Facial expression recognition based on improved completed local ternary patterns*

LUO Yuan (罗元)[1], **LIU Xing-yao** (刘星遥)[1]**, **ZHANG Yi** (张毅)[2], **CHEN Xue-feng** (陈雪峰)[1], and **CHEN Zhuo** (陈卓)[1]

1. Key Laboratory of Optoelectronic Information Sensing and Technology, Chongqing University of Posts and Telecommunications, Chongqing 400065, China

2. Engineering Research Center for Information Accessibility and Service Robots, Chongqing University of Posts and Telecommunications, Chongqing 400065, China

(Received 19 August 2018; Revised 5 December 2018)

The information of expression texture extracted by the completed local ternary patterns (CLTP) method is not accurate enough, which may cause low recognition rate. Therefore, an improved completed local ternary patterns (ICLTP) is proposed here. Firstly, the Scharr operator is used to calculate gradient magnitudes of images to enhance the detail of texture, which is beneficial to obtaining more accurate expression features. Secondly, two different neighborhoods of CLTP features are combined to obtain much information of facial expression. Finally, K nearest neighbor (KNN) and sparse representation classifier (SRC) are combined for classification and a 10-fold cross-validation method is tested in the JAFFE and CK+ databases. The results show that the ICLTP method can improve the recognition rate of facial expression and reduce the confusion between various expressions. Especially, the misrecognition rate of other six expressions recognized as neutral is reduced in the 7-class expression recognition.

Document code: A **Article ID:** 1673-1905(2019)03-0224-7

DOI https://doi.org/10.1007/s11801-019-8136-z

Facial expression recognition which is an aspect of emotion recognition plays a very important role in interpersonal communication, which not only shows emotions but also is an important way to spread emotional information and coordinate the relationship between the two parties. The studies have shown that the information transmitted through facial expressions is as high as 55% of the total amount of information in human communication, while the information transmitted through voice and language accounts for 38% and 7% of the total information, respectively. By recognizing facial expression, a large amount of valuable information can be obtained. Therefore, facial expression recognition which based on human visual features and uses facial features to classify facial expression has become an important research in the fields of human-computer interaction, emotion computing and machine vision.

It is very important for facial expression recognition to extract effective facial expression features from facial images. The local binary patterns (LBP)[1] and its variants[2,3] have attracted much attention because of their robust performance in facial expression recognition. The LBP encodes the local texture of the image by quantifying the neighbor gray levels of the local neighborhood of the center pixel. Although the LBP using grayscale values to encode image textures is computationally efficient, but it exhibits weakly performance in the presence of non-monotonic illumination variations and random noise because the LBP code can be easily changed with little changes in gray level. Local direction patterns (LDP)[4] uses a different texture encoding scheme than LBP. And LDP uses the directional edge response values instead of the gray levels for encoding. Although LDP has proven to achieve a better recognition performance than LBP, but its two-level discriminant encoding approach tends to produce inconsistent patterns in uniform and near-uniform regions. And LDP relies on the number of edge direction parameters. Later, local ternary patterns (LTP)[5] was introduced and extended the two-level discriminant code to three-level code for increasing the resolution level of the gray level, which not only enhanced the robustness of the uniform and approximately uniform regions, but also is insensitive to noise. LTP which is an extension of LBP inherits the disadvantage from LBP that may incorrectly classify two or more different patterns into the same class, thereby reducing its discrimination. The completed local ternary patterns (CLTP)[6,7] not only inherits the advantages of LTP, but also solves the problem that different patterns may fall into the same class. The CLTP not only encodes the sign information of an image, but also encodes the amplitude information and the center pixel value of an image.

* This work has been supported by the National Natural Science Foundation of China (No.51604056), and the Chongqing Science and Technology Commission (No.cstc2015jcyjBX0066).

** E-mail: 403364852@qq.com

Therefore, each pattern is jointly represented by three encoded values, so CLTP cannot classify different patterns into the same class. Although CLTP shows superior performance in texture recognition, but it shows weak performance in facial expression recognition.

An improved CLTP is proposed for improving the recognition rate of facial expression. The ICLTP method uses gradient images instead of original images for feature extraction and the robust gradient amplitude values instead of the gray value is used to encode expression information, then merges different neighborhoods of CLTP features. The improved CLTP inherits advantages of the CLTP method and also make features of facial expression contain more accurate information of facial expression which is beneficial to improve the recognition rate of facial expression.

LTP is an extension of the LBP that introduced a −1 value, which extends the encoding from two-valued patterns (0,1) to three-valued patterns (−1,0,1). The LTP has better performance in random noises and illumination variations compared with LBP. The following equations show that how to compute the LTP operator:

$$LTP_{P,R}=\sum_{p=0}^{P-1}s\left(g_p-g_c\right)2^p, \tag{1}$$

$$s\left(x\right)=\begin{cases}1, & x\geq t\\ 0, & \left|x\right|\leq t\\ -1, & x<-t\end{cases}. \tag{2}$$

If the difference between the gray value of the center pixel g_c and the gray value of the pixel $g_p\,(0,1,\cdots,P-1)$ in the neighborhood falls within the range of [$-t$, t], the value is quantized to 0 and the quantization is 1 if it is above this threshold and the quantization is −1 if it is below this threshold, where t is the control parameter and P is the total number of pixels contained in its neighborhood and R is the radius of the neighborhood. Fig.1 has shown that the encoding process of LTP. Assuming that the value of the parameter t is 5, the interval range is [83, 93].

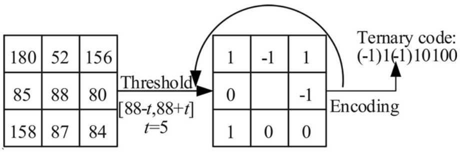


Fig.1 The encoding process of LTP

Although the original LTP algorithm is not sensitive to noise, but the dimension of the feature vector is increased from 2^P to 3^P. If P=8, the dimension is increased from 256 to 6 561, which is increased by about 25 times. As the value of P is increased, the dimension will increase exponentially. The increased number of dimensions will reduce the efficiency of LTP. To solve this problem, the LTP can be splits into its positive and negative LBP as shown in Fig.2. Therefore, the final feature vector is considered to be connected by positive and negative LBPs, and the dimension of feature vector of the image is reduced from 3^P to 2×2^P. For the sake of simplicity, the positive and negative LBP are separately described to obtain the histogram and the results are connected together at the end of the calculation.

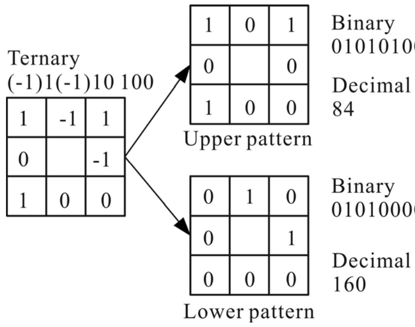


Fig.2 Splitting an LTP into positive and negative LBP

The CLTP is an extension of LTP, which introduces the encoding of amplitude information and central pixel values. The CLTP solves the shortcoming that LBP may classify two or more different patterns into the same class. In CLTP, local difference of the image can be decomposed into two parts, sign components and magnitude components, are calculated as follows:

$$\begin{aligned}&s_p^{\text{upper}}=s\left(g_p-\left(g_t+t\right)\right) \quad s_p^{\text{lower}}=s\left(g_p-\left(g_t-t\right)\right)\\ &m_p^{\text{upper}}=\left|g_p-\left(g_c+t\right)\right| \quad m_p^{\text{lower}}=\left|g_p-\left(g_c-t\right)\right|\end{aligned}, \tag{3}$$

where g_c and g_p are the gray value of central pixel and the gray value of neighborhood pixel, respectively. t is the control parameter.

The $CLTP\_S_{P,R}^{\text{upper}}$ and $CLTP\_S_{P,R}^{\text{lower}}$ can be built by using the $s_{P,R}^{\text{upper}}$ and $s_{P,R}^{\text{lower}}$ as follows:

$$\begin{aligned}&CLTP\_S_{P,R}^{\text{upper}}=\sum_{p=0}^{P-1}s\left(g_p-\left(g_c+t\right)\right)2^p\\ &s_p^{\text{upper}}=\begin{cases}1, & g_p\geq g_c+t\\ 0, & \text{otherwise}\end{cases}\end{aligned}, \tag{4}$$

$$\begin{aligned}&CLTP\_S_{P,R}^{\text{lower}}=\sum_{p=0}^{P-1}s\left(g_p-\left(g_c-t\right)\right)2^p\\ &s_p^{\text{lower}}=\begin{cases}1, & g_p<g_c-t\\ 0, & \text{otherwise}\end{cases}\end{aligned}. \tag{5}$$

The $CLTP\_S_{P,R}^{\text{upper}}$ and $CLTP\_S_{P,R}^{\text{lower}}$ are concatenated to form the $CLTP\_S_{P,R}$ as follows:

$$CLTP\_S_{P,R}=\left[CLTP\_S_{P,R}^{\text{upper}} \quad CLTP\_S_{P,R}^{\text{lower}}\right]. \tag{6}$$

The two magnitude complementary components m_p^{upper} and m_p^{lower} are used to build the $CLTP\_M_{P,R}$ as follows:

$$\begin{aligned}&CLTP\_M_{P,R}^{\text{upper}}=\sum_{p=0}^{P-1}t\left(m_p^{\text{upper}},c\right)2^p\\ &t\left(m_p^{\text{upper}},c\right)=\begin{cases}1, & \left|g_p-\left(g_c+t\right)\right|\geq c\\ 0, & \left|g_p-\left(g_c+t\right)\right|<c\end{cases}\end{aligned}, \tag{7}$$

$$CLTP\_M_{P,R}^{\text{lower}}=\sum_{p=0}^{P-1}t\left(m_p^{\text{lower}},c\right)2^p$$
$$t\left(m_p^{\text{lower}},c\right)=\begin{cases}1, & \left|g_p-\left(g_c-t\right)\right|\geq c\\ 0, & \left|g_p-\left(g_c-t\right)\right|<c\end{cases}, \tag{8}$$

$$CLTP\_M_{P,R}=\left[CLTP\_M_{P,R}^{\text{upper}}\quad CLTP\_M_{P,R}^{\text{lower}}\right], \tag{9}$$

where c is the mean value of m_p in the whole image.

The $CLTP\_C_{P,R}$ can be built by using the $CLTP\_C_{P,R}^{\text{upper}}$ and $CLTP\_C_{P,R}^{\text{lower}}$ as follows:

$$\begin{aligned}CLTP\_C_{P,R}^{\text{upper}}&=t\left(g_c^{\text{upper}},c_1\right)\\ g_c^{\text{upper}}&=g_c+t\end{aligned}, \tag{10}$$

$$\begin{aligned}CLTP\_C_{P,R}^{\text{lower}}&=t\left(g_c^{\text{lower}},c_1\right)\\ g_c^{\text{lower}}&=g_c-t\end{aligned}, \tag{11}$$

$$CLTP\_C_{P,R}=\left[CLTP\_C_{P,R}^{\text{upper}}\quad CLTP\_C_{P,R}^{\text{lower}}\right], \tag{12}$$

where c_I is the average gray value of the image.

The three operator *CLTP_S*, *CLTP_M* and *CLTP_C*, could be combined int two ways, jointly or hybridly. In the first way, all three operators are connected in parallel to form a 3-D joint histogram called *CLTP_S/M/C*. In the second way, the sign or amplitude component is connected in parallel with *CLTP_C* to form a 2-D joint histogram called *CLTP_S/C* or *CLTP_M/C*, and then the 2-D joint histogram is transformed into a 1-D histogram, and a joint histogram is formed by using the 1-D histogram concatenated with *CLTP_M* or *CLTP_S*, which called *CLTP_S_M/C* or *CLTP_M_S/C*.

The CLTP solves the problem that different patterns may fall into the same class by introducing a three-valued encoding of amplitude components and center pixel values with additional discriminating information, but the CLTP cannot guarantee that the extracted features contain sufficiently sufficient or accurate texture information of facial expression. Ahmed et al[8] believe that applying gradient operators before feature extraction can enhance the details of texture, which is beneficial to make the features contain more accurate information of texture.

Therefore, the gradient operator is applied to the original expression image to generate the gradient image, then features are extracted from the gradient image. The more powerful gradient amplitude value is used instead of the gray value for CLTP coding so that the CLTP feature contains more accurate texture information of facial expression. The gradient magnitude of the image can be calculated using a Sobel or Scharr gradient operator. The Sobel gradient operator contains two 3×3 kernels, which are approximated by the convolution with the original image:

$$\boldsymbol{I}^x=\boldsymbol{G}_x*\boldsymbol{I}=\begin{bmatrix}-1&0&1\\-2&0&2\\-1&0&1\end{bmatrix}*\boldsymbol{I}, \tag{13}$$

$$\boldsymbol{I}^y=\boldsymbol{G}_y*\boldsymbol{I}=\begin{bmatrix}-1&-2&-1\\0&0&0\\1&2&1\end{bmatrix}*\boldsymbol{I}, \tag{14}$$

where $\boldsymbol{I}^x$ and $\boldsymbol{I}^y$ represent the horizontally and vertically filtered results, respectively. And the gradient magnitude can be given by combining $\boldsymbol{I}^x$ and $\boldsymbol{I}^y$ as follows:

$$I=\sqrt{\left(\boldsymbol{I}^x\right)^2+\left(\boldsymbol{I}^y\right)^2}. \tag{15}$$

In addition, you can use the simpler formula as follows:

$$I=\left|\boldsymbol{I}^x\right|+\left|\boldsymbol{I}^y\right|. \tag{16}$$

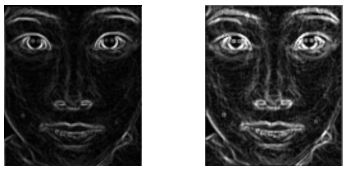
Inaccurate results may be produced when calculating gradient magnitude values of the image by using the Sobel operator which only computes an approximation of the derivatives of an image, though this estimation may be sufficient for most purposes. The Scharr operator is an optimized filter that convolves the image based on minimizing the weighted mean-squared angular error in the Fourier domain. Fig.3 is the filter masks of 3×3 Scharr kernel. The Scharr and Sobel operator have the same speed of calculation when calculating the gradient magnitude of an image.

$$\begin{bmatrix}-3&0&+3\\-10&0&+10\\-3&0&+3\end{bmatrix}\qquad\begin{bmatrix}-3&-10&-3\\0&0&0\\+3&+10&+3\end{bmatrix}$$

Horizontal Vertical

Fig.3 Filter masks for 3×3 Scharr kernel

But the Scharr operator has much higher precision than the Sobel operator so a higher precision representation of the gradient magnitude image is produced by using the Scharr operator, which is beneficial to the extraction of expression feature. As can be seen from Fig.4, the emotion texture of the gradient image generated by using the Scharr operator is clearer than the emotion texture of the gradient image generated by using the Sobel operator.



(a) (b)

Fig.4 (a) Sobel gradient magnitude image; (b) Scharr gradient magnitude image

Therefore, the gradient of the expression image can be calculated by using the Scharr gradient operator, then using the CLTP method to extract the feature. So the CLTP feature can have more accurate texture information of facial expression than the feature extracted from the original image. Fig.5 is a flow chart of Scharr-CLTP feature extraction.

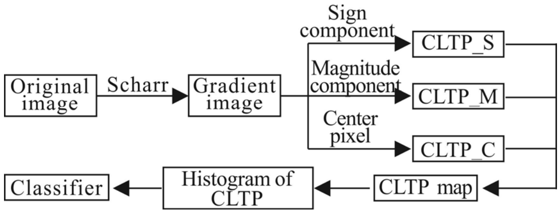


Fig.5 The flow chart of Scharr-CLTP feature extraction

The CLTP features contain accurate texture information of facial expression by encoding gradient amplitude values. However, the CLTP features will be extracted in two different neighborhoods, namely $CLTP_{8,1}$ and $CLTP_{16,3}$ features, then they will be merged as facial expression features which contains more expression texture information than a single CLTP feature. Therefore, expression features which contains more discriminative expression information can improve the recognition rate of facial expression, as shown in Fig.6.

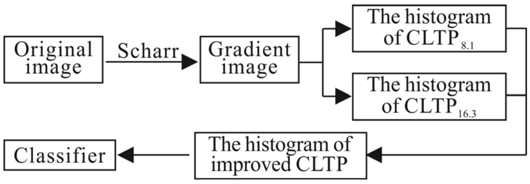


Fig.6 The flow chart of the improved CLTP expression feature extraction

The improved method adopts more accurate Scharr operator to generate gradient images instead of original images for feature extraction, then fusing two different neighborhoods of CLTP features as expression features for classification recognition which not only makes features contain accurate information of facial expression, but also contains more expression information than basic CLTP features. Firstly, Scharr gradient images are calculated and divided into 7×6 non-overlapping regions. Secondly, the $CLTP_{8,1}$ histogram of each image region is extracted. Then, each histogram is treated as a single vector and all histogram vectors are concatenated as the $CLTP_{8,1}$ histogram. Third, the same procedures are used for creating the $CLTP_{16,3}$ histogram. Finally, two histogram vectors are concatenated as a facial expression feature.

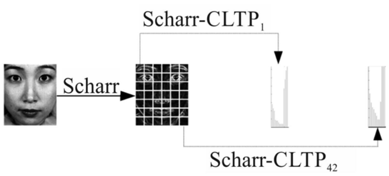


Fig.7 Shcarr-CLTP expression feature extraction

K nearest neighbor (KNN) and sparse representation classifier (SRC)[9] is used in the experiment. The basic idea of KNN-SRC is described as follows. First, a testing sample from all training samples is given to find its *K* nearest neighbors. Then, these *K* nearest neighbors are used for representing the testing sample. Finally, the residual is computed which between testing samples and the nearest neighbors belonging to each class and comparing the residual for deciding the class of the testing sample. KNN-SRC has better performance than SRC, which is mainly reflected in the following two aspects. First, KNN-SRC is more efficient than SRC in calculation. Second, a testing sample which is represented by the *K* nearest neighbors avoid problem that the training samples will more sparsely represent the testing samples.

The facial expression recognition system can recognize six basic expressions: Happy, Angry, Surprise, Fear, Disgust and Sadness. The 6-class expression classification and 7-class expression classification which add a neutral expression were experimented. The Japanese female facial expressions (JAFFE) database and the extended Cohn-Kanade (CK+) database are tested in the experiment. The JAFFE dataset contains 213 facial expression images. The format of each image is .tiff image format and the resolution of each image is 256×256 pixels. There are 10 persons and each person has seven expressions. There are about 3 or 4 sample images for each type of expression and they are all frontal images in the database. The CK database consists of 100 university students who at the time of their inclusion were between 18 to 30 years old; 65% were female, 15% were African-American, and 3% were Asian or Latino. CK+ is an extension of the CK database and the number of subjects increased to 123 and the number of image sequences increased to 593. In the experiment, 106 subjects of 309 sequences were selected for the experiment and selecting the three most expressive images from each sequence for 6-class expression recognition. In addition, the first frame image is selected as a neutral expression image for 7-class expression recognition. After choosing the images, they were cropped from the original one using the positions of two eyes and resized into 150×110 pixels. Fig.8 shows some examples of cropped facial images.

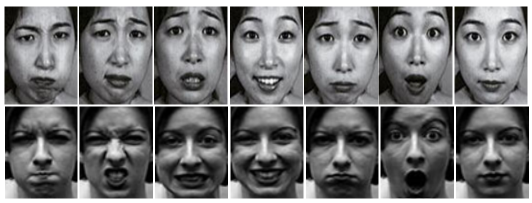


Fig.8 Some examples of cropped facial images

10-fold cross-validation approach was used in classification experiment. The tested database is divided into ten disjoint subsets. Then selecting nine of them as the training set and leaving different one as the testing set in every time. So, the average value of 10 tests performed on the facial database as the classification result.

In this paper, a 10-fold cross-validation method was tested for facial expression classification experiment in the JAFFE database. It can be seen from Tab.1 that the

ICLTP method improves the recognition rate of basic CLTP in facial expression recognition. The recognition rate of 6-class classification is improved by 4.9% and the recognition rate of 7-class classification is improved by 5.2%. We also know that the recognition rate of CLTP with different neighborhoods are different and the recognition rate of CLTP with Scharr operator is higher than CLTP without Scharr operator.

Tab.1 Recognition rates of different CLTP in the JAFFE database

Methods	6-class (%)	7-class (%)
$CLTP_{8,1}$	89.2±1.1	85.4±1.3
$CLTP_{16,3}$	90.5±0.9	86.9±1.1
Scharr-$CLTP_{8,1}$	92.4±0.7	88.5±0.9
Scharr-$CLTP_{16,3}$	93.8±0.6	90.2±0.9
ICLTP	**95.0±0.6**	**92.1±0.5**

It can be seen from Tab.2 that the ICLTP method improves the recognition rate of facial expression by 0.8% in 6-class classification and by 0.7% in 7-class classification compared with the es-LBP method which has the highest recognition rate. The ICLTP method improved the recognition rate by 11.7% in 6-class classification and by 10.4% in 7-class classification, which compared with the recognition rate of the IGLTP method. Although the ICLTP and IGLTP methods both use the Scharr operator to calculate the gradient of facial images in feature extraction process, but the IGLTP method only extracts sign components of the gradient image. The ICLTP method extracts amplitude components which has additional discriminant information and the CLTP features of different neighborhoods are merged. Therefore, expression features extracted by the ICLTP method contain more expression information than expression features extracted by the IGLTP method. And the recognition rate of facial expression using the IGLTP method is higher than using the IGLTP method.

Tab.2 Recognition rates of different methods in the JAFFE database

Methods	6-class (%)	7-class (%)
LBP[1]	87.5±5.1	81.9±5.2
LDP[4]	90.1±4.9	85.4±4.0
LDTP[10]	90.2±1.0	88.7±0.5
LDNP[11]	93.4±0.4	90.6±0.4
es-LBP[13]	94.2±0.7	91.4±0.8
GLTP[12]	77.0±1.1	74.4±1.3
IGLTP[12]	83.3±1.6	81.7±1.3
ICLTP	**95.0±0.6**	**92.1±0.5**

The confusion matrices (CMs) for 6-class and 7-class expression recognition using the JAFFE database are given in Tabs.3 and 4, respectively. It can be observed that the recognition rate of happy is the highest among 6-class and 7-class classification, which are 100% and 95.9%, respectively. Although expression of sad is easily recognized as disgust and anger, the ICLTP method improves its recognition rate and reduces the misrecognition rate of disgust or anger. The average recognition rate in 7-class classification is lower than in 6-class classification, because expression of neutral is added for tested in 7-class classification so that other expressions may be misidentified as neutral in 7-class classification. The misrecognition rate of other expressions recognized as neutral is reduced by using the ICLTP method.

Tab.3 CM of 6-class expression recognition in the JAFFE database

(%)	AN	DI	FE	HA	SA	SU
AN	**95.5**	4.5				
DI	2.1	**93.4**			4.5	
FE			**94.0**	1.2		4.8
HA				**100.0**		
SA	2.1	4.5			**93.4**	
SU			3.1			**96.9**

Tab.4 CM of 7-class expression recognition in the JAFFE database

(%)	AN	DI	FE	HA	SA	SU	NE
AN	**95.5**	4.5					
DI	3.5	**90.4**	4.4		1.7		
FE		1.1	**90.2**		3.3		5.4
HA				**95.9**			4.1
SA	2.1	5.5			**88.8**		3.6
SU			2.5			**94.2**	3.3
NE	2.6		3.1		1.5		**92.8**

The same experiment also tested in the CK+ database. It can be seen from Tabs.5 and 6 that the recognition rate of facial expression using the ICLTP or other methods is very high in the CK+ database. The recognition rate of 6-class classification is as high as above 95% and the recognition rate of 7-class classification as high as above 91%. The recognition rate of 6-class classification using the ICLTP method reached 99.4% and the recognition rate of 7-class classification reached 97.8%, which is improved by 1.9% and 0.9% compared with the basic CLTP method. The recognition rate of 6-class and 7-class classification using the ICLTP method are improved by 0.1% and 0.2% compared with the IGLTP method.

Tab.5 Recognition rates of different CLTP in the CK+ database

Methods	6-class (%)	7-class (%)
$CLTP_{8,1}$	97.0±1.1	96.6±1.3
$CLTP_{16,3}$	97.5±0.9	96.9±1.0
Scharr-$CLTP_{8,1}$	98.3±0.7	97.1±1.0
Scharr-$CLTP_{16,3}$	98.7±0.6	97.4±0.9
ICLTP	**99.4±0.7**	**97.8±0.8**

Tab.6 Recognition rates of different methods in the CK+ database

Methods	6-class (%)	7-class (%)
LBP[14]	95.1±0.5	91.8±0.4
Gabor+LBP[14]	97.4±0.2	95.5±0.3
GLTP[12]	98.9±0.2	96.9±0.2
IGLTP[12]	99.3±0.4	97.6±0.4
ICLTP	**99.4±0.7**	**97.8±0.8**

Tabs.7 and 8 show the confusion matrices of 6-class and 7-class expression recognition in the CK+ database. According to Tabs.7 and 8, the recognition rate of six expressions is over 98% and the recognition rate of seven expressions is over 94%. And we learned that the recognition rate of happy reached 100% in both 6-class and 7-class classification. The expression of sad has the same problem which has been appeared in the JAFFE database. It is easy to be recognized as disgust and anger. The recognition rate of sad using the IGLTP method was 97.3% while using the ICLTP method was 98.0% and the misrecognition rate of other expressions was reduced by 0.7%. The misrecognition rate using the ICLTP method is lower than using the IGLTP method.

Tab.7 CM of 6-class expression recognition in the CK+ database

(%)	AN	DI	FE	HA	SA	SU
AN	**99.4**	0.3			0.3	
DI	0.5	**99.4**			0.1	
FE			**99.4**			0.6
HA				**100.0**		
SA	0.9	1.1			**98.0**	
SU			0.6			**99.4**

Tab.8 CM of 7-class expression recognition in the CK+ database

(%)	AN	DI	FE	HA	SA	SU	NE
AN	**95.3**	0.2	0.1				4.4
DI	0.7	**98.6**					0.7
FE			**99.2**			0.6	0.2
HA				**100.0**			
SA	0.3	0.5	0.2		**94.0**		5.0
SU			1.0			**98.1**	0.9
NE	0.8	0.9	0.2		0.9		**97.2**

In this experiment, facial expression features of JAFFE and CK+ database were extracted by GLTP[12], IGLTP[12] and ICLTP, respectively. Finally, KNN-SRC was used for classification. The classification result is calculated as an average value from 10 tests performed on the data using 10-fold cross-validation approach.

The experimental comparison result of facial expression recognition running time is given in Tab.9 and 10. As can be seen from Tabs.9 and 10, the running time of the ICLTP method is increased, which compared with other methods in the training and testing phases. the total running time is nearly doubled compared with the IGLTP method which uses the same gradient operator. This is because the ICLTP method extracts the amplitude information of expression images in feature extraction, so features extracted by the ICLTP method contains additional expression texture information. In the training phase, features extracted by the ICLTP method also own more information of facial expression than by the IGLTP method to be processed by the classifier, so the running time is also increased. Since features extracted by the ICLTP method has more expression information, the recognition rate using the ICLTP method is higher than using the IGLTP method. However, the ICLTP method can obtain high recognition rate by increasing the running time. Therefore, the next step is to shorten the running time of facial expression recognition.

Tab.9 Running time of different methods (seconds) in the CK+ database

Methods	Training		Testing	
	6-class	7-class	6-class	7-class
GLTP	41.35	74.35	0.22	0.26
IGLTP	26.26	53.53	0.12	0.15
ICLTP	48.44	79.24	0.45	0.53

Tab.10 Running time of different methods (seconds) in the JAFFE database

Methods	Training		Testing	
	6-class	7-class	6-class	7-class
GLTP	27.39	54.20	0.37	0.51
IGLTP	18.46	24.86	0.31	0.36
ICLTP	38.20	58.93	0.43	0.53

In this paper, the precise Scharr operator is used to calculate the image gradient. Then two different neighborhoods of CLTP features are extracted from gradient images and merged together to generate facial expression features for classification recognition. The results tested on the JAFFE and CK+ database show that the ICLTP method improves the recognition rate of facial expression. And the recognition rate of the ICLTP method is higher than other methods from the comparative literatures. Therefore, the ICLTP method is an accurate method for feature extraction.

References

[1] Shan C, Gong S and Mcowan P W, Image & Vision Computing **27**, 803 (2009).

[2] Zhao G and Pietikäinen M, Pattern Recognition Letters **30**, 1117 (2009).

[3] Doshi N P, Schaefer G and Hossain S, Improved Dominant Local Binary Pattern Texture Features, IEEE International Conference on Informatics, Electronics and Vision, 1157 (2016).

[4] Jabid T, Kabir M H and Chae O, ETRI Journal **32**, 784 (2010).

[5] Tan X and Triggs B, IEEE Transactions on Image Processing **19**, 1635 (2010).

[6] Rassem T H and Khoo B E, The Scientific World Journal **2014**, 254 (2014).

[7] Rassem T H, Mohammed M F and Khoo B E, Performance Evaluation of Completed Local Ternary Patterns (CLTP) for Medical, Scene and Event Image Categorisation, IEEE International Conference on Software Engineering and Computer Systems, 33 (2015).

[8] Ahmed F and Hossain E, Chinese Journal of Engineering **2013**, 1 (2013).

[9] Ameur B, Masmoudi S and Derbel A G, Fusing Gabor and LBP Feature Sets for KNN and SRC-based Face Recognition, IEEE International Conference on Advanced Technologies for Signal and Image Processing, 453 (2016).

[10] A.R. Rivera, J.R. Castillo and O. Chae, Pattern Recognition Letters **51**, 94 (2015).

[11] A.R. Rivera, J.R. Castillo and O. Chae, IEEE Transactions on Image Processing **22**, 1740 (2013).

[12] Holder R P and Tapamo J R, EURASIP Journal on Image & Video Processing **2017**, 42 (2017).

[13] W.L. Chao, J.J. Ding and J.Z. Liu, Signal Processing **117**, 1 (2015).

[14] Sun Y and Yu J, Facial Expression Recognition by Fusing Gabor and Local Binary Pattern Features, MultiMedia Modeling, Springer International Publishing, 2017.