21st International Conference on Pattern Recognition (ICPR 2012)
November 11-15, 2012. Tsukuba, Japan

Multi-subregion Face Recognition using Coarse-to-Fine Quad-tree Decomposition

Cuicui Zhang, Xuefeng Liang, and Takashi Matsuyama
Graduate School of Informatics, Kyoto University, Japan
zhang@vision.kuee.kyoto-u.ac.jp, {xliang, tm}@i.kyoto-u.ac.jp

Abstract

One problem of existing appearance-based face recognition methods (e.g. PCA, LDA) is their weak ability of coping with local variations caused by facial expressions, motion deformation, data missing, etc. Multi-subregion fusion methods, which divide the face into a set of subregions, aim at this issue, and were reported a better performance. However, it leaves two open questions: 1. what subregions are good partitions on the face, and 2. How to fuse these subregions could achieve the expected performance. In this paper, we address these two questions and propose a local discrimination driven face partition method based on a coarse-to-fine Quad-tree decomposition. Unlike other multi-subregion approaches relying on prior knowledge, our method partitions the face according to the data property. Thus, it can adapt to varied databases. Meanwhile, our method introduces an optimized solution that fuses selected subregions to reach higher recognition accuracy. The cross-database experiments including one 3D database and three 2D databases demonstrate the efficiency and effectiveness of the proposed method.

1. Introduction

The appearance-based technique has been extensively studied, and acknowledged as one kind of the most successful face recognition approaches. In holistic appearance-based methods, an image with size $n \times m$ pixels is represented by a long vector in the $n \times m$ dimensional space. Principal Component Analysis (PCA) [9] and Linear Discriminant Analysis (LDA) [2] are two widely accepted representatives under this framework. However, they are not a universal solution in all cases. For instance, motion deformation, illumination changing, data missing make modifications on the representation coefficients on the face, and lead these methods to have weak ability of coping with local facial variations. Multi-subregion fusion methods ([7], [3], [1]) were proposed as a solution for this problem. They divide the face into a set of disjoint subregions, perform recognition on each subregion and fuse the results. However, if the subregion is small, it may return low recognition accuracy as less discriminative features are involved. In [8], the fusion of multiple overlapped regions was investigated. It defined a set of 30 regions with large overlaps, see Fig. 4. Experiments showed it outperformed others on well registered databases. Unfortunately, it leaves two open questions to us: First, what subregions are good partitions on the face? The 30 regions in [8] are manually designed depending on the prior knowledge based on experimental evaluation, which is time consuming and requires redesign for different databases; Second, how to select subregions to build up the best combination is also depending on extensive experimental evaluations.

We address the above two questions and propose a new face partition method based on coarse-to-fine Quad-tree decomposition. Since the decomposition relies on the property of the template of that database, the obtained subregions can adapt to various databases without redesign. In our method, the criterion of Quad-tree decomposition is using LDA-motivated total variance, which ensures the robustness to local noise and efficiency of computation. Meanwhile, we introduce an optimized solution, which borrows the idea for the 0-1 Knapsack problem, to select subregions and fuse them to achieve expected performance. We evaluate our method against the method in [8] on four databases: Uchimura 3D, ATT, IFD, and JFFE database. Experiments show our method has a better ability of handling local variations, and well adapts to different databases.

The reminder of this paper is organized as follows: the coarse-to-fine Quad-tree decomposition for facial region partition is introduced in section 2. Section 3 presents an optimized solution for subregion selection and fusion. Experimental results are described in section 4. And Section 5 concludes the paper.

2. Coarse-to-fine Quad-tree decomposition for facial region partition

Instead of dividing the face region into a uniform grid, Quad-tree partitions the face region by means of local discriminative variance. In other words, the variance is discriminative feature density. Larger partition means the block is with lower feature density. Similarly, the smaller partition means the block is with higher feature density. To make the partition more stable to local noises, we consider the variance on all faces cross the entire database. Motivated by the idea of LDA which encodes discriminative information by maximizing the between-class scatter matrix S_b and minimizing the within-class scatter matrix S_w (See Eq. 1). We define a template face by Eq. 2 to represent the distribution of discriminative information for the database (see Fig. 1). Thus, the total variance of entire database is the variance of the template (see Eq. 3).

$$\begin{cases} S_b = \sum_{i=1}^{c} N_i(\mu_i - \mu)(\mu_i - \mu)^T, \\ S_w = \sum_{i=1}^{c} \sum_{x_k \in X_i} (x_k - \mu_i)(x_k - \mu_i)^T, \end{cases} \tag{1}$$

$$template = diag(\frac{S_b}{S_w}), \tag{2}$$

$$totalVar = variance(diag(\frac{S_b}{S_w})), \tag{3}$$

where μ is the mean image of all classes, μ_i is the mean image of class X_i, N_i is the number of samples in class X_i, and x_k is the k-th sample of class X_i.

Quad-tree decomposition is performed on the template using Eq.4. If the variance of a region R (block) is higher than a threshold variance ($T * totalVar$), then R is split into smaller blocks (Eq. 4.1). Unfortunately, the variances may not distribute uniformly. Often, the average variance of R is low, but the variance of its certain sub-blocks may be higher than the threshold variance. In this case, R is split too (Eq. 4.2).

$$\begin{cases} doSplit(R) = totalVar(R) > T * totalVar, & (4.1) \\ doSplit(R) = doSplit(R) | doSplit(subR). & (4.2) \end{cases}$$

Local variances usually vary on different databases, so one universal threshold is not applicable in our method. Even in one database, it is rather difficult to find the best partition using one threshold. We, therefore, give a set of thresholds in a descending order, and introduce coarse-to-fine face partitions. The face image is split into less and bigger blocks when threshold is large, but into more and smaller blocks when threshold is small.

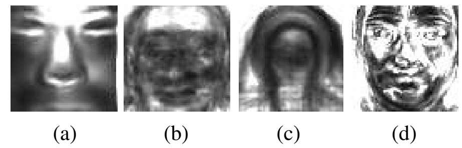

Figure 1: Templates for four databases: (a) Uchimura 3D, (b) ATT, (c) IFD, (d) JFFE database.

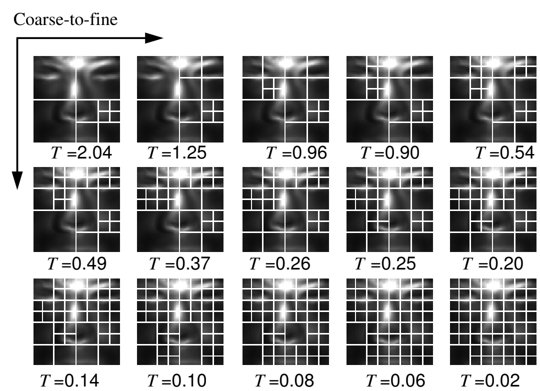

Figure 2: The coarse-to-fine Quad-tree partition of template on Uchimura 3D database. The threshold T is in a descending order from left to right, from top to bottom.

In this work, thresholds are set between a maximum value when the whole face cannot be partitioned and a minimum value when the face is partitioned completely. Figure 2 shows Quad-tree partitions on the template range image of Uchimura 3D database [10]. As explained above, larger partitions (blocks) have less discriminative features. There is no need to keep its original size. We then resize a block to $((d/2) \times (d/2))$ if d is greater than the minimum 3, where d is the width of the block (in pixel). Finally, coarse-to-fine partitions and block resizing result in a set of subregions whose sizes are smaller than original face image.

3. Facial subregion selection using 0-1 Knapsack solution

Subregions, which come from coarse-to-fine partitions, are portions of the face with differences. Therefore, they perform differently according to the discriminative features involved. Normally, single subregion is unlikely to achieve the expected performance. On the contrary, fusion of multiple subregions often works better. In our work, we borrow the idea of Dynamic Programming solution for 0-1 Knapsack problem, covert the problem of subregion selection and fusion into an optimization problem. That is: Given a set of subregions $S = S_1, S_2, \cdots, S_n$, each subregion S_i has two parameters: the weight W_i and the value V_i. We aim at choosing a subset O of S such that the total weight of selected subregions does not exceed the capacity of O and the total value is maximized. The weight and the value of subregion are defined as the size (in pixel) and discriminative power, respectively. As aforementioned in Sect.2, the discriminative information of a database is represented by the matrix of $S_b./S_w$. Here, the trace of subregional $S^i_b./S^i_w$ (see Eq. 5) is employed as the discriminative power of the i-th subregion. The capacity of O is defined by the total size of the 30 regions in Fig. 4, because we want to test if our method could work better than [8] using the same computational cost.

$$V_i = trace(\frac{S^i_b}{S^i_w}). \tag{5}$$

4. Experiments and analysis

Our method was evaluated on four databases: one 3D database (Uchimura 3D database [10]) and three widely used 2D databases: ATT [6], IFD [5], and JFFE database [4]. The face data in these databases are under varied conditions including a variety of head poses (Uchimura, ATT, IFD), illumination changing(Uchimura), partial data missing (Uchimura), facial expressions (ATT, IFD, JFFE), and facial details (e.g. with glasses or not: ATT). We used the cropped images as shown in Fig. 3 for recognition. Our method partitioned templates to $17 \sim 25$ subregions depending on the databases. A template was created on half amount and randomly chosen images from that database. The others were used as test data.

Figure 3: The sample images of four databases: (1) Uchimura 3D, (2) ATT, (3) IFD, (4) JFFE database.

- Uchimura 3D database: 38 face classes, 10 samples per class, varied head poses and lighting conditions, 3D registration by [10].
- ATT database: 40 face classes, 10 samples per class, varied lighting conditions, facial expressions and facial details (glasses / no glasses).

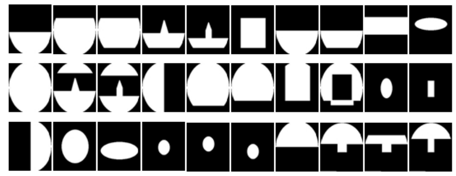

Figure 4: The manually designed 30 regions in [8].

- Indian Face Database (IFD): 40 face classes, 11 samples per class, varied head poses, 4 facial expressions.
- The Japanese Female Facial Expression Database (JFFE): 10 classes, 7 samples of 7 facial expressions per class.

Table 1: Recognition rate of our Quad-tree based face partition method compared to the 30-region method [8] and the conventional PCA-LDA method.

Method \ Database		3D	ATT	IFD	JFFE
PCA-LDA		**99.9**	93.7	88.1	95.0
30-region+ PCA-LDA	30 regions	98.9	91.9	87.3	95.0
	28 regions	98.9	93.7	86.0	95.0
	20 regions	98.9	95.0	87.7	95.0
	max single	99.3	94.4	87.7	**99.9**
Quad-tree+PCA-LDA		**99.9**	**96.8**	**91.4**	**99.9**

The performance of our method is compared with the 30-region method in [8] that is claimed as the best muti-subregion method currently, and the conventional PCA-LDA method without facial partition. In [8], since author did not know how to select the best combination of 30 regions either, we did experiments and found the following four combinations outperformed others: the all 30 regions, the 28 regions of highest performances, the 20 regions of highest performances, and the maximum single region (the single region has the best performance). Thus, they were used in comparison. And in [8], they used max voting for the fusion of multiple regions. Experiments show the maximum single region occasionally performs best. In our method, the better one between the fusion of multiple regions and the maximum single performance was employed. Table 1 illustrates the result. On Uchimura 3D and IFD database, we found none of combinations of the 30-region method could outperform the conventional PCA-LDA. The reason for Uchimura 3D database is that no region in 30-region method covers the whole facial area. Therefore, the lack of face boundary, which is an important discriminative clue for recognition, downgrades the performance. Since our method partition the face image region from coarse to fine, at least one subregion can cover the whole face image, no important discriminative clue is lost. On IFD database, head poses are pretty diverse such that it is difficult to do facial registration. Since those 30 regions are designed based on registration, they are very likely to ignore the discriminative information on the data without well registration. On the contrary, our method adapts the shape of partitioned subregion according to the data property. On the ATT database, the 30-region method outperforms the conventional PCA-LDA, but it depends on which regions are chosen. We can see that the accuracy using the 20 regions are higher than that using 28 regions and 30 regions. Observing the recognition rate of each single region, we find that the rates span a large range. These selected regions are the top 20, and involve much less local variations. As our method selects the combination of subregions using the optimized solution for 0-1 Knapsack problem, it can select the appropriate combination automatically. For the JFFE database, one single subregion achieves a better performance than the fusion of multiple regions and the PCA-LDA. By observing the shape of that subregion, we find that it removes most local variations, and happens to be very suitable for the face data in that database. In our method, the coarse-to-fine partition also generates a most appropriate subregion for the face data. So, this single subregion is employed other than merely focusing on fusion of multiple subregions. With comparison on four databases, we can conclude that the performance of the 30-region method is not stable because of its limitation of working better on well registered data and lower ability of adapting the subregions to diverse face images. In addition, since authors do not know how to select the best combination of multiple subregions, it is a serious issue when applying this method. Compared to the 30-region method, our method achieves the best performance on all the databases in Table 1. This shows that our method can be widely applied to different databases. Because our Quad-tree based facial partition method is processed on the template image, which is obtained from S_b and S_w, and deemed as a summery of the database. Thus, our method can generate appropriate subregions and adapt to various databases. Moreover, the dynamic programming solution for 0-1 knapsack problem guarantees our method to select the best combination of multiple subregions automatically. Since the total size of our subregions equals to those selected regions in [8], this means we have the same computational cost. Above experiment proofs our method has better ability of handling local variations and adapting to various face databases.

5. Conclusion

This paper proposes a new face region partition method based on a coarse-to-fine Quad-tree decomposition to handle local variations. It answers two questions remaining in other multi-subregion methods: 1. what subregions are good partitions? To make the decomposition more robust to local noises, a LDA-motivated total variance is used to build up a template face for Quad-tree decomposition. According to the template property, coarse-to-fine partition results in a set of different subregions who have higher discriminative information. Comparing with existing multi-subregion methods, our method does not rely on prior knowledge obtained from experimental evaluation. Therefore, it adapts subregions to databases. 2. how to select and fuse these subregions? We propose an optimization solution to select the best combination of different subregions. Experimental results on four databases show that our method achieves at a better performance than the up-to-now best face partition method using 30 overlapping regions.

References

[1] N. Alyüz, B. Gökberk, and L. Akarun. Regional registration and curvature descriptors for expression resistant 3D face recognition. *SIU*, pages 544–547, 2009.

[2] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman. Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. *PAMI*, 20(7):711–720, 1997.

[3] C. Boehnen, T. Peters, and P. Flynn. 3D signatures for fast 3D face recognition. *ICB*, pages 12–21, 2009.

[4] M. N. Dailey, C. Joyce, M. J. Lyons, M. Kamachi, H. Ishi, and J. Gyoba. Evidence and a computational explanation of cultural differences in facial expression recognition. *Emotion*, 10(6):874–893, 2010.

[5] V. Jain and A. Mukherjee. The indian face database. *http://vis-w.cs.umass.edu/ vidit/IndianFaceDatabase/*.

[6] A. V. Nefian and M. H. Hayes. Hidden markov models for face recognition. *Proc. Int. Conf. Acoustics, Speech, Signal Processing*, pages 2721–2724, 1998.

[7] C. C. Queirolo, L. Silva, O. R. P. Bellon, and M. P. Segundo. 3D face recognition using simulated annealing and the surface interpenetration measure. *PAMI*, 32:206–219, 2010.

[8] L. Spreeuwers. Fast and accurate 3D face recognition using registration to an intrinsic coordinate system and fusion of multiple region classifiers. *IJCV*, 93(3):389–414, 2011.

[9] M. Turk and A. Pentland. Eigenfaces for recognition. *J. Cognitive Neuroscience*, 3(1), 1991.

[10] C. Zhang, K. Uchimura, G. Koutaki, and C. M. Zhang. 3D face recognition using multi-level multi-feature fusion. *PSIVT*, 2010.