Boosting Radial Strings for 3D Face Recognition with Expressions and Occlusions

Xun Yu, Yongsheng Gao
School of Engineering
Griffith University
Nathan, QLD, Australia
xun.yu@griffithuni.edu.au, yongsheng.gao@griffith.edu.au

Jun Zhou
School of Information and Communication Technology
Griffith University
Nathan, QLD, Australia
jun.zhou@griffith.edu.au

***Abstract*—In this paper, we present a new radial string representation and matching approach for 3D face recognition under expression variations and partial occlusions. The radial strings are an indexed collection of strings emanating from the nose tip of a face scan. The matching between two radial strings is conducted through a dynamic programming process, in which a partial matching mechanism is established to find those un-occluded substrings effectively. Moreover, the most discriminative and stable radial strings are selected optimally by the well-known AdaBoost algorithm to achieve a composite classifier for 3D face recognition under facial expression changes. Experimental results on the GavabDB and the Bosphorus databases show that the proposed approach achieves promising results for human face recognition with expressions and occlusions.**

Keywords*— *face recognition; string matching; facial curves; machine learning; feature selection

I. Introduction

Face is very popular as biometric modality since its natural, contact-free, nonintrusive and psychologically supported. However, 2D image-based face recognition technologies still meet the huge challenge to identify humans when considering non-cooperative and uncontrolled scenarios. As the technology for capturing the shape of facial surface becomes simpler and cheaper, 3D image-based face recognition has emerged as a major research trend in the last few years. Compared to conventional 2D face recognition, 3D face recognition is expected to be less sensitive to illumination and pose variances, because the actual geometric information residing in the 3D data is inherently robust to changes in illumination and pose which adversely affect 2D data [1]. However, handling of expressions and occlusions is still a challenging task for most of the existing 3D face recognition approaches.

In this work, we present a novel radial string representation and similarity measuring technique to recognize 3D facial scans under expressions and occlusions. Basically, a 3D face is represented by an indexed collection of attributed strings emanating from the nose tip. The matching of two 3D faces is performed by the matching of the corresponding attributed string sets. A string-to-string matching scheme is developed by using dynamic programming (DP), which includes a partial matching mechanism that can effectively eliminate those occluded parts during the matching process. Moreover, we use the well-known AdaBoost algorithm to learn a strong classifier which identifies and then combine the most stable and discriminative radial strings for 3D face recognition, especially under expression variances.

In recent years, a large number of papers have been concerned with 3D face recognition based on local geometric features. Compared to holistic face recognition approaches, local face recognition approaches are more robust to expression variances and partial data. In [2], a 3D face was encoded into a 3D ridge image, which showed the locations of ridge curves around the important facial regions on the face. Berretti et al. [3] used concave and convex regions to represent a 3D face scan, a compact graph was then constructed to measure spatial relationships between those regions. Moreno et al. [4] extracted thirty local geometrical features from a 3D face surface by using the HK algorithm [5], and both the PCA and SVM matching schemes were tested in their work.

Unlike the problem of expression handling has been extensively studied in recent years, there are only relatively few studies directly attacking the occlusion problem in the literature. Colombo et al. introduced an occlusion detection strategy [6] into their face recognition approach. Alyuz et al. [7] used the Average Region Models (ARMs) for 3D face recognition under occlusions. The facial area was manually divided into several meaningful components, and the registration of faces was performed by separate dense alignments to relative ARMs. Berretti et al. [8] represented 3D facial scans by a set of facial curves, which were generated by connecting SIFT keypoints. The most saliency curves were selected to perform partial face matching. Drira et al. [9] represented facial surfaces by radial curves emanating from the nose tips. A Riemannian framework was developed to analysis these curves and to measure facial deformations. The representation and the elastic Riemannian framework are robust to handle expressions, pose variations, missing parts and partial occlusions. Lei et al. [10] explored an efficient 3D face recognition that encoded a face scan by a set of local-feature descriptors called KMTS, then a TPWCRC framework was proposed to address the problem of 3D partial face recognition.

The rest of paper is organized as follows: Section 2 describes radial string extraction. In Section 3, we explain the edit cost operations and the similarity measurement of radial strings. Section 4 gives the radial string selection step by using the AdaBoost algorithm. Section 5 presents the experimental results. Finally, we conclude this work in Section 6.

978-1-5090-2896-2/16/$31.00 ©2016 IEEE

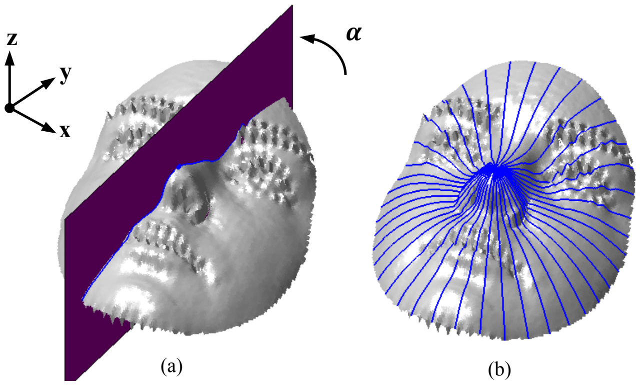


Fig. 1. Extraction of radial curves. (a) Illustates the intersection between the 3D face scan and a plan to form two radial curves. (b) Illustrates the collection of radial curves

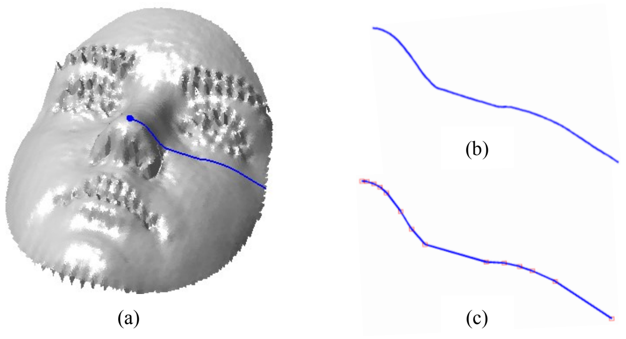


Fig. 2. (a) A 3D face scan with the radial curve at 18 degree. (b) Enlarge the radial curve in (a). (c) The corresponding raidal string to radial curve (b), where red squares are the corner points.

II. Radial String Extraction

Give a normalized facial range image F, a radial curve C is obtained by slicing the facial scan F by a plane P that rotates with the respect to the nosetip (chosen as the origin of the facial scan image) and makes an angle α with the reference xz-plane (see Fig. 1(a)). By repeating this step to extract radial curves, a 3D face representation is constructed by considering N radial curves taken at equal angular interval in [0,360](see Fig. 1(b)).

However, without considering the inherent local structural characteristics inside radial curves, such radial curve based face representation lacks the local structural discriminative power. In this paper, we propose a new 3D face representation based on radial strings, which combines the structural information with spatial information. In practice, a corner detection process based on Douglas-Peucker algorithm [11] is then applied to generate radial strings after extracting radial curves, which are formed by connected line segments. Fig. 2(a) illustrates a 3D face model where a radial curve C originated from the nose tip. In Fig. 2(b), we enlarge the radial curve C and show the corresponding curve string S in Fig. 2(c), where line segment $s(l,\boldsymbol{n},x,y,z)$ is the primitive. l, $\boldsymbol{n}$ and (x,y,z) are the length, directional vector and midpoint location of the line segment, respectively. The directional vector $\boldsymbol{n}$ is defined as the vector that directs to the ending corner point from the starting corner point of the line segment.

III. Radial String Matching

The goal of string matching algorithms is to find a sequence of elementary edit operations which transform one string into another at a minimal cost. However, the conventional string matching algorithm [12] cannot tackle the problem of face recognition under partial occlusions. In order to overcome this limitation, we propose a new attributed string matching algorithm to perform partial matching for 3D face recognition under partial occlusions.

In this section, we first introduce the edit cost functions used in our algorithm and then provide details of the proposed string-to-string matching algorithm.

A. Edit Cost Functions

We start by considering two radial strings S^A and S^B where $S^A\langle i\rangle$ and $S^B\langle j\rangle$ are the ith and jth primitives in S^A and S^B with attributes $(l_i,\boldsymbol{n}_i,x_i,y_i,z_i)$ and $(l_j,\boldsymbol{n}_j,x_j,y_j,z_j)$ respectively. Following to this, a substring of S is denoted by $S\langle i\!:\!j\rangle$ starting at position i and ending at position j in radial string S.

The cost function for change operation from $S^A\langle i\rangle$ to $S^B\langle j\rangle$ is defined as

$$C(S^A\langle i\rangle \rightarrow S^B\langle j\rangle) = \left|l_i - l_j\right| + f(\langle\boldsymbol{n}_i,\boldsymbol{n}_j\rangle) + \left\|(x_i-x_j, y_i-y_j, z_i-z_j)\right\|_2 \quad (1)$$

where $\langle\boldsymbol{n}_i,\boldsymbol{n}_j\rangle \in [0,90°]$ is the angle difference between two primitives, and $f(x) = x^2/w$ is a non-linear function to penalize large angle deviation resulted from inter-class difference but ignore small variation derived from data noise or intra-class difference. w is a weight to be determined experimentally in section 5.

A merge operation [13] with modified merge cost is used to tackle the problem of adding, missing and shifting of feature points. The cost of merging a sequence of k primitives $S^A\langle i-k+1\!:\!i\rangle$ into one merged primitive $S_k^A\langle i\rangle$ is defined as

$$C(S^A\langle i-k+1\!:\!i\rangle \rightarrow S_k^A\langle i\rangle) = f\Big(\frac{k-1}{l^k}\sum_{q=i-k+1}^{i}\langle\boldsymbol{n}^k,\boldsymbol{n}_q\rangle \times l_q\Big) \quad (2)$$

where l^k and $\boldsymbol{n}^k$ are the length and the directional vector of merged primitive S_k^A , l_q and $\boldsymbol{n_q}$ are the length and the directional vector of primitive $S^A\langle q\rangle$ in $S^A\langle i-k+1\!:\!i\rangle$ before merging. So the cost function for a change operation after merge can be rewritten as

$$C(S_k^A\langle i\rangle \rightarrow S_l^B\langle j\rangle) = \left|l_i^k - l_j^l\right| + f(\langle\boldsymbol{n}_i^k,\boldsymbol{n}_j^l\rangle) + \left\|(x_i-x_j, y_i-y_j, z_i-z_j)\right\|_2 \quad (3)$$

If $k=1$ and $l=1$, no merge is performed and the above change operation reduces to the conventional one-to-one change operation (see Eq. (1)).

B. *Similarity Measure via Dynamic Programming*

In order to compare the shapes of two radial strings, we compute the similarity between them by utilizing the edit cost functions via DP. Let $S^A = S^A\langle 1\rangle \cdots S^A\langle m\rangle$ and $S^B = S^B\langle 1\rangle \cdots S^B\langle n\rangle$ be radial strings from a probe and a gallery facial scans, where m and n are numbers of primitives in S^A and S^B. Accordingly to this, a similarity matrix **S** with a size of $(m+1)\times(n+1)$ is set up through the DP, which is defined as follows:

$$\mathbf{S} = \begin{pmatrix} s(0,0) & s(0,1) & \cdots & s(0,n) \\ s(1,0) & s(1,1) & \cdots & s(1,n) \\ \vdots & \vdots & \vdots & \vdots \\ s(m,0) & s(m,1) & \cdots & s(m,n) \end{pmatrix} \tag{4}$$

where each element $s(i,j)$ stores the maximum accumulated similarity value between the strings (or substrings) $S^A\langle 1{:}i\rangle$ and $S^B\langle 1{:}j\rangle$, as defined in Eq. (5).

$$s(i,j) = \max_{k,l}(\Lambda(k,l)) \tag{5}$$

where $\Lambda(k,l)$ is the accumulated similarity resulting from edit operations as defined below.

$$\Lambda(k,l) = \begin{cases} s(i-k, j-l) + \Delta & if\ \Delta \geq 0 \\ 0 & otherwise \end{cases} \tag{6}$$

To seamlessly include the partial matching mechanism in our approach, the similarity parameter Δ in (6) is defined as

$$\begin{aligned} \Delta = \Upsilon - \ [&C(S_k^A\langle i\rangle \rightarrow S_l^B\langle j\rangle) \\ &+C(S^A\langle i-k+1{:}i\rangle \rightarrow S_k^A\langle i\rangle) \\ &+C(S^B\langle j-l+1{:}j\rangle \rightarrow S_l^B\langle j\rangle)] \end{aligned} \tag{7}$$

where Υ is a threshold to decide whether $S_k^A\langle i\rangle$ and $S_l^B\langle j\rangle$ are matched substrings. If the combined cost of merging and changing operations is greater than Υ (which gives $\Delta < 0$), these primitives are considered as outliers. This is achieved by setting $\Lambda(k,l) = 0$ when $\Delta < 0$.

Algorithm 1 gives the detailed process of creating the similarity matrix **S** , where $limit_i^A$ and $limit_j^B$ are the controlling upper limits on the number of primitives to be merged into a new one in S^A and S^B respectively. In our experiments, the controlling upper limits are set as 5 to balance the recognition performance and computing cost. To compute the similarity score between S^A and S^B , we fuse all the similarity values of substrings from S^A with substrings from S^B , which is defined as follows:

$$Score(S^A, S^B) = \textstyle\sum_{i=1}^{f} S_i \tag{8}$$

The term S_i is the similarity (the ith maximal element in **S** matrix table) of the ith best similar substrings between two radial strings and f is the number of best similar substrings.

Algorithm 1. Generate Similarity Matrix **S**

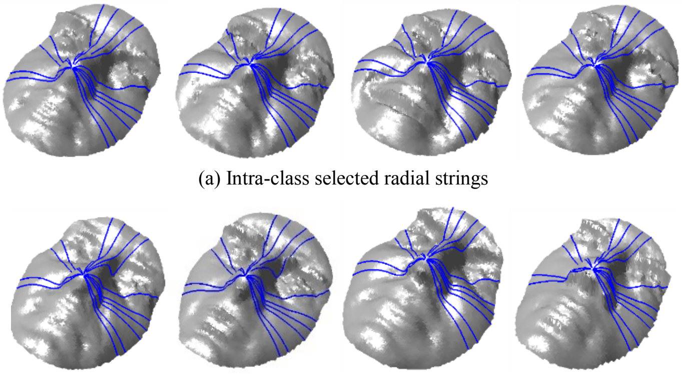
```
Input: $S^A = S^A\langle 1\rangle \cdots S^A\langle m\rangle$, $S^B = S^B\langle 1\rangle \cdots S^B\langle n\rangle$
Initialization: $s(0,0) = 0$, $s(i,0) = 0$, $s(0,j) = 0$
For $i = 1\ to\ m,\ j = 1\ to\ n$ do
    For $k = 1\ to\ min(limit_i^A, i), l = 1\ to\ min(limit_j^B, j)$ do
        $\Delta = \Upsilon - [C(S_k^A\langle i\rangle \rightarrow S_l^B\langle j\rangle)$
                $+C(S^A\langle i-k+1{:}i\rangle \rightarrow S_k^A\langle i\rangle)$
                $+C(S^B\langle j-l+1{:}j\rangle \rightarrow S_l^B\langle j\rangle)]$
        $\Lambda(k,l) = \begin{cases} s(i-k,j-l)+\Delta & if\ \Delta \geq 0 \\ 0 & otherwise \end{cases}$
    $s(i,j) = \max_{k,l}(\Lambda(k,l))$
Output: $\mathbf{S}(S^A, S^B)$
```

IV. Boosting for Radial String Selection

As described in Section 3, we introduced the string matching algorithm to compute the similarity between two radial strings, in which a partial matching mechanism is established to effectively eliminate those occluded parts and find the most discriminative parts during the matching process. However, after we approximately represent 3D facial scans by indexed collections of radial strings, comparing two 3D facial scans under expression variations may still run into trouble. This is because the shapes of radial strings are affected by changes in facial expression. For example, a smile greatly changes the shapes of the strings in the lower part of a face. In order to address the recognition problems with expression variances, a feature selection step is introduced in our approach to identify the most stable and discriminative strings.

In our approach, we leverage the classic machine learning algorithm, AdaBoost introduced by Freund and Schapire in [14], to learn a strong classifier based on the weighted selection of weak classifiers. In our case, the individual strings are used to build the weak classifiers. In order to apply real AdaBoost for face recognition, we adopt the intra-personal and inter-personal difference method to convert face recognition to a two-class classification problem. Formally, we consider a set of pairs $(x_n^k, y_n)_{1\leq n\leq N}$ corresponding to similarity scores between corresponding radial strings, where y_n can equal to 1 or -1 referring to intra-personal similarity score and inter-personal similarity score, respectively. For those radial strings, the boosting can then be used to optimize their performance. AdaBoost is based on iterative selection of weak classifiers by using a distribution of training samples. In each iteration, one string is selected by choosing the corresponding weak classifier which has the lowest error. After T iterations, the most discriminative T strings are selected. The pseudocode of the AdaBoost algorithm is given in Algorithm 2.

The GavabDB database [15] is used to train AdaBoost classifier for face recognition under expression variance. A subset of the GavabDB database containing 305 scans from 61 subjects is used in our experiments. For each subject, there are



(a) Intra-class selected radial strings

(b) Inter-class selected radial strings

Fig. 4. The most stable and discrminative raidal strings slected by AdaBoost for face recogntion.

Algorithm 2. AdaBoost Algorithm

Input: Training set $\{(x_1^k, y_1), \cdots, (x_N^k, y_N)\}$, where x_i^k is the similarity score of kth radial string and $y_i = 1, -1$ indicates intra-personal and inter-personal samples respectively.

Initialization: $\omega_{1,i} = 1/2m, 1/2l$ for $y_i = 1, -1$ where m and l are the numbers of match and non-match score, respectively.

```
1: For $i = 1, \ldots, T$
2:    Normalize the weights, $\omega_{1,i} \leftarrow \omega_{1,i} / \sum_{i=1}^{N} \omega_{1,i}$ so that $\omega_t$
      is a probability distribution.
3:    For each string $S_j$, train a weak classifier $h_j$ which uses a
      single string. The error $\varepsilon_j$ is evaluated with the weight $\omega_t$:
          $\varepsilon_j = \sum_i \omega_{t,i} |h_j(x_i) - y_i|$
4:    Choose the classifier $h_j$ with the lowest error.
5:    Update the weights:
          $\omega_{t+1,i} = \omega_{t,i} \gamma_t^{1-e_i}$
      Where $\gamma_t = (\varepsilon_t)/(1 - \varepsilon_t)$ and $e_i = 0$ if example $x_i$ is
      classified correctly, $e_i = 1$ otherwise.
```

Output: The final strong classifier $\boldsymbol{H}$ is a weighted linear combination of classifiers with the lowest error found in each iteration.

$$\boldsymbol{H}(x) = \operatorname{sign}\left[\sum_{t=1}^{T} \log \frac{1}{\gamma_t} h_t(x)\right]$$

two frontal neutral scans and three expressive scans with a smile, a laughing, and an arbitrary expression respectively, and the first neutral scan is taken as gallery model, while the remaining is used as probes. For each radial string, we compute the 305 × 305 similarity score matrix. Then, the similarity score matrix is split into two parts: Gallery versus Probe (61 × 244) for testing and Probe versus Probe (244 × 244) for training. Thus the testing and training samples are disjoint in our experiments.

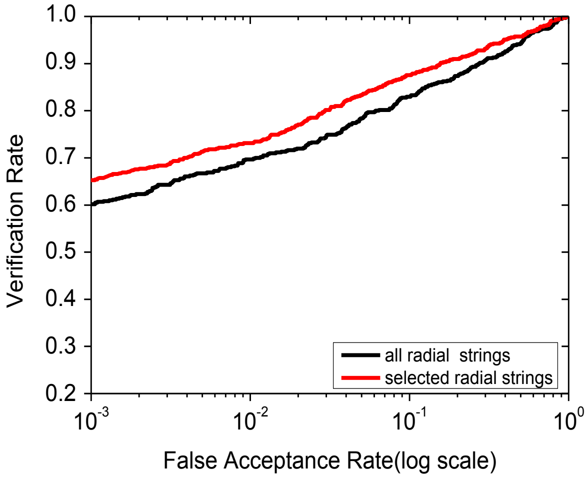


Fig. 3. ROC curves produced from the training set.

As illustrated in Fig. 4, we find that the selected radial strings returned by AdaBoost do not pass through two regions, the mouth and eye/eyebrow regions, which are easily affected by expressions such as laughing and surprising, respectively. Here, we present the illustrations in both cases: the first row gives the selected radial string of the identical person with different expressions (intra-class), while the second row shows the selected radial strings of different subjects (inter-class).

In order to demonstrate the effectiveness of the string selection step, the ROC curves [16] generated by the Probe versus Probe score matrix are plotted in Fig. 3 for both conditions of using selected radial strings and all radial strings. Note that the string selection step increases the 0.1% FAR verification rate from 60.2% to 66.3%.

After the radial string selection by AdaBoost, the matching between two face scans F^A and F^B is conducted by the fusion of all the scores of selected radial strings.

$$Score(F^A, F^B) = \sum_{n=1}^{N_s} Score(S_n^A, S_n^B) \tag{9}$$

where N_s is the number of selected radial strings in each face scan.

V. Experimental Results

In the following, we firstly analyze the effect of parameters to our algorithm. Then, we test our proposed approach with a set of experiments on the GavabDB [15] and the Bosphorus [17] databases.

All face images are preprocessed in the same manner as [18]. A template with three manually selected features points (the nose tip and two inner eye corners) is used to initially align the cropped face images and a recursive-ICP algorithm [9] is then applied to finely align those face images.

A. Databases

The GavabDB database consists of 549 3D facial surface images from 61 individuals. It was created by the GAVAB research group of computer science department at the University of King Juan Carlos. All the subjects of this database are Caucasian (16 female and 45 male) and most of them aged between 18 to 40. For each subject, there are 9 different images captured under different settings. More specifically, two frontal images with neutral expression, four neutral images with pose (looking left, right, up and down), and three frontal gesture images with different facial expressions (smile, laugh and random gesture chosen by the person respectively).

The Bosphorus database was constructed by the Boğaziçi University to provide a comprehensive 3D face databases with expressions, poses, and partial occlusions. In this database, there are 4652 face scans of 105 subjects (60 men and 45 women). Most of the subjects are Caucasian and aged between 25 and 35. The Bosphorus database contains two versions: v1 and v2 sets. In the Bosphorus v1 set, there are 34 subjects with 31 scans per each: 10 types of expressions, 13 different poses, 4 partial occlusions, and 4 neutral scans. Multiple neutral scans for each subject are also available so that 3D face recognition algorithms can be evaluated for neutral scans. Compared to the Bosphorus v1 set, the Bosphorus v2 set has more facial expression changes. In the Bosphorus v2 set, there are 47 subjects having 53 scans. Each subject has 34 scans for different expressions, 13 scans for pose variations, 4 partial occluded and 1 (or 2) neutral faces.

B. Determination of Parameters

In this section, we determine parameters involved in the proposed method, i.e. the parameter of weight w in Eq. (1) and the parameter of threshold ϒ in Eq. (7).

In theory, the parameter ϒ is a threshold to decide the strength of partial matching. When ϒ is close to 0, all string pairs are considered as outliers and are discarded in the matching process, unless they are identical string pairs. As ϒ increases, the percentage of outlier strings decreases. The parameter w is a weight to balance the contribution of spatial and structural distance. When the value of w is small, the contribution of spatial distance to the overall edit operation cost is small. With the increase of w, the contribution from the spatial distance increases.

The parameter ϒ and w are determined experimentally on a training dataset. 105 people with 2 neutral 3D faces per person from the Bosphorus database are used to create a training dataset. The first neutral 3D face of each subject is used to construct the model database, and the second neutral 3D face is used as query image. The system performance with different values of ϒ and w is illustrated in Fig. 5. It can be seen that the recognition rate increases when ϒ increases. The system reaches its best performance when ϒ = 8, Then it drops when ϒ increases further. Hence, we set ϒ = 8 and w = 120 in the rest of experiments in this study.

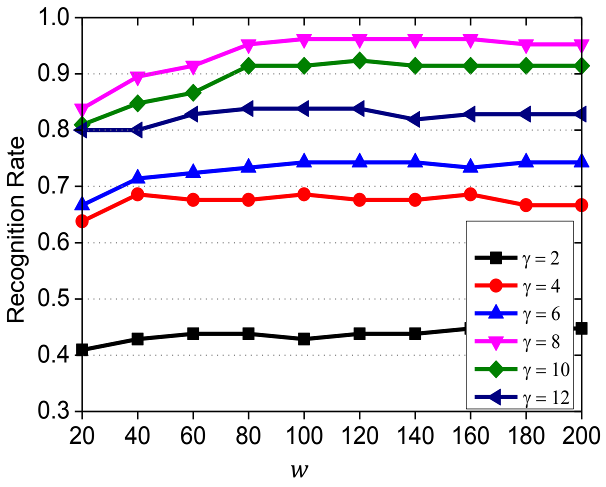


Fig. 5. The effect of *w* and ϒ on the recognition rate.

C. Results on the GavabDB Database

In this section, experiments on the GavabDB database are conducted to evaluate the effects of facial expression on the system performance. As mentioned in Section 3, we separate a subset of the GavabDB database into two parts for testing and training, respectively. In our testing experiment, we use the remaining Gallery versus Probe (61 × 244) similarity matrices and then divide the probe set into two categories: the neutral (contains 61 neutral scans) and the expressive (contains 183 expressive scans).

The experimental results for different categories of probe faces are summarized in Table 1. It is encouraging to find that, compared with other three local geometrical features based methods from Moreno et al. [4], Mahoor and Abdel-Mottaleb [2] and Berretti et al. [3], the proposed approach gives the highest rank-1 recognition rate 82.9 % for expressive faces. This is due to both the radial string representation and feature selection step in our approach.

TABLE I. Comparison of rank-1 recognition rates using different methods: (A) Neutral, (B) Expressive, (C)Neutral+Expressive

	Moreno[4]	Mahoor [2]	Berretti [3]	Ours
(A)	90.16%	95%	94%	95.0%
(B)	77.90%	72%	81%	82.9%
(C)	-	-	82.66%	86.1%

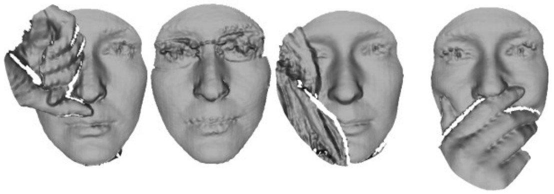


Fig. 6. Scans of one subject in the Bosphorus database with four occlusion types.

TABLE II. COMPARISON OF RANK-1 RECOGNITION RATES ON THE SUBSET OF OCCLUSION ON THE BOSPHORUS DATABASE.

Approaches	eye	mouth	glasses	hair	all
Colombo[6]	91.1%	74.7%	94.2%	90.4%	87.6%
Drira [9]	98.9%	78.5%	94.2%	85.2%	89.2%
Lei [10]	90.2%	94.3%	96.2%	98.0%	94.2%
Ours	96.1%	96.1%	94.2%	97.2%	95.3%

D. *Results on the Bosphorus Database*

In this section, to compare with the benchmark methods on the Bosphorus database, we follow the same experiment protocol used by Colombo et al. [6]. A neutral scan for each person is selected to form a gallery set of size 105. The probe set contains 360 scans that have four different types of occlusions as illustrated in Fig. 6.

Table 2 shows the recognition results for different approaches under various types of occlusions. In the four experiments under varying occlusions, the proposed method has significantly outperformed other methods in one type of occlusion, and match or close to the best results in three other cases. The overall recognition rate is 7.7%, 6.1%, and 1.1% higher than the methods from Colombo et al. [6], Drira et al. [9] and Lei et al. [10], respectively. This has confirmed that our method can effectively handle partial occlusions.

VI. CONCLUSIONS

In this work, we proposed a local geometric feature based 3D face recognition approach and tested its performance on the GavabDB and the Bosphorus databases. A facial scan is represented by a set of radial strings that integrates local structural information with spatial information of a 3D surface. The matching of two 3D facial scans is conducted by matching two radial string sets through string-to-string matching, which is able to effectively find the most similar substrings for each corresponding radial string pair. To further improve the recognition performance especially under facial expression changes, a radial string selection step was implemented in our approach by using the AdaBoost algorithm. The performance of the proposed approach has been evaluated and compared with several state-of-the-art approaches. Preliminary experimental results demonstrated the feasibility and effectiveness of our approach for tackling problems of facial expressions and partial occlusions.

REFERENCES

[1] K. Bowyer, K. Chang, and P. Flynn, "A survey of approaches and challenges in 3D and multi-modal 3D+2D face recognition," *Computer Vision and Image Understanding,* vol. 101, pp. 1-15, 2006.

[2] M. Mahoor and M. Abdel-Mottaleb, "Face recognition based on 3D ridge images obtained from range data," *Pattern Recognition,* vol. 45, pp. 445-451, 2009.

[3] S. Berretti, A. D. Bimbo, and P. Pala, "3d face recognition by modeling the arrangement of concave and convex regions," in *Adaptive Multimedia Retrieval: User, Context, and Feedback*, ed: Springer, 2006, pp. 108-118.

[4] A. Moreno, A. Sanchez, J. Velez, and J. Diaz, "Face recognition using 3D local geometrical features: PCA vs. SVM," in *The 4th International Symposium on Image and Signal Processing and Analysis*, 2005, pp. 185-190.

[5] J. Hopcroft and R. Karp, "An n^5/2 algorithm for maximum matchings in bipartite graphs," *SIAM Journal on Computing,* vol. 2, pp. 225-231, 1973.

[6] A. Colombo, C. Cusano, and R. Schettini, "Recognizing faces in 3d images even in presence of occlusions," in *The 2nd IEEE International Conference on Biometrics: Theory, Applications and Systems*, 2008, pp. 1-6.

[7] N. Alyuz, B. Gokberk, and L. Akarun, "A 3D face recognition system for expression and occlusion invariance," in *The 2nd IEEE International Conference on Biometrics: Theory, Applications and Systems*, 2008, pp. 1-7.

[8] S. Berretti, A. D. Bimbo, and P. Pala, "Sparse matching of salient facial curves for recognition of 3-D faces with missing parts," *IEEE Transactions on Information Forensics and Security,* vol. 8, pp. 374-389, 2013.

[9] H. Drira, B. B. Amor, A. Srivastava, M. Daoudi, and R. Slama, "3D face recognition under expressions, occlusions, and pose variations," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 35, pp. 2270-2283, 2013.

[10] Y. Lei, Y. Guo, M. Hayat, M. Bennamoun, and X. Zhou, "A Two-Phase Weighted Collaborative Representation for 3D partial face recognition with single sample," *Pattern Recognition,* vol. 52, pp. 218-237, 2016.

[11] D. Douglas and T. Peucker, "Algorithms for the reduction of the number of points required to represent a digitized line or its caricature," *Cartographica: The International Journal for Geographic Information and Geovisualization,* vol. 10, pp. 112-122, 1973.

[12] Y. Gao and M. Leung, "Human face profile recognition using attributed string," *Pattern Recognition,* vol. 35, pp. 353-360, 2002.

[13] W. Tsai and S. Yu, "Attributed string matching with merging for shape recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 7, pp. 453-462, 1985.

[14] Y. Freund and R. Schapire, "A desicion-theoretic generalization of on-line learning and an application to boosting," *Lecture Notes in Computer Science,* vol. 904, pp. 23-37, 1995.

[15] A. Moreno and A. Sánchez, "GavabDB: a 3D face database," presented at the Proc. 2nd COST275 Workshop on Biometrics on the Internet, Vigo (Spain), 2004.

[16] T. Fawcett, "An introduction to ROC analysis," *Pattern recognition letters,* vol. 27, pp. 861-874, 2006.

[17] A. Savran, N. Alyüz, H. Dibeklioğlu, O. Çeliktutan, B. Gökberk, B. Sankur, *et al.*, "Bosphorus database for 3D face analysis," in *Biometrics and Identity Management*, ed: Springer, 2008, pp. 47-56.

[18] A. Mian, M. Bennamoun, and R. Owens, "An efficient multimodal 2D-3D hybrid approach to automatic face recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 29, pp. 1927-1943, 2007.