines the average center point C_{Ae} lying on the nose vector will be explained.

The nose vector is rotated around the nose ridge (Fig. 5.) by an average angle computed from the angle between the two vectors V_{am} and V_{ar}. V_{am} and V_{ar} are the vectors pointing from the nose ridge point to the corresponding points on the mirror face and the real face respectively. The corresponding points are the closest points on the real face and the mirror face. The nose vector V_{no} is rotated by the rotation angle $\theta/2$ centered at the nose ridge point. The angle θ is the average of α_i which are the angles between V_{ami} and V_{ari} . By rotating the nose vector, the average center point C_{Ae} is moved to a new position C'_{Ae}. The symmetry plane and the mirror face are then computed again. The average center point is more stable when the average error distance between the corresponding points on the mirror face and the real face is smaller than the previous iteration. After the center point C_{Ae} has moved to the new center point C'_{Ae}, the farthest point from the center point C_{Ae} is not farthest from the new center point C'_{Ae}. Therefore, the nose ridge which is the farthest point from C'_{Ae} is detected again by using the nose ridge detection algorithm described in subsection 2.1.2.(B). This algorithm is repeated until the average error distance shows a bigger value than the previous one.

This algorithm is shown by the following pseudo code:

[Symmetry plane correction algorithm]

calculate the average distance error between the mirror face and the real face;
set an initial value for the average distance error;
do{
calculate using two vectors V_{am} and V_{ar};
move C_{Ae} to C'_{Ae} by rotating the nose vector with angle $\theta/2$;
make the symmetry plane and calculate the mirror face;
calculate the average distance error between the mirror face and the real face;
the previous point)
apply the nose ridge detection algorithm described in subsection 2.1.2(B);

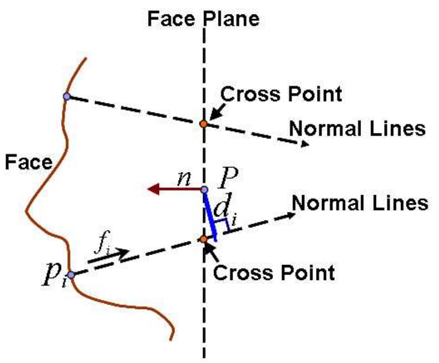

Fig.7: *Derivation of face plane*

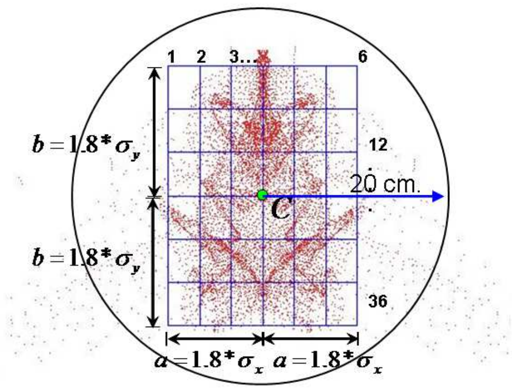

Fig.8: *The crossing point distribution in M regions*

}*while (the average distance error is smaller than the previous one)*

End of algorithm

2.2 Face Plane Computation

After the virtual face was reconstructed in section 2.1., the facial surface is extracted as shown in Fig. 1(c). The face plane algorithm is applied to represent the facial expression in terms of crossing points [12]. An advantage of the face plane is that the relative position from the face is fixed adaptively.

2.2.1 Derivation of the Face Plane

The face plane is a virtual plane across the head. A face plane is derived by normal vectors. These normal vectors point from the facial surface toward the center of the head. A normal vector is calculated using four neighboring points on the facial surface. The derivation of the face plane is described as follows.

3D data of a point on the facial surface is described as $pi(i = 1, \ldots, N)$ (see Fig.7). The normalized vector f_i on the normal line points from the point pi to the center of the head. The point P is assumed to be a point inside the head. A distance d_i is the orthogonal projection distance between the point P and the normal line f_i as shown in the following equations.

$$\begin{aligned} d_i^2 &= \|P - p_i\|^2 - (P - p_i, f_i)^2 \\ &= (P - p_i)^t (E - f_i \cdot f'_i (P - p_i) \end{aligned} \quad (1)$$

where E is the unit matrix and $(\bullet, \bullet)$ is the inner product. The optimization can be carried out by differentiating $Q = \sum_i d_i^2$with respecting to P.

$$\frac{dQ}{dP} = 2 \sum_i (E - f_i \cdot f_i')(P - p_i) = 0 \quad (2)$$

Then the optimal point P can be obtained.

$$P = \left[\sum_i (E - f_i \cdot f_i') \right]^{-1} \cdot \sum_j (E - f_i \cdot f_i') p_j \quad (3)$$

The face plane passes through this point. But the orientation of the plane has not been determined. Therefore, the normal vector n of the face plane which makes the next equation maximum can be obtained. This makes an eigen problem under the condition $\|n\| = 1$.

$$S = \sum_i (f_i, n)^2 = n' \left(\sum_i f_i \cdot f_i^f \right) n \quad (4)$$

By solving this problem, three eigenvalues and three corresponding eigenvectors can be obtained as $\lambda_1 \geq \lambda_2 \geq \lambda_3$ and ϕ_1, ϕ_2, ϕ_3 respectively. ϕ_1 will be chosen as the direction of the face plane based on the least distance. The face plane is fixed by P and n.

2.2.2 Crossing Point Computation

Based on the face plane computed in 2.2.1., crossing points can be calculated. As the normalized vector f_i points from the point p_i, the equation of the normal line is described as follows;

$$h_i = p_i + t_i f_i \quad (5)$$

where h is a point on the normal line, t_i is a real number. Then, suppose a virtual plane in a head;

$$(n, P - h_i^*) = 0 \quad (6)$$

where n is the normal vector of the face plane, h_i^* is a point on the face plane. From Eqs.(5) and (6), we can obtain t_i

$$t_i = \frac{(n, p_i - P)}{(n, f_i)} \quad (7)$$

By substituting t_i in Eq. (5), we can obtain the crossing point h_i^*(see Fig. 1.(d)).

2.3 Crossing Point Analysis Scheme

As the facial expression is represented in terms of the crossing points, the crossing points are analyzed by using two schemes for the recognition: crossing point distribution scheme and displacement vector scheme.

2.3.1 Crossing Point Distribution Method (CPD)

In this subsection, the crossing point distribution is considered. The face plane is divided into M regions (see Fig. 8). A set of crossing points in each region is used as a pattern feature.

To divide the face plane into M regions, the center C is first defined as the average coordinate of crossing points (Fig. 8). The standard deviations on the x-axis (σ_x) and the y-axis (σ_y) of the crossing points within the area of 20 cm from the center are calculated. The width and height of the region of interest (ROI) are defined as $2*1.8*\sigma_x$ and $2*1.8*\sigma_y$ respectively, where the factor 1.8 is determined by a pilot test. The crossing points in each region are counted and normalized in Eq. (8).

$$
\begin{aligned}
W &= (w_1, w_2, w_3, \ldots, w_M)^t \\
\bar{w} &= \frac{1}{n}\sum_{a=1}^{n} w_a \\
V_c &= (w_1-\bar{w}, w_2-\bar{w}, w_3-\bar{w}, \ldots, w_n-\bar{w})^t \\
V_\alpha^k &= \left(\frac{w_1-\bar{w}}{\|V_c\|}, \frac{w_2-\bar{w}}{\|V_c\|}, \frac{w_3-\bar{w}}{\|V_c\|}, \ldots, \frac{w_M-\bar{w}}{\|V_c\|}\right)^t
\end{aligned} \tag{8}
$$

where w_a is the number of the crossing points in each region. a is the region number starting from 1 to M. $V_\alpha^{(k)}$ is the normalized feature vector of a facial expression. α is the data number. k is the category number of facial expression. The normalized feature vector $V_\alpha^{(k)}$ is used for the facial expression recognition.

2.3.2 Displacement Vector Method (DVT)

Based on an assumption that the person is known, each facial expression image is compared with the neutral face of the same person. A pair of points on the both facial surfaces which have the same xy-coordinate of the neutral face and the expression face are considered. The pair of crossing points on the face plane corresponding to the pair of points on the two facial surfaces will be used for computing a displacement vector. A displacement vector is the movement vector of the crossing point pointing from the crossing point of the neutral face to the one of the expression face (see Fig. 9.). In other words, the displacement vector reflects the movement of facial surface.

To compute the displacement vectors, the size of the neutral face and the expression face has to be the same. For the face size normalization, two eye positions of the both faces are detected by a texture analysis. The expression face is then normalized to the neutral face. Pair points on the surface of the neutral face and expression face which have the same xy-coordinate are described as and p_i^e , $(i = 1, \ldots, N)$, respectively. Two crossing points c_i^n and $c_i^e$are the positions on the face plane corresponding to p_i^n and p_i^e , respectively. A displacement vector d_i^e is a vector pointing from c_i^n to c_i^e .

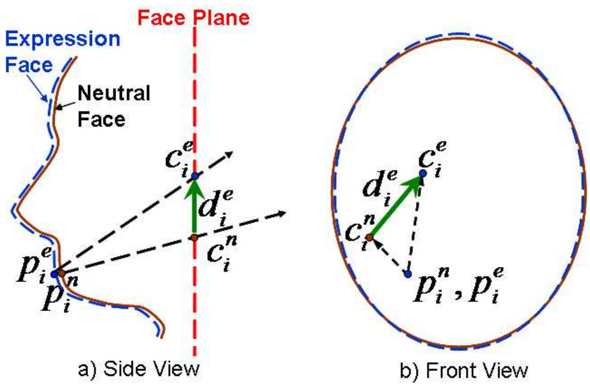

Fig.9: *The concept of displacement vector*

A surface of the expression face is divided into M regions based on the size of neutral face (see Fig. 1.(f)). The displacement vector set V is consisted of M average displacement vectors v_m.

$$
\begin{aligned}
d_i^e &= c_i^e - c_i^n && (9) \\
v_m^e &= \frac{1}{s}\sum_{i=1}^{s} d_i^e && (10) \\
V^e &= (v_1^e, \ldots, v_M^e) && (11)
\end{aligned}
$$

where c_i is the crossing point, d_i is the displacement vector, v_m is the average displacement vector for each region $(m = 1, \ldots, M)$, s is the number of the crossing points in the region m, e is the category of the expression face and n represents the neutral face.

3. EXPERIMENTAL RESULTS AND DISCUSSION

3.1 RESULTS AND DISCUSSION 3.1. Facial

The experiments were done for recognizing four facial expressions (neutral, anger, surprise and smiling) from 22 persons. A 2.5D partial face data set is captured from a viewpoint ranging from -45° to +45° with step 22.5°. However, the range of the viewpoint is limited because over half the facial area is used to evaluate the symmetry plane. As the result of testing the performance of the 3D reconstruction algorithm, we obtained a high reconstruction rate over 97% in [9]. So, in this paper, we focus on the facial expression recognition using successful data of the 3D reconstruction. A support vector machine (SVM) [13] with a radial basis function is used for the recognition by Leave-One-Out method. Data of a person from 22 persons is used as unknown expression data while the

Table 1: *Experimental results by using the crossing point distribution scheme and the displacement vector scheme*

Facial expression		Crossing point distribution method					Displacement vector method				
		−45°	−22.5°	0°	22.5°	45°	−45°	−22.5°	0°	22.5°	45°
Total number of sample	Anger	21	22	22	22	18	21	22	22	22	18
	Neutral	20	22	22	22	22	20	22	22	22	22
	Smiling	20	21	22	21	19	20	21	22	21	19
	Surprise	16	21	22	17	20	16	21	22	17	20
Number of correct recognition	Anger	7	9	17	10	7	13	14	17	16	14
	Neutral	11	12	11	12	14	11	19	19	20	19
	Smiling	12	17	15	16	14	17	20	22	21	19
	Surprise	8	13	19	14	13	10	16	20	14	15
Recognition accuracy (%)	Anger	33.3	40.9	77.3	45.5	38.9	61.9	63.6	77.3	72.7	77.8
	Neutral	55.0	54.5	50.0	54.5	63.6	55.0	86.4	86.4	90.9	86.4
	Smiling	60.0	81.0	68.2	76.2	73.7	85.0	95.2	100.0	100.0	100.0
	Surprise	50.0	61.9	86.4	82.4	65.0	62.5	76.2	90.9	82.4	75.0
Average accuracy (%)		49.35	59.30	70.45	63.41	60.76	66.23	80.23	88.64	86.59	84.81
		60.9					81.6				

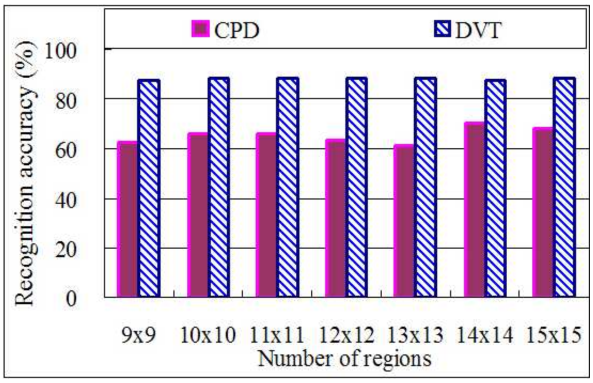

Fig.10: *Facial expression recognition accuracy from different numbers of regions by using the crossing point distribution method (CPD) and the displacement vector method (DVT). The number of the regions is varied from 9×9 to 15×15. The numbers of region which show the highest accuracy of the CPD method and DVT method are 14×14 and 11×11, respectively.*

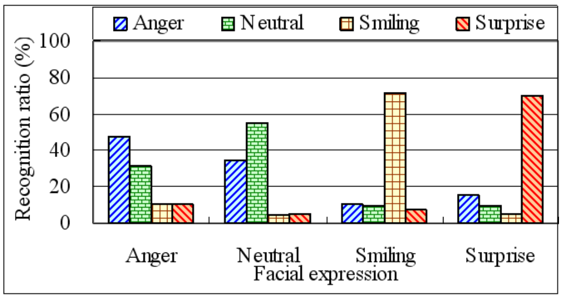

Fig.11: *The incorrect recognition ratio of each expression face by using the crossing point distribution scheme*

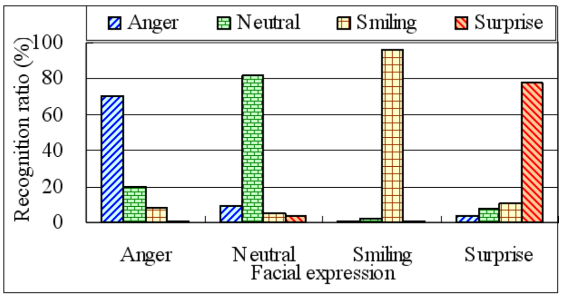

Fig.12: *The incorrect recognition ratio of each expression face by using the displacement vector scheme*

others are used as training data. The unknown expression data is tested and changed cyclically until total 22 persons are recognized. The unknown data is classified into four categories: neutral, anger, surprise and smiling. The algorithm was implemented in the MATLAB complier version R2006a.

To test the performance of the crossing point analysis methods, the following two experimental conditions are considered: 1) crossing point distribution scheme, 2) displacement vector scheme. To maximize the recognition accuracy, we tested to optimize numbers of region on the face plane of two methods by varying the number of region from 81 (9×9) to 225 (15×15) in which the frontal faces from 22 persons were used. Numbers of regions M which showed the highest accuracy for the crossing point distribution method and the displacement vector method are 196 (14×14) and 121 (11×11) regions, respectively, as shown in Fig. 10.

Table 2: Computational time of the crossing point distribution scheme (CPD) and the displacement vector scheme (DVT)

Process	Computational Time (s)	
	CPD	DVT
3D Face Reconstruction (Section 2.1)	16.1	16.1
Face Plane Computation (Section 2.2)	32.8	32.8
Crossing Point Analysis (Section 2.3.1)	16.4	16.4
Displacement Vector (Section 2.3.1)		25.9
Total	65.3	91.2

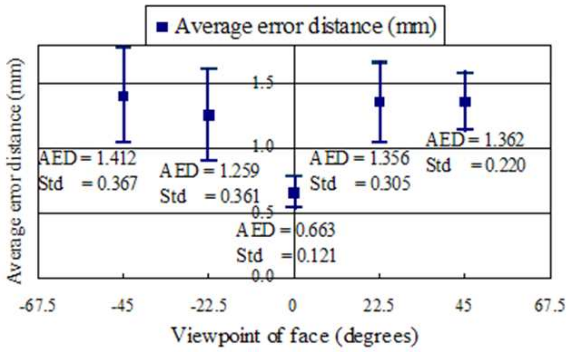

Fig.13: *Average error distance (AED) for all correct surface matching results and the standard deviation (Std) for each angle of view.*

The experimental results of the crossing point distribution scheme and the displacement vector scheme of each facial expression and each viewpoint of facial data are shown in Table 1. The average accuracies of the crossing point distribution method and the displacement vector method are 60.9% and 81.6%, respectively.

In the experimental results, the accuracy of the smiling face is the highest because the characteristic of the crossing point distribution is distinctive from the other. The accuracy of the anger face is the lowest because only the partial surface around the eyebrows changes from the neutral face. Therefore, there are some incorrect recognition between the anger face and the neutral face as shown in Figs 11 and 12.

Furthermore, we also tested the recognition performance of the two schemes by using the training data, which means that the subject being tested also appears in the training set. The experimental condition was set as same as the experimental condition of the unknown data test. The SVM method with a radial basis function is used for recognition. The recognition accuracies of crossing point distribution method and displacement vector method were 100% for four facial expressions.

The computational times of the crossing point distribution method and the displacement vector method are 65.33 and 91.16 seconds, respectively, for one data set using an Intel Core(TM) 2 Duo CPU 3.25GHz processor with 3.25 GB of RAM. The computational cost of each step is shown in Table 2. In the crossing point distribution scheme and the displacement vector scheme, the face plane computation process shows highest computational time. This is because the normal vector n of a face plane in Eq. 4 will be computed repeatedly until the direction of normal is stable. However, the process of computing the displacement vector in the displacement vector scheme also shows high computational time. This is because every point on the surface of expression face will be matched with a point on the reference face which has the same xy-coordinate. The computational time improvement will be discussed in section 3.4.2.

3.2 Comparison of Viewpoint

The results from the two methods in Table 1 show the highest accuracy at 0 degrees. The accuracy for the greater angle tends to be lower. These problems occurred due to the imprecision of the reconstructed face. Although we used only the successful data of the 3D reconstructed face for the recognition in 3.1, each successful data still have a slight error of the reconstruction. To evaluate the facial reconstruction error, a virtual face was aligned to the frontal real face according to six degrees of freedom (x, y, z, roll, pitch and yaw). The nose tip points and the nose ridge vectors of the two faces were used as the initial parameters to match the surfaces of the two faces. The errors between the two faces were minimized by a Nelder-Mead simplex method [14]. The matching distance was then computed.

The average error distances of each viewpoint of the face are shown in Fig. 13. The average error distances for greater angles tend to be bigger than the one of the frontal face. The errors of the reconstructed faces make the recognition accuracy for the greater angle of viewpoints slightly low.

3.3 Comparison of Two Schemes

In this subsection, the performances of two schemes, the crossing point distribution scheme and the displacement vector scheme, are discussed.

The recognition accuracies of the displacement vector scheme and the crossing point distribution scheme are compared as shown in Figs. 14 and 15. Fig. 14 shows the accuracy for each viewpoint. Fig. 15 shows the accuracy for each facial expression. The accuracy of the displacement vector scheme is higher because the displacement vector represents the direction change due to

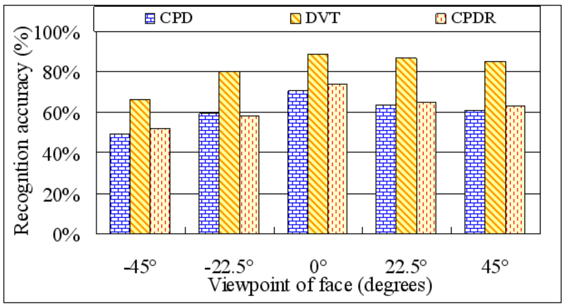

Fig.14: *The expression recognition accuracy from different angles of the face rotation by using the crossing point distribution method (CPD), the displacement vector method (DVT) and the crossing point distribution method with reference faces (CPDR).*

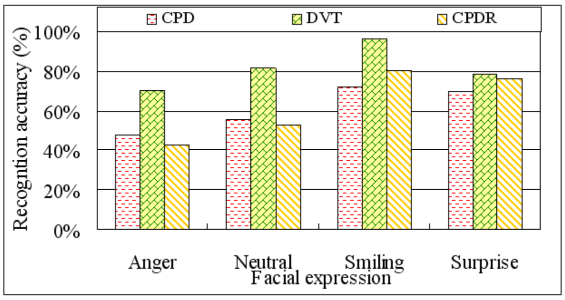

Fig.15: *The expression recognition accuracy from different facial expressions by using the CPD method, the DVT method and the CPDR method*

facial expression change. However, the displacement vector scheme needs the reference neutral face. While the crossing point scheme do not use the reference face. The computational time of the displacement vector scheme is also longer than the crossing point distribution scheme. As the displacement vector is computed from two points on the facial surface of the expression face and the neutral face which have the same xy-coordinate, the expression face has to be aligned to the neutral face. Therefore, the longer computational time is spent for aligning the size of expression face to the neutral face.

3.4 Discussion

3.4.1 Crossing Point Distribution Method with Reference Faces (CPDR)

In this section, a reference face is also used in the crossing point distribution scheme. Each facial expression is compared with a reference neutral face of the same person (see Eq. (12)). Data used for recognition in this method is:

$$\mathbf{Y}_{\alpha}^{(k)} = \mathbf{V}_{\alpha}^{(k)} - \mathbf{V}_{\alpha}^{(n)} \tag{12}$$

where $\mathbf{Y}_{\alpha}^{(k)}$ is the difference vector. $\mathbf{V}_{\alpha}^{(k)}$ is the normalized feature vector by Eq. (8). V^n is the

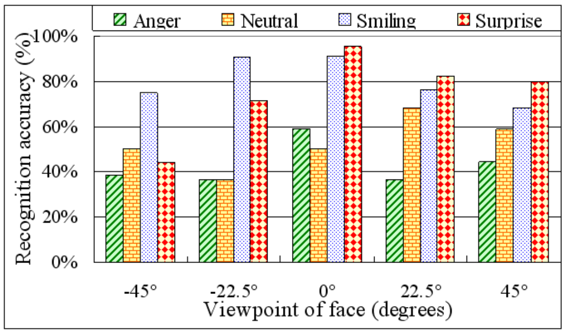

Fig.16: *The accuracy of facial expression recognition for different face directions of -45°, -22.5°, 0°, +22.5° and +45° by using the crossing point distribution method with reference faces.*

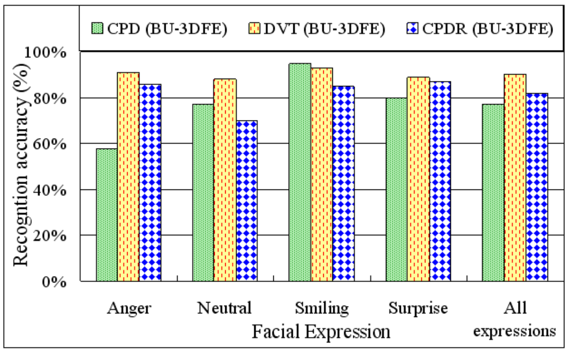

Fig.17: *The accuracy of facial expression recognition by using the BU-3DFE database.*

normalized feature vector of a reference neutral face. The difference vector $\mathbf{Y}_{\alpha}^{(k)}$ is for the facial expression recognition.

The recognition accuracy for each viewpoint of the face is shown in Fig. 16. The average accuracy using the reference crossing point distribution scheme is 62.6%. We also compare the result with the crossing point distribution and the displacement vector method as shown in Figs 14 and 15. The recognition accuracy by this method is a little higher than the result by the crossing point distribution method.

3.4.2 Recognition Accuracy by using the BU3DFE Database

The performance of the proposed method is compared by using a publicly available database. The Binghamton university 3D facial expression database (BU-3DFE) proposed by Lijun [15] is only one 3D facial expression database available currently and is used to test the performance of the primitive surface feature distribution method [8]. The database consists of data from 100 subjects. Each subject performed seven expressions. A full frontal face is reconstructed from two views about +45° and -45° of 3D face data. The primitive surface feature distribution method is a subject-independent facial expression recognition method. By using the primitive surface feature distribution method and the BU-3DFE database, the recognition accuracy was 83.6%.

In this experiment, four facial expressions, which are anger, neutral, smiling and surprise, are used. By using the crossing point distribution method (CPD), the displacement vector method (DVT) and the crossing point distribution method with reference faces (CPDR), the correct recognition rates are 77.5%, 90.3% and 82.0%, respectively as shown in Fig. 17. This experiment shows that the quality of BU-3DFE database is better than that of our database, comparing with the experimental results in section 3.1. Although the DVT method has limitation in subject-dependent and the CPD method has a little lower accuracy than the primitive surface feature distribution method, our algorithm can be used for input face data which only one viewpoint is available.

According to the primitive surface feature distribution method, seven local regions covering the most expressive areas on a human face are used for recognition. In the proposed method, data of a whole face is used. Therefore, in the further enhancement of the proposed method, local regions on the facial surface which are expressive will be used instead of whole face data for improving the computational time and the recognition accuracy.

4. CONCLUSIONS

In this paper, we have proposed a novel algorithm for facial expression recognition from a 2.5D partial face image. An input 2.5D partial face image is captured from any viewpoint ranging from -45° to +45° around the y-axis. A 3D expression face is reconstructed by using our 3D face reconstruction algorithm. The facial expression is represented in terms of a change of crossing points on a face plane. Two crossing point analysis schemes, a crossing point distribution scheme and a displacement vector scheme, are used to analyze the crossing points for the recognition. The experiments were done for four facial expressions (neutral, anger, surprise and smiling) from 22 persons. Average recognition accuracies using the crossing point distribution scheme and the displacement vector scheme were 60.9% and 81.6% respectively. Reference faces are necessary for the displacement vector scheme while the crossing point distribution scheme does not need reference faces.

The BU-3DFD standard database was also used to test the proposed algorithm. By using the crossing point distribution method, the displacement vector method and the crossing point distribution method with reference faces, the correct recognition rates were 77.5%, 90.3% and 82.0%, respectively. Although the displacement vector method has a limitation in subject-dependent and the crossing point distribution method show a little lower accuracy than the primitive surface feature distribution method [8], our algorithm can be used for a partial face data which only one viewpoint is available while the primitive surface feature distribution method is appropriate for only the complete frontal face data.

Our algorithm can be applied to applications of robot-vision or human-machine interface in case a partial face data set can be captured. For the further work, additional methods such as a method for selecting the expressive areas on a human face and hierarchical classification algorithm will be used to improve the performance of the system.

References

[1] M. Pantic, and L.J.M. Rothkrantz, "Automatic Analysis of Facial Expressions: the State of the Art," *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 22, No. 12, pp. 1424-1445, Dec. 2000.

[2] B. Fasel, and J. Luettin, "Automatic Facial Expression Analysis: A Survey," *Pattern Recognition*, 36(1):259-275, 2003.

[3] X. Lu, A.K. Jain and D. Colbry, "Matching 2.5D Face Scans to 3D Models," *IEEE Trans. on Pattern Anal. Mach. Intell.*; 28(1), pp. 31-43 (2006).

[4] M. Pantic, and L.J.M. Rothkrantz, "Expert System for Automatic Analysis of Facial Expression," *Image and Vision Computing Journal*, 18(11):881-995, 2000.

[5] H. Micheal, A. Wiratanaya and K.F. Kraiss, "Facial Expression Modeling from Still Images using a Single Generic 3D Head Model," *Proceeding of the 28th DAGM Symposium*, Berling, pp. 324-333, September, 2006.

[6] I.A. Essa, A.P. Pentland, "Coding, Analysis, Interpretation, and Recognition of Facial Expressions," *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 19, No. 7. pp. 757-763, July, 1997.

[7] Li and A. Barreto, "An Integrated 3D Face-Expression Recognition Approach," *Proceeding of ICASSP2006*, pp. 1132-1135, Toulouse, 2006.

[8] J. Wang, L. Yin, X. Wei and Y. Sun, "3D Facial Expression Recognition Based on Primitive Surface Feature Distribution," *Proceeding of CVPR'06*, New York, pp. 1399-1406, October, 2006.

[9] Th. Charoenpong, T. Shogo and H. Hase, "3D Face Reconstruction by 2.5D Partial Face Data," *The Journal of the IIEEJ*, Vol. 37, No. 4, pp. 476-485, 2008.

[10] R. Halir, and J. Flusser, "Numerically Stable Direct Least Squares Fitting of Ellipses," *In Skala, V (ed) Proc. of Int. Conf. in Central Europe on Computer Graphics, Visualization and Interactive Digital Media*, pp. 125-132, 1998.

[11] M. Benz, X. Laboureux, T. Maier, et al, "The Symmetry of Faces," *Proc. of VMV'2002*, Erlangen, Germany, pp. 43-50, 2002.

[12] H. Hase, M. Yoneda, T. Kasamatsu and J. Kato, "A Proposal for a Face Plane," *Proceeding of the* 7^{th} *International Conference on Image Processing*, pp. 333-336, Greece, 2001.

[13] C.-C. Chang and C.-J. Lin, "LIBSVM: a library for support vector machines". http://www.csie.ntu.edu.tw/ cjlin/libsvm.

[14] Nelder, J. A. and Mead, R. "A Simplex Method for Function Minimization." *Computer Journal.* 7, 308-313, 1965.

[15] Y. Lijun, W. Xiaozhou, S. Yi, W. Jun, J.R. Matthew, "A 3D Facial Expression Database For Facial Behavior Research," *7th Int. Conf. on Automatic Face and Gesture Recognition (FGR06)*, pp. 211 - 216, 2006.

Theekapun Charoenpong received his BEng degree in electronics engineering from King Mongkut's Institute of Technology Ladkrabang, Thailand, in 2001, MEng degree in electronics and information engineering from King Mongkut's University of Technology Thonburi, Thailand, in 2005 and a D.Eng. degree in Graduate School of Engineering, University of Fukui, Japan, in 2008. Now, he works as a lecturer at Biomedical Engineering Programme, Faculty of Engineering, Srinakharinwiroj University, Thailand.

Shogo Tokai received the BS, MS and Ph.D. degrees from Nagoya University, in 1991, 1993, and 1996 respectively. He is currently an associate professor at the Department of Fiber Amenity Engineering, University of Fukui. His research interests include computer graphics, computer vision, and multiple view video processing. He is a member of the Information Processing Society of Japan.

Hiroyuki Hase received his B.E. degree in electrical engineering from Toyama University in 1971, and Ph.D degree from Tohoku University in 1989. From 1975 to 1989, he was an assistant in Department of Electronics and Information Engineering at Toyama University. From 1989 to 1993, he was a lecturer in the same department. From 1993 to 2003, he was an associate professor in the Department of Intellectual Information Systems Engineering at Toyama University. From April 2003, he is a professor of the Graduate School of Information Science, University of Fukui. His current research interests are document image analysis, character recognition and facial expression analysis. He is a member of the Information Processing Society of Japan, the Institute of Image Information and Television Engineers, and the Institute of Electronics, Information and Communication Engineers.