

# Local Directional Number Pattern for Face Analysis: Face and Expression Recognition

Adin Ramirez Rivera, *Student Member, IEEE*, Jorge Rojas Castillo, *Student Member, IEEE*,  
and Oksam Chae\*, *Member, IEEE*

**Abstract**—This paper proposes a novel local feature descriptor, Local Directional Number Pattern (LDN), for face analysis: face and expression recognition. LDN encodes the directional information of the face’s textures (*i.e.*, the texture’s structure) in a compact way, producing a more discriminative code than current methods. We compute the structure of each micro-pattern with the aid of a compass mask, that extracts directional information, and we encode such information using the prominent direction indexes (directional numbers) and sign—which allows us to distinguish among similar structural patterns that have different intensity transitions. We divide the face into several regions, and extract the distribution of the LDN features from them. Then, we concatenate these features into a feature vector, and we use it as a face descriptor. We perform several experiments in which our descriptor performs consistently under illumination, noise, expression, and time lapse variations. Moreover, we test our descriptor with different masks to analyze its performance in different face analysis tasks.

**Index Terms**—Local pattern, directional number pattern, image descriptor, face descriptor, feature, face recognition, expression recognition

## I. INTRODUCTION

In face analysis, a key issue is the descriptor of the face appearance [1], [2]. The efficiency of the descriptor depends on its representation and the ease of extracting it from the face. Ideally, a good descriptor should have a high variance among classes (between different persons or expressions), but little or no variation within classes (same person or expression in different conditions). These descriptors are used in several areas, such as, facial expression and face recognition.

There are two common approaches to extract facial features: geometric-feature-based and appearance-based methods [3]. The former [4], [5] encodes the shape and locations of different facial components, which are combined into a feature vector that represents the face. An instance of these methods are the graph-based methods [6]–[10], which use several facial components to create a representation of the face and process it. Moreover, the Local-Global Graph algorithm [6]–[8] is an interesting approach that uses Voronoi tessellation and Delaunay graphs to segment local features and builds a graph

for face and expression recognition. These features are mixed into a local graph, and then the algorithm creates an skeleton (global graph) by interrelating the local graphs to represent the topology of the face. Furthermore, facial features are widely used in expression recognition, as the pioneer work of Ekman and Friesen [11] identifying six basic emotions produced a system to categorize the expressions, known as Facial Action Coding System [12], and later it was simplified to the Emotional Facial Action Coding System [13]. However, the geometric-feature-based methods usually require accurate and reliable facial feature detection and tracking, which is difficult to accommodate in many situations. The appearance-based methods [14], [15] use image filters, either on the whole-face, to create holistic features, or some specific face-region, to create local features, to extract the appearance changes in the face image. The performance of the appearance-based methods is excellent in constrained environment but their performance degrade in environmental variation [16].

In the literature, there are many methods for the holistic class, such as, Eigenfaces [17] and Fisherfaces [18], which are built on Principal Component Analysis (PCA) [17]; the more recent 2D PCA [19], and Linear Discriminant Analysis [20] are also examples of holistic methods. Although these methods have been studied widely, local descriptors have gained attention because of their robustness to illumination and pose variations. Heisele *et al.* showed the validity of the component-based methods, and how they outperform holistic methods [21]. The local-feature methods compute the descriptor from parts of the face, and then gather the information into one descriptor. Among these methods are Local Features Analysis [22], Gabor features [23], Elastic Bunch Graph Matching [24], and Local Binary Pattern (LBP) [14], [25]. The last one is an extension of the LBP feature, that was originally designed for texture description [26], applied to face recognition. LBP achieved better performance than previous methods, thus it gained popularity, and was studied extensively. Newer methods tried to overcome the shortcomings of LBP, like Local Ternary Pattern (LTP) [27], and Local Directional Pattern (LD<sub>i</sub>P) [28]–[30]. The last method encodes the directional information in the neighborhood, instead of the intensity. Also, Zhang *et al.* [31], [32] explored the use of higher order local derivatives (LD<sub>e</sub>P) to produce better results than LBP. Both methods use other information, instead of intensity, to overcome noise and illumination variation problems. However, these methods still suffer in non-monotonic illumination variation, random noise, and changes in pose, age, and expression conditions. Although some methods, like Gradientfaces [33],

Copyright (c) 2012 IEEE. Personal use of this material is permitted. However, permission to use this material for any other purposes must be obtained from the IEEE by sending a request to pubs-permissions@ieee.org.

This work was supported in part by the National Research Foundation of Korea, Grant funded by the Korean Government, under Grant 2012-0005523.

The authors are with the Department of Computer Engineering, Kyung Hee University, South Korea, 1 Seocheon-dong, Giheung-gu, Yongin-si, Gyeonggi-do 446-701, e-mails {adin, jarc, oschae}@khu.ac.kr.

\* Corresponding author

Manuscript accepted December 2012.

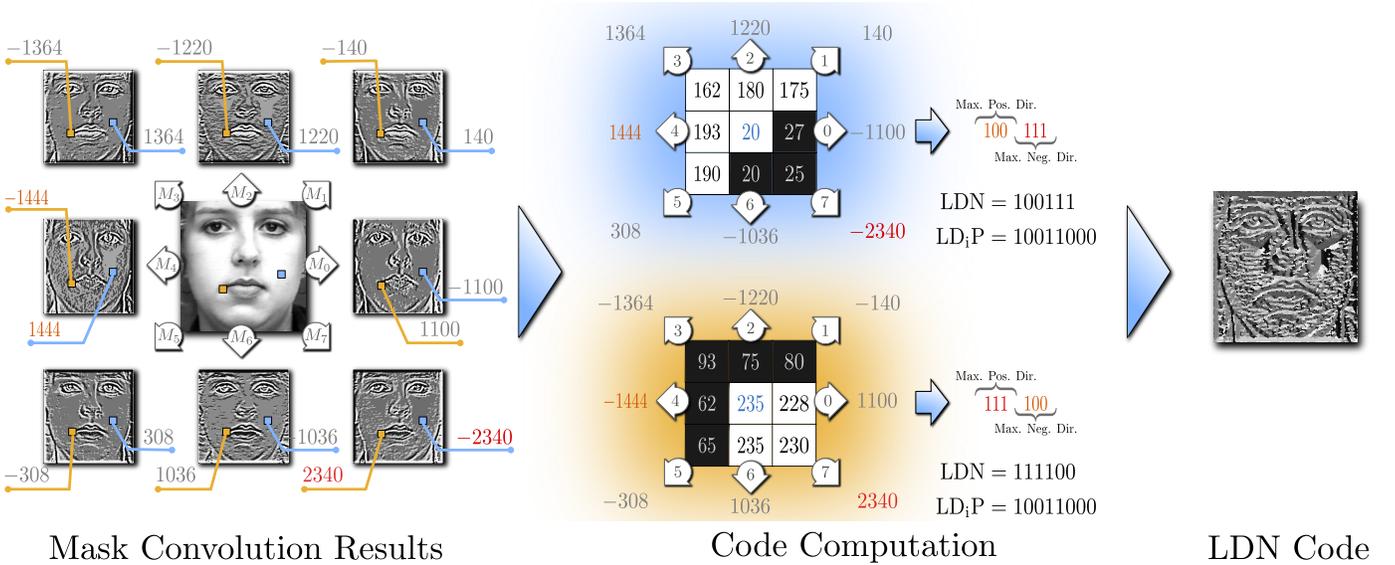


Fig. 1. LDN code computation. The (Kirsch) compass masks are convoluted with the original image to extract the edge response images (shown in the left). From these images, we choose the prominent directional numbers (positive and negative directions) to encode the texture in the neighborhood. We show an example of a neighborhood in the middle-top, that corresponds to the colored marks on the edge response images. It shows the different response values, the top directional numbers (in red and orange), and the final LDN code (shown in the right). Moreover, LDN can detect changes in the intensity regions by producing a different code (as shown in the middle-bottom) while other directional patterns cannot (like LD<sub>i</sub>P), as they produce the same code for different textures.

have a high discrimination power under illumination variation, they still have low recognition capabilities for expression and age variation conditions. However, some methods explored different features, such as, infrared [34], near infrared [32], and phase information [35], [36], to overcome the illumination problem while maintaining the performance under difficult conditions.

In this paper, we propose a face descriptor, Local Directional Number Pattern (LDN), for robust face recognition that encodes the structural information and the intensity variations of the face’s texture. LDN encodes the structure of a local neighborhood by analyzing its directional information. Consequently, we compute the edge responses in the neighborhood, in eight different directions with a compass mask. Then, from all the directions, we choose the top positive and negative directions to produce a meaningful descriptor for different textures with similar structural patterns. This approach allows us to distinguish intensity changes (*e.g.*, from bright to dark and vice versa) in the texture, that otherwise will be missed—see fig. 1. Furthermore, our descriptor uses the information of the entire neighborhood, instead of using sparse points for its computation like LBP. Hence, our approach conveys more information into the code, yet it is more compact—as it is six bit long. Moreover, we experiment with different masks and resolutions of the mask to acquire characteristics that may be neglected by just one, and combine them to extend the encoded information. We found that the inclusion of multiple encoding levels produces an improvement in the detection process.

This paper is structured as follows: In Section II we introduce our proposed coding scheme. Then, in Section III we detail the use of the proposed descriptor for face and expression recognition. We evaluate the performance of the proposed descriptor and discuss its results in Section IV. Finally, we present concluding remarks in Section V.

## II. LOCAL DIRECTIONAL NUMBER PATTERN

The proposed Local Directional Number Pattern (LDN) is a six bit binary code assigned to each pixel of an input image that represents the structure of the texture and its intensity transitions. As previous research [37], [38] showed, edge magnitudes are largely insensitive to lighting changes. Consequently, we create our pattern by computing the edge response of the neighborhood using a compass mask, and by taking the top directional numbers, that is, the most positive and negative directions of those edge responses. We illustrate this coding scheme in fig. 1. The positive and negative responses provide valuable information of the structure of the neighborhood, as they reveal the gradient direction of bright and dark areas in the neighborhood. Thereby, this distinction, between dark and bright responses, allows LDN to differentiate between blocks with the positive and the negative direction swapped (which is equivalent to swap the bright and the dark areas of the neighborhood, as shown in the middle of fig. 1) by generating a different code for each instance, while other methods may mistake the swapped regions as one. Furthermore, these transitions occur often in the face, for example, the top and bottom edges of the eyebrows and mouth have different intensity transitions. Thus, it is important to differentiate among them; LDN can accomplish this task as it assigns a specific code to each of them.

### A. Difference with previous work

Current methods have several shortcomings. For example, LBP [25] encodes the local neighborhood intensity by using the center pixel as a threshold for a sparse sample of the neighboring pixels. The few number of pixels used in this method introduce several problems: First, it limits the accuracy of the method. Second, the method discards most of the

information in the neighborhood. Finally, it makes the method very sensitive to noise. Moreover, these drawbacks are more evident for bigger neighborhoods. Consequently, to avoid these problems more information from the neighborhood can be used, as other methods do [27], [28], [31], [35], [36]. Although the use of more information makes these methods more stable, they still encode the information in a similar way as LBP: by marking certain characteristics in a bit string. And despite the simplicity of the bit string coding strategy, it discards most information of the neighborhood. For example, the directional (LD<sub>i</sub>P) [28] and derivative (LD<sub>e</sub>P) [31] methods miss some directional information (the responses' sign) by treating all directions equally. Also, they are sensitive to illumination changes and noise, as the bits in the code will flip and the code will represent a totally different characteristic. To avoid these problems, we investigate a new coding scheme, that implicitly uses the sign of the directional numbers to increase the encoded structural information, with two different masks: a derivative-Gaussian (to avoid the noise perturbation, and to make our method robust to illumination changes, as previous methods showed [33]) and a Kirsch compass mask. Figure 1 shows how LDN produces different codes in different scenarios, while LD<sub>i</sub>P [28] produces the same code (note that LD<sub>e</sub>P will have a similar result). Thus, the use of the directional numbers produces a more robust code than a simple bit string. Moreover, the use of principal directions may be similar to a weighted coding scheme, in the sense that not all directions have the same importance. In contrast, previous weighting methods [34] treat the code (again) as a bit string, picking all the information of the neighborhood, and weight only the inclusion of each code into the descriptor. However, we (equally) use the two principal directional numbers of each neighborhood (and code them into a single number) instead of assigning weights to them. Consequently, we pick the prominent information of each pixel's neighborhood. Therefore, our method filters and gives more importance to the local information before coding it, while other methods weight the grouped (coded) information.

In summary, the key points of our proposed method are: (1) the coding scheme is based on directional numbers, instead of bit strings, which encodes the information of the neighborhood in a more efficient way; (2) the implicit use of sign information, in comparison with previous directional and derivative methods we encode more information in less space, and, at the same time, discriminate more textures; and (3) the use of gradient information makes the method robust against illumination changes and noise.

### B. Coding scheme

In our coding scheme, we generate the code, LDN, by analyzing the edge response of each mask,  $\{M^0, \dots, M^7\}$ , that represents the edge significance in its respective direction, and by combining the dominant directional numbers. Given that the edge responses are not equally important, the presence of a high negative or positive value signals a prominent dark or bright area. Hence, to encode these prominent regions, we implicitly use the sign information, as we assign a fixed

$$\begin{array}{cccc} \begin{bmatrix} -3 & -3 & 5 \\ -3 & 0 & 5 \\ -3 & -3 & 5 \end{bmatrix} & \begin{bmatrix} -3 & 5 & 5 \\ -3 & 0 & 5 \\ -3 & -3 & -3 \end{bmatrix} & \begin{bmatrix} 5 & 5 & 5 \\ -3 & 0 & -3 \\ -3 & -3 & -3 \end{bmatrix} & \begin{bmatrix} 5 & 5 & -3 \\ 5 & 0 & -3 \\ -3 & -3 & -3 \end{bmatrix} \\ M_0 & M_1 & M_2 & M_3 \\ \begin{bmatrix} 5 & -3 & -3 \\ 5 & 0 & -3 \\ 5 & -3 & -3 \end{bmatrix} & \begin{bmatrix} -3 & -3 & -3 \\ 5 & 0 & -3 \\ 5 & 5 & -3 \end{bmatrix} & \begin{bmatrix} -3 & -3 & -3 \\ -3 & 0 & -3 \\ 5 & 5 & 5 \end{bmatrix} & \begin{bmatrix} -3 & -3 & -3 \\ -3 & 0 & 5 \\ -3 & 5 & 5 \end{bmatrix} \\ M_4 & M_5 & M_6 & M_7 \end{array}$$

Fig. 2. Kirsch compass masks.

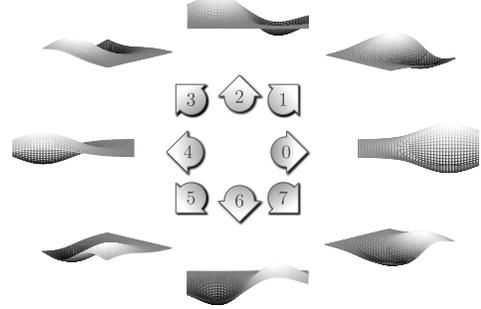


Fig. 3. Derivative of Gaussian compass masks, computed by Eq. (6).

position for the top positive directional number, as the three most significant bits in the code, and the three least significant bits are the top negative directional number, as shown in fig. 1. Therefore, we define the code as:

$$\text{LDN}(x, y) = 8i_{x,y} + j_{x,y}, \quad (1)$$

where  $(x, y)$  is the central pixel of the neighborhood being coded,  $i_{x,y}$  is the directional number of the maximum positive response, and  $j_{x,y}$  is the directional number of the minimum negative response defined by:

$$i_{x,y} = \arg \max_i \{\mathbb{I}^i(x, y) \mid 0 \leq i \leq 7\}, \quad (2)$$

$$j_{x,y} = \arg \min_j \{\mathbb{I}^j(x, y) \mid 0 \leq j \leq 7\}, \quad (3)$$

where  $\mathbb{I}^i$  is the convolution of the original image,  $I$ , and the  $i$ th mask,  $M^i$ , defined by:

$$\mathbb{I}^i = I * M^i. \quad (4)$$

### C. Compass masks

We use the gradient space, instead of the intensity feature space, to compute our code. The former has more information than the later, as it holds the relations among pixels implicitly (while the intensity space ignores these relations). Moreover, due to these relations the gradient space reveals the underlying structure of the image. Consequently, the gradient space has more discriminating power to discover key facial features. Additionally, we explore the use of a Gaussian to smooth the image, which makes the gradient computation more stable. These operations make our method more robust; similarly previous research [28], [31], [33] used the gradient space to compute their code. Hence, our method is robust against illumination due to the gradient space, and to noise due to the smoothing.

To produce the LDN code, we need a compass mask to compute the edge responses. In this paper, we analyze our

proposed code using two different asymmetric masks: Kirsch and derivative-Gaussian (shown in figs. 2 and 3). Both masks operate in the gradient space, which reveals the structure of the face. Furthermore, we explore the use of Gaussian smoothing to stabilize the code in presence of noise by using the derivative-Gaussian mask.

The Kirsch mask [39] is rotated  $45^\circ$  apart to obtain the edge response in eight different directions, as shown in fig. 2. We denote the use of this mask to produce the LDN code by  $\text{LDN}^K$ . Moreover, inspired by the Kirsch mask [39], we use the derivative of a skewed Gaussian to create an asymmetric compass mask that we use to compute the edge response on the smoothed face. This mask is robust against noise and illumination changes, while producing strong edge responses. Hence, given a Gaussian mask defined by:

$$G_\sigma(x, y) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right), \quad (5)$$

where  $x, y$  are location positions, and  $\sigma$  is the width of the Gaussian bell; we define our mask as:

$$M_\sigma(x, y) = G'_\sigma(x + k, y) * G_\sigma(x, y), \quad (6)$$

where  $G'_\sigma$  is the derivative of  $G_\sigma$  with respect to  $x$ ,  $\sigma$  is the width of the Gaussian bell,  $*$  is the convolution operation, and  $k$  is the offset of the Gaussian with respect to its center—in our experiments we use one fourth of the mask diameter for this offset. Then, we generate a compass mask,  $\{M_\sigma^0, \dots, M_\sigma^7\}$ , by rotating  $M_\sigma$ ,  $45^\circ$  apart, in eight different directions. Thus, we obtain a set of masks similar to those shown in fig. 3. Due to the rotation of the mask,  $M_\sigma$ , there is no need of computing the derivative with respect to  $y$  (because it is equivalent to the  $90^\circ$  rotated mask) or other combination of these variables. We denote the code generated through this mask as  $\text{LDN}_\sigma^G$ , where  $\sigma$  determines the parameter for the Gaussian.

### III. FACE DESCRIPTION

Each face is represented by a LDN histogram (LH) as shown in fig. 4(a). The LH contains fine to coarse information of an image, such as edges, spots, corners and other local texture features. Given that the histogram only encodes the occurrence of certain micro-patterns without location information, to aggregate the location information to the descriptor, we divide the face image into small regions,  $\{R^1, \dots, R^N\}$ , and extract a histogram  $H^i$  from each region  $R^i$ . We create the histogram,  $H^i$ , using each code as a bin, and then accumulating all the codes in the region in their respective bin by:

$$H^i(c) = \sum_{\substack{(x,y) \in R^i \\ \text{LDN}(x,y)=c}} v, \quad \forall c, \quad (7)$$

where  $c$  is a LDN code, and  $(x, y)$  is a pixel position in the region  $R^i$ ,  $\text{LDN}(x, y)$  is the LDN code for the position  $(x, y)$ , and  $v$  is the accumulation value—commonly the accumulation value is one. Finally, the LH is computed by concatenating those histograms:

$$\text{LH} = \prod_{i=1}^N H^i, \quad (8)$$

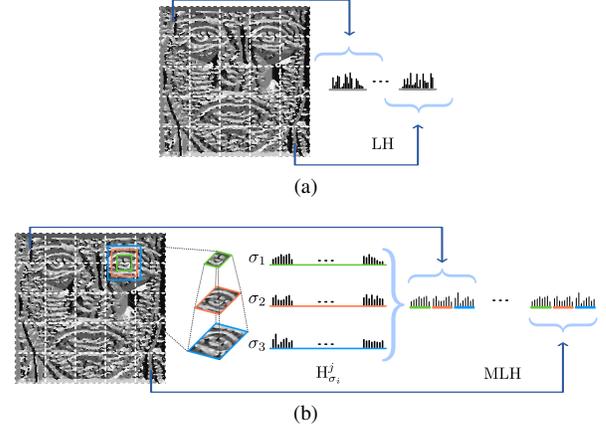


Fig. 4. Face descriptor using uniform grid for histogram extraction. (a) LDN histogram (LH), and (b) multi-LDN histogram (MLH).

where  $\prod$  is the concatenation operation, and  $N$  is the number of regions of the divided face. The spatially combined LH plays the role of a global face feature for the given face.

The use of the derivative-Gaussian mask allows us to freely vary the size of the mask. The change in the size allows the coding scheme,  $\text{LDN}^G$ , to capture different characteristics of the face. Hence, a fine to coarse representation is achieved by computing the  $\text{LDN}_\sigma^G$  code at  $n$  different  $\sigma_i$  (which we represent by  $\text{LDN}_{\sigma_1, \dots, \sigma_n}^G$ ), and by concatenating the histogram of each  $\sigma_i$ ,  $H_{\sigma_i}^j$ , which is computed in the same way as Eq. (7) by using  $\text{LDN}_\sigma^G$ , we can merge the characteristics at different resolutions [as shown in fig. 4(b)]. We call this mixture of resolutions a multi-LDN histogram (MLH), and it is computed by:

$$\text{MLH}_{\sigma_1, \dots, \sigma_n} = \prod_{j=1}^N \prod_{i=1}^n H_{\sigma_i}^j, \quad (9)$$

where  $\prod$  is the concatenation operation,  $H_{\sigma_i}^j$  is the histogram of the  $\text{LDN}_{\sigma_i}^G$  code at the  $R^j$  region, and  $n$  is the number of  $\sigma$ 's used—in our experiments we limit ourselves to three. The change in the mask's size allows our method to capture features in the face that otherwise may be overlooked. As previous research showed [40], it is vital to provide descriptive features for long range pixel interaction. However, previous works do not take into account the long range pixel interaction that takes place outside the coverage of their neighborhood system. We find that combining the local shape information, the relation between the edge responses, and relating the information from different resolutions can better characterize the face's characteristics.

In other words, we represent the face using a single-feature histogram, by using LH, or with a multi-feature histogram, by using MLH. The LDN code in LH can be  $\text{LDN}^K$  or  $\text{LDN}_\sigma^G$ , and the code in MLH must be a  $\text{LDN}_{\sigma_1, \dots, \sigma_n}^G$ .

#### A. Face Recognition

The LH and MLH are used during the face recognition process. The objective is to compare the encoded feature vector from one person with all other candidate's feature vector with

the Chi-Square dissimilarity measure. This measure between two feature vectors,  $F_1$  and  $F_2$ , of length  $N$  is defined as:

$$\chi^2(F_1, F_2) = \sum_{i=1}^N \frac{(F_1(i) - F_2(i))^2}{F_1(i) + F_2(i)}. \quad (10)$$

The corresponding face of the feature vector with the lowest measured value indicates the match found.

### B. Expression Recognition

We perform the facial expression recognition by using a Support Vector Machine (SVM) to evaluate the performance of the proposed method. SVM [41] is a supervised machine learning technique that implicitly maps the data into a higher dimensional feature space. Consequently, it finds a linear hyperplane, with a maximal margin, to separate the data in different classes in this higher dimensional space.

Given a training set of  $M$  labeled examples  $T = \{(x_i, y_i) \mid i = 1, \dots, M\}$ , where  $x_i \in \mathbb{R}^n$  and  $y_i \in \{-1, 1\}$ , the test data is classified by:

$$f(x) = \text{sign} \left( \sum_{i=1}^M \alpha_i y_i K(x_i, x) + b \right), \quad (11)$$

where  $\alpha_i$  are Lagrange multipliers of dual optimization problem,  $b$  is a bias, and  $K(\cdot, \cdot)$  is a kernel function. Note that SVM allows domain-specific selection of the kernel function. Although many kernels have been proposed, the most frequently used kernel functions are the linear, polynomial, and Radial Basis Function (RBF) kernels.

Given that SVM makes binary decisions, multi-class classification can be achieved by adopting the one-against-one or one-against-all techniques. In our work, we opt for one-against-one technique, which constructs  $k(k-1)/2$  classifiers, that are trained with data from two classes [42]. We perform a grid-search on the hyper-parameters in a 10-fold cross-validation scheme for parameter selection, as suggested by Hsu *et al.* [43]. The parameter setting producing the best cross-validation accuracy was picked.

## IV. EXPERIMENTS

We performed several experiments to evaluate the performance of the proposed coding scheme for face recognition and expression classification. We analyzed the former under expression, time lapse, pose, and illumination variation. Also, we tested the proposed code for expression recognition with six and seven expressions.

Regarding the length of the proposed descriptor, the basic LDN has 56 different values, and the length of the final descriptor will be a multiple of this length. Consequently,  $\text{LDN}^K$  has a length of 56, and the  $\text{LDN}^G$  codes have a length of  $56n$ , where  $n$  is the number of sigmas used (in our experiments we set  $n = 3$ ). Note that similar methods have descriptors with greater lengths. For example, the basic length of: LBP [25] (in the uniform case) is 59,  $\text{LD}_i\text{P}$  [28] is 56,  $\text{LD}_e\text{P}$  [31] is 1024, LPQ [36] is 256, LTP [33] (coded as two uniform LBP codes) is 128, and general LTP is  $3^8$ . However, multi-scale codes, like HGPP [35], have huge lengths, as the

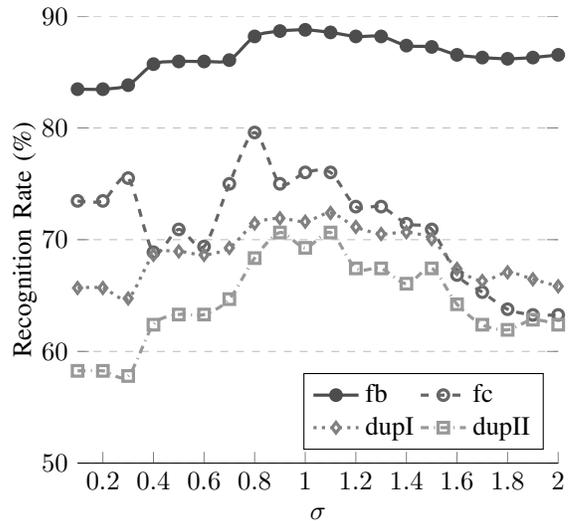


Fig. 5. Recognition accuracy of  $\text{LDN}^G$ , by varying  $\sigma$ , on the FERET database.

global version (GGPP) length is  $256n_s$ , and the local version (LGPP) length is  $256n_s n_o$  (where  $n_s = 5$  is the number of scales, and  $n_o = 8$  is the number of orientations). Furthermore, HGPP is a combination of the local and global versions, which will combine its lengths (note that the use of real and imaginary values will double the length). Due to the length of the HGPP descriptor, we will not compare against it in the following section—see the end of next section for more details on the differences with the literature. Additionally, all these lengths should be multiplied by the grid size. In comparison, our multi-scale descriptor is extremely compact, and the single scale is more compact than other descriptors. Moreover, the execution time of our code is, in average, 37 ms on images of size  $100 \times 100$  for one mask—we produced this time using a Dual-Core CPU with 2.5 GHz, using un-optimized MATLAB code. Moreover, the codes that use  $n$  different sigmas will take, in average,  $n$  times more.

### A. Face Recognition

We tested our method for face recognition in several databases: FERET [44], Yale B [45], Extended Yale B [46], LFW [47], and CAS-PEAL [48]. Moreover, we cropped and normalized all images to  $100 \times 100$  pixels, based on the ground truth positions of the two eyes and mouth when available, or used a face detector to crop the face. In our experiments, every image is partitioned into  $10 \times 10$  regions for all the methods.

1) *FERET results*: We tested the performance of the methods, for the face recognition problem, in accordance to the CSU Face Identification Evaluation System with images from the FERET [44] database. In this problem, given a gallery containing labeled face images of several individuals (one or more face images for each person), we classify a new set of probe images. Thus, we used *fa* image set as gallery and the other four sets as probe images. These sets are *fb*, for expression variation, *fc*, for illumination variation, *dupI* and *dupII*, for time lapse variation. (Note that in the FERET methodology

these datasets are for age variation testing. However the time between images is not significant for age variation. Instead, we associate this factor with time lapse variation.)

First, given that  $\text{LDN}_{\sigma}^G$  depends on its parameter  $\sigma$ , we test different  $\sigma$ 's to analyze the performance of the code when varying this parameter. As fig. 5 reveals, the code presents an increment in the interval  $0.5 \leq \sigma \leq 1.5$ . Thus, we test the multi-LDN code,  $\text{LDN}_{\sigma_1, \sigma_2, \sigma_3}^G$ , for different combinations in this interval. We choose to investigate the combination of the small neighborhoods ( $3 \times 3$ ,  $5 \times 5$ ,  $7 \times 7$ ) in  $\text{LDN}_{0.3, 0.6, 0.9}^G$ , medium neighborhoods ( $5 \times 5$ ,  $7 \times 7$ ,  $9 \times 9$ ) in  $\text{LDN}_{0.5, 1.0, 1.5}^G$ , and large neighborhoods ( $7 \times 7$ ,  $9 \times 9$ ,  $11 \times 11$ ) in  $\text{LDN}_{1.0, 1.3, 1.6}^G$ .

Table I shows the results of all the different methods in the FERET database. The  $\text{LDN}_{\sigma_1, \sigma_2, \sigma_3}^G$  codes outperform the results of  $\text{LDN}^K$ , and other methods in the expression and time lapse variation data sets (fb, dupI, and dupII). For the intensity variation data set (fc), LBP has the same accuracy as the best  $\text{LDN}^G$  code, but not as good as the  $\text{LDN}^K$  code. However, for extreme illumination variation LBP's and  $\text{LDN}^K$ 's performance considerably drop in comparison to  $\text{LDN}^G$  codes—*c.f.* fig. 9. Moreover, GGPP and LPQ produce the best results in fc data set, because they do not rely on intensity. Instead, GGPP and LPQ use phase as main feature to build their code, which makes them more robust to illumination. However, like LBP, LPQ's accuracy considerably drops in extreme lighting variations—*c.f.* fig. 9. Furthermore, note that the directional patterns (LD<sub>i</sub>P and LD<sub>e</sub>P) produce poor results in this data set. Moreover, given that we are evaluating the raw power of the descriptors, and for a fair comparison, we did not pre-process any image. Consequently, some methods exhibit lower accuracy than expected, such as,  $\text{LBP}_w$ , LTP, and LD<sub>e</sub>P. Additionally, the reduced accuracy of  $\text{LBP}_w$  may due to its design (for infrared images), and the changes in intensity prove too challenging for the method. Regarding the  $\text{LDN}^G$  combined codes, the medium neighborhood combination,  $\text{LDN}_{0.5, 1.0, 1.5}^G$ , performs better than the other two. This high accuracy is due to the  $\sigma$  combination that recovers small to large characteristics, instead of picking only small or large characteristics. Therefore, we can say that the improvement of this assemble is due to the balance of its masks sizes that range from small to large regions. This behavior is also supported by the high performance of these middle  $\sigma$ 's as shown in fig. 5.

Despite  $\text{LDN}^K$  code being more robust against illumination than previous methods (as shown in table I and fig. 9), such as, LD<sub>e</sub>P, LD<sub>i</sub>P, LTP, or LBP, it has an inferior recognition capability in comparison with  $\text{LDN}^G$ . Moreover, we found that due to the higher discrimination power of  $\text{LDN}^K$  code, it overlooks the face identities and tends to match stronger features in the face, such as, expression's structures. This behavior is shown in fig. 6, in which we can see that the code represents better the facial expression characteristics, as it matches similar expression among different people. This discrimination comes from the use of the Kirsch mask, which extracts more robust structural features than our proposed derivative-Gaussian mask. Therefore, we can accommodate the

TABLE I  
COMPARISON OF THE RECOGNITION ACCURACY OF THE LDN CODE AND OTHER METHODS IN THE FERET DATABASE.

Method	fb	fc	dupI	dupII
LBP [25]	80.99	84.69	64.90	48.62
$\text{LBP}_w$ [34]	79.93	84.18	50.55	19.72
LTP [27]	84.30	36.22	52.26	22.94
LD <sub>i</sub> P [28]	83.12	71.94	66.61	58.26
LD <sub>e</sub> P [31]	85.10	79.35	63.45	61.21
LPQ [36]	84.89	88.78	63.34	46.79
GGPP [35]	87.60	<b>92.86</b>	70.67	66.97
$\text{LDN}^K$	82.88	86.22	65.21	50.46
$\text{LDN}_{0.3, 0.6, 0.9}^G$	87.84	84.69	72.86	69.27
$\text{LDN}_{0.5, 1.0, 1.5}^G$	<b>88.55</b>	81.12	<b>73.32</b>	<b>71.10</b>
$\text{LDN}_{1.0, 1.3, 1.6}^G$	88.43	78.06	72.08	70.18



Fig. 6. False detected faces using  $\text{LDN}^K$  code. The first row shows the query face, and the second row the retrieved face. Note that the code matches the expressions instead of the faces.

mask that is used to extract the features according to the target application.

2) *Noise evaluation:* To evaluate the robustness of the proposed method against noise, we corrupted the probe face images, in the FERET database, with white Gaussian noise, and then try to identify them using the same process as described before. We perform this experiment with different levels of noise, and the results are shown in fig. 7. The robustness of LDN, against noise, is notable as it outperforms the other methods for every level of noise in every data set. LD<sub>i</sub>P and LBP have problems overcoming the errors introduced by the noise. However, LDN, due to the use of the directional numbers, has a higher recognition rate.

Among the different  $\text{LDN}^G$  schemes that we tested, the combination of the medium neighborhoods ( $\text{LDN}_{0.5, 1.0, 1.5}^G$ ) has a higher average recognition rate than the other two. However, it is not as robust as  $\text{LDN}_{1.0, 1.3, 1.6}^G$  in the presence of noise. In contrast, for noise-free environments, we found that the inclusion of medium neighborhoods ( $\text{LDN}_{0.5, 1.0, 1.5}^G$ ) provides better results. As this combination includes characteristics of several resolutions that are different enough (as they include information that differentiates them from others), yet consistent with each other, to represent the face's textures. This balance of the size combination is not outstanding in the other two  $\text{LDN}^G$  schemes. Nevertheless, these other  $\text{LDN}^G$  schemes produce better results than previous methods. Moreover, each  $\text{LDN}^G$  scheme has different characteristics that can be exploited in certain conditions. For example,  $\text{LDN}_{0.3, 0.6, 0.9}^G$  has a high performance under illumination variation, as the

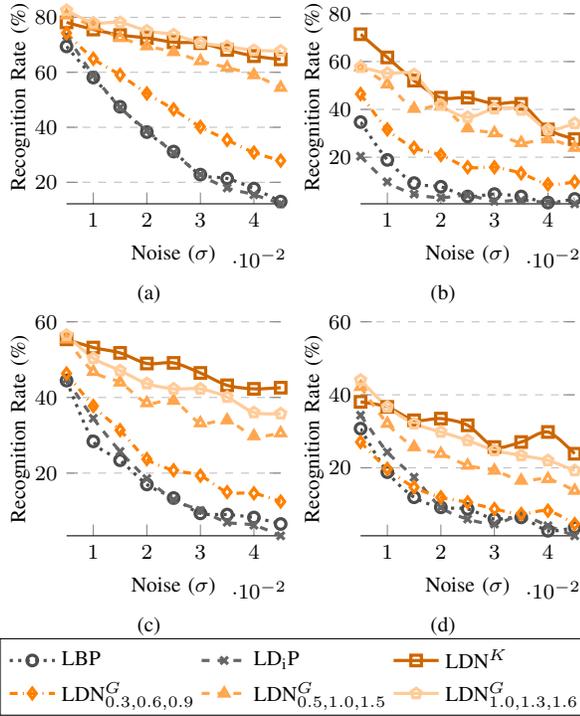


Fig. 7. Recognition accuracy in presence of noise on the FFERET data sets: (a) fb, (b) fc, (c) dupI, and (d) dupII.



Fig. 8. Samples images of a single subject of the (Extended) Yale B database under illumination variation, with their facial characteristics hidden by the shadows. And their respective LDN coded faces, in which those facial features can be easily distinguished. (In this example we used LDN<sup>K</sup>.)

results in the fc data set in table I show, and LDN<sup>G</sup><sub>1.0,1.3,1.6</sub> has a high performance under extreme illumination variation and noisy conditions. Note that the LDN<sup>K</sup> code performs equally good or better as the best LDN<sup>G</sup> codes in the presence of noise. A high presence of noise in the face makes the textures on the face more difficult to detect, thus the use of the Kirsch and derivative-Gaussian masks stabilize the texture codes.

3) *(Extended) Yale B results*: Furthermore, we used the Yale B [45] and the Extended Yale B [46], which is an improvement over the former, databases for illumination variation evaluation. The former contains images of ten subjects, and the latter contains images of 38 subjects, both with nine poses and 64 illuminations per pose. And we used the frontal face images of all subjects, each with 64 different illuminations. The faces are divided into five subsets based on the angle of the light source directions. The five subsets are: *Sub 1* ( $0^\circ$  to

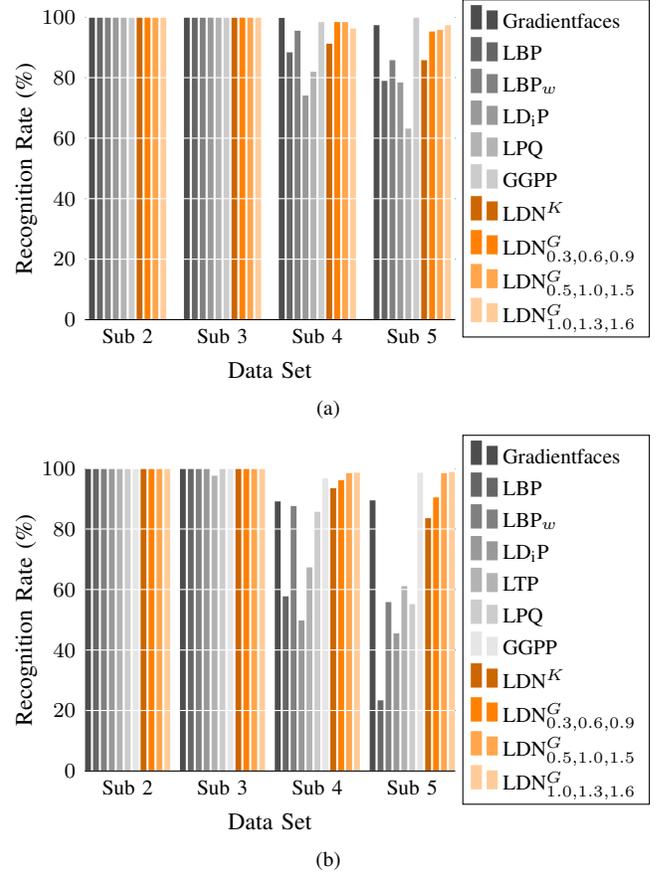


Fig. 9. Recognition accuracy of the different methods in the (a) Yale B and (b) Extended Yale B databases. Note that the all the methods use no pre-processing.

$12^\circ$ ), *Sub 2* ( $13^\circ$  to  $25^\circ$ ), *Sub 3* ( $26^\circ$  to  $50^\circ$ ), *Sub 4* ( $51^\circ$  to  $77^\circ$ ), and *Sub 5* (above  $78^\circ$ ). We used *Sub 1* image set as gallery and the other four sets as probe images.

The difficulty of these databases increases for the subsets four and five, due to the illumination angles that cover half of the face with shadows. Figure 8 shows that the LDN coded faces reveal the facial features in presence of shadows in the face. Moreover, we present results without pre-processing to evaluate the robustness of the descriptors alone. Nevertheless, our method recovers face features in the dark areas, as it does not rely on intensity, like LBP or LTP. We evaluated our method against other methods: Gradientfaces [33], LD<sub>i</sub>P [28], LBP [25], LBP<sub>w</sub> [34], LTP [27], LQP [36], and GGPP [35]; and the results are shown in fig. 9. Most methods perform well in the normal Yale B database, except in the Sub 5 data set, in which the proposed method, Gradientfaces, and GGPP outperform the other methods. All methods, except for LTP, are flawless in the first two sets in the Extended Yale B, which have minor illumination changes. However, for the last two sets the recognition rate of LD<sub>i</sub>P, LBP, LBP<sub>w</sub>, LTP, and LPQ decreases significantly. On the other hand, LDN takes advantage of its compass masks, which are more robust against illumination changes, and uses the directional number encoding scheme to produce a more discriminative code. Note that the proposed LDN performs better than LPQ, which use

phase information. Additionally, GGPP has two more scales, and a greater length than our method to build its descriptor.

The recognition rate difference of  $LDN^G$ , in average on the last two data sets of the normal Yale B, between Gradientfaces and GGPP is 0.7% and 1.3%, respectively. Additionally, on the extended database,  $LDN^G$  is 1.1% better than GGPP, in average on the last two data sets. The  $LDN^K$  code is better than previous codes, but not as good as its Gaussian counterpart. Although Gradientfaces has a higher accuracy than LDN codes in the illumination variation problem (in the normal Yale B), it is not robust against expression and time lapse variation. Gradientfaces has a non-acceptable recognition rate of 7% in fb, and 1% in dupI and dupII in the FERET database. However, LDN codes showed to be more reliable in different variation conditions.

4) *LFW results*: Additionally, we evaluated the proposed method using the Labeled Faces in the Wild (LFW) database [47], which comprises a collection of annotated faces captured from news articles on the web. This database was published with a specific benchmark, which focuses on the face recognition task of pair matching. In this task, given two face images, the goal is to decide whether the two pictures are of the same individual. This is a binary classification problem, in which the two possible outcomes are *same* or *not-same*.

We used a straightforward approach for pair matching, in which we considered the distance between two image descriptors, and we learned a threshold that classifies whether the distances correspond to a matching pair. This method can be generalized by using a binary SVM. We trained a SVM classifier on the 5400 one-dimensional vectors each containing the distance between the two images of a pair. Then, we used this classifier to predict whether the 600 test pairs are matches or not. This experiment is repeated for the ten train/test splits, and we record mean recognition rate as well as the standard deviation of it.

In table II we report the recognition rate for the Euclidean distance for each of the following descriptors: LBP [25], Three and Four Patch LBP (TPLBP and FPLBP) [49], Gabor (C1) [50], and the proposed methods using Kirsch ( $LDN^K$ ) and different derivative-Gaussian masks ( $LDN^G_{0.3,0.6,0.9}$ ,  $LDN^G_{0.5,1.0,1.5}$ ,  $LDN^G_{1.0,1.3,1.6}$ ). The proposed method, for large neighborhoods, achieves better recognition rate than other methods based on LBP. However, the proposed codes, based on small and medium neighborhood, achieve a smaller recognition rate. Note that we use only one distance measure, and Wolf *et al.* [49] showed that the combination of different distance measures produces better results. Hence, our close and better results will be boosted using such combined approach. Although these results are not the best in this benchmark, as noted by Wolf *et al.* too [49], the local descriptors are faster than the methods with higher accuracy.

5) *CAS-PEAL results*: Also, we tested the gallery/probe-sets methodology in the CAS-PEAL face database [48], which contains 99594 images of 1040 individuals (595 males and 445 females) with varying pose, expression, accessory, and lighting (PEAL). The CAS-PEAL-R1, a subset of the CAS-PEAL face database, has been released for the purpose of research, which contains 9060 images of 1040 persons.

TABLE II

MEAN ( $\pm$  STANDARD ERROR) RECOGNITION RATES ON THE FUNNELED PAIR MATCHING BENCHMARK OF LFW (IMAGE-RESTRICTED TRAINING, “VIEW 1”).

Method	Euclidian Distance
LBP [25]	67.67 $\pm$ 0.68
TPLBP [49]	68.75 $\pm$ 0.44
FPLBP [49]	68.65 $\pm$ 0.56
Gabor [50]	62.93 $\pm$ 0.47
$LDN^K$	61.77 $\pm$ 0.38
$LDN^G_{0.3,0.6,0.9}$	66.75 $\pm$ 0.35
$LDN^G_{0.5,1.0,1.5}$	67.67 $\pm$ 0.32
$LDN^G_{1.0,1.3,1.6}$	<b>69.08 <math>\pm</math> 0.31</b>

The CAS-PEAL database provides several comparison environments. The proposed method LDN gives good results for most of the data sets, and it is close to the best methods. Our proposed methods are outperform by the  $LD_iP$ , LPQ, and GGPP methods in the Accessory, Age, Background, and Expression data sets by 0.6%, 1.5%, 0.2% and 1.2%, respectively. Note that the Lighting data set in this database is really challenging. In contrast to the Yale B database which contains only images with shadows, the CAS-PEAL Lighting data set has dark (with shadows) and bright (with flashes) images. This combination of images makes this data set more challenging. As we see in the results, the methods can detect less than half of the data set. Nevertheless, the proposed method still outperforms other methods that use intensity, and even outperforms the GGPP and LPQ methods that use phase information as feature. However, the use of phase information makes GGPP to produce really good results.

Furthermore, note that there are other versions of the algorithm proposed by Zhang *et al.* [35] (HGPP and its weighted variations) from which the authors reported better results (produced in their environment) than those presented here—HGPP has an average accuracy of 87.57%, and 91.01% in FERET and CAS-PEAL databases, respectively. However, for a fair comparison, we used the method with the shortest descriptor (GGPP with a length of 1280 times the grid size) and produced results in our environment. Note that the other methods have larger descriptors that are difficult to compute, and are not feasible for an application. As Zhang *et al.* suggested [35], we also tested the quantized version of the GGPP descriptor (16 times shorter), but its recognition accuracy on the CAS-PEAL database drops drastically to 76.72%, 89.39%, 94.21%, 96.73%, 82.87%, and 16.36%, respectively. In this case, our descriptors outperform all these results. (Note that the same happens in the FERET database, case in which the recognition drops to 83.47%, 83.16%, 61.62%, and 52.75%, respectively. And in both versions of the Yale B database; in the worst case—in the extended database—the recognition drops to 99.78%, 76.81%, and 61.91% for the data sets Sub3 to Sub5.) Moreover, the difference in the detection accuracy that we presented and the one shown by Zhang *et al.* [35], may due to the face detector we used (consequently, the detected face region varies considerably), our deliberated lack of pre-processing (to test the power of the descriptor), and the parameters used to re-size and divide the face.

TABLE III  
RECOGNITION ACCURACY OF SEVERAL METHODS ON THE CAS-PEAL DATABASE.

Methods	Acc.	Age	Back.	Dist.	Expr.	Light.
LBP [25]	75.06	89.39	98.73	97.45	87.45	14.62
LD <sub>2</sub> P [28]	78.21	90.91	<b>99.64</b>	96.73	87.58	17.83
LPQ [36]	81.18	87.88	99.46	97.09	<b>88.60</b>	21.09
GGPP [35]	<b>82.76</b>	<b>96.97</b>	97.29	97.45	87.77	30.85
LDN <sup>K</sup>	80.00	95.45	97.83	<b>98.18</b>	84.20	27.99
LDN <sup>G</sup> <sub>0.3,0.6,0.9</sub>	79.43	93.94	99.46	97.45	86.05	39.50
LDN <sup>G</sup> <sub>0.5,1.0,1.5</sub>	81.66	92.42	99.46	97.45	87.39	<b>40.57</b>
LDN <sup>G</sup> <sub>1.0,1.3,1.6</sub>	82.14	93.94	99.46	97.09	87.26	39.81

### B. Expression Recognition

We performed experiments to evaluate the performance of the proposed algorithm under six, and seven facial expressions using SVM for classification. We tested our method in five different databases: CK [51], CK+ [52], JAFFE [53], MMI [54], [55] and CMU-PIE [56], [57]. Moreover, we cropped and normalized all the images to  $110 \times 150$  pixels, based on the ground truth positions of the eyes and mouth (when available), or by using a face detector. In our experiments, every image is partitioned into  $4 \times 7$  regions. We compared the performance of the proposed LDN based method against several encoding schemes. To evaluate the generalization performance to novel subjects, we adopted a 10-fold cross-validation testing scheme in our experiments. More specifically, we partitioned the data sets randomly into ten groups of roughly equal number of subjects. Nine groups were used as the training data to train the classifiers, while the remaining group was used as the test data. The above process was repeated ten times for each group in turn to be omitted from the training process. We reported the average recognition results on the test sets. In this section we discuss the evaluation process and the results.

We conducted the recognition using SVM with different kernels to classify the facial expressions. Table IV compares the performances with the SVM classifier based on different LDN features, where the degree of the polynomial kernel is one, and the standard deviation for the RBF kernel is  $2^{11}$  for 6-class recognition and  $2^{13}$  for 7-class recognition. For this task, we analyzed the four variations of LDN that we presented before: LDN<sup>K</sup> and the three variations of LDN<sup>G</sup>.

1) (Extended) Cohn-Kanade results: The Cohn-Kanade Facial Expression (CK) database [51] consists of 100 university students. Subjects were instructed to perform a series of 23 facial displays, six of which were based on description of prototype emotions. Image sequences from neutral to target display were digitized. In our setup, we selected 408 image sequences from 96 subjects, each of which was labeled as one of the six basic emotions. For 6-class prototypical expression recognition, the three most expressive image frames were taken from each sequence that resulted into 1224 expression images. In order to build the neutral expression set, the first frame (neutral expression) from all 408 sequences was selected to make the 7-class expression data set (1632 images). Furthermore, we used the extended Cohn-Kanade database (CK+) [52], which includes 593 sequences for seven basic expressions (happiness, sadness, surprise, anger, disgust, fear, and contempt). In our experiments, we selected the most

TABLE IV  
EXPRESSION RECOGNITION ACCURACY OF LDN CODES USING SVM WITH DIFFERENT KERNELS, ON SEVERAL DATABASES: CK, CK+, JAFFE, MMI, AND CMU-PIE.

(a) LDN <sup>K</sup>				
Database		Kernel (%)		
		Linear	Polynomial	RBF
CK	6-Class	98.4 ± 1.4	99.1 ± 0.7	99.2 ± 0.8
	7-Class	92.3 ± 3.0	95.1 ± 4.1	94.8 ± 3.1
CK+	7-Class	82.0 ± 0.8	81.7 ± 0.7	82.3 ± 0.8
JAFFE	6-Class	92.9 ± 1.7	93.4 ± 2.2	92.3 ± 1.7
	7-Class	90.1 ± 3.0	91.1 ± 3.0	89.2 ± 2.8
MMI	6-Class	92.9 ± 3.0	94.1 ± 2.7	93.8 ± 3.1
CMU-PIE	2-Class	84.6 ± 0.3	86.2 ± 0.4	88.8 ± 0.3

(b) LDN <sup>G</sup> <sub>0.3,0.6,0.9</sub>				
Database		Kernel (%)		
		Linear	Polynomial	RBF
CK	6-Class	96.8 ± 0.3	98.6 ± 0.2	98.7 ± 0.3
	7-Class	94.3 ± 0.3	95.5 ± 0.3	95.6 ± 0.7
CK+	7-Class	85.6 ± 0.8	79.5 ± 0.9	85.6 ± 0.8
JAFFE	6-Class	93.4 ± 0.6	92.9 ± 0.2	92.9 ± 0.1
	7-Class	90.6 ± 0.2	91.6 ± 0.4	90.6 ± 0.4
MMI	6-Class	94.9 ± 3.2	95.2 ± 3.5	94.1 ± 2.9
CMU-PIE	2-Class	91.9 ± 0.3	92.1 ± 0.2	92.9 ± 0.2

(c) LDN <sup>G</sup> <sub>0.5,1.0,1.5</sub>				
Database		Kernel (%)		
		Linear	Polynomial	RBF
CK	6-Class	96.8 ± 0.3	98.9 ± 0.2	98.9 ± 0.2
	7-Class	94.3 ± 0.2	96.4 ± 0.3	96.6 ± 0.6
CK+	7-Class	89.0 ± 0.7	66.4 ± 1.0	89.0 ± 0.7
JAFFE	6-Class	92.9 ± 0.4	91.3 ± 0.3	92.4 ± 0.3
	7-Class	90.1 ± 0.3	89.7 ± 0.2	88.7 ± 0.2
MMI	6-Class	95.2 ± 2.7	95.8 ± 2.4	94.6 ± 3.2
CMU-PIE	2-Class	94.2 ± 0.3	93.1 ± 0.2	93.9 ± 0.3

(d) LDN <sup>G</sup> <sub>1.0,1.3,1.6</sub>				
Database		Kernel (%)		
		Linear	Polynomial	RBF
CK	6-Class	97.9 ± 0.4	99.5 ± 0.2	99.1 ± 0.2
	7-Class	94.6 ± 0.2	96.4 ± 0.4	96.6 ± 0.6
CK+	7-Class	89.3 ± 0.6	63.3 ± 0.9	89.3 ± 0.7
JAFFE	6-Class	93.4 ± 0.4	92.4 ± 0.3	93.4 ± 0.4
	7-Class	90.1 ± 0.4	90.6 ± 0.3	90.1 ± 0.2
MMI	6-Class	95.5 ± 3.0	95.5 ± 3.6	94.1 ± 3.9
CMU-PIE	2-Class	94.4 ± 0.2	93.6 ± 0.2	94.3 ± 0.2

expressive image frame from 325 sequences from 118 subjects from the database for evaluation.

The best recognition rates of the proposed methods, in comparison with other methods, are shown in table V, in which the LDN<sup>G</sup> codes perform better in the 6- and 7-class problem on the CK database. To obtain a better picture of the recognition accuracy of individual expression types, we present the confusion matrices for 6- and 7-class expression recognition using the CK database for the best LDN codes. Tables VI and VII show the recognition for the LDN<sup>G</sup><sub>1.0,1.3,1.6</sub> code. These results show that the 6-class recognition can be

TABLE V  
RECOGNITION RATE ON THE COHN-CANADE (CK) AND JAPANESE FEMALE FACIAL EXPRESSION (JAFFE) DATABASE FOR SEVERAL METHODS.

Method	CK database		JAFFE database	
	6-Class (%)	7-Class (%)	6-Class (%)	7-Class (%)
LBP [14]	92.6 ± 2.9	88.9 ± 3.5	86.7 ± 4.1	80.7 ± 5.5
LD <sub>p</sub> P [28]	98.5 ± 1.4	94.3 ± 3.9	85.8 ± 1.1	85.9 ± 1.8
Gabor [58]	89.8 ± 3.1	86.8 ± 3.1	85.1 ± 5.0	79.7 ± 4.2
LDN <sup>K</sup>	<b>99.2 ± 0.8</b>	94.8 ± 3.1	92.3 ± 1.6	89.2 ± 2.8
LDN <sup>G</sup> <sub>0.3,0.6,0.9</sub>	98.7 ± 0.3	95.6 ± 0.7	92.9 ± 0.1	<b>90.6 ± 0.4</b>
LDN <sup>G</sup> <sub>0.5,1.0,1.5</sub>	98.9 ± 0.2	<b>96.6 ± 0.6</b>	92.3 ± 0.3	88.7 ± 0.2
LDN <sup>G</sup> <sub>1.0,1.3,1.6</sub>	99.1 ± 0.2	<b>96.6 ± 0.6</b>	<b>93.4 ± 0.4</b>	90.1 ± 0.2

TABLE VI  
CONFUSION MATRIX OF 6-CLASS FACIAL EXPRESSION RECOGNITION USING SVM (RBF) WITH LDN<sup>G</sup><sub>1.0,1.3,1.6</sub> IN THE CK DATABASE.

(%)	Anger	Disgust	Fear	Joy	Sadness	Surprise
Anger	<b>98.53</b>	0.00	0.00	0.00	0.98	0.49
Disgust	0.00	<b>100.00</b>	0.00	0.00	0.00	0.00
Fear	0.00	0.00	<b>98.98</b>	0.51	0.51	0.00
Joy	0.00	0.00	0.00	<b>100.00</b>	0.00	0.00
Sadness	3.17	0.00	0.00	0.00	<b>96.83</b>	0.00
Surprise	0.00	0.00	0.00	0.00	0.00	<b>100.00</b>

solved with high accuracy; but the greatest confusion occurs for the sadness expression being confused with the anger expression. However, as we include the neutral expression in the 7-class recognition problem, the accuracy of other five expressions decreases because some facial expression samples are confused with a neutral expression.

To further evaluate our proposed methods, we used the extended version of the Cohn-Kanade (CK+) database. In this case, we compared our descriptor against several geometric-base methods. Lucey *et al.* [52] reported two methods when they proposed their database, namely similarity-normalized shape (SPTS), and canonical appearance features (CAPP). Also, Chew *et al.* [59] proposed a constrained local model (CLM) based method. Moreover, Jeni *et al.* [60] also proposed a CLM method by using shape related information only (CLM-SRI). Furthermore, we compared against a method based on emotion avatar image (EAI) [61] that leverages the out of plane rotation. Table VIII(a) shows that our method outperforms all the other methods. Note that all the other methods are geometric based, which use a more complex representation of the face. Yet, our proposed LDN outperforms them with a simple representation. Jeni *et al.* [60] used a temporal normalization step which yields an accuracy of 96%. However, for a fair comparison against all the other methods we leave this score outside of the table, and used the result that do not use the temporal normalization.

Additionally, we present the confusion matrix of our best LDN descriptor in table IX. The worst confusion occurs among the anger, contempt, and sadness emotions. These emotions have similar characteristics, which difficult their detection from a single frame.

2) *JAFFE results*: Additionally, we used the Japanese Female Facial Expression (JAFFE) database [53], which contains only 213 images of female facial expression expressed by ten

TABLE VII  
CONFUSION MATRIX OF 7-CLASS FACIAL EXPRESSION RECOGNITION USING SVM (RBF) WITH LDN<sup>G</sup><sub>1.0,1.3,1.6</sub> IN THE CK DATABASE.

(%)	Anger	Disgust	Fear	Joy	Sadness	Surprise	Neutral
Anger	<b>89.81</b>	0.00	0.00	0.00	0.00	0.00	10.19
Disgust	0.00	<b>97.06</b>	0.00	0.00	0.00	0.00	2.94
Fear	0.00	0.00	<b>97.97</b>	0.00	0.00	0.00	2.03
Joy	0.00	0.00	0.00	<b>99.62</b>	0.00	0.00	0.38
Sadness	0.00	0.00	0.00	0.00	<b>97.84</b>	0.00	2.16
Surprise	0.00	0.00	0.00	0.00	0.00	<b>99.59</b>	0.41
Neutral	3.32	0.00	0.51	0.00	1.27	0.00	<b>94.90</b>

TABLE VIII  
RECOGNITION ACCURACY FOR EXPRESSIONS ON THE (A) CK+, (B) MMI, AND (C) CMU-PIE DATABASES.

	(a)	(b)	(c)
Method	CK+	Method	MMI
SPTS [52]	50.4	LBP [14]	86.9
CAPP [52]	66.7	CPL [62]	49.4
SPTS+CAPP	83.3	CSPL [62]	73.5
CLM [59]	74.4	AFL [62]	47.7
CLM-SRI [60]	88.6	ADL [62]	47.8
EAI [61]	82.6	LDN <sup>K</sup>	94.1
LDN <sup>K</sup>	82.3	LDN <sup>G</sup> <sub>0.3,0.6,0.9</sub>	95.2
LDN <sup>G</sup> <sub>0.3,0.6,0.9</sub>	85.6	LDN <sup>G</sup> <sub>0.5,1.0,1.5</sub>	<b>95.8</b>
LDN <sup>G</sup> <sub>0.5,1.0,1.5</sub>	89.0	LDN <sup>G</sup> <sub>1.0,1.3,1.6</sub>	95.5
LDN <sup>G</sup> <sub>1.0,1.3,1.6</sub>	<b>89.3</b>	LDN <sup>G</sup> <sub>1.0,1.3,1.6</sub>	<b>94.4</b>

subjects. Each image has a resolution of 256 × 256 pixels with almost the same number of images for each categories of expression. The head in each image is usually in frontal pose, and the subject's hair was tied back to expose all the expressive zones of her face.

We observed that the recognition accuracy in JAFFE database, shown in table IV, is relatively lower than the CK database. One of the main reasons behind this accuracy is that some expressions in the JAFFE database are very similar with other expressions. Thus, depending on whether these expression images are used for training or testing, the recognition result is influenced. Furthermore, we compared the proposed method against three other methods, and we show their recognition rates on the table V. As observed, our approach outperforms the others methods.

3) *MMI results*: Moreover, we tested the expression recognition problem on the MMI face database [54], [55], which contains more than 1500 samples of both static images and image sequences of faces in frontal and in profile view displaying various facial expressions of emotion, single AU activation, and multiple AU activation. In our experiments we used the Part II of the database, which comprises 238 clips of 28 subjects (sessions 1767 to 2004) where all expressions (anger, disgust, fear, happiness, sadness, and surprise) were recorded twice. People who wear glasses were recorded once while wearing their glasses, and once without.

We compared our proposed methods against two recent studies: a boosted LBP [14] and several patch-based approaches based on the former method [62]. Zhong *et al.* [62] proposed two methods Common Patches (CPL) and Common and Specific Patches (CSPL) with LBP to produce a more

TABLE IX

CONFUSION MATRIX OF 7-CLASS FACIAL EXPRESSION RECOGNITION USING SVM (RBF) WITH  $LDN_{1,0,1.3,1.6}^G$  IN THE CK+ DATABASE.

(%)	Anger	Contempt	Disgust	Fear	Happy	Sadness	Surprise
Anger	71.70	3.77	3.77	1.89	1.89	15.09	1.89
Contempt	5.26	73.68	0.00	5.26	0.00	10.53	5.26
Disgust	6.56	0.00	93.44	0.00	0.00	0.00	0.00
Fear	0.00	0.00	0.00	90.48	0.00	9.52	0.00
Happy	0.00	1.41	0.00	2.82	95.77	0.00	0.00
Sadness	10.53	5.26	0.00	5.26	0.00	78.95	0.00
Surprise	0.00	0.00	0.00	1.20	0.00	1.20	97.59

TABLE X

CONFUSION MATRIX OF 6-CLASS FACIAL EXPRESSION RECOGNITION USING SVM (RBF) WITH  $LDN_{0.5,1.0,1.5}^G$  IN THE MMI DATABASE.

(%)	Anger	Disgust	Fear	Happiness	Sadness	Surprise
Anger	100.00	0.00	0.00	0.00	0.00	0.00
Disgust	0.00	95.45	0.00	0.00	4.55	0.00
Fear	0.00	0.00	95.45	0.00	0.00	4.55
Happiness	0.00	3.13	3.13	90.63	3.13	0.00
Sadness	0.00	0.00	0.00	0.00	100.00	0.00
Surprise	0.00	2.63	5.26	2.63	0.00	89.47

localized descriptor. Moreover, they use Adaboost (ADL) to learn certain patches in the face, and code them using LBP; also they use all available patches (AFL) to create the descriptor and recognize the expressions. Furthermore, table VIII(b) shows that the proposed method outperforms previous methods. Additionally, note that our method is not boosted in any way, unlike these other methods. Moreover, we use a wide variety of images, and all expressions to evaluate our performance, while Shan *et al.* [14] used a reduced set.

For a better comprehension of the performance of our approach on the MMI database, table X shows the confusion matrix of the best LDN descriptor. We note that from all the expressions, the surprise expression get confused with disgust, fear, and happiness. This confusion is due to the similarity among the expressions, as some people only rise their eyebrows when surprised, while others open their mouth, which may lead to some confusion. Hence, to improve this detection, temporal information may be incorporated. Furthermore, we show in table IV the recognition rates of the variations of the proposed method.

4) *CMU-PIE results*: Furthermore, we also used the CMU PIE database [56], [57], which includes 41368 face images of 68 people captured under 13 poses, 43 illuminations conditions, and with four different expressions: neutral, smile, blinking, and talk. For our experiments, we tested two expressions: smile and neutral, as blinking and talking requires temporal information, which is out of the scope of this publication. Moreover, we used the poses that are near frontal (camera 27) with horizontal (cameras 05 and 29) and vertical rotation (cameras 07 and 09).

We show on table XI the confusion matrices of the best LDN descriptors. Also, table IV shows the results for all the proposed descriptors and kernels on the CMU-PIE database. We found that the large variation in the head pose influences the result, as we see the  $LDN^K$  descriptor is below 90% ac-

TABLE XI

CONFUSION MATRIX OF 2-CLASS FACIAL EXPRESSION RECOGNITION USING SVM (RBF) WITH (A)  $LDN^K$  AND (B)  $LDN_{1,0,1.3,1.6}^G$  IN THE CMU-PIE DATABASE.

(a)			(b)		
(%)	Neutral	Smile	(%)	Neutral	Smile
Neutral	87.18	12.82	Neutral	93.80	6.20
Smile	8.25	91.75	Smile	5.90	94.10

curacy, while the best of our descriptors achieves an accuracy of 94.4%. Moreover, table VIII(c) shows the comparison of the proposed methods against different other local descriptors. Note that our methods with middle and large neighborhoods outperform the rest. It seems that the Kirsch mask cannot recover the subtle differences between the smile and neutral expression, as the results of  $LD_iP$  and  $LDN^K$  have the lowest accuracy.

## V. CONCLUSION

In this paper we introduced a novel encoding scheme, LDN, that takes advantage of the structure of the face's textures and that encodes it efficiently into a compact code. LDN uses directional information that is more stable against noise than intensity, to code the different patterns from the face's textures. Additionally, we analyzed the use of two different compass masks (a derivative-Gaussian and Kirsch) to extract this directional information, and their performance on different applications. In general, LDN, implicitly, uses the sign information of the directional numbers which allows it to distinguish similar texture's structures with different intensity transitions—*e.g.*, from dark to bright and vice versa.

We found that the derivative-Gaussian mask is more stable against noise and illumination variation in the face recognition problem, which makes  $LDN^G$  a reliable and stable coding scheme for person identification. Furthermore, we found that the use of Kirsch mask makes the code suitable for expression recognition, as the  $LDN^K$  code is more robust to detect structural expression features than features for identification. Moreover, we proposed a face descriptor that combines the information from several neighborhoods at different sizes to encode micro patterns at those levels. Consequently, LDN recovers more information, and uses it to increase its discriminating power. Furthermore, we found that the combination of different sizes (small, medium and large) gives better recognition rates for certain conditions. For example, the combination of  $5 \times 5$ ,  $7 \times 7$ , and  $9 \times 9$  neighborhoods, in the  $LDN^G$  code, yields better results for expression and time lapse variation, in general. And for noise intense environments large neighborhood's sizes perform better than other combinations, and that in such environments the Kirsch mask performs as well as the derivative-Gaussian mask.

Also, we evaluated LDN under expression, time lapse and illumination variations, and found that it is reliable and robust throughout all these conditions, unlike other methods. For example, Gradientfaces had excellent results under illumination variation but failed with expression and time lapse variation. Also, LBP and  $LD_iP$  recognition rate deteriorates faster than LDN in presence of noise and illumination changes.

## REFERENCES

- [1] R. Chellappa, C. Wilson, and S. Sirohey, "Human and machine recognition of faces: A survey," *Proceedings of the IEEE*, vol. 83, no. 5, pp. 705–741, May 1995.
- [2] W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld, "Face recognition: A literature survey," *ACM Comput. Surv.*, vol. 35, no. 4, pp. 399–458, Dec. 2003.
- [3] S. Z. Li, A. K. Jain, Y. L. Tian, T. Kanade, and J. F. Cohn, "Facial expression analysis," in *Handbook of Face Recognition*. Springer New York, 2005, pp. 247–275, 10.1007/0-387-27257-7-12.
- [4] H. Hong, H. Neven, and C. von der Malsburg, "Online facial expression recognition based on personalized galleries," in *Automatic Face and Gesture Recognition, 1998. Proceedings. Third IEEE International Conference on*, Apr. 1998, pp. 354–359.
- [5] I. Kotsia and I. Pitas, "Facial expression recognition in image sequences using geometric deformation features and support vector machines," *IEEE Trans. Image Process.*, vol. 16, no. 1, pp. 172–187, Jan. 2007.
- [6] N. G. Bourbakis and P. Kakumanu, "Skin-based face detection-extraction and recognition of facial expressions," in *Applied Pattern Recognition*, 2008, pp. 3–27.
- [7] N. Bourbakis, A. Esposito, and D. Kavradi, "Extracting and associating meta-features for understanding people's emotional behaviour: Face and speech," *Cognitive Computation*, vol. 3, pp. 436–448, 2011, 10.1007/s12559-010-9072-1.
- [8] P. Kakumanu and N. Bourbakis, "A local-global graph approach for facial expression recognition," in *Tools with Artificial Intelligence, 2006. ICTAI '06. 18th IEEE International Conference on*, Nov. 2006, pp. 685–692.
- [9] A. Cheddad, D. Mohamad, and A. A. Manaf, "Exploiting voronoi diagram properties in face segmentation and feature extraction," *Pattern Recognition*, vol. 41, no. 12, pp. 3842–3859, 2008.
- [10] X. Xie and K.-M. Lam, "Facial expression recognition based on shape and texture," *Pattern Recognition*, vol. 42, no. 5, pp. 1003–1011, 2009.
- [11] P. Ekman, *Emotions Revealed: Recognizing Faces and Feelings to Improve Communication and Emotional Life*. Times Books, 2003.
- [12] P. Ekman and W. Friesen, *Facial Action Coding System: A Technique for the Measurement of Facial Movement*. Palo Alto: Consulting Psychologists Press, 1978.
- [13] W. V. Friesen and P. Ekman, "Emfac-7: Emotional facial action coding system," 1983, University of California at San Francisco.
- [14] 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.
- [15] Y. L. Tian, "Evaluation of face resolution for expression analysis," in *Computer Vision and Pattern Recognition Workshop, 2004. CVPRW '04. Conference on*, Jun. 2004, p. 82.
- [16] M. Pantic and L. J. M. Rothkrantz, "Automatic analysis of facial expressions: The state of the art," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 12, pp. 1424–1445, Dec. 2000.
- [17] M. Turk and A. Pentland, "Eigenfaces for Recognition," *Journal of Cognitive Neuroscience*, vol. 3, no. 1, pp. 71–86, 1991.
- [18] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman, "Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 7, pp. 711–720, Jul. 1997.
- [19] J. Yang, D. Zhang, A. F. Frangi, and J. Yu Yang, "Two-dimensional PCA: A new approach to appearance-based face representation and recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, pp. 131–137, 2004.
- [20] K. Etemad and R. Chellappa, "Discriminant analysis for recognition of human face images," *Journal of Optical Society of America A*, vol. 14, pp. 1724–1733, 1997.
- [21] B. Heisele, T. Serre, and T. Poggio, "A component-based framework for face detection and identification," *Int. J. Comput. Vision*, vol. 74, pp. 167–181, Aug. 2007.
- [22] P. Penev and J. Atick, "Local feature analysis: A general statistical theory for object representation," *Network: Computation in Neural Systems*, vol. 7, no. 3, pp. 477–500, 1996.
- [23] D. Gabor, "Theory of communication," *J. Inst. Elect. Eng.*, vol. 93, pp. 429–457, 1946.
- [24] L. Wiskott, J.-M. Fellous, N. Kuiger, and C. von der Malsburg, "Face recognition by elastic bunch graph matching," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 7, pp. 775–779, Jul. 1997.
- [25] T. Ahonen, A. Hadid, and M. Pietikäinen, "Face description with local binary patterns: Application to face recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 12, pp. 2037–2041, Dec. 2006.
- [26] T. Ojala, M. Pietikäinen, and D. Harwood, "A comparative study of texture measures with classification based on feature distributions," *Pattern Recognition*, vol. 29, no. 1, pp. 51–59, Jan. 1996.
- [27] X. Tan and B. Triggs, "Enhanced local texture feature sets for face recognition under difficult lighting conditions," *IEEE Trans. Image Process.*, vol. 19, no. 6, pp. 1635–1650, Jun. 2010.
- [28] T. Jabid, M. H. Kabir, and O. Chae, "Local directional pattern (LDP) for face recognition," in *Proceedings of the IEEE International Conference on Consumer Electronics*, 2010, pp. 329–330.
- [29] —, "Robust facial expression recognition based on local directional pattern," *ETRI Journal*, vol. 32, pp. 784–794, 2010.
- [30] M. Kabir, T. Jabid, and O. Chae, "A local directional pattern variance (LDPv) based face descriptor for human facial expression recognition," in *Advanced Video and Signal Based Surveillance (AVSS), 2010 Seventh IEEE International Conference on*, Sep. 2010, pp. 526–532.
- [31] B. Zhang, Y. Gao, S. Zhao, and J. Liu, "Local derivative pattern versus local binary pattern: Face recognition with high-order local pattern descriptor," *IEEE Trans. Image Process.*, vol. 19, no. 2, pp. 533–544, Feb. 2010.
- [32] B. Zhang, L. Zhang, D. Zhang, and L. Shen, "Directional binary code with application to polyu near-infrared face database," *Pattern Recognition Letters*, vol. 31, no. 14, pp. 2337–2344, 2010.
- [33] T. Zhang, Y. Y. Tang, B. Fang, Z. Shang, and X. Liu, "Face recognition under varying illumination using gradientfaces," *IEEE Trans. Image Process.*, vol. 18, no. 11, pp. 2599–2606, 2009.
- [34] Z. Xie and G. Liu, "Weighted local binary pattern infrared face recognition based on weber's law," in *Image and Graphics (ICIG), 2011 Sixth International Conference on*, Aug. 2011, pp. 429–433.
- [35] B. Zhang, S. Shan, X. Chen, and W. Gao, "Histogram of gabor phase patterns (HGPP): A novel object representation approach for face recognition," *IEEE Trans. Image Process.*, vol. 16, no. 1, pp. 57–68, Jan. 2007.
- [36] C. H. Chan, J. Kittler, N. Poh, T. Ahonen, and M. Pietikäinen, "(multiscale) local phase quantisation histogram discriminant analysis with score normalisation for robust face recognition," in *Computer Vision Workshops (ICCV Workshops), 2009 IEEE 12th International Conference on*, Oct. 2009, pp. 633–640.
- [37] H. Chen, P. Belhumeur, and D. Jacobs, "In search of illumination invariants," in *Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on*, vol. 1, 2000, pp. 254–261.
- [38] H. Ling, S. Soatto, N. Ramanathan, and D. Jacobs, "A study of face recognition as people age," in *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, Oct. 2007, pp. 1–8.
- [39] R. A. Kirsch, "Computer determination of the constituent structure of biological images," *Computers & Biomedical Research*, pp. 315–328, 1970.
- [40] S. Liao, M. Law, and A. Chung, "Dominant local binary patterns for texture classification," *IEEE Trans. Image Process.*, vol. 18, no. 5, pp. 1107–1118, May 2009.
- [41] C. Cortes and V. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, no. 3, pp. 273–297, 1995.
- [42] C. W. Hsu and C. J. Lin, "A comparison of methods for multiclass support vector machines," *IEEE Trans. Neural Netw.*, vol. 13, no. 2, pp. 415–425, Mar. 2002.
- [43] C. W. Hsu, C. C. Chang, and C. J. Lin, *A practical guide to support vector classification*, Taipei, Taiwan, 2003.
- [44] P. Phillips, H. Moon, S. Rizvi, and P. Rauss, "The FERET evaluation methodology for face-recognition algorithms," *Pattern Analysis and Machine Intelligence*, vol. 22, no. 10, pp. 1090–1104, Oct. 2000.
- [45] A. Georghiadis, P. Belhumeur, and D. Kriegman, "From few to many: Illumination cone models for face recognition under variable lighting and pose," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 6, pp. 643–660, 2001.
- [46] K.-C. Lee, J. Ho, and D. J. Kriegman, "Acquiring linear subspaces for face recognition under variable lighting," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 5, pp. 684–698, 2005.
- [47] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller, "Labeled faces in the wild: A database for studying face recognition in unconstrained environments," University of Massachusetts, Amherst, Tech. Rep. 07–49, Oct. 2007.
- [48] W. Gao, B. Cao, S. Shan, X. Chen, D. Zhou, X. Zhang, and D. Zhao, "The CAS-PEAL large-scale chinese face database and baseline evaluations," *IEEE Trans. Syst., Man, Cybern. A*, vol. 38, no. 1, pp. 149–161, Jan. 2008.
- [49] L. Wolf, T. Hassner, and Y. Taigman, "Descriptor based methods in the wild," in *Real-Life Images workshop at the European Conference on Computer Vision (ECCV)*, Oct. 2008.

- [50] E. Meyers and L. Wolf, "Using biologically inspired features for face processing," *International Journal of Computer Vision*, vol. 76, no. 1, pp. 93–104, 2008.
- [51] T. Kanade, J. Cohn, and Y. L. Tian, "Comprehensive database for facial expression analysis," in *Automatic Face and Gesture Recognition, 2000. Proceedings. Fourth IEEE International Conference on*, 2000, pp. 46–53.
- [52] P. Lucey, J. Cohn, T. Kanade, J. Saragih, Z. Ambadar, and I. Matthews, "The extended Cohn-Kanade dataset (CK+): A complete dataset for action unit and emotion-specified expression," in *Computer Vision and Pattern Recognition Workshops (CVPRW), 2010 IEEE Computer Society Conference on*, Jun. 2010, pp. 94–101.
- [53] M. Lyons, S. Akamatsu, M. Kamachi, and J. Gyoba, "Coding facial expressions with gabor wavelets," in *Automatic Face and Gesture Recognition, 1998. Proceedings. Third IEEE International Conference on*, Apr. 1998, pp. 200–205.
- [54] M. Pantic, M. Valstar, R. Rademaker, and L. Maat, "Web-based database for facial expression analysis," in *Multimedia and Expo, 2005. ICME 2005. IEEE International Conference on*, Jul. 2005.
- [55] M. F. Valstar and M. Pantic, "Induced disgust, happiness and surprise: An addition to the MMI facial expression database," in *Proceedings of Int'l Conf. Language Resources and Evaluation, Workshop on EMOTION*, Malta, May 2010, pp. 65–70.
- [56] T. Sim, S. Baker, and M. Bsat, "The CMU pose, illumination, and expression (PIE) database," in *Proceedings of the 5th International Conference on Automatic Face and Gesture Recognition, 2002*.
- [57] —, "The CMU pose, illumination, and expression database," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 12, pp. 1615–1618, Dec. 2003.
- [58] M. S. Bartlett, G. Littlewort, I. Fasel, and J. R. Movellan, "Real time face detection and facial expression recognition: Development and applications to human computer interaction," in *Computer Vision and Pattern Recognition Workshop, 2003. CVPRW '03. Conference on*, vol. 5, Jun. 2003, p. 53.
- [59] S. Chew, P. Lucey, S. Lucey, J. Saragih, J. Cohn, and S. Sridharan, "Person-independent facial expression detection using constrained local models," in *Automatic Face Gesture Recognition and Workshops (FG 2011), 2011 IEEE International Conference on*, Mar. 2011, pp. 915–920.
- [60] L. Jeni, D. Takacs, and A. Lorincz, "High quality facial expression recognition in video streams using shape related information only," in *Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on*, Nov. 2011, pp. 2168–2174.
- [61] S. Yang and B. Bhanu, "Understanding discrete facial expressions in video using an emotion avatar image," *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, vol. 42, no. 4, pp. 980–992, Aug. 2012.
- [62] L. Zhong, Q. Liuz, P. Yangy, B. Liuy, J. Huangx, and D. N. Metaxasy, "Learning active facial patches for expression analysis," in *Computer Vision and Pattern Recognition, 2012. Proceedings. IEEE Conference on*, 2012.



**Adin Ramirez Rivera** (S'12) received his B.Sc. degree in Computer Engineering from San Carlos University (USAC), Guatemala in 2009. He is currently working towards his Ph.D. degree in the Department of Computer Engineering, Kyung Hee University, South Korea. His research interests are image enhancement, object detection, expression recognition, and pattern recognition.



**Jorge Rojas Castillo** (S'12) received his B.Sc. degree (with highest honors) in Electrical Engineering from San Carlos University (USAC), in 2009. After his graduation, he worked as an Energy Measurement Engineer in the area of Electrical Power in Administrador del Mercado Mayorista in Guatemala from 2009 to 2011. He completed his M.Sc. in the Department of Computer Engineering at Kyung Hee University, South Korea in 2013. His research interests include texture analysis, image processing, computer vision, and pattern recognition.



**Oksam Chae** (M'92) received his B.Sc. degree in Electronics Engineering from Inha University, South Korea in 1977. He completed his M.S. and Ph.D. degree in Electrical and Computer Engineering from Oklahoma State University, USA in 1982 and 1986, respectively. From 1986 to 88, he worked as a research engineer in Texas Instruments Image Processing Laboratory, USA. From 1988, he has been working as a professor in the Department of Computer Engineering, Kyung Hee University, South Korea. His research interest includes multimedia data processing environment, intrusion detection system, sensor networks and medical image processing in dentistry. He is a member of IEEE, SPIE, KES and IEICE.