

## Research Article

# Automated Facial Expression Recognition Using Gradient-Based Ternary Texture Patterns

Faisal Ahmed<sup>1</sup> and Emam Hossain<sup>2</sup>

<sup>1</sup> Department of Computer Science and Engineering, Islamic University of Technology, Board Bazar, Gazipur 1704, Bangladesh

<sup>2</sup> Department of Computer Science and Engineering, Ahsanullah University of Science and Technology, Dhaka 1208, Bangladesh

Correspondence should be addressed to Faisal Ahmed; fahmed@iut-dhaka.edu

Received 17 October 2013; Accepted 24 November 2013

Academic Editors: K. Ariyur and H. Lu

Copyright © 2013 F. Ahmed and E. Hossain. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Recognition of human expression from facial image is an interesting research area, which has received increasing attention in the recent years. A robust and effective facial feature descriptor is the key to designing a successful expression recognition system. Although much progress has been made, deriving a face feature descriptor that can perform consistently under changing environment is still a difficult and challenging task. In this paper, we present the gradient local ternary pattern (GLTP)—a discriminative local texture feature for representing facial expression. The proposed GLTP operator encodes the local texture of an image by computing the gradient magnitudes of the local neighborhood and quantizing those values in three discrimination levels. The location and occurrence information of the resulting micropatterns is then used as the face feature descriptor. The performance of the proposed method has been evaluated for the person-independent face expression recognition task. Experiments with prototypic expression images from the Cohn-Kanade (CK) face expression database validate that the GLTP feature descriptor can effectively encode the facial texture and thus achieves improved recognition performance than some well-known appearance-based facial features.

## 1. Introduction

Over the last two decades, automated recognition of human facial expression has been an active research area with a wide variety of potential applications in human-computer interaction, data-driven animation, surveillance, and customized consumer products [1, 2]. Since the classification rate is heavily dependent on the information contained in the feature representation, an effective and discriminative feature set is the most important constituent of a successful facial expression recognition system [3]. Even the best classifier will fail to attain satisfactory performance if supplied with inconsistent or inadequate features. However, in real-world applications, facial images can easily be affected by different factors, such as variations in lighting condition, pose, aging, alignment, and occlusion [4]. Hence, designing a robust feature extraction method that can perform consistently in changing environment is still a challenging task.

Based on the types of features used, facial feature extraction methods can be roughly divided into two different categories: geometric feature-based methods and appearance-based methods [1, 2]. In geometric feature-based methods, the feature vector is formed based on the geometric relationships, such as positions, angles, or distances between different facial components [2]. Among the different techniques introduced so far, one of the most popular geometric methods is the facial action coding system (FACS) [5] that recognizes facial expression with the help of a set of action units (AU). Each of these action units corresponds to the physical aspect of a particular facial muscle. Later, fiducial point-based representations [6–8] were also investigated by several researchers. However, the effectiveness of geometric methods is heavily dependent on the accurate detection of facial components, which is a difficult task in changing environment. Hence, geometric feature-based methods are difficult to accommodate in many real-world scenarios [2].

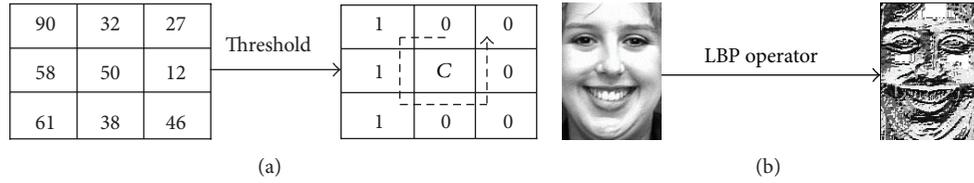


FIGURE 1: (a) Illustration of the basic  $LBP_{8,1}$  encoding scheme. Here, the  $LBP_{8,1}$  code 01110000 is used to label the center pixel  $C$ . (b) Sample LBP encoded image.

Appearance-based methods extract the facial appearance by convoluting the whole face image or some specific facial regions with image filter or filter bank [1, 2]. Some widely used appearance-based methods include principal component analysis (PCA) [9], independent component analysis (ICA) [10, 11], and Gabor wavelets [12, 13]. Although PCA and ICA feature descriptors can effectively capture the variability of the training images, their performances deteriorate in changing environment [14, 15]. On the other hand, extraction of Gabor features by convoluting face images with multiple Gabor filters of various scales and orientations is computationally expensive. Recently, local appearance descriptors based on local binary pattern (LBP) [16] and its variants [17] have attained much attention due to their robust performance in uncontrolled environment. The LBP operator encodes the local texture of an image by quantizing the neighbor gray levels of a local neighborhood with respect to center value and thus forms a binary pattern that acts as a template for micro level information such as edges, spots, or corners. However, the LBP method performs weakly under the presence of large illumination change and random noise [4], since a little variation in the gray level can easily change the LBP code. Later, local ternary pattern (LTP) [4] was introduced to increase the robustness of LBP in uniform and near-uniform regions by adding an extra intensity discrimination level and extending the binary LBP value to a ternary code. More recently, Sobel-LBP [15] has been proposed to improve the performance of LBP by applying Sobel operator to enhance the edge information prior to applying LBP for feature extraction. However, in uniform and near-uniform regions, Sobel-LBP generates inconsistent patterns as it uses only two discrimination levels just like LBP. Local directional pattern (LDP) [2, 14] employed a different texture encoding approach, where directional edge response values around a position are used instead of gray levels. Although this approach achieves better recognition performance than local binary pattern, LDP tends to produce inconsistent patterns in uniform and near-uniform facial regions and is heavily dependent on the selection of the number of prominent edge directions parameter [3].

Considering the limitations of the existing local texture descriptors, this paper presents a new texture pattern, namely, the gradient local ternary pattern (GLTP) for person-independent facial expression recognition. The proposed GLTP operator encodes the local texture information by quantizing the gradient magnitude values of a local neighborhood using three different discrimination levels. The proposed encoding scheme is able to differentiate between

smooth and high-textured facial parts, which ensure the formation of texture micropatterns that are consistent with the local image characteristics (smooth or high-textured). The performance of the GLTP feature descriptor is empirically evaluated using a support vector machine (SVM) classifier. Experiments with seven prototypic expression images from the Cohn-Kanade (CK) face expression database [18] validate that the GLTP feature descriptor can effectively encode the facial texture and thus achieves improved recognition performance than some widely used appearance-based facial feature representation.

## 2. LBP and LTP: A Review

Local binary pattern (LBP) is a simple yet effective local texture description technique. LBP was originally introduced by Ojala et al. [19] for grayscale and rotation-invariant texture analysis. Later, many researchers have successfully adopted LBP in different face-related problems, such as face recognition [20] and facial expression analysis [16]. The basic LBP method operates on a local neighborhood around each pixel of an image and thresholds the neighbor gray levels with respect to the center. The result is then concatenated binomially, and the center pixel is labeled with the resultant value. Formally, the LBP operator can be represented as:

$$LBP_{P,R}(x_c, y_c) = \sum_{p=0}^{P-1} s(i_p - i_c) 2^p, \quad (1)$$

$$s(x) = \begin{cases} 1, & x \geq 0, \\ 0, & x < 0. \end{cases} \quad (2)$$

Here,  $i_c$  is the gray value of the center pixel  $(x_c, y_c)$ ,  $i_p$  is the gray value of the surrounding neighbors,  $P$  is the total number of neighbors, and  $R$  is the radius of the neighborhood. Bilinear interpolation is used to estimate the gray level of a neighbor if it does not fall exactly on a pixel position. The histogram of the LBP encoded image or image block is then used as the feature descriptor. The basic LBP encoding process is illustrated in Figure 1.

One limitation of the LBP encoding is that the LBP codes are susceptible to noise since a little change in the intensities of the neighbors can entirely alter the resulting binary code. To address this issue, Tan and Triggs [4] proposed the local ternary pattern (LTP), which extends the binary LBP code to a 3-valued ternary code in order to provide more consistency in uniform and near-uniform regions. In the LTP encoding

-1	-2	-1
0	0	0
1	2	1

(a)

-1	0	1
-2	0	2
-1	0	1

(b)

FIGURE 2: Sobel masks, (a) horizontal mask and (b) vertical mask.

process, gray values in a zone of width  $\pm t$  about the center pixel are quantized to 0, and those above  $+t$  and below  $-t$  are quantized to  $+1$  and  $-1$ , respectively. Hence, the indicator  $s(x)$  in (2) is substituted by a 3-valued function:

$$s'(i_p, i_c) = \begin{cases} 1, & i_p \geq i_c + t, \\ 0, & |i_p - i_c| < t, \\ -1, & i_p \leq i_c - t. \end{cases} \quad (3)$$

Here,  $t$  is a user-specified threshold. The combination of these three discrimination levels in a local neighbourhood yields the final LTP value.

### 3. Proposed Method: Gradient Local Ternary Pattern (GLTP)

In practice, the LBP operator encodes the local texture primitives such as edges or spots by thresholding the local neighborhood at the value of the center pixel into a binary pattern. Zhao et al. [15] argued that applying Sobel operator prior to LBP feature extraction further enhances the texture details and thus facilitates more accurate texture encoding. Hence, they proposed the Sobel-LBP method [15], where the Sobel operator is first applied on the image to compute the gradient magnitude values, and then the basic LBP method is used to encode the gradient values. However, both LBP and Sobel-LBP employ two discrimination levels (0 and 1) for texture encoding and thus fail to generate consistent patterns in uniform and near-uniform regions, where the difference between the center and the neighbor gray levels is negligible. To address this limitation, we propose the gradient local ternary pattern (GLTP), a new texture descriptor that combines the advantages of Sobel-LBP and LTP operators. Our proposed method utilizes the more robust gradient magnitude values instead of gray levels with a three-level encoding scheme to discriminate between smooth and high-textured facial regions. Thus, the proposed method ensures generation of robust texture patterns which are consistent with the local image property (smooth or high-textured region), even under the presence of illumination variations.

**3.1. GLTP Encoding.** The proposed GLTP operator first calculates the gradient magnitudes of each pixel position of

an image, which enhances the local texture features, such as edges, spots, or corners. The gradient magnitude  $G_{x,y}$  at  $(x, y)$  position of an image  $f(x, y)$  can be computed using the following equation:

$$G_{x,y} = |G_x| + |G_y|. \quad (4)$$

Here,  $G_x$  and  $G_y$  are the two elements of the gradient vector and can be obtained by applying Sobel operator on the image  $f(x, y)$ . The Sobel operator convolves an image  $f(x, y)$  with a horizontal mask and a vertical mask to obtain the value of  $G_x$  and  $G_y$ . The two Sobel masks are shown in Figure 2.

In a uniform or near-uniform local region, the gradient magnitude of all the pixels will be the same or almost similar. However, in high-textured regions, pixels located on an edge or spot will have relatively higher gradient magnitudes than the other pixels in the local neighborhood. Hence, the GLTP operator employs a threshold region  $\pm t$  around the center gradient value of a  $3 \times 3$  local neighborhood in order to differentiate between smooth and high-textured facial regions. Here, neighbor gradient values falling in the  $\pm t$  threshold region around the center gradient value  $G_c$  are quantized to 0; those below  $G_c - t$  and those above  $G_c + t$  are quantized to  $-1$  and  $+1$ , respectively, as shown in the following equation:

$$S_{\text{GLTP}}(G_c, G_i) = \begin{cases} -1, & G_i < G_c - t, \\ 0, & G_c - t \leq G_i \leq G_c + t, \\ +1, & G_i > G_c + t. \end{cases} \quad (5)$$

Here,  $G_c$  is the gradient magnitude of the center  $(x_c, y_c)$  of a  $3 \times 3$  neighborhood,  $G_i$  is the gradient magnitude of the surrounding neighbors, and  $t$  is a threshold. Finally, the GLTP code is obtained by concatenating the results. The basic gradient LTP encoding scheme is illustrated in Figure 3.

**3.2. Positive and Negative GLTP Codes.** One consequence of using three-level encoding is that the number of possible GLTP patterns ( $3^8$ ) is much higher than the number of possible LBP patterns ( $2^8$ ), which results in a high-dimensional feature vector. Different approaches [4, 21] have been proposed to reduce the number of possible ternary patterns. Here, we have adopted the approach proposed by Tan and Triggs [4], where each ternary code is split into its

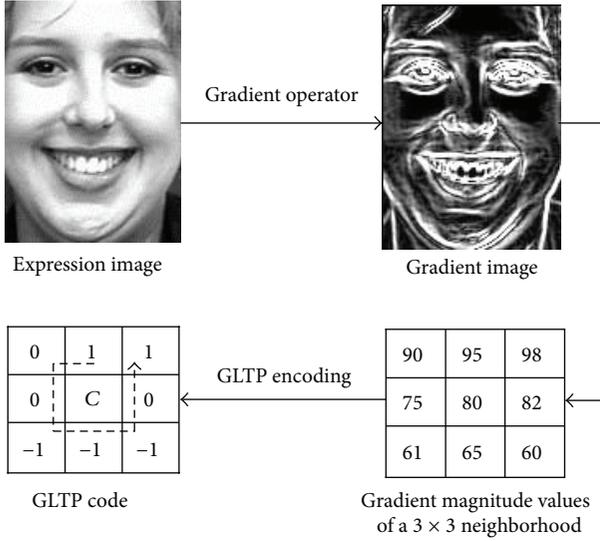


FIGURE 3: Illustration of the proposed GLTP encoding scheme. Here, the GLTP code is 100(-1)(-1)(-1)01 for threshold  $t = 10$ .

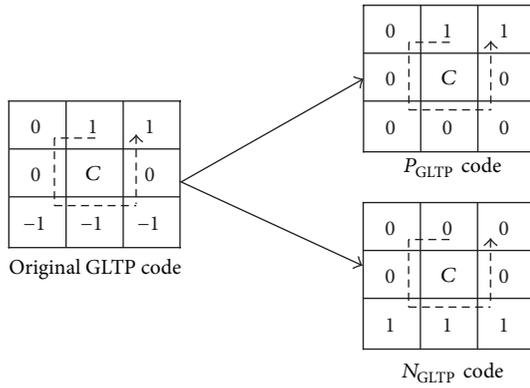


FIGURE 4: Generation of  $P_{\text{GLTP}}$  and  $N_{\text{GLTP}}$  codes from the original GLTP code.

corresponding positive ( $P_{\text{GLTP}}$ ) and negative ( $N_{\text{GLTP}}$ ) parts and treated as individual binary patterns, as shown in

$$\begin{aligned}
 P_{\text{GLTP}} &= \sum_{i=0}^7 S_P(S_{\text{GLTP}}(i)) \times 2^i, \\
 S_P(v) &= \begin{cases} 1, & \text{if } v > 0, \\ 0, & \text{otherwise,} \end{cases} \\
 N_{\text{GLTP}} &= \sum_{i=0}^7 S_N(S_{\text{GLTP}}(i)) \times 2^i, \\
 S_N(v) &= \begin{cases} 1, & \text{if } v < 0, \\ 0, & \text{otherwise.} \end{cases}
 \end{aligned} \tag{6}$$

Here,  $P_{\text{GLTP}}$  and  $N_{\text{GLTP}}$  are the corresponding positive and negative parts of the GLTP code  $S_{\text{GLTP}}$ . The process is illustrated in Figure 4.

#### 4. Facial Feature Description Based on GLTP Codes

Applying the GLTP operator on a facial image will produce two encoded image representations: one for the  $P_{\text{GLTP}}$  and the other for the  $N_{\text{GLTP}}$ . First, histograms are computed from these two encoded images using

$$H_{P_{\text{GLTP}}}(\tau) = \sum_{r=1}^M \sum_{c=1}^N f(P_{\text{GLTP}}(r, c), \tau), \tag{7}$$

$$H_{N_{\text{GLTP}}}(\tau) = \sum_{r=1}^M \sum_{c=1}^N f(N_{\text{GLTP}}(r, c), \tau), \tag{8}$$

$$f(a, \tau) = \begin{cases} 1 & a = \tau \\ 0 & \text{otherwise.} \end{cases} \tag{9}$$

Here,  $\tau$  is the positive or negative GLTP code value. Histograms computed from the  $P_{\text{GLTP}}$  and  $N_{\text{GLTP}}$  encoded images are then concatenated spatially to produce the GLTP histogram, which represents the occurrence information of the  $P_{\text{GLTP}}$  and  $N_{\text{GLTP}}$  binary patterns. The flowchart for computing the GLTP histogram is shown in Figure 5.

Spatial histograms computed from the whole encoded image do not reflect the location information of the micropatterns, only their occurrence frequencies are represented [1]. However, it is understandable that a histogram representation that combines the location information of the GLTP micropatterns with their occurrence frequencies is able to describe the local texture more accurately and effectively [22, 23]. Therefore, in order to incorporate some degree of location information with the GLTP histogram, each facial image is divided into a number of regions, and individual GLTP histograms (representing the occurrence information of the micropatterns from the corresponding local region) computed from each of the regions are concatenated to obtain a spatially combined GLTP histogram. In the facial expression recognition system, this combined GLTP histogram is used as the facial feature vector. The process of generating the combined GLTP histogram is illustrated in Figure 6.

#### 5. Expression Recognition Using Support Vector Machine (SVM)

Shan et al. [16] presented a comparative analysis of four different machine learning techniques for the facial expression recognition task, namely, template matching, linear discriminant analysis, linear programming, and support vector machine. Among these methods, support vector machine (SVM) achieved the best recognition performance. Hence, in our study, we use SVM to classify facial expressions based on the GLTP features.

Support vector machine (SVM) is a well-established machine learning approach, which has been successfully adopted in different data classification problems. The concept of SVM is based on the modern statistical learning theory. For data classification, SVM first implicitly maps the data into a higher dimensional feature space and then constructs

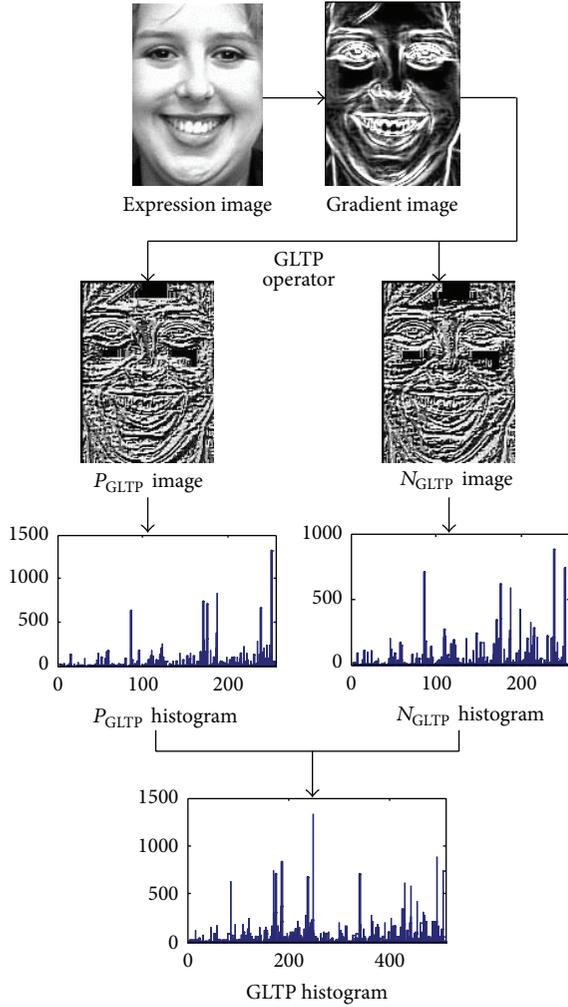


FIGURE 5: Pictorial illustration of the GLTP histogram generation process.

a hyperplane in such a way that the separating margin between the samples of two classes is optimal. This separating hyperplane then functions as the decision surface.

Given a set of labeled training samples  $T = \{(\mathbf{x}_i, l_i), i = 1, 2, \dots, L\}$ , where  $\mathbf{x}_i \in \mathbf{R}^P$  and  $l_i \in \{-1, 1\}$ , a new test data  $\mathbf{x}$  is classified by

$$f(\mathbf{x}) = \text{sign} \left( \sum_{i=1}^L \alpha_i l_i K(\mathbf{x}_i, \mathbf{x}) + b \right). \quad (10)$$

Here,  $\alpha_i$  are Lagrange multipliers of dual optimization problem,  $b$  is a threshold parameter, and  $K$  is a kernel function. SVM constructs a hyperplane which lies on the maximum separating margin with respect to the training samples with  $\alpha_i > 0$ . These samples are called the support vectors.

SVM takes binary decisions by constructing the separating hyperplane between the positive and negative examples. To achieve multiclass classification, we can adopt either the one-against-rest or several two-class decision problems. In this study, the one-against-rest approach was employed. We

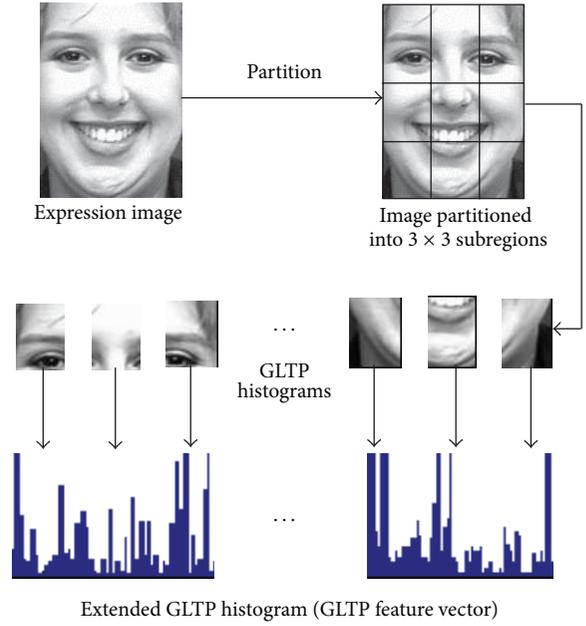


FIGURE 6: Each expression image is partitioned into a number of regions, and individual GLTP histograms generated from each of the regions are concatenated to form the feature vector.

used radial basis function (RBF) kernel for the classification problem. The radial basis function  $K$  can be defined as

$$K(\mathbf{x}_i, \mathbf{x}) = \exp(-\gamma \|\mathbf{x}_i - \mathbf{x}\|^2), \quad \gamma > 0, \quad (11)$$

$$\|\mathbf{x}_i - \mathbf{x}\|^2 = (\mathbf{x}_i - \mathbf{x})^t (\mathbf{x}_i - \mathbf{x}).$$

Here,  $\gamma$  is a kernel parameter. We carried out a grid search for selecting appropriate parameter value, as suggested in [24].

## 6. Experiments and Results

**6.1. Experimental Setup and Dataset Description.** To evaluate the effectiveness of the proposed face feature descriptor, experiments were conducted on images collected from a well-known image database, namely, the Cohn-Kanade (CK) facial expression database [18]. In the CK database, a sample set of 100 students, aging from 18 to 30 during image acquisition, were included. A majority of the subjects (65%) were female; 15% of the samples were African-American, and 3% were Asian or of Latin descent. Each of the students displayed facial expressions starting from nonexpressiveness to one of the aforementioned six prototypic emotional expressions in the image acquisition process. These image sequences were then digitized into  $640 \times 480$  or  $640 \times 690$  pixel resolutions. In our setup, a set of 1224 facial image sequences were selected from 96 subjects, and each of the images was given a label describing the subject's facial expression. The dataset containing the 6 classes of expressions was then extended by 408 images of neutral facial images to obtain the 7-class expression dataset. Figure 7 shows sample prototypic expression images from the CK database.

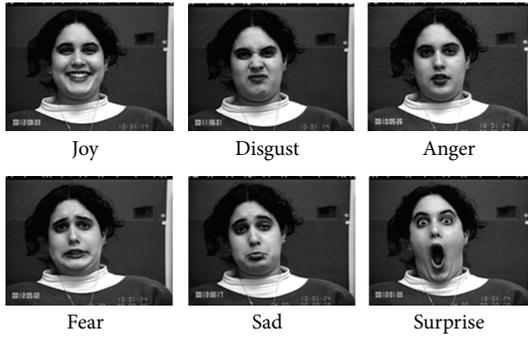


FIGURE 7: Sample 6-class expression images from the CK database [18].



FIGURE 8: Cropping of a sample face image from the original one.

We cropped the selected images from the original ones based on the ground truth of the positions of two eyes, which were then normalized to  $150 \times 110$  pixels. Figure 8 shows a sample cropped facial image from CK database. A tenfold cross-validation was carried out to compute the classification rate of the proposed method. In tenfold cross-validation, ten subsets comprising equal number of instances are formed by partitioning the whole dataset randomly. The classifier is first trained on the nine subsets, and then the remaining set is used for testing. This process is repeated for 10 times, and the average classification rate is computed. The threshold value  $t$  was set to 10 empirically.

**6.2. Experimental Results.** The classification rate of the proposed method can be influenced by adjusting the number of regions into which the expression images are to be split [2]. We have considered three cases in our experiments as opted in [2], where images were divided into  $3 \times 3$ ,  $5 \times 5$ , and  $7 \times 6$  regions. We have compared our proposed method with 3 widely used local texture descriptors, namely, local binary pattern (LBP) [16], local ternary pattern (LTP) [4], and local directional pattern (LDP) [2]. Tables 1 and 2 show the classification rates of these local texture descriptors for the 6-class and the 7-class expression recognition problem, respectively. It can be observed that dividing an image with higher number of regions will produce higher classification rate, since the feature descriptor then contains more location and spatial information of the local patterns. However, the feature vector length will also be higher in such cases, which affects the computational efficiency. Hence, selection of the number of regions is a trade-off between computational efficiency and classification rate.

TABLE 1: Recognition rate (%) for the CK 6-class expression dataset using different local texture descriptors.

Operator	Classification rate (%) for different number of regions		
	$3 \times 3$	$5 \times 5$	$7 \times 6$
LBP	79.3	89.7	90.1
LTP	87.3	92.3	93.6
LDP	80.2	91.9	93.7
GLTP	90.5	96.4	97.2

TABLE 2: Recognition rate (%) for the CK 7-class expression dataset using different local texture descriptors.

Operator	Classification rate (%) for different number of regions		
	$3 \times 3$	$5 \times 5$	$7 \times 6$
LBP	73.8	80.9	83.3
LTP	81.3	88.5	88.9
LDP	75.7	86.3	88.4
GLTP	84.0	90.6	91.7

TABLE 3: Confusion matrix of CK 6-class recognition using GLTP for images partitioned into  $7 \times 6$  regions.

	Anger (%)	Disgust (%)	Fear (%)	Joy (%)	Sad (%)	Surprise (%)
Anger	98.4	0	0	0.8	0	0.8
Disgust	0.5	94.4	0	0	5.1	0
Fear	0	0.7	97.1	0	0.4	1.8
Joy	1.1	1.1	0	97.8	0	0
Sad	0	4.5	0	0	95.5	0
Surprise	0	0	0	0	0	100

For both the 6-class and the 7-class expression recognition problems, the proposed GLTP feature descriptor achieves the highest recognition rate for images partitioned into different number of regions. For the 6-class dataset, GLTP achieves an excellent recognition rate of 97.2%. On the other hand, for the 7-class dataset, the recognition rate is 91.7%. Here, inclusion of neutral expression images results in a decrease in the accuracy. For both the 6-class and the 7-class recognition problems, the highest classification rate is obtained for images partitioned into  $7 \times 6$  regions. The confusion matrix of recognition using the GLTP descriptor for the 6-class and the 7-class datasets is shown in Tables 3 and 4, respectively, which provides a better picture of the recognition accuracy of individual expression types. It can be observed that, for the 6-class recognition, all the expressions can be recognized with high accuracy. For the 7-class dataset, while anger, disgust, fear, joy, and surprise can be recognized with high accuracy, the recognition rates of sadness and neutral expressions are lower than the average. Evidently, inclusion of neutral expression images results in a decrease in the accuracy, since many sad expression images are confused with the neutral expression images and vice versa.

TABLE 4: Confusion matrix of CK 7-class recognition using GLTP for images partitioned into  $7 \times 6$  regions.

	Anger (%)	Disgust (%)	Fear (%)	Joy (%)	Sad (%)	Surprise (%)	Neutral (%)
Anger	95.6	1.8	0	0	0	2.6	0
Disgust	0	93.7	2.1	0	1.5	2.7	0
Fear	0	2.5	94.0	0	0	1.3	2.2
Joy	0.6	0.6	0	94.7	0	3.5	0.6
Sad	0	0	0	0	85.1	0	14.9
Surprise	1.8	1.8	0	5.2	0.5	90.7	0
Neutral	0	0	1.7	0	9.8	0	88.5

The reason behind the superiority of GLTP face descriptor is the utilization of robust gradient magnitude values with a three-level encoding approach, which facilitates the discrimination between smooth and high-textured face regions and thus ensures generation of consistent texture micropatterns even under the presence of illumination variation and random noise.

## 7. Conclusion

This paper presents a new local texture pattern, the gradient local ternary pattern (GLTP), for robust facial expression recognition. Since gradient magnitude values are more robust than gray levels under the presence of illumination variations, the proposed method encodes the gradient values of a local neighborhood with respect to a threshold region around the center gradient, which facilitates robust description of local texture primitives, such as edges, spots, or corners under different lighting conditions. In addition, with the help of the threshold region defined around the center, the proposed method can effectively differentiate between smooth and high-textured facial areas, which enable the formation of GLTP codes consistent with the local texture property. Experiments with prototypic expression images from the Cohn-Kanade expression database demonstrate that the proposed GLTP operator can effectively represent facial texture and thus achieves superior performance than some widely used local texture patterns. In the future, we plan to apply the GLTP feature descriptor in other face-related recognition problems such as face recognition and gender classification for intelligent consumer products and applications development.

## References

- [1] F. Ahmed, H. Bari, and E. Hossain, "Person-independent facial expression recognition based on Compound Local Binary Pattern (CLBP)," *International Arab Journal of Information Technology*, vol. 11, no. 2, 2013.
- [2] 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.
- [3] F. Ahmed and M. H. Kabir, "Directional ternary pattern (DTP) for facial expression recognition," in *Proceedings of the IEEE International Conference on Consumer Electronics (ICCE '12)*, pp. 265–266, Las Vegas, Nev, USA, January 2012.
- [4] X. Tan and B. Triggs, "Enhanced local texture feature sets for face recognition under difficult lighting conditions," in *IEEE International Workshop on Analysis and Modeling of Faces and Gestures*, vol. 4778 of *Lecture Notes in Computer Science*, pp. 168–182, 2007.
- [5] P. Ekman and W. Friesen, *Facial Action Coding System: A Technique for Measurement of Facial Movement*, Consulting Psychologists Press, Palo Alto, Calif, USA, 1978.
- [6] Z. Zhang, "Feature-based facial expression recognition: sensitivity analysis and experiments with a multilayer perceptron," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 13, no. 6, pp. 893–911, 1999.
- [7] G. D. Guo and C. R. Dyer, "Simultaneous feature selection and classifier training via linear programming: a case study for face expression recognition," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 346–352, June 2003.
- [8] M. Valstar, I. Patras, and M. Pantic, "Facial action unit detection using probabilistic actively learned support vector machines on tracked facial point data," in *IEEE CVPR Workshop*, vol. 3, pp. 76–84, 2005.
- [9] C. Padgett and G. Cottrell, "Representation face images for emotion classification," *Advances in Neural Information Processing Systems*, vol. 9, pp. 894–900, 1997.
- [10] M. S. Bartlett, J. R. Movellan, and T. J. Sejnowski, "Face recognition by independent component analysis," *IEEE Transactions on Neural Networks*, vol. 13, no. 6, pp. 1450–1464, 2002.
- [11] C. C. Fa and F. Y. Shih, "Recognizing facial action units using independent component analysis and support vector machine," *Pattern Recognition*, vol. 39, no. 9, pp. 1795–1798, 2006.
- [12] M. J. Lyons, "Automatic classification of single facial images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 21, no. 12, pp. 1357–1362, 1999.
- [13] Y. Tian, "Evaluation of face resolution for expression analysis," in *IEEE Workshop on Face Processing in Video*, 2004.
- [14] T. Jabid, H. Kabir, and O. Chaei, "Local Directional Pattern (LDP) for face recognition," in *Proceedings of the International Conference on Consumer Electronics (ICCE '10)*, pp. 329–330, Las Vegas, Nev, USA, January 2010.
- [15] S. Zhao, Y. Gao, and B. Zhang, "Sobel-LBP" in *IEEE International Conference on Image Processing*, pp. 2144–2147, 2008.
- [16] C. Shan, S. Gong, and P. W. McOwan, "Facial expression recognition based on Local Binary Patterns: a comprehensive study," *Image and Vision Computing*, vol. 27, no. 6, pp. 803–816, 2009.
- [17] G. Zhao and M. Pietikäinen, "Boosted multi-resolution spatiotemporal descriptors for facial expression recognition," *Pattern Recognition Letters*, vol. 30, no. 12, pp. 1117–1127, 2009.
- [18] T. Kanade, J. Cohn, and Y. Tian, "Comprehensive database for facial expression analysis," in *IEEE International Conference on Automated Face and Gesture Recognition*, pp. 46–53, 2000.
- [19] T. Ojala, M. Pietikäinen, and T. Mäenpää, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 7, pp. 971–987, 2002.
- [20] T. Ahonen, A. Hadid, and M. Pietikäinen, "Face description with local binary patterns: application to face recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 12, pp. 2037–2041, 2006.

- [21] D. He and N. Cercone, "Local triplet pattern for content-based image retrieval," in *International Conference on Image Analysis and Recognition*, pp. 229–238, 2009.
- [22] S. Gundimada and V. K. Asari, "Facial recognition using multisensor images based on localized kernel eigen spaces," *IEEE Transactions on Image Processing*, vol. 18, no. 6, pp. 1314–1325, 2009.
- [23] F. Ahmed, "Gradient directional pattern: a robust feature descriptor for facial expression recognition," *IET Electronics Letters*, vol. 48, no. 19, pp. 1203–1204, 2012.
- [24] C.-W. Hsu and C.-J. Lin, "A comparison of methods for multiclass support vector machines," *IEEE Transactions on Neural Networks*, vol. 13, no. 2, pp. 415–425, 2002.



**Hindawi**

Submit your manuscripts at  
<http://www.hindawi.com>

