Local Edge/Corner Feature Integration for Illumination Invariant Face Recognition

Almabrok Essa and Vijayan Asari

Department of Electrical and Computer Engineering
University of Dayton, Dayton, Ohio, USA
Email: essaa1@udayton.edu, vasari1@udayton.edu

***Abstract*—In this paper, we present a new appearance based feature descriptor, named Local edge/corner Feature Integration (LFI), which efficiently summarizes the local structure of face images. LFI is a nonparametric descriptor that utilizes a combined edge/corner detection strategy. The proposed method uses the approach suggested by Frei and Chen for corner and edge detection with nine different masks. After we obtain the information about corners and edges of the image, for each pixel position, we describe the relationship of pixels to their local neighborhood from the local edge/corner features using the edges and corners information separately. Then, we concatenate these patterns together to form the final LFI feature vector. The performance evaluation of the proposed LFI algorithm is conducted on several publicly available databases and observed promising recognition rates.**

Keywords–Face recognition; Frei-Chen edge detector; modular histogram; chi-square similarity measure; libsvm classifier; local edge/corner feature integration (LFI).

I. Introduction

During the past few years, face recognition has received a great deal of attention and become one of the most popular research areas in the fields of computer vision, image processing, pattern recognition, and machine learning. The key of each face recognition system is the utilization of the feature extraction technique that must be able to extract features from the face image, which are distinct and stable under different conditions during the image acquisition process.

In the recent years, much research work has been done on extracting image features. Many computer vision applications employ the texture analysis algorithms. Two of the highest performing texture algorithms that based on the concept of local pattern descriptors, namely Local Binary Pattern (LBP) and Local Directional Pattern (LDP), which describe the relationship of pixels to their local neighborhood. They detect only the important local textures by labeling each pixel with the code of texture primitive that best matches the local neighborhood. Fig. 1 shows some of these texture primitives that can be detected by the local pattern descriptors that include spots, line ends, flat area, edges, and corners [1].

LBP is a nonparametric method which extracts local structures of images efficiently by comparing each pixel with its neighboring pixels. If a neighbor pixel has a higher gray value than the center pixel (or the same gray value) then a 1 is assigned to that pixel, which is otherwise a 0. Finally, the LBP binary code for the center pixel is produced by concatenating the eight $1s$ or $0s$, which can be converted to a decimal number to produce the new value of that central pixel. The original LBP operator was introduced by Ojala et al. for texture analysis [2], and has proved a simple yet powerful approach to describe local structures. LBP operator has a number of extensions that have been extensively used in many applications, such as face image analysis [3][4], image and video retrieval [5][6], environment modeling [7][8], visual inspection [9][10], motion analysis [11][12], and biomedical and aerial image analysis [13][14]. LBP-based facial image analysis has been one of the most popular and successful applications in recent years. Nevertheless, LBP considers only first order intensity pattern change in a local neighborhood which fails to extract detailed information especially during changes in face image due to the noise and illumination variation problems.

LDP encodes the directional information in the neighborhood instead of the intensity as LBP does with higher computational cost. LDP is a gray-scale pattern that characterizes the spatial structure of a local image texture. It computes the edge response values in eight different directions at each pixel position by convolving the image with the Kirsch masks in eight different orientations, centered on its own position. Then it uses the relative strength magnitude to encode the image texture. The presence of a corner or an edge shows high response values in some particular directions. Therefore, in order to generate the LDP code, we need to know the n most prominent directions. Then, the top n directional bit responses are set to 1 and the rest $(8-n)$ bits of 8-bit LDP pattern are set to 0 [15][16]. Since the edge responses are more noise and illumination insensitive than intensity values, the resultant LDP feature maintains more information than LBP and describes the local primitives stably, including different types of curves, corners, and junctions. However, LDP technique still suffers in non-monotonic illumination variation and random noise.

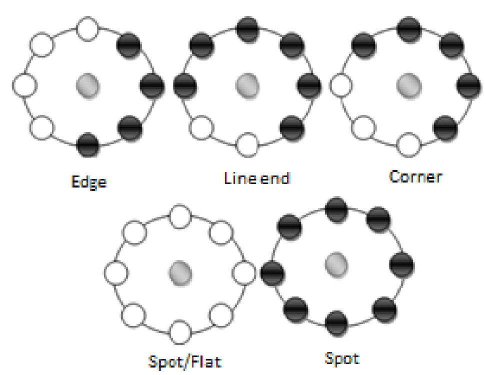

Figure 1. Different texture primitives detected by local pattern descriptors.

 ISBN: 978-1-61208-520-3

In this paper, we present a new local pattern descriptor, named Local edge/corner Feature Integration (LFI) that is simple but effective, and it can be a potential tool to extract image features. LFI is a nonparametric method which extracts local structures of images efficiently by comparing each pixel with its neighboring pixels from edge/corner responses separately, then combining these thresholding responses to form the final code. Unlike LDP whose codes are generated by setting the top n directional bit responses to 1 and the rest to 0, which may ignore some important information in the local neighborhood. LFI uses the information of edge/corner changes around pixels and labels the pixels by thresholding a 3×3 neighboring pixels with the central pixel separately then considering the results as binary codes. After that concatenates these binary codes to form the final LFI feature vector.

The rest of the paper is organized as follows. In Section 2, the mathematical details of the proposed LFI algorithm is provided. Discussion on the datasets and experimental results are presented in Section 3. Finally, the conclusion is drawn in Section 4.

II. Local Edge/Corner Feature Integration (LFI)

This work aims to improve the face recognition accuracy under illumination-variant environments by detecting much stable edges especially in dark areas, which can be done by the help of Frei-Chen edge detector [17]. The proposed LFI technique can be summarized into three stages: edge/corner detection, binary encoding and decoding, and feature integration. Fig. 2 illustrates the framework of the proposed technique. The details of each stage are described below.

A. *Corner/Edge Detection*

We suggest to utilize the properties of Frei-Chen edge detector to extract more detailed corner and edge information from input image. Frei-Chen edge detector works as nine convolution masks that work on a 3×3 window size denoted as K_i for $i = 1, 2, ..., 9$ as shown in Fig. 3. The first four masks K_i for $i = 1, ..., 4$ are used to find the edges' subspace, the first two of them K_1 and K_2 represent the isotropic smoothed gradient weighting function, which will be supported by the second two K_3 and K_4 to span the above edge's subspace by contributing to the magnitude of the edge's subspace components. The second four kernels K_i for $i = 5, ..., 8$ are utilized to find the corners' subspace. By summing all of these four, all possible discrete realizations of the points can be detected. The last one K_9 is used to compute the mean which we use as a normalization factor [17].

Mathematically, given an input image $I(x, y)$, the nine different edge, corner, and mean responses g_i can be computed by

$$g_i = I(x, y) * K_i, \qquad i = 1, 2, ..., 9 \tag{1}$$

Where $*$ represents a convolution operation. Fig. 4 shows an example of Frei-Chen kernels filtered images. In the figure, the upper row and the first image starting from the left in the middle row are the edge filtered images, the second four images are the corner filtered images, and the last one is the mean filtered image. All nine edge, corner, and mean responses g_i are extracted with their corresponding masks K_i for $i = 1, 2, ..., 9$ respectively.

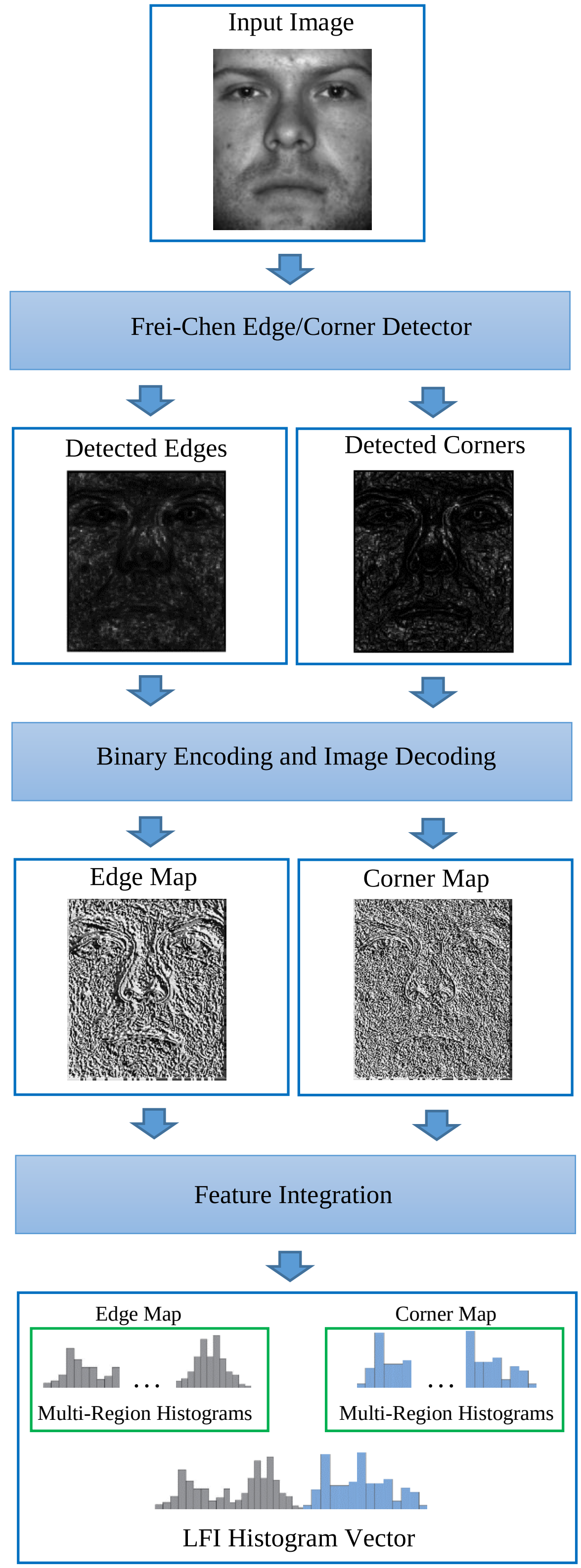

Figure 2. Overview of the proposed approach.

$$\begin{bmatrix} 1 & \sqrt{2} & 1 \\ 0 & 0 & 0 \\ -1 & -\sqrt{2} & -1 \end{bmatrix} \begin{bmatrix} 1 & 0 & -1 \\ \sqrt{2} & 0 & -\sqrt{2} \\ 1 & 0 & -1 \end{bmatrix} \begin{bmatrix} 0 & -1 & \sqrt{2} \\ 1 & 0 & -1 \\ -\sqrt{2} & 1 & 0 \end{bmatrix}$$

K_1 K_2 K_3

$$\begin{bmatrix} \sqrt{2} & -1 & 0 \\ -1 & 0 & 1 \\ 0 & 1 & -\sqrt{2} \end{bmatrix} \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} -1 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & -1 \end{bmatrix}$$

K_4 K_5 K_6

$$\begin{bmatrix} 1 & -2 & 1 \\ -2 & 4 & -2 \\ 1 & -2 & 1 \end{bmatrix} \begin{bmatrix} -2 & 1 & -2 \\ 1 & 4 & 1 \\ -2 & 1 & -2 \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

K_7 K_8 K_9

Figure 3. The nine Frei-Chen masks used to find the edge, corner, and mean responses of each image.

In terms of the edge detection denoted as E, we choose the first four filtered images g_i for $i = 1, ..., 4$ and project the image onto it. The projection equation can be given as

$$E = \sqrt{\frac{\sum_{i=1}^{4} g_i^2}{\sum_{i=1}^{9} g_i^2}} \tag{2}$$

When it comes to the corner detection that can be denoted as C, we choose the second four filtered images g_i for $i = 5, ..., 8$ and project the image onto it, which can be done by

$$C = \sqrt{\frac{\sum_{i=5}^{8} g_i^2}{\sum_{i=1}^{9} g_i^2}} \tag{3}$$

Fig. 5 shows the detected edges and corners after applying the two projection equations above.

B. Image Encoding and Decoding

After the edges and corners are detected separately as mentioned above, which can be seen in Fig. 5, a binary coding strategy is applied by exploiting the center pixel value in each 3×3 neighborhood regions, to encode the local structures information in the neighborhood. To form the edge or corner patterns, we compare each pixel with its neighboring pixels. If a neighbor pixel has a higher edge/corner value than the center pixel (or the same value) then a 1 is assigned to that pixel, which is otherwise a 0. The edge/corner binary code for that center pixel is produced by concatenating the eight $1s$ or $0s$. Finally, to retrieve the edge and corner features map, we change that binary codes into the corresponding decimal codes D, which can be defined as

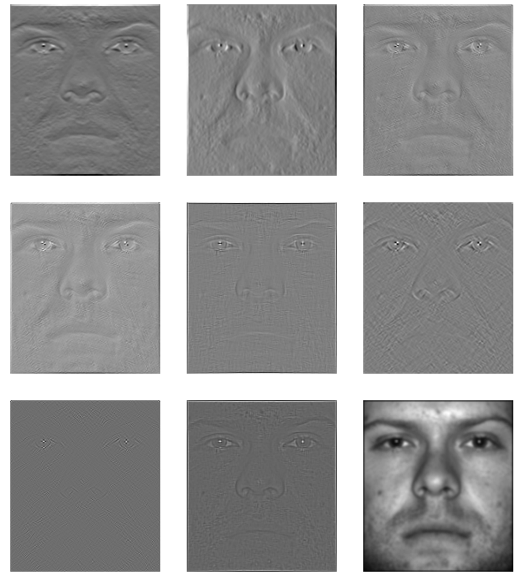

Figure 4. Projection of an image onto Frei-Chen edge, corner, and mean masks.

Figure 5. Edges and corners detected. Left input image, middle the detected edges, and right the detected corners.

$$D = \sum_{p=1}^{8} f(d_p - d_c) \times 2^{p-1} \tag{4}$$

and

$$f(x) = \begin{cases} 1 & \text{if } x \geq 0 \\ 0 & \text{if } x < 0 \end{cases} \tag{5}$$

where d_c and d_p denote the edge or corner values of the central pixel and its neighbors respectively. We use the detected edges image E and the detected corners image C, to generate a pattern for each pixel position. Fig. 6 shows a raw image, edge feature map, and the corner feature map after applying the binary coding and decoding strategy.

Figure 6. Coding and decoding strategy visualization. Left input image, middle the detected edges map, and right the detected corners map.

C. Feature Integration

To generate the final LFI feature vector, we map each edge/corner patterns to their corresponding histogram bin, then a 256 bin histogram would be computed for each edge and corner patterns separately. Finally, all the histograms will be concatenated to form the final LFI histogram vector for each input image. By this way, the LFI histogram contains all the information about the distribution of the local micro patterns such as edge, corener, line-end, flat, and spot, which can be used to statistically describe the image characteristics.

III. Experimental Results

For evaluation, we use two publicly available face datasets, named extended Yale B database [18][19] and AT&T (ORL) dataset [20]. In terms of the feature extraction process, to consider the local information of face components, we divide each image into small blocks as can be seen in Fig. 7. After that, we extract the information of each block separately using our proposed technique LFI and represent it as a local LFI histogram. Finally, we concatenate these local histograms to form a global histogram for each input image that contains information about the distribution of the local micro-patterns of the image, and can be used to statistically describe the face image characteristics. The length of this feature vector (global histogram) depends on the number of blocks (regions) of each image.

When it comes to the face recognition process, the objective is to compare the encoded feature vector from one image with all other candidate feature vectors of the dataset using two different method for classification. The first one, is a library for support vector machines (LIBSVM) [21], and the second one is, chi-square metric χ^2, which is a measure between two feature vectors, $H1$ and $H2$, of length N, that can be defined as

$$\chi^2(H_1, H_2) = \sum_{i=1}^{N} \frac{(H_1(i) - H_2(i))^2}{H_1(i) + H_2(i) + \epsilon} \tag{6}$$

where ϵ is a very small value that used to avoid division by 0.

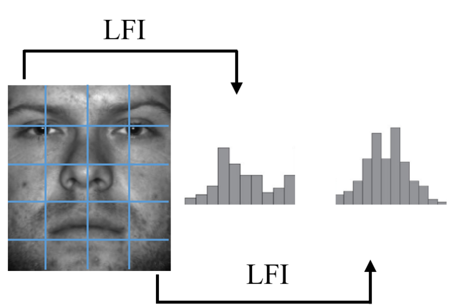

Figure 7. A face images is divided into small blocks and the features are extracted using LFI and a histogram is built for each area. Then all the histograms are concatenated.

A. Extended Yale B Database

The extended Yale B database has a total of 2280 face images for 38 subjects representing 60 illumination conditions per subject under the frontal pose, all the images resized to 64×64. Fig. 8 shows sample faces of this dataset. In the figure, from the bottom raw it is hard even for human being to recognize the person in some cases, especially the right bottom sample we cannot even say if there is a face or any other object in the image. Fig.s 9 and 10 show the edge and corner map of the face images in Fig. 8. It is clear that using Frei-Chen edge/corner detector, the edges and corners in dark areas of the image are likely to be detected. Therefore, the illumination variation problems will be overcome, which significantly helps to improve the face recognition performance under uncontrolled illumination/lighting environments.

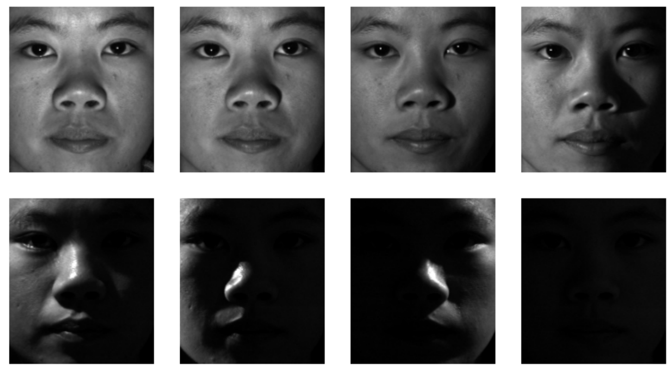

Figure 8. Samples of one subject from the Extended Yale B database.

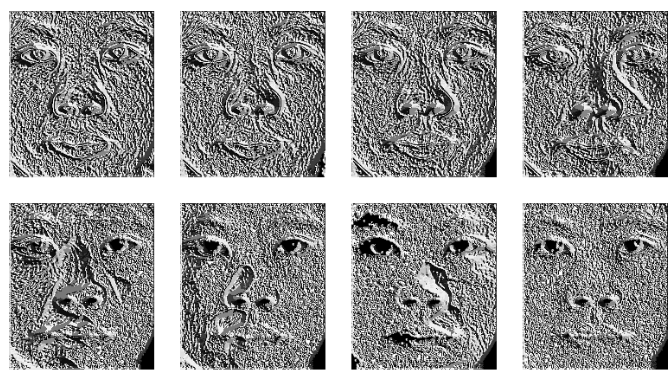

Figure 9. LFI edge map of one subject from the Extended Yale B database.

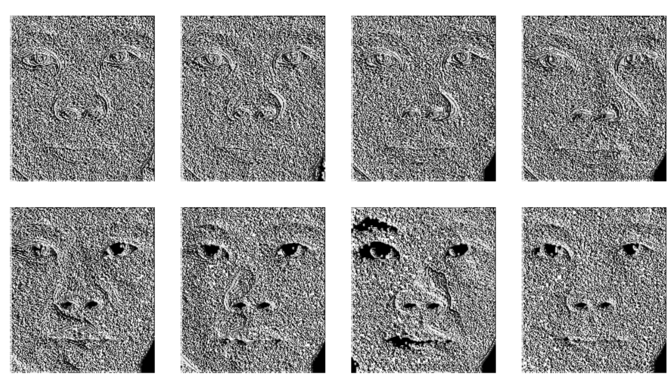

Figure 10. LFI corner map of one subject from the Extended Yale B database.

 ISBN: 978-1-61208-520-3

To avoid any bias, by using the χ^2 we select one image per subject of the data for training and the rest of the data for testing. The experiments were repeated 60 times as there are a total of 60 samples/subject, then the average results are calculated. On the other hand, by using $LIBSVM$ we randomly select half of the data for training and the other half for testing. The experiments were repeated 30 times, then the average results are calculated for comparison. The performance results of well known face recognition algorithms like local ternary patterns (LTP) [22], Weber-face [23] and gradientface (GradFace) [24], as well as LBP and LDP [16], with the proposed method on extended Yale B dataset are presented in Table I. Note that, the results we compared with are as we got from their original references which are mentioned in the table. Meanwhile, part of the extended Yale B dataset (standard Yale B dataset) was used in [23][24].

TABLE I. PERFORMANCE RESULTS OF WELL KNOWN FACE RECOGNITION ALGORITHMS TOGETHER WITH THE PROPOSED METHOD ON EXTENDED YALE B DATASET.

Refernce	Method	Highest Recognition Accuracy
Proposed	LFI / LIBSVM	**99.29 %**
Proposed	LFI / χ^2	**99.24 %**
[24]	GradFace	98.96 %
[22]	RLTP	98.71 %
[23]	Weber-face	98.30 %
[22]	LTP	98.25 %
[24]	LTV	97.93 %
[16]	LDP+2D-PCA	96.43 %
[22]	LBP	96.07 %
[16]	LBP+2D-PCA	91.54 %
[16]	LDP+PCA	81.34 %

B. AT&T Dataset (ORL)

The ORL database contains a total of 400 face images corresponding to 10 different images of 40 distinct subjects. Some sample faces are shown in Fig. 11. The images are taken at different times with different specifications, including slightly varying in illumination, different facial expressions such as open and closed eyes, smiling and non-smiling, and facial details like wearing glasses. All the images resized to 64×64. Table II summarizes the highest recognition rates of the proposed local edge/corner feature integration method compared to well known face recognition algorithms together like (GLCM+LDP+EDGE) [25], and State Preserving Extreme Learning Machine (SPELM) [26], and a combined phase congruency and Gabor wavelet techniques (PC/GW) [27], as well as LBP and LDP, with the proposed method on ORL dataset with the use of χ^2 similarity measure and $LIBSVM$. Note that, the results we compared with are as we got from their original references which are mentioned in the table, since we do not have any original codes of of these algorithms. The procedure of splitting the training and testing data has been done as in the previous experiment. Therefore, we select one image per subject of the data for training and the rest of the data for testing to avoid any bias. The experiments were repeated 10 times, then the average results were calculated for comparison using χ^2. Additionally, we select seven images of the data randomly for training the $LIBSVM$ classifier and the rest for testing. The experiments were repeated 10 times, then the average results were calculated.

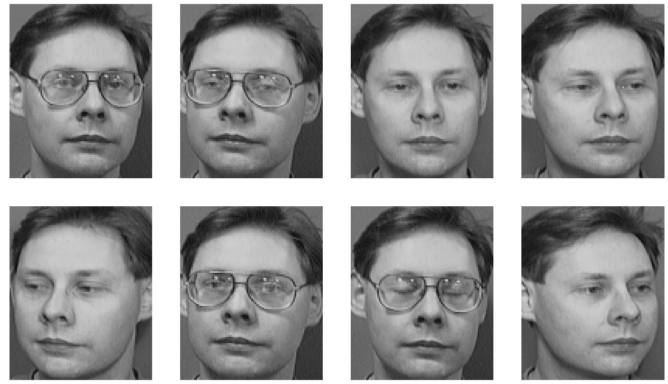

Figure 11. Samples of a subject from the ORL database

TABLE II. PERFORMANCE RESULTS OF WELL KNOWN FACE RECOGNITION ALGORITHMS TOGETHER WITH THE PROPOSED METHOD ON ORL DATASET.

Refernce	Method	Highest Recognition Accuracy
Proposed	LFI / LIBSVM	**99.17 %**
Proposed	LFI / χ^2	**98.88 %**
[25]	GLCM+LDP+EDGE	98.75 %
[27]	GW+PC+PCA	98.00 %
[27]	GW+PC	98.00 %
[26]	Gabor+SPELM	97.97 %
[25]	LDP+EDGE	96.60 %
[25]	GLCM+LDP	92.70 %
[26]	PHOG+SPELM	92.45 %
[25]	GLCM+EDGE	90.50 %
[25]	LDP	88.50 %
[27]	PCA	88.00 %
[25]	LBP	87.80 %

IV. CONCLUSION

In this paper, we have introduced a new feature descriptor technique named local edge/corner feature integration. Throughout the performance evaluation, we found that LFI is robust for face recognition regardless of extremely variations of illumination/lighting environments as in extended Yale B database, and slightly differences of pose conditions as in AT&T dataset. In addition, compared to the other state-of-the-art methods, we can say that our method provides better accuracy in most test cases. From the results above, it is clear that the LFI provides a stronger discriminative capability in describing detailed texture information than the LBP and LDP. In general, considering all comparison results, we can assess that LFI can be a promising candidate for face recognition applications. The work is progressing to investigate the ability of the proposed technique LFI with different applications such as dynamic texture recognition.

REFERENCES

[1] T. Mäenpää and M. Pietikäinen, "Texture analysis with local binary patterns," Handbook of Pattern Recognition and Computer Vision, vol. 3, pp. 197–216, 2005.

[2] T. Ojala, M. Pietikäinen, and D. Harwood, "A comparative study of texture measures with classification based on featured distributions," Pattern recognition, vol. 29, no. 1, pp. 51–59, 1996.

[3] T. Ahonen, A. Hadid, and M. Pietikäinen, "Face recognition with local binary patterns, computer vision, eccv 2004 proceedings," Lecture Notes in Computer Science, vol. 3021, pp. 469–481, 2004.

[4] A. Hadid, M. Pietikainen, and T. Ahonen, “A discriminative feature space for detecting and recognizing faces,” in Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on, vol. 2. IEEE, pp. II–797, 2004.

[5] D. Huijsman and N. Sebe, “Content-based indexing performance: A class size normalized precision,” in Recall, Generality Evaluation, International Conference on Image Processing (ICIP’03), vol. 3, pp. 733–736, 2003.

[6] D. Grangier and S. Bengio, “A discriminative kernel-based approach to rank images from text queries,” IEEE transactions on pattern analysis and machine intelligence, vol. 30, no. 8, pp. 1371–1384, 2008.

[7] W. Ali, F. Georgsson, and T. Hellstrom, “Visual tree detection for autonomous navigation in forest environment,” in Intelligent Vehicles Symposium, 2008 IEEE. IEEE, pp. 560–565, 2008.

[8] L. Nanni and A. Lumini, “Ensemble of multiple pedestrian representations,” IEEE Transactions on Intelligent Transportation Systems, vol. 9, no. 2, pp. 365–369, 2008.

[9] T. Mäenpää, J. Viertola, and M. Pietikäinen, “Optimising colour and texture features for real-time visual inspection,” Pattern Analysis & Applications, vol. 6, no. 3, pp. 169–175, 2003.

[10] M. Turtinen, M. Pietikainen, and O. SilvÉn, “Visual characterization of paper using isomap and local binary patterns,” IEICE transactions on information and systems, vol. 89, no. 7, pp. 2076–2083, 2006.

[11] M. Heikkila and M. Pietikainen, “A texture-based method for modeling the background and detecting moving objects,” IEEE transactions on pattern analysis and machine intelligence, vol. 28, no. 4, pp. 657–662, 2006.

[12] V. Kellokumpu, G. Zhao, and M. Pietikäinen, “Human activity recognition using a dynamic texture based method.” in BMVC, vol. 1, p. 2, 2008.

[13] A. Oliver, X. Lladó, J. Freixenet, and J. Martí, “False positive reduction in mammographic mass detection using local binary patterns,” in International Conference on Medical Image Computing and Computer-Assisted Intervention. Springer, pp. 286–293, 2007.

[14] S. Kluckner, G. Pacher, H. Grabner, H. Bischof, and J. Bauer, “A 3d teacher for car detection in aerial images,” in 2007 IEEE 11th International Conference on Computer Vision. IEEE, pp. 1–8, 2007.

[15] T. Jabid, M. H. Kabir, and O. Chae, “Robust facial expression recognition based on local directional pattern,” ETRI journal, vol. 32, no. 5, pp. 784–794, 2010.

[16] D.-J. Kim, S.-H. Lee, and M.-K. Sohn, “Face recognition via local directional pattern,” International Journal of Security and Its Applications, vol. 7, no. 2, pp. 191–200, 2013.

[17] W. Frei and C.-C. Chen, “Fast boundary detection: A generalizationand new algorithm,” IEEE Transactions on Computers, vol. 26, no. 10, 1977.

[18] A. S. Georghiades and P. N. Belhumeur, “Illumination cone models for faces recognition under variable lighting,” in Proceedings of CVPR98, 1998.

[19] K.-C. Lee, J. Ho, and D. J. Kriegman, “Acquiring linear subspaces for face recognition under variable lighting,” Pattern Analysis and Machine Intelligence, IEEE Transactions on, vol. 27, no. 5, pp. 684–698, 2005.

[20] F. S. Samaria and A. C. Harter, “Parameterisation of a stochastic model for human face identification,” in Applications of Computer Vision, 1994., Proceedings of the Second IEEE Workshop on. IEEE, pp. 138–142, 1994.

[21] C.-C. Chang and C.-J. Lin, “Libsvm: a library for support vector machines,” ACM Transactions on Intelligent Systems and Technology (TIST), vol. 2, no. 3, p. 27, 2011.

[22] J. Ren, X. Jiang, and J. Yuan, “Relaxed local ternary pattern for face recognition.” in ICIP, pp. 3680–3684, 2013.

[23] B. Wang, W. Li, W. Yang, and Q. Liao, “Illumination normalization based on weber’s law with application to face recognition,” IEEE Signal Processing Letters, vol. 18, no. 8, pp. 462–465, 2011.

[24] T. Zhang, Y. Y. Tang, B. Fang, Z. Shang, and X. Liu, “Face recognition under varying illumination using gradientfaces,” IEEE Transactions on Image Processing, vol. 18, no. 11, pp. 2599–2606, 2009.

[25] A. Kar, D. Bhattacharjee, D. K. Basu, M. Nasipuri, and M. Kundu, “An adaptive block based integrated ldp, glcm, and morphological features for face recognition,” arXiv preprint arXiv:1312.1512, 2013.

[26] M. Z. Alom, P. Sidike, V. K. Asari, and T. M. Taha, “State preserving extreme learning machine for face recognition,” in 2015 International Joint Conference on Neural Networks (IJCNN). IEEE, pp. 1–7, 2015.

[27] E. Bezalel and U. Efron, “Efficient face recognition method using a combined phase congruency/gabor wavelet technique,” in Optics & Photonics 2005. International Society for Optics and Photonics, pp. 59 081K–59 081K, 2005.