Turkish Journal of Electrical Engineering & Computer Sciences

http://journals.tubitak.gov.tr/elektrik/

Research Article

Turk J Elec Eng & Comp Sci
(2019) 27: 516 – 531

doi:10.3906/elk-1804-58

Local directional-structural pattern for person-independent facial expression recognition

Farkhod MAKHMUDKHUJAEV, Md. Tauhid Bin IQBAL, Byungyong RYU, Oksam CHAE*
Department of Computer Science and Engineering, College of Electronics and Information, Kyung Hee University, Yongin-si, Republic of Korea

Received: 08.04.2018 • **Accepted/Published Online:** 10.10.2018 • **Final Version:** 22.01.2019

Abstract: Existing popular descriptors for facial expression recognition often suffer from inconsistent feature description, experiencing poor accuracies. We present a new local descriptor, local directional-structural pattern (LDSP), in this work to address this issue. Unlike the existing local descriptors using only the texture or edge information to represent the local structure of a pixel, the proposed LDSP utilizes the positional relationship of the top edge responses of the target pixel to extract more detailed structural information of the local texture. We further exploit such information to characterize expression-affiliated crucial textures while discarding the random noisy patterns. Moreover, we introduce a globally adaptive thresholding strategy to exclude futile flat patterns. Hence, LDSP offers a stable description of facial expressions with the explicit representation of the expression-affiliated features along with the exclusion of random futile textures. We visualize the efficacy of the proposed method in three folds. First, the LDSP descriptor possesses a moderate code-length owing to the exclusion of the futile patterns, yielding less computation time than other edge descriptors. Second, for person-independent expression recognition in benchmark datasets, LDSP demonstrates higher accuracy than existing descriptors and other state-of-the-art methods. Third, LDSP shows better performance than other descriptors against noise and low resolution, exhibiting its robustness under such uneven conditions.

Key words: Local directional-structural pattern, facial expression recognition, feature extraction, person-independent expression recognition

1. Introduction

Automatic facial expression recognition (AFER) is an integral part of miscellaneous human–machine applications [1]. The task of AFER is challenging in person-independent environments, where in the training stage, a classifier does not have the advantage of memorizing any person's prior information that is to appear in the testing stage. Many real-life online systems, such as CCTV and surveillance videos, do not possess such prior information, signifying the importance of achieving better expression recognition accuracy in such person-independent environments. Simultaneously, achieving consistent performance in the presence of noise and low resolution is also important in order to comply with such online systems [2]. However, a stable representation of the facial image describing the expression-affiliated most crucial information is the key to achieve consistent performance in this regard, which is perhaps still absent in the current literature, contributing to the poor state-of-the-art results so far.

Existing methods for facial feature descriptions can be mainly categorized into two classes: sparse and dense. Sparse methods represent the facial image using the information from specific facial components [3, 4].

*Correspondence: oschae@khu.ac.kr

However, these methods often have poor accuracies due to the detection error of such facial components, as mentioned in [5]. Dense methods extract the appearance change information of the facial image and generate a dense description from it. Dense methods can further be classified into two classes: global and local. The global methods, such as eigenfaces [6], Fisherfaces [7], and 2D PCA [8] extract global appearance information from the whole face to represent the facial description, and thereby, owing to this global approach, they often miss important microlevel appearance-change information, as stated in [9, 10]. Local methods, on the contrary, aim to describe the microlevel appearance change information by means of the local coding scheme. To be precise, there are two kinds of local approaches: texture-based descriptors and edge-based descriptors. Among the texture-based descriptors, local binary pattern (LBP) [2] is the most used one, which is advantageous for its computational simplicity and its efficacy in monotonic illumination changes. However, LBP is found to be sensitive to noise and uneven illuminations, as pointed out in [11]. Besides, edge-based methods use the edge responses from compass masks [12] and pick top k-directional responses to formulate the feature codes. Examples of some of the prominent edge-based descriptors may include the local directional number pattern (LDN) [9], local directional pattern (LDP) [11], local principal texture pattern (LPTP) [13], and positional ternary pattern (PTP) [14, 15]. Although these descriptors exhibit promising results so far, achieving stable and robust definition of the structural detail of the local texture is still an issue for them. One key concern in this regard is that these descriptors only use principal edge directional information to represent the local texture. Such sole use of edge direction may fail to appropriately represent detailed structural information of the pixel in different situations. Another critical issue of these descriptors is the redundant code generation for different featureless textures, such as flat textures and noisy edge-like patterns, which are considered as expressionless textures in practice. Codes from such textures either perturb the other textured code-bins or generate unnecessary redundant code-bins, which, in turn, create ambiguity in the feature description and hinder the consistency of the descriptor. Apart from the above-mentioned local descriptors, recently deep learning methods have also shown promising performances in recognizing facial expressions [16–20]. However, the major concern regarding the use of these methods within current facial expression datasets is the unavailability of sufficient training samples that is required for proper training of the network-model. Some of the deep-methods, however, combat this issue by training with number of additional artificially-generated data [17–20], at the cost of high computational effort. In this paper, we mainly consider the strength of the feature for recognizing facial expressions without the effect of such training procedures with additional data, and hence we consider such an approach as out of the scope of this work.

In this work, we present a new local descriptor, local directional-structural pattern (LDSP), in order to address the above-mentioned issues of the existing edge descriptors. Unlike the existing descriptors using only the edge responses to generate code, the proposed descriptor utilizes the positional relationship among the top edge responses to extract detailed structural information of the local texture. In the coding, LDSP exploits such information to explicitly characterize the expression-affiliated most crucial textures while discarding the random noisy patterns. Moreover, we introduce a globally adaptive thresholding strategy in order to filter the featureless flat textures. Filtering these featureless patterns allows LDSP to hold less code-length, yielding less computation time than other existing descriptors. Experimental results show that LDSP outperforms existing descriptors and other state-of-the-art methods in several datasets for person-independent expression recognition.

We summarize the contributions as follows:

a) We present a new descriptor, LDSP, utilizing the positional relationship of the top edge responses to encode the expression-related crucial features while avoiding random futile patterns.

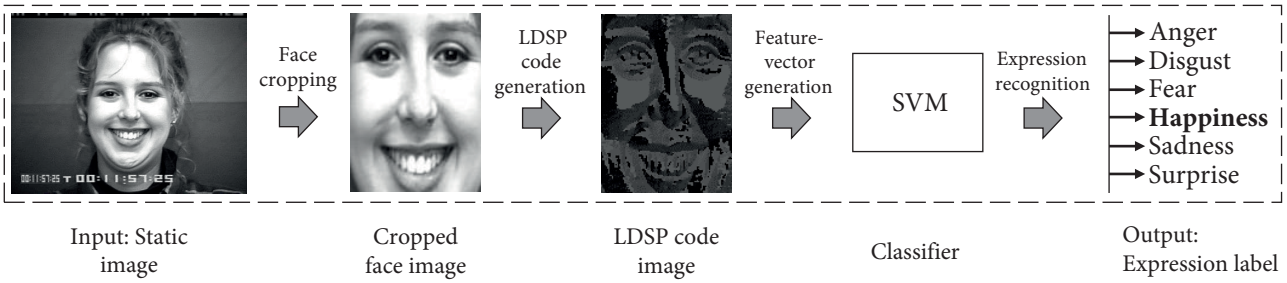


Figure 1. The overall flow of the proposed method.

b) We introduce a globally adaptive thresholding scheme to discard the featureless flat patterns having negligible contributions to the expression changes.

c) The exclusion of the featureless random patterns leads to a concise feature-vector length facilitating moderate computational time.

d) We exhibit the preeminence of our method in recognizing facial expressions in a person-independent environment in state-of-the-art datasets.

e) LDSP demonstrates better performance than other descriptors in the presence of noise and low resolution and thereby contributes to the overall improvement of facial expression recognition.

2. Methodology

The proposed LDSP generates 5-bit feature code at a target pixel, representing the structural detail of local texture. In expression recognition, vital facial features that change according to different expressions mostly appear as some of the basic texture-primitives such as edge and corner, and hence these texture-primitives are crucial to capture. To be precise, edges are evident in the boundary of different expressive features, such as eye border, lip border, wrinkle, and nose bridge, whereas eye corners and lip corners appear as corner-like texture. Existing edge descriptors only use microlevel edge-directional information to generate the code, which may fail to appropriately represent the structural detail of the above textures. We argue that an explicit representation, clearly characterizing such texture-primitives, can be more influential in this regard. To generate such a representation, we take advantage of the edge responses generated by compass masks and analyze the positional relationship among the top few edge responses to extract detailed structural information of the above texture-primitives. In practice, the top few edge responses appear in a certain way in edge and corner textures, whereas in random noisy edge-like textures, they appear differently, which we utilize to characterize these features in LDSP code. Moreover, with a new globally adaptive thresholding mechanism, we filter the featureless flat texture of the facial image having no effect on the expression changes. In this way, LDSP clearly encodes the structural information of the expressive features and discards the featureless patterns. The overall flow of this work is illustrated in Figure 1.

2.1. LDSP coding scheme

We start computing LDSP code at a target pixel by applying Kirsch compass masks [12] to it. Kirsch masks, as shown in Figure 2, convolve in eight different orientations to obtain eight corresponding directional edge response values as:

$$R_i = M_i * I(x, y), \quad 0 \leq i \leq 7, \tag{1}$$

$$\underset{M_0}{\begin{bmatrix}-3&-3&5\\-3&0&5\\-3&-3&5\end{bmatrix}}\quad\underset{M_1}{\begin{bmatrix}-3&5&5\\-3&0&5\\-3&-3&-3\end{bmatrix}}\quad\underset{M_2}{\begin{bmatrix}5&5&5\\-3&0&-3\\-3&-3&-3\end{bmatrix}}\quad\underset{M_3}{\begin{bmatrix}5&5&-3\\5&0&-3\\-3&-3&-3\end{bmatrix}}$$

$$\underset{M_4}{\begin{bmatrix}5&-3&-3\\5&0&-3\\5&-3&-3\end{bmatrix}}\quad\underset{M_5}{\begin{bmatrix}-3&-3&-3\\5&0&-3\\5&5&-3\end{bmatrix}}\quad\underset{M_6}{\begin{bmatrix}-3&-3&-3\\-3&0&-3\\5&5&5\end{bmatrix}}\quad\underset{M_7}{\begin{bmatrix}-3&-3&-3\\-3&0&5\\-3&5&5\end{bmatrix}}$$

Figure 2. Kirsch compass masks.

where I is the target pixel, M_i is the ith Kirsch mask among eight masks $(M_0, M_1, ..., M_7)$, and R_i is their corresponding edge response values. From all these responses, we pick the top k responses, which we utilize to extract further structural detail of the local texture. We define the kth highest response value and its position as:

$$D_k = \arg\max_k\{R_i : 0 \le i \le 7\}, \quad P_k = i, \tag{2}$$

where the value of the kth highest response among the eight R_i responses is represented by D_k, and its position, i, is denoted by P_k.

Since we analyze the positional relationship among the top k Kirsch edge responses to extract further information, it is very important to choose the number of k carefully. Existing descriptors such as LDP use three edge responses [11]; however, the top two edge responses yield consistent performances in recent works owing to their efficacy in characterizing the local texture structure [14, 15, 21]. Motivated by this, we utilize the positional information of the top two edge responses (P_1 and P_2) and analyze their relation to extract further detail of the local structure of the crucial texture-primitives, such as edge and corner. The structure of Kirsch masks is designed in such a way that, for the edge pattern, the second top response (P_2) appears next to the top response (P_1); besides, P_2 appears orthogonally to P_1 for the corner pattern, characterizing the corner-like orthogonal texture-structure [12, 21]. We provide a detailed illustration of such characteristics of Kirsch masks in Figure 3. We take advantage of these traits of Kirsch masks in order to explicitly characterize the texture-primitives. In the coding, we represent these structural traits using the positional distance between P_1 and P_2. To be specific, within the eight neighbors, in the case of P_2 appearing next to P_1, the positional distance is 1, while, in case of their orthogonal appearance, the distance is 2. Another important factor to note is that within the local neighborhood, the appearance of P_2 to P_1 is either clockwise or counter-clockwise. We consider all these combinations and get a total of four specific patterns contributing to characterize the structure of edge and corner patterns, which we also illustrate in Figure 3. We define these patterns as follows:

$$S = \begin{cases} \left.\begin{array}{ll} 0, & \text{if,} \quad P_2 \circlearrowleft P_1 \\ 1, & \text{otherwise} \end{array}\right\} |P_1 - P_2| = 1 \\ \left.\begin{array}{ll} 2, & \text{if,} \quad P_2 \circlearrowleft P_1 \\ 3, & \text{otherwise} \end{array}\right\} |P_1 - P_2| = 2 \\ 4, \quad \text{if,} \quad |P_1 - P_2| > 2, \end{cases} \tag{3}$$

where S stands for the structural feature code of the texture-primitives that is mainly defined by the absolute difference between P_1 and P_2 values followed by considering their clockwise or counter-clockwise appearance.

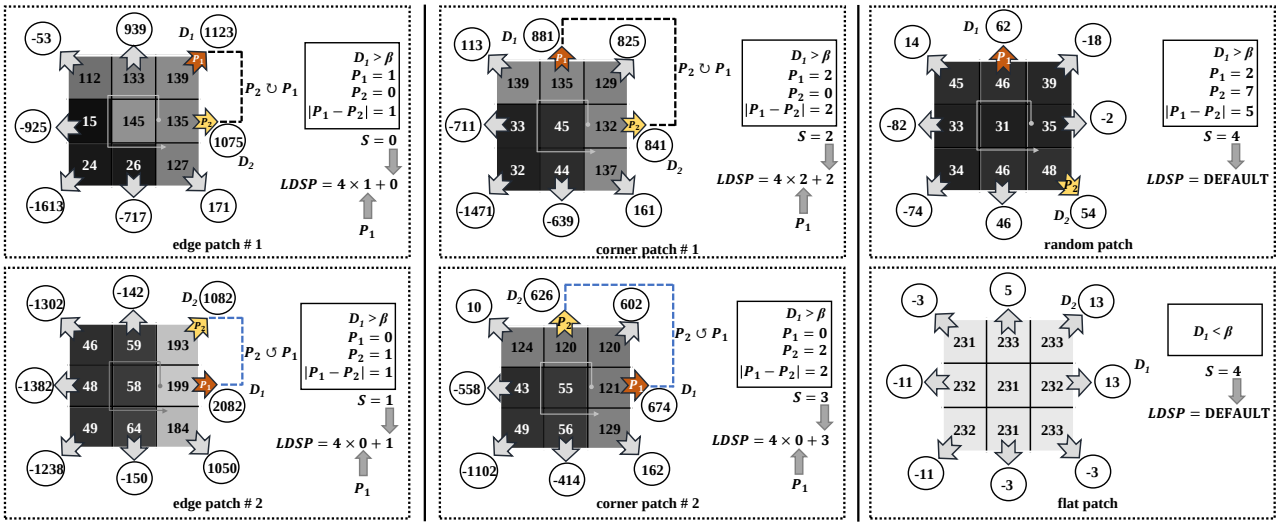


Figure 3. Generation of LDSP codes at different textures (edge, corner, flat, random texture) are shown. Sample image patches of these textures and their corresponding eight Kirsch responses (in the circle) are shown within the respective dotted rectangles. In the figure, P_1 and P_2 denote the position of the top two directions, respectively. D_1 and D_2 are the top response values, and S is the structural feature code. $P_2 \circlearrowright P_1$ (referred to by black dotted line) denotes that P_2 appears clockwise to P_1, whereas $P_2 \circlearrowleft P_1$ (referred to by blue dotted line) denotes that P_2 appears counter-clockwise to P_1.

Since for both the edge and corner patterns the absolute difference value ($|P_1 - P_2|$) is limited to either 1 or 2, we only consider the pixels having $|P_1 - P_2| \leq 2$. We regard other patterns having $|P_1 - P_2| > 2$ as random noisy patterns, representing futile information. We set $S = 4$ for these patterns and do not compute the LDSP code for these pixels afterwards. We now generate the final LDSP code by concatenating the binary form of the top edge direction, P_1, and the structural feature code, S. The top edge direction gives the signature of the primary edge direction of the pixel and the structural feature code provides explicit detail of the pixel's texture structure. We formally define the desired LDSP code for the target pixel as follows:

$$LDSP = \begin{cases} 4 \times P_1 + S, & \text{if, } D_1 > \beta \ \& \ S < 4 \\ \text{DEFAULT}, & \text{otherwise.} \end{cases} \tag{4}$$

It important to note that in order to compute the LDSP code for a pixel, we set two conditions. One is having a value of $S < 4$, which limits the coding only for the significant texture-primitives contributing to expression changes. Because we assign $S = 4$ for the random noisy patterns, this restriction subsequently avoids such random patterns from being encoded. The second one is having a significantly higher top-edge response, D_1, by denoting that D_1 should be higher than a threshold, β. This threshold filters the weak responses from flat pixels, which are expressionless in practice. β can be set in many ways; however, one of the adaptive ways is introduced in the following subsection. In this way, we avoid encoding the insignificant flat and random noisy patterns, and we assign a DEFAULT code to accumulate them in a single bin. In Eq. 4, 3 bits are required for the top direction, P_1, and 2 bits are for the structural feature code, S. Hence, the final LDSP code is 5 bits long and varies from 0 to 31 decimal code-values. We use DEFAULT = 32 in the coding. In Figure 3, we illustrate some examples of the code computation for different textures, including edge, corner, flat, and random noisy patterns.

2.2. Globally adaptive thresholding

An important factor of Eq. 4 is the selection of the threshold, β. We set β adaptively assuming that there are $Z\%$ of flat pixels in an image. Hence, we count the $Z\%$ of weak responses from the flat pixels and use this rate as a boundary to pick the threshold, β, for that image. To be specific, at first, we calculate the cumulative distribution of the primary response values from all pixels of the input image. Since the weak responses from the flat region belong to the initial part of the cumulative distribution, we now select the smallest bin having $Z\%$ of the total distribution and select it as our desired threshold. Formally, we define it as follows:

$$\beta = \arg\min_{x} \frac{\sum_{i=0}^{x} C(i)}{\sum_{j=0}^{N} C(j)} > Z\%, \tag{5}$$

where x is the expected bin that we use as the threshold, β, to filter the flat pixels. $C(.)$ is the cumulative histogram calculated from the top response values of each pixel, and N is the total histogram bin number. However, Eq. 5 needs to select the optimal $Z\%$ for the automatic recognition purpose. Selection of the optimal Z value is discussed in Section 3.2.

2.3. Feature-vector generation

After computing the LDSP code at each pixel of an image, a histogram of the code-bins of that image can be regarded as the feature vector. However, to gather more spatial information, typically the image is divided into different spatial regions, and histograms of all these regions are concatenated to generate the final feature vector [2, 9, 11, 14]. For this purpose, we divide the facial image into several nonoverlapping $x \times y$ sized uniform blocks, followed by generating a histogram for each block. The kth component of the histogram, H^Z, for block B^Z is computed by:

$$H^Z(k) = \sum_{c \in B^Z} f(c,k), \quad f(m,n) = \begin{cases} 1, & m = n \\ 0, & m \neq n. \end{cases} \tag{6}$$

We now concatenate all the H^Z histograms into one histogram, H, using

$$H = \Omega_{Z=1}^{b} H^Z, \tag{7}$$

where Ω is the concatenation operator and b is the total number of blocks. We regard H as the final feature vector in our approach, which we pass to the classifier for the classification task.

2.4. Analysis of LDSP code

In this section, we analyze some key advantages of the LDSP code against other existing descriptors under consideration. In particular, we discuss the comparative advantage of LDSP code in generating discriminative codes for different textures, and afterwards, we discuss its comparative efficacy in achieving robustness in uneven noisy environments.

We first show the efficacy of LDSP against other descriptors in generating distinctive codes for different textures. For this purpose, we show several sets of examples in Figure 4. In Figure 4a, we show that existing descriptors, such as LBP, LDP, LDN, and PTP, confuse an edge patch with a flat patch by generating identical code for them. Figure 4b also shows the same aspect when considering a corner and a flat patch. Generation of such identical code for two completely different textures creates ambiguity in the feature description and affects the classification result. However, the given descriptors only use the principal directions to encode the textures,

which may easily generate the same code for different textures, contributing to such ambiguity in the feature description. Nevertheless, in both cases, the proposed LDSP generates different codes for edges and corners, respectively, than that generated in the given flat patches. In particular, LDSP takes advantage of the positional relationship of the top edge directions to characterize the structures of edge and corner textures yielding their distinctive codes. Simultaneously, default code is generated for the flat patches with the proposed thresholding in order to differentiate them from other textured-patches. In addition, we provide two different random patches in Figure 4c, where for both the patches, LBP, LDP, LDN, and PTP generate respective codes. However, codes from such random patches may perturb other textured bins in the feature histogram and thereby affect the description of actual expressions. Nevertheless, LDSP analyzes the positional relationship between the top edge directions (according to Eq. (3)) and generates default code for the given patches to differentiate them from other textured patches. The above given examples, we believe, strongly show the comparative efficacy of LDSP against other existing descriptors in generating distinguishable codes for different textures.

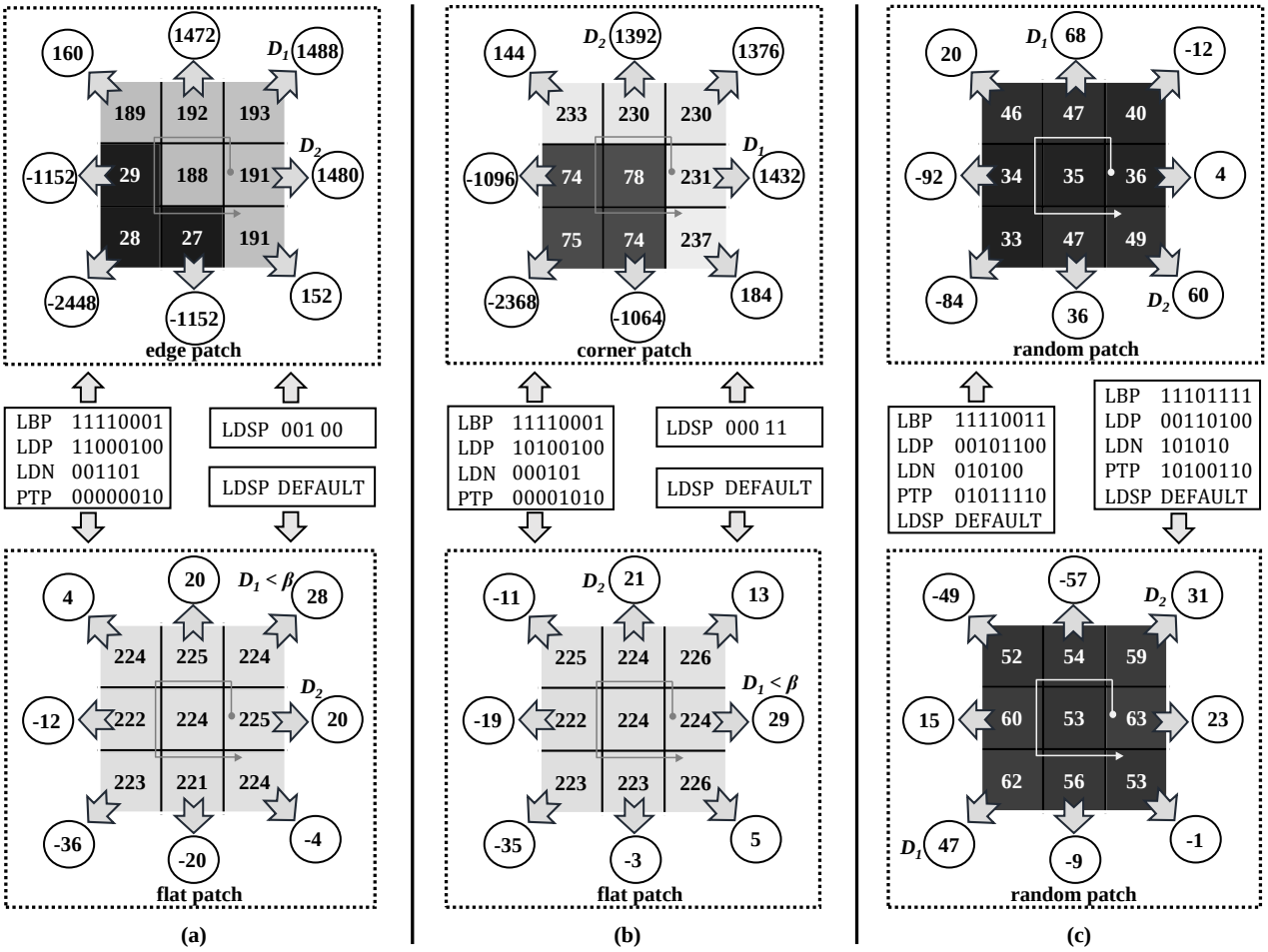


Figure 4. Comparative analysis of the code of different descriptors in different texture patches: (a) edge and flat texture patch; (b) corner and flat texture patch; (c) two different random patches.

We also analyze the stability of the proposed descriptor in the presence of noise. For this purpose, for an input image, we add zero-mean Gaussian noise with different noise variances (0.0001 to 0.0010) to generate a set of noisy images. We now compute the LDSP feature vector for these images and calculate chi-square

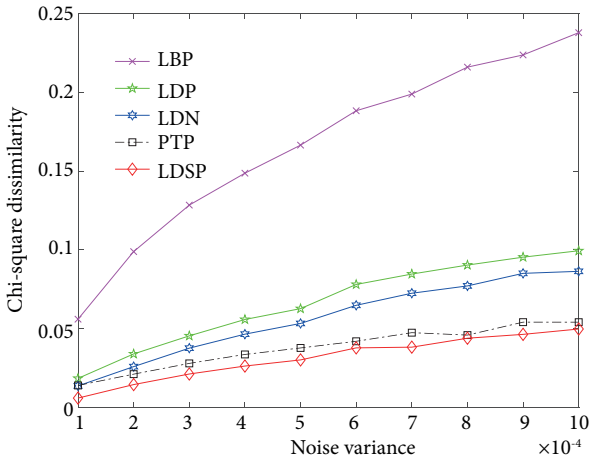


Figure 5. Average scaled chi-square dissimilarities between the feature vector of input code image and corresponding noisy code images at different noise variances.

dissimilarity [22] between the code feature vector of the input image and other noisy images. The process is repeated for 1000 randomly collected images from the working datasets and we report the average dissimilarity values at each noise level. We also produce such results for other descriptors in order to compare them with the proposed one. We present the average dissimilarity values at different noise levels for all these descriptors in Figure 5. We observe that at each of the noise levels, the dissimilarities for LDSP are comparatively lower than other descriptors, exhibiting its stability under noise. The main reason for such stable results is that LDSP explicitly discards the featureless futile patterns such as random noisy patterns and flat textures. Such textures may change their structure randomly under noise and thereby may create ambiguity in the feature description. Therefore, unlike other descriptors, the uncertain effects of such textures are less in the LDSP descriptor, contributing to its consistency under noise.

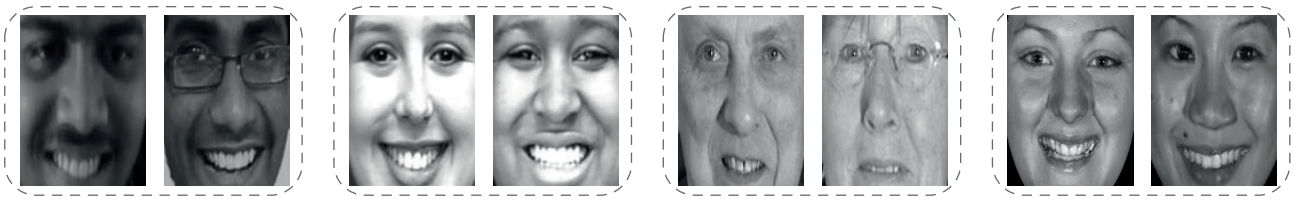
3. Experimental results and analysis

In this section, we analyze the performance of the proposed descriptor for person-independent facial expression recognition in existing popular datasets. First we describe our experimental settings in detail, and afterwards, we describe the comparative performance of the proposed method against other different methods with the existing datasets. In the experiments, we conduct the comparisons from different perspectives, such as recognition result, computation time, and performance under low resolution and noise.

3.1. Experimental settings

To ensure person-independence in the recognition of facial expressions, we adopt a leave-one-subject-out (LOSO) cross-validation strategy. This strategy excludes the expression images of each subject (person) from the training set, and then tests them. The process is repeated for every subject in the dataset and the average result is reported. In this procedure, the testing phase does not have any prior information of a person trained before and hence ensures person-independence.

We initiate the testing procedure by cropping face images and resizing them to 110 $\times$ 150 resolution, as done in other existing works [2, 9, 11]. We crop the face image based on the positions of eyes and mouth provided by ground truth or manual selections. We now divide the face image into $x \times y$ regions to generate codes from



(a) (b) (c) (d)

Figure 6. Example of facial expression images from (a) ISED, (b) CK+, (c) MMI, and (d) BU-3DFE datasets.

each region, as explained in the previous section. For the classification, we use a support vector machine (SVM) classifier with radial basis function (RBF) kernel [23]. Because SVM mainly conducts binary classification, in order to comply with the multiclass classification, we adopt the one-against-one method, as referred to by Hsu et al. [23]. However, to select the optimal parameters, we conduct a grid search on the hyperparameters within a cross-validation approach and pick the parameter values giving the best cross-validation results.

We carried out person-independent experiments with several benchmark datasets for both the spontaneous and posed (acted) expressions. For the spontaneous expressions, we use the ISED [24] dataset, whereas CK+ [25], MMI [26], and BU-3DFE [27] are used for posed expressions. Several facial expression images of these datasets are demonstrated in Figures 6a–6d. We also test the performance of the proposed descriptor in the presence of low resolution and a noisy environment to show its robustness in such conditions.

3.2. Optimal parameter selection

There are a couple of parameters of the proposed method, such as threshold, β, and block-size, $x \times y$, which need to be set. For parameter selection purposes, we follow the procedure described in [11]. That is, we generate a synthetic validation dataset by randomly selecting a number of images (we select 500 images) from the working datasets and test the performance of the proposed method for different parameter values in this image set. The values giving the best recognition rate are selected as the optimal parameters.

In LDSP, we select the threshold, β (as in Eq. (5)), adaptively based on the percentage ($Z\%$) of flat pixels in an image. Hence, we have to set an optimal value of Z in order to select the best β value. We test the performance of LDSP in the above-mentioned synthetic image set for different Z values varying from 0% to 25% within different block sizes. The results are shown in Figure 7. We observe that the highest result is achieved considering $Z = 15\%$ flat pixels and a block size of 8×9. It is worth noting that considering a higher Z-percentage of flat pixels may increase the threshold value, β, which, in turn, eliminates a considerable number of pixels from the image. Therefore, in the case of dealing with a smaller-sized facial block, a high threshold may reduce the number of samples from that block, causing sampling error while generating the respective block histogram [28]. Nevertheless, considering larger-sized blocks may suffer from limited spatial information in this regard. Figure 7 demonstrates that considering smaller-sized blocks (e.g., 12×13, 13×14, 14×15) with a comparably higher threshold (e.g., $Z = 20\%$ and $Z = 25\%$ flat pixels) causes gradual degradation in performance, whereas considering larger-sized blocks (e.g., 4×5, 5×6) suffers from less accuracy due to the limited spatial information. Therefore, we have to choose the values of threshold (β) and block size ($x \times y$) very carefully so that we have adequate samples after thresholding and simultaneously preserve sufficient spatial information in the image blocks. However, the recognition accuracies of different parameter values and the above analysis suggest that considering $Z = 15\%$ flat pixels and the block size of 8×9 ensures a good trade-off between

preserving adequate samples and spatial information, and thereby we consider these values as the optimal values for the above two parameters.

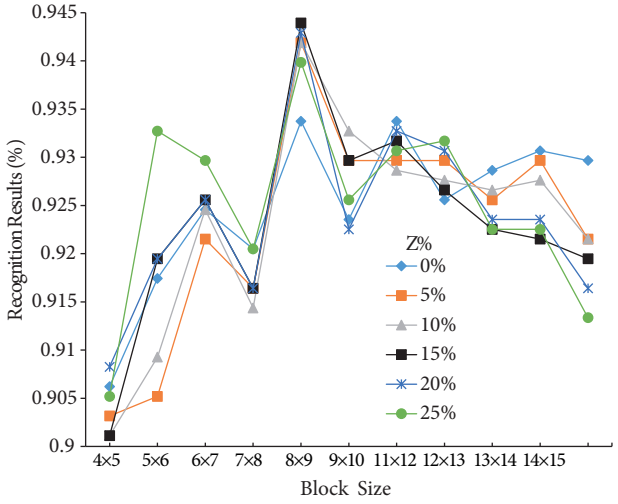


Figure 7. Recognition results of LDSP at different $(x \times y)$-sized uniform blocks and different Z values, in a collection of 500 images. Images are collected randomly from working datasets.

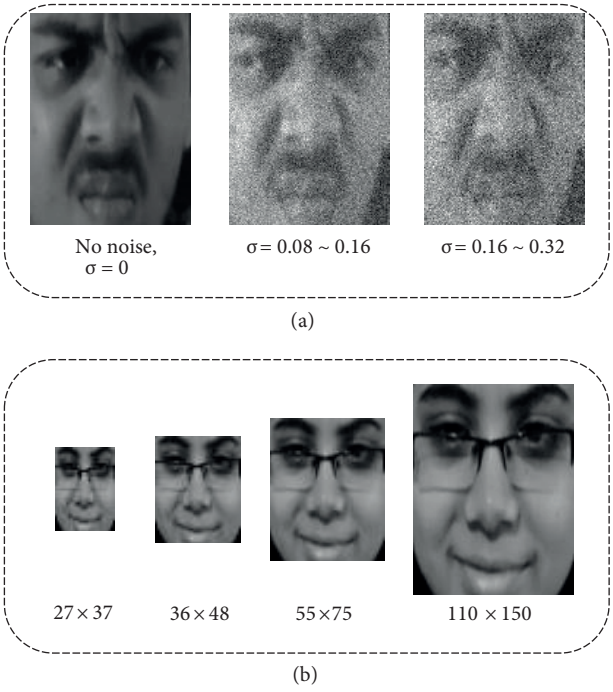


Figure 8. Sample ISED images: (a) at different noise variations, σ, and (b) at four different resolutions.

3.3. Recognition results

3.3.1. ISED results

The Indian Spontaneous Expression Database (ISED) [24] provides near-frontal spontaneous expressive images with the emotion level for the peak expression frames of all videos. The given peak expression faces of 50 subjects from all the video clips are used in the baseline experiment. There are four expression classes in ISED, including happiness, surprise, sadness, and disgust. Among the selected peak images, 227, 73, 48, and 80 images belong to the happiness, surprise, sadness, and disgust, respectively.

We test LDSP in the ISED dataset for person-independent facial expression recognition against different descriptions, under our considerations, such as LBP, LDP, LDN, LPTP, and PTP. We use the peak images for this experiment, as done in the baseline method [24], and generate the results for all these descriptors. Representative results are shown in Table 1 (see the results in the "Without Noise (110×150)" column), where we observe better accuracy of the proposed LDSP than other descriptors. We note that the given person-independent results in this table are different from the results presented in the base paper [24], which were produced without ensuring the person-independence.

In Section 2.4, we have shown the stability of LDSP against other descriptors under noise in terms of chi-square dissimilarity of the feature vector at different noise levels. However, in this section, we explicitly show the recognition accuracies of LDSP against other descriptors in such noisy environments. To test the

recognition accuracy under noise, we artificially add random Gaussian noise with zero mean and standard deviation, σ, within two random intervals (0.08–0.16 and 0.16–0.32) for each image of the ISED dataset. In our approach, noises within the given intervals are randomly distributed in the dataset images in order to comply with natural uneven situations. Examples of such noise-perturbed images at different σ values are given in Figure 8a. Consequently, we conduct person-independent recognition for LDSP with other descriptors including LDP, LDN, LPTP, and PTP. We present the results in Table 1 (see the results with "Under Noise"), where it is evident that the proposed descriptor achieves better performance than other descriptors in the presence of noise. As described before, the exclusion of the uncertain patterns, like flat and random noisy textures, contributes the most to such consistent performance of LDSP.

Furthermore, we test the performance of LDSP for low-resolution ISED images. Achieving better performance at low resolution is important since most surveillance systems and real-time video analysis systems deal with low-resolution video input. To test the performance, we generate four different sets of ISED dataset images after varying the image resolutions. In particular, we divide the images into 110×150, 55×75, 36×48, and 27×37 resolutions, respectively, to generate four different ISED image sets. Figure 8b provides example of an image at four different resolutions. However, similar to the previous testing, we test LDSP against other descriptors including LDP, LDN, LPTP, and PTP. Results presented in Table 1 show that the proposed LDSP achieves higher accuracy than other descriptors, showing its efficiency at different image resolutions.

3.3.2. CK+ results

The extended Cohn–Kanade (CK+) [25] dataset contains 593 image sequences from 123 subjects having seven expressions: anger, contempt, disgust, fear, happiness, sadness, and surprise. In each image sequence, the person's expression starts from the neutral frame (onset) to the peak expressive frame (apex). In our settings, we select 327 image sequences out of 593, where every sequence is labeled with one of the above-mentioned expressions. We performed 7 class expression recognitions on CK+. Following the studies in [9, 11, 15], the three most expressive frames of each sequence are considered, resulting in 981 facial expression images. In addition, we added the first neutral frame of each video with the previous 7-class images to perform the 8-class experiment in CK+.

We test the performance of the proposed descriptor on the CK+ dataset for 7 classes, where we compare the proposed LDSP descriptor with different local descriptors and other state-of-the-art methods, under our considerations. We present the comparative results in Table 2, where we observe that LDSP achieves higher accuracy than other descriptors, such as, LBP, LDP, LDN, LPTP, and PTP, for 7-class recognition. Lee et al. [29] presented some of the descriptors' results, including LBP, LPQ, and Gabor with sparse representation classifier

Table 1. Person-independent expression recognition results (%) for 428 images of the ISED dataset with varying noise and image resolutions.

Descriptors	Without noise	With noise		Resolution		
	(110 × 150)	$\sigma = 0.08 \sim 0.16$	$\sigma = 0.16 \sim 0.32$	55 × 75	36 × 48	27 × 37
LBP	76.47	56.66	49.53	74.98	72.67	72.40
LDP	74.61	57.28	52.94	72.75	70.28	68.11
LDN	75.85	61.61	55.10	74.92	68.11	67.80
LPTP	72.46	61.92	57.27	70.59	69.97	69.04
PTP	76.16	62.22	58.51	73.07	72.45	70.50
LDSP	**77.78**	**65.01**	**60.00**	**75.10**	**72.96**	**72.44**

Table 2. Person-Independent expression recognition results (%) for 981 images for 7-class and 1308 images for 8-class CK+ dataset. Results with citations are from corresponding papers.

Methods	7-class
LBP	85.84
LDP	88.07
LDN	88.58
LPTP	91.64
PTP	91.03
SPTS [25]	50.4
CAPP [25]	66.7
SPTS + CAPP [25]	83.3
LBP + SRC [29]	79.97
LPQ + SRC [29]	80.78
Gabor + SRC [29]	82.82
Lee et. al [29]	92.34
SRC + ICV [30]	90.5
MSR [31]	91.4
CLGDNP [32]	94.27
LDSP	**94.49**

Methods	8-class
LBP	89.38
LDP	85.04
LDN	85.12
LPTP	86.20
PTP	88.99
HOG [16]	89.53
Gabor [16]	88.61
SIFT [16]	86.39
Zero-biased CNN [17]	81.88
FN2EN [18]	88.70
LDSP	**90.75**

(SRC), which were also found inferior to the proposed descriptor. Nevertheless, we compare our results against other state-of-the-art methods [25, 29–32]. Among these methods, Lucey et al. [25] reported several sparse methods using geometric information of the facial components, namely similarity-normalized shape (SPTS), canonical appearance features (CAPP), and their combination (SPTS + CAPP). On the contrary, to represent the expressions of facial images, Lee et al. [29, 30] utilized the intraclass variation information, while Raymod et al. [31] utilized manifold-based sparse representation (MSR). Inspired by the Gabor feature and LDN, Zhang et al. [32] used Gabor magnitude and phase information by means of LDN. Despite these methods using comparatively more complex facial representations than our method, the proposed LDSP shows better accuracy than all these methods, providing a strong indication of its efficacy in recognizing facial expressions.

We also perform 8-class experiments on CK+, where we compare LDSP with the previously mentioned descriptors, including LBP, LDP, LDN, LPTP, and PTP, along with some other appearance-based methods, e.g., HOG [16], Gabor [16], and SIFT [16]. Moreover, we compare our method against a couple of recent deep learning methods, such as zero-biased CNN [17] and FN2EN [18]. We note that while comparing the results of [17, 18], we only consider the results generated without any augmented data since comparing the results using augmented data is different from our experimental settings, leading to unfair comparison. However, we visualize better performance of our method against all the above methods, showing the overall efficacy of LDSP in recognizing facial expressions for different classes.

3.3.3. MMI results

The MMI face dataset [26] contains more than 1500 samples of both image sequences and static images in frontal and profile views depicting various expressions. Part II of this dataset is used in our experiments, which consists of frontal images taken from 28 subjects. These image sequences are labeled with one of the six basic expression classes. In some image sequences, the subject's expression is taken with and without glasses.

Table 3. Person-independent expression recognition results (%) for 504 images of the MMI dataset.

Methods	6-class results
LBP	69.05
LDP	63.89
LDN	65.87
LPTP	62.10
PTP	67.05
LBP + SRC [29]	59.18
LPQ + SRC [29]	62.72
Gabor + SRC [29]	61.89
Best from [29]	**70.12**
LDSP	69.05

Table 4. Person-independent expression recognition results (%) for 2400 images of BU-3DFE dataset.

Methods	6-class results
LBP	56.2
LDP	61.3
LDN	56.5
LPTP	67.8
PTP	66.88
Hu et al. [33]	66.5
BDA/GMM [34]	68.28
Tariq et al. [35]	68.3
LDSP	**68.75**

Table 5. Comparison of feature-vector generation time.

Descriptors	Feature dimension	Computation time
LBP	59	3.5 ms
LDP	59	4.8 ms
LDN	64	4.6 ms
LPTP	128	4.5 ms
PTP	256	4.4 ms
LDSP	32	4.3 ms

We perform a 6-class person-independent experiment on the MMI dataset and present the results in Table 3. It is interesting to observe that the proposed descriptor achieves better accuracy than other edge-based descriptors including LDP, LDN, LPTP, and PTP and ties with LBP. Nevertheless, LDSP works better than LBP, LPQ, and Gabor when they are applied with SRC [29]. Moreover, the best result reported in [29] is slightly better than the proposed method. We analyze the reason for such discrepancies in the results of the proposed method in MMI dataset and observe that results of LDSP for the expressions of older subjects are less accurate owing to less distinction among different expressions of the older subjects. Moreover, many of the subjects wear eyeglasses, which are often characterized as edges in our approach. In practice, such information from eyeglasses is irrelevant in characterizing the expressions; hence, including this futile information may create ambiguity in the feature description. These, perhaps, are some of the reasons for the lower accuracy of LDSP in MMI. However, improving the performance of the proposed method in such situations can be an interesting research problem, which we leave here as our future endeavor.

3.3.4. BU-3DFE results

The BU-3DFE dataset [27] provides 2400 face images with 6 basic expressions taken from 100 subjects. Since face images of this database vary in ethnicity, ancestries, and intensity of expressions, BU-3DFE is considered challenging in recognizing expressions.

The recognition results presented in Table 4 show that the proposed method achieves much higher accuracy than other descriptors, including LBP, LDP, LDN, LPTP, and PTP. The reason behind the lower accuracy of other descriptors is that each BU-3DFE image has a black background that does not possess any relevant expression-affiliated information. Because the other descriptors generate codes for each pixel, they include meaningless codes from the black background in the feature vector, which may create uncertainty in the classification stage. However, we exclude these meaningless flat texture codes by applying a thresholding mechanism, as described in Section 2.2. This thresholding strategy avoids the effect of such featureless patterns in classification and contributes to the better performance of the proposed method. We also observe that the proposed method achieves better accuracy than some other existing works [33–35], advocating for its efficacy in recognizing the expressions of the BU-3DFE dataset.

3.4. Computation time

We show the feature-vector computation time of proposed LDSP descriptor. We conduct this experiment with Visual Studio 2015 on an Intel core *i*5 @2.67 GHz with 8 GB RAM. We select 100 random images from the working datasets, and for each image, we record the time needed to extract its code feature vector. Finally, we calculate the average time in milliseconds and present it in Table 5 for different descriptors. It is evident that LBP is computationally the most efficient method among others owing to its computational simplicity. Edge descriptors including LDP, LDN, LPTP, and PTP require comparatively more time than LBP due to the convolution process of compass masks. However, the proposed LDSP requires less computation time than the above-mentioned edge descriptors because of having smaller code length. Due to the exclusion of different featureless patterns, LDSP generates 5-bit long code resulting in 32 codes in total, which is quite lower than others, contributing to its computational efficiency.

4. Conclusion and future work

In this paper, we propose a new descriptor, LDSP, for the person-independent facial expression recognition task. The proposed LDSP descriptor generates 5-bit code for a pixel by utilizing the positional relationship among its top edge responses. LDSP exploits such information to extract further structural detail of the crucial texture-primitives related to expression changes. Moreover, with a globally adaptive thresholding scheme, LDSP discards the featureless flat textures from the feature description. In this way, we explicitly characterize the expression-related most crucial textures while discarding the flat and random noisy patterns. Moreover, due to discarding the flat and random patterns, LDSP facilitates with moderate computation time. With such a coding scheme, the proposed descriptor offers compact and efficient feature representations that are fast to compute and memory-efficient, yet exhibit good discriminability and robustness. We conduct person-independent facial expression recognition with LDSP in several popular datasets where LDSP is found to work better than other existing descriptors. We also observe better performance of the proposed method in the presence of noise and low resolution, which advocates for the overall efficacy of the proposed method in recognizing facial expressions. In this work, we dealt with the static images having posed expressions. In this regard, extending the current work for the recognition of spontaneous expressions from video sequences can be a potential research direction. The effect of deep learning-based methods in expression recognition is another issue, which we seldom discuss in this paper. Nevertheless, the recent success of deep methods in computer vision may drive one to embed the proposed descriptor within deep models in order to achieve better expression recognition accuracies.

Acknowledgment

This research was supported by the MSIT (Ministry of Science and ICT), Korea, under the Grand Information Technology Research Center support program (IITP-2018-2015-0-00742) supervised by the IITP (Institute for Information & Communications Technology Promotion), and a National Research Foundation of Korea (NRF) grant funded by the Korean government (MSIT) (No. NRF-2015R1A2A2A01006412).

References

[1] Tian YL, Kanade T, Cohn JF. Facial expression analysis. In: Li SZ, Jain AK, editors. Handbook of Face Recognition. New York, NY, USA: Springer, 2005. pp. 247-275.

[2] Shan CF, Gong SG, McOwan PW. Facial expression recognition based on local binary patterns: a comprehensive study. Image Vision Comput 2009; 27: 803-816.

[3] Bourbakis N, Esposito A, Kavraki D. Extracting and associating meta-features for understanding people's emotional behaviour: face and speech. Cogn Comput 2011; 3: 436–448.

[4] Kotsia I, Pitas I. Facial expression recognition in image sequences using geometric deformation features and support vector machines. IEEE T Image Process 2007; 16: 172-187.

[5] Pantic M, Bartlett MS. Machine analysis of facial expressions. In: Delac K, Grgic M, editors. Face Recognition. Rijeka, Croatia: Intech Open, 2007. pp. 377-416.

[6] Turk M, Pentland A. Eigenfaces for recognition. J Cognitive Neurosci 1991; 3: 71-86.

[7] Belhumeur PN, Hespanha JP, Kriegman DJ, Eigenfaces vs. Fisherfaces: recognition using class specific linear projection. IEEE T Pattern Anal 1997; 19: 711-720.

[8] Yang J, Zhang D, Frangi AF, Yang JY. Two-dimensional PCA: a new approach to appearance-based face representation and recognition. IEEE T Pattern Anal 2004; 26: 131-137.

[9] Rivera AR, Castillo JR, Chae O. Local directional number pattern for face analysis: face and expression recognition. IEEE T Image Process 2013; 22: 1740-1752

[10] Heisele B, Serre T, Poggio T. A component-based framework for face detection and identification. Int J Comput Vision 2007; 74: 167-181.

[11] Jabid T, Kabir MH, Chae O. Robust facial expression recognition based on local directional pattern. ETRI J 2010; 32: 784-794.

[12] Kirsch RA. Computer determination of the constituent structure of biological images. Comput Biomed Res 1971; 4: 315–328.

[13] Rivera AR, Castillo JR, Chae O. Recognition of face expressions using local principal texture pattern. In: IEEE 2012 International Conference on Image Processing; 30 September–3 October 2012; Orlando, FL, USA. New York, NY, USA: IEEE. pp. 2613–2616.

[14] Iqbal MT, Ryu B, Song G, Chae O. Positional ternary pattern (PTP): an edge based image descriptor for human age recognition. In: IEEE 2016 International Conference on Consumer Electronics; 7–11 January 2016; Las Vegas, NV, USA. New York, NY, USA: IEEE. pp. 289-292.

[15] Iqbal MT, Ryu B, Song G, Kim J, Makhmudkhujaev F, Chae O. Exploring positional ternary pattern (PTP) for conventional facial expression recognition from static images. In: 2016 Conference Proceedings; 29 June–1 July 2016; Jeju, Republic of Korea. pp. 853-855.

[16] Liu M, Li S, Shan S, Chen X. AU-aware deep networks for facial expression recognition. In: IEEE 2013 International Conference and Workshops on Automatic Face and Gesture Recognition; 22–26 April 2013; Shanghai, China. New York, NY, USA: IEEE. pp. 1-6.

[17] Khorrami P, Paine T, Huang T. Do deep neural networks learn facial action units when doing expression recognition? In: IEEE 2015 International Conference on Computer Vision Workshops; 7–13 December 2015; Santiago, Chile. New York, NY, USA: IEEE. pp. 19-27.

[18] Ding H, Zhou SK, Chellappa R. Facenet2expnet: regularizing a deep face recognition net for expression recognition. In: IEEE 2017 International Conference on Automatic Face and Gesture Recognition; 30 May–3 June 2017; Washington, DC, USA. New York, NY, USA: IEEE. pp. 118-126.

[19] Lopes AT, de Aguiar E, De Souza AF, Oliveira-Santos T. Facial expression recognition with convolutional neural networks: coping with few data and the training sample order. Pattern Recogn 2017; 61: 610-28.

[20] Mollahosseini A, Chan D, Mahoor MH. Going deeper in facial expression recognition using deep neural networks. In: IEEE 2016 Winter Conference on Applications of Computer Vision; 7–10 March 2016; Lake Placid, NY, USA. New York, NY, USA: IEEE. pp. 1-10.

[21] Iqbal MT, Shoyaib M, Ryu B, Abdullah-Al-Wadud M, Chae O. Directional age-primitive pattern (DAPP) for human age group recognition and age estimation. IEEE T Inf Foren Sec 2017; 12: 2505-2517.

[22] Chardy P, Glemarec M, Laurec A. Application of inertia methods to benthic marine ecology: practical implications of the basic options. Estuar Coast Mar Sci 1976; 4: 179-205.

[23] Hsu CW, Lin CJ. A comparison of methods for multiclass support vector machines. IEEE T Neural Networ 2002; 13: 415-25.

[24] Happy SL, Patnaik P, Routray A, Guha R. The Indian spontaneous expression database for emotion recognition. IEEE T Affect Comput 2017; 8: 131-142.

[25] Lucey P, Cohn JF, Kanade T, Saragih J, Ambadar Z, Matthews I. The extended Cohn-Kanade dataset (CK+): A complete dataset for action unit and emotion-specified expression. In: IEEE 2010 Computer Society Conference on Computer Vision and Pattern Recognition Workshops; 13–18 June 2010; San Francisco, CA, USA. New York, NY, USA: IEEE. pp. 94-101.

[26] Valstar M, Pantic M. Induced disgust, happiness and surprise: an addition to the MMI facial expression database. In: Third International Workshop on EMOTION (Satellite of LREC): Corpora for Research on Emotion and Affect; 19–21 May 2010; Valletta, Malta. p. 65-70.

[27] Yin L, Wei X, Sun Y, Wang J, Rosato MJ. A 3D facial expression database for facial behavior research. In: IEEE 2006 International Conference on Automatic Face and Gesture Recognition; 10–12 April 2006; Southampton, UK. New York, NY, USA: IEEE. pp. 211-216.

[28] Ylioinas J, Hadid A, Guo Y, Pietikäinen M. Efficient image appearance description using dense sampling based local binary patterns. In: Asian Conference on Computer Vision; 5–9 November 2012; Daejeon, Korea. Berlin, Germany: Springer. pp. 375-388.

[29] Lee SH, Baddar WJ, Ro YM. Collaborative expression representation using peak expression and intra class variation face images for practical subject-independent emotion recognition in videos. Pattern Recogn 2016; 54: 52-67.

[30] Lee SH, Plataniotis KN, Ro YM. Intra-class variation reduction using training expression images for sparse representation based facial expression recognition. IEEE T Affect Comput 2014; 5: 340-351.

[31] Ptucha R, Tsagkatakis G, Savakis A. Manifold based sparse representation for robust expression recognition without neutral subtraction. In: IEEE 2011 International Conference on Computer Vision Workshops (ICCV Workshops); 6–13 November 2011; Barcelona, Spain. New York, NY, USA: IEEE. pp. 2136-2143.

[32] Zhang Z, Lu G, Yan J, Li H, Sun N, Li X. Compact local Gabor directional number pattern for facial expression recognition. Turk J Elec Eng & Comp Sci 2018; 26: 1236-1248.

[33] Hu Y, Zeng Z, Yin L, Wei X, Tu J, Huang TS. A study of non-frontal-view facial expressions recognition. In: 19th International Conference on Pattern Recognition; 8–11 December 2008; Tampa, FL, USA. New York, NY, USA: IEEE. pp. 1-4.

[34] Zheng W, Tang H, Lin Z, Huang TS. Emotion recognition from arbitrary view facial images. In: European Conference on Computer Vision; 5–11 September 2010; Heraklion, Greece. Berlin, Germany: Springer-Verlag. pp. 490-503.

[35] Tariq U, Huang TS. Features and fusion for expression recognition — A comparative analysis. In: IEEE 2012 Computer Society Conference on Computer Vision and Pattern Recognition Workshops; 16–21 June 2012; Providence, RI, USA. New York, NY, USA: IEEE. pp. 146-152.