Rotation-Invariant Image and Video Description With Local Binary Pattern Features

Guoying Zhao, Timo Ahonen, Jiří Matas, and Matti Pietikäinen, *Fellow, IEEE*

***Abstract*—In this paper, we propose a novel approach to compute rotation-invariant features from histograms of local noninvariant patterns. We apply this approach to both static and dynamic local binary pattern (LBP) descriptors. For static-texture description, we present LBP histogram Fourier (LBP-HF) features, and for dynamic-texture recognition, we present two rotation-invariant descriptors computed from the LBPs from three orthogonal planes (LBP-TOP) features in the spatiotemporal domain. LBP-HF is a novel rotation-invariant image descriptor computed from discrete Fourier transforms of LBP histograms. The approach can be also generalized to embed any uniform features into this framework, and combining the supplementary information, e.g., sign and magnitude components of the LBP, together can improve the description ability. Moreover, two variants of rotation-invariant descriptors are proposed to the LBP-TOP, which is an effective descriptor for dynamic-texture recognition, as shown by its recent success in different application problems, but it is not rotation invariant. In the experiments, it is shown that the LBP-HF and its extensions outperform noninvariant and earlier versions of the rotation-invariant LBP in the rotation-invariant texture classification. In experiments on two dynamic-texture databases with rotations or view variations, the proposed video features can effectively deal with rotation variations of dynamic textures (DTs). They also are robust with respect to changes in viewpoint, outperforming recent methods proposed for view-invariant recognition of DTs.**

***Index Terms*—Classification, dynamic texture, feature, Fourier transform, local binary patterns (LBP), rotation invariance, texture.**

I. Introduction

TEXTURE analysis is a basic vision problem [26], [30] with application in many areas, e.g., object recognition, remote sensing, and content-based image retrieval. In many practical applications, textures are captured in arbitrary orientations.

Manuscript received November 24, 2010; revised April 18, 2011 and August 02, 2011; accepted October 22, 2011. Date of publication November 11, 2011; date of current version March 21, 2012. This work was supported in part by the Academy of Finland and in part by Infotech Oulu. The work of J. Matas was supported by the Czech Science Foundation under Project P103/10/1585. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Erhardt Barth.

G. Zhao and M. Pietikäinen are with the Center for Machine Vision Research, Department of Computer Science and Engineering, University of Oulu, 90014 Oulu, Finland (e-mail: gyzhao@ee.oulu.fi; mkp@ee.oulu.fi).

T. Ahonen was with the Center for Machine Vision Research, University of Oulu, 90014 Oulu, Finland. He is now with the Nokia Research Center, Palo Alto, CA 94306 USA (e-mail: timo.ahonen@nokia.com).

J. Matas is with the Center for Machine Perception, Department of Cybernetics, Faculty of Electrical Engineering, Czech Technical University in Prague, 166 27 Prague, Czech Republic (e-mail: matas@cmp.felk.cvut.cz).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIP.2011.2175739

For static textures, rotation-invariant features are independent of the angle of the input texture image [22], [27], [30]. Robustness to image conditions such as illumination is often required/desirable. Describing the appearance locally, e.g., using cooccurrences of gray values or with filter bank responses and then forming a global description by computing statistics over the image region is a well-established technique [26]. This approach has been extended by several authors to produce rotation-invariant features by transforming each local descriptor to a canonical representation invariant to rotations of the input image [2], [22], [27]. The statistics describing the whole region are then computed from these transformed local descriptors. The published work on rotation-invariant texture analysis is extensive.

We have chosen to build our rotation-invariant texture descriptor on the local binary pattern (LBP). The LBP is an operator for image description that is based on the signs of differences of neighboring pixels. It is fast to compute and invariant to monotonic grayscale changes of the image. Despite being simple, it is very descriptive, which is attested by the wide variety of different tasks it has been successfully applied to. The LBP histogram has proven to be a widely applicable image feature for, e.g., texture classification, face analysis, video background subtraction, and interest region description. LBPs have been used for rotation-invariant texture recognition before. The original one is in [22], where the neighboring n binary bits around a pixel are clockwise rotated n times that the maximal number of the most significant bits is used to express this pixel. The more recent dominant LBP method [18], which makes use of the most frequently occurred patterns to capture descriptive textural information, also has the rotation-invariant characteristics. Guo *et al.* developed an adaptive LBP [13] by incorporating the directional statistical information for rotation-invariant texture classification. In [14], LBP variance (LBPV) was proposed to characterize the local contrast information into the 1-D LBP histogram. The performance evaluation using the rotation-invariant LBP, the coordinated-cluster representation, and the improved LBP was conducted on granite texture classification [11]. The sign and the magnitude of LBP, and the binary code of the intensity of center pixels were combined together in CLBP [15] to improve the texture classification. However, the intensity information is very sensitive to illumination changes; thus, this method needs image normalization to remove global intensity effects before feature extraction.

Dynamic textures (DTs) are image sequences with visual pattern repetition in time and space, such as sea waves, smoke, foliage, fire, shower, and whirlwind. For DT analysis, feature description is the key element. Local features have been obtaining increasing attention due to their ability of using microtextons to describe the motions, while there is an argument against global

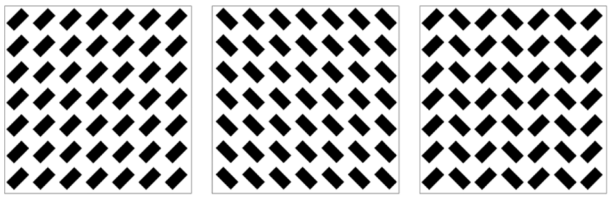

Fig. 1. (a) and (b) Rotations of static textures. (c) Different texture.

spatiotemporal transforms on the difficulty to provide rotation invariance [7]. DTs in video sequences can be arbitrarily oriented. The rotation can be caused by the rotation of cameras and the self-rotation of the captured objects. Rotation-invariant DT analysis is an important but still open research problem. It aims at providing features that are invariant to the rotation angle of the input-texture-image sequences along the time axis. Moreover, these features should also capture the appearance and the motions, should be robust to other challenges such as illumination changes, and should allow multiresolution analysis. Fazekas and Chetverikov [10] studied the normal- and complete-flow features for DT classification. Their features are rotation invariant, and the results on the ordinary DT without rotations are promising. Lu *et al.* proposed a method using spatiotemporal multiresolution histograms based on velocity and acceleration fields [19]. Velocity and acceleration fields of different spatiotemporal resolution image sequences are accurately estimated by the structure tensor method. This method is also rotation-invariant and provides local directionality information. However, both of these methods cannot deal with illumination changes and did not consider the multiscale properties of the DT. Although there are some methods that are rotation invariant in theory, such as [10] and [19], but to our best knowledge, there are very few results reported about their performance evaluation using rotated sequences.

The main contribution of this paper is the observation that invariants globally constructed for the whole region by histogramming noninvariant are superior to most other histogram-based invariant texture descriptors, which normalize rotation locally. In [14], the authors also considered how to get a rotation-invariant strategy from nonrotation-invariant histograms. Our approach is different from that. The method in [14] keeps the original rotation-variant features but finds a match strategy to deal with the rotation. Our method generates new features from rotation-variant features and does not need any special match strategy.

Most importantly, as each local descriptor (e.g., filter bank response) is transformed to canonical representation independently, the relative distribution of different orientations is lost. Fig. 1(a) and (b) represents different rotations of the same texture, whereas Fig. 1(c) is clearly a different texture. Considering the case that each texture element (black bar) is rotated into canonical orientation independently, Fig. 1(a) and (b) will correctly get the same representation, but also, the difference between textures in Fig. 1(a) and (c) will be lost. Furthermore, as the transformation needs to be performed for each texton, it must be computationally simple if the overall computational cost needs to be low.

We apply this idea to static-texture recognition (Sections II and III) and dynamic-texture recognition (Sections IV–VI). Preliminary results for static-texture recognition were presented in [1].

On the basis of the LBP, we propose a novel LBP histogram Fourier (LBP-HF) features for static-texture recognition. The LBP-HF is a rotation-invariant image descriptor based on uniform LBPs [22]. Unlike the earlier local rotation-invariant features, which are histograms of the rotation-invariant version of LBPs, the LBP-HF descriptor is formed by first computing a noninvariant LBP histogram over the whole region and then constructing rotationally invariant features from the histogram. This means that rotation invariance is globally attained, and the features are thus invariant to rotations of the whole input signal, but they still retain information about the relative distribution of different orientations of uniform LBPs. Again, considering Fig. 1, if the rotation is globally compensated for, textures in Fig. 1(a) and (b) get the same description, but the difference between Fig. 1(a) and (c) is retained. In addition, this approach is generalized to embed any uniform features, e.g., sign and magnitude components of the LBP, into this framework to improve the description ability.

Later, this idea is extended to the spatiotemporal domain, i.e., two variants of rotation-invariant LBP-TOP operators are developed and the experiments on two databases show their effectiveness for rotation variations and view changes in the dynamic-texture recognition.

II. Rotation-Invariant Image Descriptors

Here, we will focus on the rotation-invariant image features for static-texture description. Because it is based on the uniform LBP, first, the LBP methodology is briefly reviewed.

A. The LBP Operator

The LBP operator [22] is a powerful means of texture description. The original version of the operator labels the image pixels by thresholding the 3×3 neighborhood of each pixel with the center value and summing the thresholded values weighted by powers of two.

The operator can be also extended to use neighborhoods of different sizes [22] (see Fig. 2). To do this, a circular neighborhood denoted by (P, R) is defined. Here, P represents the number of sampling points, and R is the radius of the neighborhood. These sampling points around pixel (x, y) lie at coordinates $(x_p, y_p) = (x + R\cos(2\pi p/P), y - R\sin(2\pi p/P))$. When a sampling point does not fall at integer coordinates, the pixel value is bilinearly interpolated. Now, the LBP label for the center pixel (x, y) of image $f(x, y)$ is obtained through

$$\mathrm{LBP}_{P,R}(x, y) = \sum_{p=0}^{P-1} s\left(f(x, y) - f(x_p, y_p)\right) 2^p \qquad (1)$$

where $s(z)$ is the thresholding function and

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

Further extensions to the original operator are the so-called *uniform* patterns [22]. An LBP is called uniform if the binary pattern contains at most two bitwise transitions from 0 to 1 or vice versa when the bit pattern is considered circular. In the

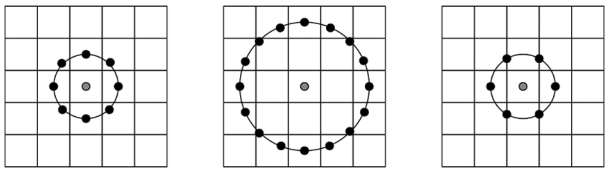

Fig. 2. Three circular neighborhoods: (8, 1), (16, 2), (6, 1). The pixel values are bilinearly interpolated whenever the sampling point is not in the center of a pixel.

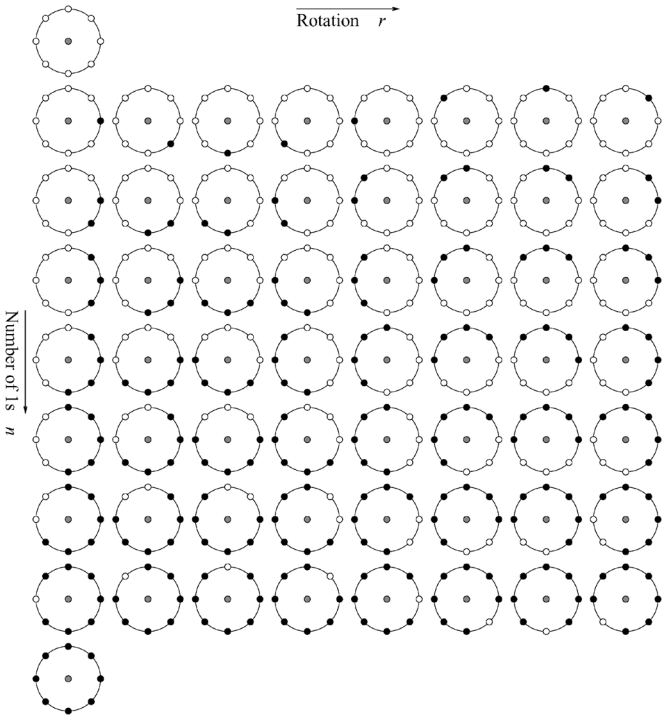

Fig. 3. Fifty-eight different uniform patterns in the (8, R) neighborhood.

computation of the LBP histogram, uniform patterns are used so that the histogram has a separate bin for every uniform pattern and all nonuniform patterns are assigned to a single bin. The 58 possible uniform patterns in neighborhood of eight sampling points are shown in Fig. 3.

The original rotation-invariant LBP operator based on uniform patterns, denoted here as $\mathrm{LBP}^{\mathrm{riu2}}$, is achieved by circularly rotating each bit pattern to the minimum value. For instance, the bit sequences 10000011, 11100000, and 00111000 arise from different rotations of the same local pattern, and they all correspond to the normalized sequence 00000111. In Fig. 3, this means that all the patterns from one row are replaced with a single label.

B. Rotation-Invariant Descriptors From LBP Histograms for Static-Texture Analysis

Let us denote a specific uniform LBP pattern by $U_P(n,r)$. Pair (n,r) specifies a uniform pattern so that n is the number of 1 bit in the pattern (corresponds to the row number in Fig. 3) and r is the rotation of the pattern (the column number in Fig. 3). Table I lists the notations and the corresponding meanings used here.

TABLE I
NOTATIONS AND THEIR CORRESPONDING MEANINGS IN THE DESCRIPTION OF LBP-HF

Notations	Meaning
$U_P(n,r)$	uniform LBP pattern
n	number of 1-bits in the pattern
r	rotation of the pattern
P	number of neighboring sampling points
a	discrete steps of rotation
$h_I(U_P(n,r))$	number of occurrences of uniform pattern $U_P(n,r)$ in image I
$H(n,\cdot)$	DFT of nth row of the histogram $h_I(U_P(n,r))$
$\overline{H(n_2,u)}$	complex conjugate of $H(n_2,u)$
$\lvert H(n,u)\rvert$	Fourier magnitude spectrum

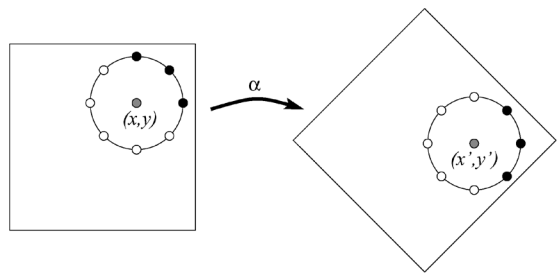

Fig. 4. Effect of image rotation on points in circular neighborhoods.

Now, if the neighborhood has P sampling points, n gets values from 0 to $P+1$, where $n = P+1$ is the special label marking all the nonuniform patterns. Furthermore, when $1 \le n \le P-1$, the rotation of the pattern is in range $0 \le r \le P-1$.

Let $I^{\alpha^\circ}(x,y)$ denote the rotation of image $I(x,y)$ by α degrees. Under this rotation, point (x,y) is rotated to location (x',y'). If we place a circular sampling neighborhood on points $I(x,y)$ and $I^{\alpha^\circ}(x',y')$, we observe that it also rotates by α° (see Fig. 4).

If the rotations are limited to integer multiples of the angle between two sampling points, i.e., $\alpha = a(360^\circ/P), a = 0,1,\ldots,P-1$, this rotates the sampling neighborhood exactly by a discrete steps. Therefore, the uniform pattern $U_P(n,r)$ at point (x,y) is replaced by the uniform pattern $U_P(n, r+a \bmod P)$ at point (x',y') of the rotated image.

Now, consider the uniform LBP histograms $h_I(U_P(n,r))$. The histogram value h_I at bin $U_P(n,r)$ is the number of occurrences of the uniform pattern $U_P(n,r)$ in image I.

If image I is rotated by $\alpha = a(360^\circ/P)$, based on the aforementioned reasoning, this rotation of the input image causes a cyclic shift in the histogram along each of the rows, i.e.,

$$h_{I^{\alpha^\circ}}\left(U_p(n, r+a \bmod P)\right) = h_I\left(U_p(n,r)\right). \tag{3}$$

For example, in the case of the eight-neighbor LBP, when the input image is rotated by 45°, the value from the histogram bin $U_8(1,0) = 00000001b$ moves to bin $U_8(1,1) = 00000010b$, the value from bin $U_8(1,1)$ to bin $U_8(1,2)$, etc.

Based on the property, which states that rotations induce shift in the polar representation (P,R) of the neighborhood, we propose a class of features that are invariant to the rotation of the input image, namely, such features computed along the input histogram rows, which are invariant to cyclic shifts.

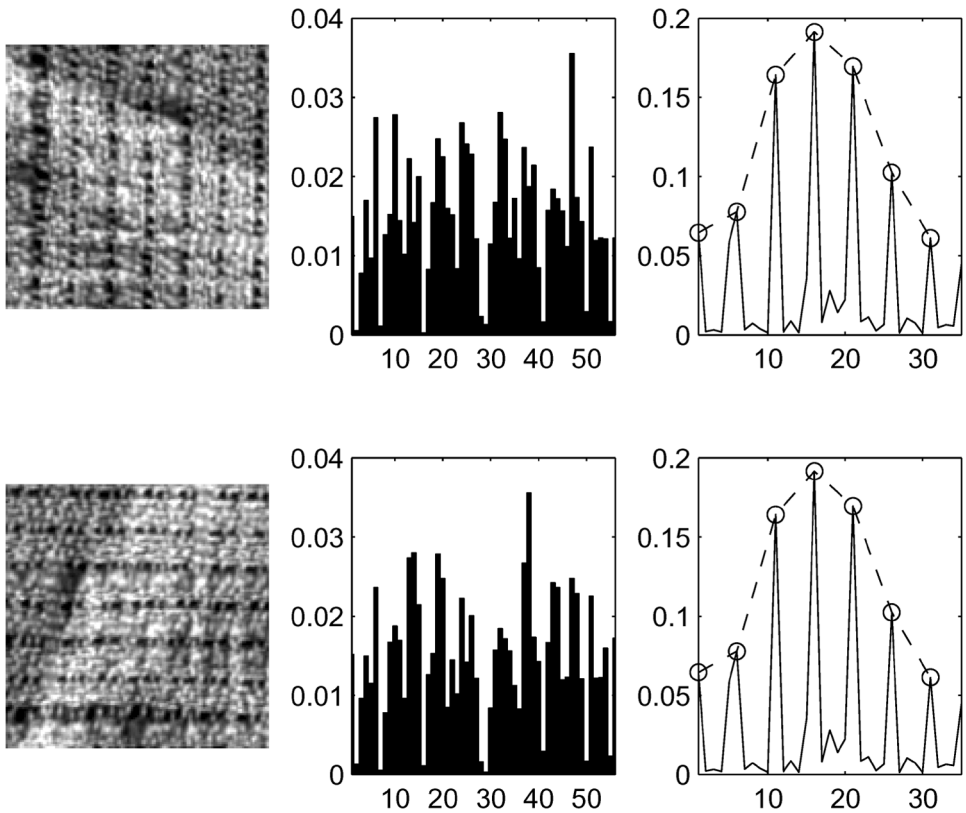

Fig. 5. First column: Texture image at orientations 0° and 90°. Second column: Bins 1–56 of the corresponding LBP^{u2} histograms. Third column: Rotation-invariant features (solid line) $|H(n,u)|$, $1 \le n \le 7$, $0 \le u \le 5$, and (circles, dashed line) LBP^{riu2}. Note that the LBP^{u2} histograms for the two images are markedly different, but the $|H(n,u)|$ features are nearly equal.

We use the discrete Fourier transform (DFT) to construct these features. Let $H(n,\cdot)$ be the DFT of the nth row of histogram $h_I(U_P(n,r))$, i.e.,

$$H(n,u) = \sum_{r=0}^{P-1} h_I\left(U_p(n,r)\right) e^{-i2\pi ur/P}. \tag{4}$$

Now, for DFT, it holds that a cyclic shift of the input vector causes a phase shift in the DFT coefficients. If $h'(U_P(n,r)) = h(U_P(n,r-a))$, then

$$H'(n,u) = H(n,u)e^{-i2\pi ua/P} \tag{5}$$

and therefore, with any $1 \le n_1, n_2 \le P-1$

$$\begin{aligned} H'(n_1,u)\overline{H'(n_2,u)} &= H(n_1,u)e^{-i2\pi ua/P}\overline{H(n_2,u)}e^{i2\pi ua/P} \\ &= H(n_1,u)\overline{H(n_2,u)} \end{aligned} \tag{6}$$

where $\overline{H(n_2,u)}$ denotes the complex conjugate of $H(n_2,u)$.

This shows that, with any $1 \le n_1, n_2 \le P-1$, and $0 \le u \le P-1$, the features, i.e.,

$$\text{LBP}^{u2}\text{-HF}(n_1,n_2,u) = H(n_1,u)\overline{H(n_2,u)} \tag{7}$$

are invariant to cyclic shifts of the rows of $h_I(U_P(n,r))$, and consequently, they are also invariant to rotations of the input image $I(x,y)$. The Fourier magnitude spectrum, which we call LBP-HF features, i.e.,

$$|H(n,u)| = \sqrt{H(n,u)\overline{H(n,u)}} \tag{8}$$

can be considered a special case of these features. Furthermore, it should be noted that the Fourier magnitude spectrum contains LBP^{riu2} features as a subset since

$$|H(n,0)| = \sum_{r=0}^{P-1} h_I(U_p(n,r)) = h_{\text{LBP}^{\text{riu2}}}(n). \tag{9}$$

An illustration of these features is in Fig. 5.

The LBP-HF can be thought as a general framework, in which $U_P(n,r)$ in (4) does not have to be the occurrence of that pattern, and instead, it can be any features corresponding to that uniform pattern. As long as the features are organized in the same way as uniform patterns so that they satisfy (3), they can be embedded into (4) to replace $U_P(n,r)$ and generate new rotation-invariant descriptors. One example is CLBP [15]. CLBP contains the sign-LBP (the sign of the difference of neighboring pixel against central pixel, i.e., it is equal to LBP) and magnitude-LBP (the magnitude of the difference of the neighboring pixel against the central pixel) components. Sign LBP can be

TABLE II
ABBREVIATION OF THE METHODS IN EXPERIMENTS AND THEIR CORRESPONDING MEANING

Abbreviation	Method
LBP^{u2}	Uniform sign LBP
LBP^{riu2}	Rotation invariant uniform sign LBP
LBP_M^{u2}[15]	Uniform magnitude LBP
LBP_M^{riu2}[15]	Rotation invariant uniform magnitude LBP
$LBP\_S\_M^{riu2}$	Concatenation of Rotation invariant uniform sign LBP and magnitude LBP
$LBP-HF$	Uniform LBP histogram Fourier
$LBPHF\_M$	Uniform magnitude LBP histogram Fourier
$LBPHF\_S\_M$	Concatenation of sign LBP histogram Fourier and magnitude LBP histogram Fourier

TABLE III
TEXTURE RECOGNITION RATES ON OUTEX_TC_00012 DATA SET

(P,R)	LBP^{u2}	LBP^{riu2}	$LBP-HF$	LBP_M^{u2}	LBP_M^{riu2}	$LBPHF\_M$	$LBP\_S\_M^{riu2}$	$LBPHF\_S\_M$
$(8,1)$	0.569	0.646	**0.741**	0.496	0.610	**0.622**	0.714	**0.786**
$(16,2)$	0.589	0.789	**0.903**	0.567	0.731	**0.856**	0.860	**0.940**
$(24,3)$	0.569	0.830	**0.924**	0.594	0.799	**0.874**	0.904	**0.949**

calculated using (1), and magnitude LBP can be obtained using the following equation [15]:

$$\text{LBP\_}M_{P,R}(x,y) = \sum_{p=0}^{P-1} s\left(|f(x,y) - f(x_p,y_p)| - c\right) 2^p \tag{10}$$

where c is a threshold to be adaptively determined. It can be set as the mean value of $|f(x,y) - f(x_p,y_p)|$ from the whole image.

Both two parts can be also organized into uniform sign-LBP (equal to the uniform LBP) and uniform magnitude-LBP components. We can embed these two parts to (4) separately, concatenate the produced histogram Fourier features, and obtain LBPHF_S_M.

III. EXPERIMENTS ON STATIC-TEXTURE CLASSIFICATION

For static textures, we carried out experiments on Outex_TC_00012 database [23] for rotation-invariant texture classification. Experiments with other databases are presented in [1].

The proposed rotation-invariant LBP-HF features were compared against noninvariant LBP^{u2} and the original rotation-invariant version LBP^{riu2}. To show the generalization of the LBP-HF, we also put LBPHF_M and LBPHF_S_M into comparison.

Table II lists the abbreviation of the methods used in the comparison and their corresponding meaning.

For a fair comparison, we used the chi-square metric since many previous works, e.g., [14] and [15], also used it, assigning a sample to the class of the model minimizing the L_{Chi} distance, i.e.,

$$L_{\text{Chi}}(h^S, h^M) = \sum_{b=1}^{B} \left(h^S(b) - h^M(b)\right)^2 / \left(h^S(b) + h^M(b)\right) \tag{11}$$

where $h^S(b)$ and $h^M(b)$ denote bin b of the sample and model features, respectively.

The implementation of LBP-HF features for MATLAB can be found in http://www.cse.oulu.fi/CMV/Downloads/LBP-Matlab. The feature vectors are of the following form:

$$\begin{aligned} fv_{\text{LBP}-\text{HF}} = [&|H(1,0)|, \ldots, |H(1,P/2)|, \\ &\ldots, |H(P-1,0)|, \ldots, |H(P-1,P/2)|, \\ &h\left(U_P(0,0)\right), h\left(U_P(P,0)\right), \\ &h\left(U_P(P+1,0)\right)]. \end{aligned}$$

We derived from the setup of [22] by using nearest-neighbor (NN) classifier instead of 3NN because no significant performance difference between the two was observed.

We evaluated our methods on the Outex database. Rotation variation is common in captured images. The Outex database is widely used to evaluate texture methods for dealing with rotation variations [13]–[15], [22].

We used the Outex_TC_00012 [23] test set intended for testing rotation-invariant texture classification methods. This test set consists of 9120 images representing 24 different textures imaged under different rotations and lightings. The test set contains 20 training images for each texture class. The training images are under single orientation, whereas different orientations are present in the total of 8640 testing images. We report here the total classification rates over all test images.

The results of the experiment are shown in Table III. As we can observe, rotation-invariant features provide better classification rates than noninvariant features (here, they are LBP^{u2} and LBP_M^{u2}). The performance of LBP-HF features and LBPHF_M is clearly higher than that of LBP^{u2} and LBP^{riu2}, and LBP_M^{u2}, and $\text{LBP\_M}^{\text{riu2}}$. When combining the LBP-HF and magnitude LBP together (LBPHF_S_M), much better results are obtained (i.e., 0.949 for LBPHF_S_M with 24 neighboring points and a radius of 3) than for all the other methods. By comparing the results of first column with the fourth column, the second column with the fifth column, and the third column with the sixth column, we can see that sign information usually plays more important roles than magnitude information, which is also consistent with the analysis in [15].

By varying (P, R), the multiresolution analysis can be utilized to get improved classification accuracy, e.g., LBPHF_S_M$_{16,2+24,3}$ can achieve a 96.2% accuracy. Similar improvement can be seen in [1] and [13]–[15]. Moreover, the proposed LBP-HF method can be applied to other features, such as LBPV [14] and the central pixel in CLBP [15], not just limited to sign and magnitude, as shown in the aforementioned experiments. As long as they are organized in the same way as uniform patterns so that they satisfy (3), they can be embedded into (4) to replace $U_P(n, r)$ and generate new rotation-invariant descriptors.

IV. Dynamic-Texture Recognition and LBP-TOP

In the previous sections, the LBP-HF features were constructed for static-image analysis and obtained very good results on static-texture classification. In the following sections, we will extend it to an appearance–motion (AM) description for dealing with rotation variation in video sequences. DT recognition is utilized as a case study. The recognition and the segmentation of DTs have attracted growing interest in recent years [5], [8], [24]. DTs provide a new tool for motion analysis. Now, the general assumptions used in motion analysis that the scene is Lambertian, rigid, and static can be relaxed [28].

Recently, two spatiotemporal operators based on LBPs [31] have been proposed for dynamic-texture description, i.e., volume LBPs (VLBP) and LBP histograms from three orthogonal planes (LBP-TOP), which are XY, XT, and YT planes. These operators combine motion and appearance together, and are robust to translation and illumination variations. They can be also extended to multiresolution analysis. A rotation-invariant version of the VLBP has been also proposed, providing a promising performance for DT sequences with rotations [32]. However, the VLBP considers cooccurrences of neighboring points in subsequent frames of a volume at the same time, which makes its feature vector too long when the number of neighboring points used is increased. The LBP-TOP does not have this limitation. It has performed very well in different types of computer vision problems, such as dynamic-texture recognition [31], segmentation [6] and synthesis [12], facial-expression recognition [31], visual speech recognition [33], activity recognition [17], [20], [21], and analysis of facial paralysis [16]. However, LBP-TOP is not rotation invariant, which limits its wide applicability.

Fig. 6(a)–(c) (left) shows one image in the XY plane, the XT plane, which gives the visual impression of one row changing in time, and the motion of one column in the temporal space, respectively. For each pixel in images from these three planes or slices, a binary code is produced by thresholding its neighborhood in a circle or an ellipse from XY, XT, and YT slices independently with the value of the center pixel. A histogram is created to collect up the occurrences of different binary patterns from three slices, which are denoted as XY-LBP, XT-LBP, and YT-LBP, and then concatenated into a single histogram, as demonstrated in last row of Fig. 6 (left). In such a representation, the DT is encoded by the LBP, while the appearance and the motion in two directions of the DT are considered, incorporating spatial-domain information and two spatiotemporal cooccurrence statistics together. For LBP-TOP,

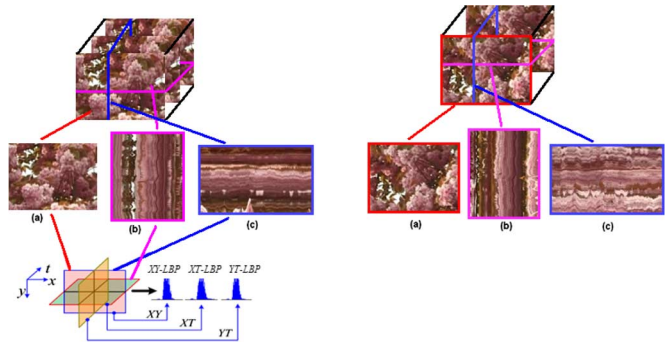

Fig. 6. Computation of LBP-TOP for "watergrass" with (left) 0° and (right) 60° rotation.

the radii in axes X, Y, and T, and the number of neighboring points in the XY, XT, and YT planes or slices can be also different, which can be marked as R_X, R_Y, and R_T, and P_{XY}, P_{XT}, and P_{YT}. The corresponding LBP-TOP feature is denoted as LBP-TOP$_{P_{XY},P_{XT},P_{YT},R_X,R_Y,R_T}$. Sometimes, the radii in three axes are same and so do the number of neighboring points in XY, XT, and YT planes. In that case, we use LBP-TOP$_{P,R}$ for the abbreviation, where $P = P_{XY} = P_{XT} = P_{YT}$ and $R = R_X = R_Y = R_T$.

In this way, a description of the DT is effectively obtained based on the LBP from three different planes. The labels from the XY plane contain information about the appearance, and in the labels from the XT and YT planes, the cooccurrence statistics of motion in horizontal and vertical directions are included. These three histograms are concatenated to build a global description of the DT with the spatial and temporal features. However, the AM planes XT and YT in LBP-TOP are not rotation invariant, which makes LBP-TOP hard to handle the rotation variations. This needs to be addressed for the DT description. As shown in Fig. 6 (right), the input video in the top row is with 60° rotation from that in Fig. 6 (left); thus, the XY, XT, and YT planes in the middle row are different from that of Fig. 6 (left), which obviously makes the computed LBP codes different from each other. Even if we sample the texture information in 8 or 16 planar orientations, the orders of these planes would not change with the rotation of images.

On the basis of LBP-TOP, we propose two rotation-invariant descriptors for LBP-TOP, based on using the DFT for rotation-invariant DT recognition. One is computing the 1-D histogram Fourier transform for the uniform patterns along all the rotated motion planes. The other one is computing the 2-D Fourier transform for the patterns with the same number of "1's" along its rotation in bins and along all the rotated motion planes as well, which avoids using the redundant information and makes the computation complexity much lower than the first one. We compare them with earlier methods in the classification of rotated DT sequences. The robustness of the descriptors on viewpoint variations is also studied using a recently introduced test set for view-invariant dynamic-texture recognition [24].

V. Rotation-Invariant LBP-TOP

Both LBP-XT and LBP-YT describe the appearance and the motion. When a video sequence rotates, these two planes

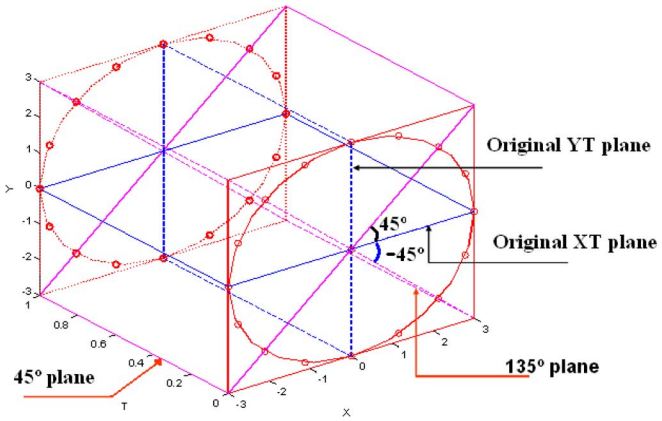

Fig. 7. Rotated planes from which the LBP is computed.

do not accordingly rotate, which makes the LBP-TOP operator not rotation invariant. Moreover, rotation only happens around the axis parallel to the T axis; thus, considering the rotation-invariant descriptor inside planes does not make any sense. Instead, we should consider the rotations of the planes, not only just the orthogonal planes (XT and YT rotations of 90°) but also the planes with different rotation angles, such as the purple planes with rotations of 45° and 135° in Fig. 7. Thus, the AM planes consist of P_{XY} rotation planes around the T axis. The radius in X and Y should be same but can be different from that in T. Only two types for the number of neighboring points are included; one is P_{XY}, which determines how many rotated planes will be considered, and the other one is P_T, which is the number of neighboring points in AM planes. The original XT and YT are not two separate planes anymore; instead, they are AM planes obtained by rotating the same plane 0° and 90°, respectively.

The corresponding feature is denoted as $\text{LBP-TOP}^{\text{ri}}_{P_{XY},P_T,R_{XY},R_T}$. Suppose the coordinates of the center pixel $g_{t_c,c}$ are (x_c, y_c, t_c), we compute the LBP from P_{XY} spatiotemporal planes. The coordinates of the neighboring points $g_{d,p}$ sampled from the ellipse in the XYT space with $g_{t_c,c}$ as center and R_{XY} and R_T as the length of axes are given by $(x_c + R_{XY}\cos(2\pi d/P_{XY})\cos(2\pi p/P_T), y_c - R_{XY}\sin(2\pi d/P_{XY})\cos(2\pi p/P_T), t_c + R_T\sin(2\pi p/P_T))$, where $d(d = 0,\ldots,(P_{XY}-1))$ is the index of the AM plane and $p(p = (0,\ldots,(P_T-1))$ represents the label of neighboring point in plane d.

A. One-Dimensional HFLBP-TOP

After extracting the uniform LBP for all the rotated planes, we compute the Fourier transform for every uniform pattern along all the rotated planes. Fig. 8 demonstrates the computation. For $P_T = 8, 59$ uniform patterns can be obtained, as shown in the left column. For all the rotated $P_{XY} = 8$ planes, the DFT is applied for every pattern along all planes to produce the frequency features. Equation (12) illustrates the following computation:

$$H_1(n,u) = \sum_{d=0}^{P_{XY}-1} h_1(d,n)e^{-i2\pi ud/P_{XY}} \qquad (12)$$

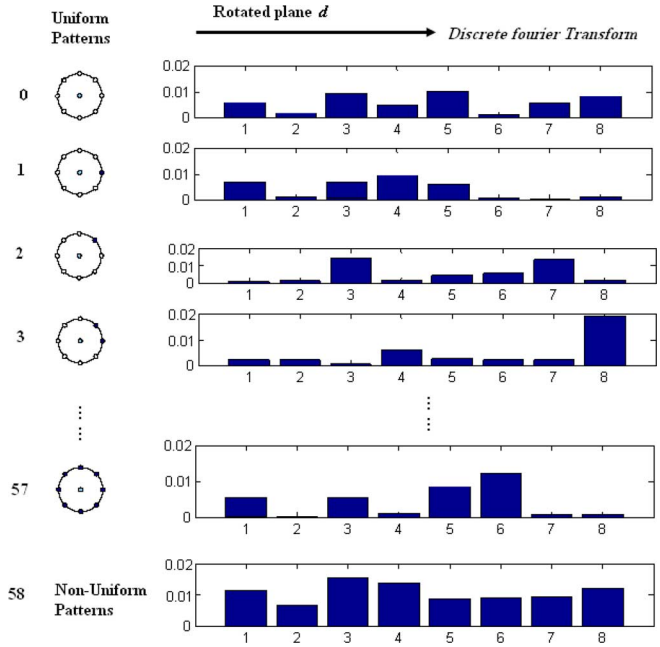

Fig. 8. LBP histograms for uniform patterns in different rotated motion planes with $P_{XY} = 8$ and $P_T = 8$.

where n is the index of uniform patterns $[(0,\ldots,N),\ N = 58$ for $P_T = 8]$ and u $(u = 0,\ldots,P_{XY}-1)$ is the frequency. $d\,[d = 0,\ldots,(P_{XY}-1)]$ is the index of rotation degrees around the line passing through the current central pixel $g_{t_c,c}$ and parallel to the T axis. $h_1(d,n)$ is the value of pattern n in the uniform LBP histogram at plane d. To get the low frequencies, u can use the value from 0 to $(P_{XY}/s+1)$ (e.g., $s = 4$ in experiments).

When $u = 0$, $H_1(n,0)$ means the sum of pattern n through all the rotated motion planes, which can be thought as another kind of rotation-invariant descriptor of simply summing the histograms from all the rotated planes. Since it uses 1-D histogram Fourier transform for LBP-TOP, we call it the 1-D **histogram Fourier LBP-TOP (1DHFLBP-TOP)**.

The feature vector $V1$ of 1DHFLBP-TOP is of the following form:

$$V1 = [|H_1(0,0)|,\ldots,|H_1(0,P_{XY}/s+1)|,\ldots, |H_1(N-1,0)|,\cdots,|H_1(N-1,P_{XY}/s+1)|].$$

N is the number of uniform patterns with neighboring points P_{XY}. Here, s is the segments of frequencies. Not all the P_{XY} frequencies are used. Instead, only the low frequencies, saying $[0\ P_{XY}/s+1]$, are utilized. The total length of $V1$ is $LRI1 = N \times (P_{XY}/s+2)$.

B. Two-Dimensional HFLBP-TOP

We can notice that, for descriptor 1DHFLBP-TOP, the LBPs from a plane rotated g degrees $(180 > g >= 0)$ and $g+180$ are mirrored along the T-axis through the central point, as line cg in Fig. 9, but they are not same. Thus, to get the rotation-invariant descriptor, all the rotated planes should be used, which increases the computational load.

However, for planes rotated g degrees and $g + 180°$, the information is same; thus, there is no need to use both of them.

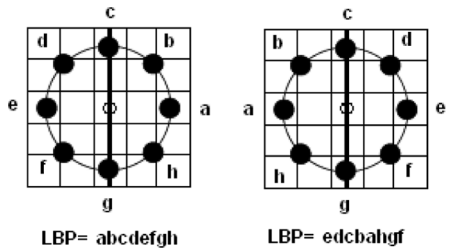

Fig. 9. LBP from plane with (left) g degrees and (right) $g + 180°$. They are mirrored.

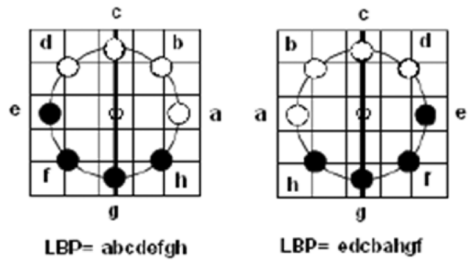

Fig. 10. Uniform pattern (left) before and (right) after mirror.

We notice this in the left and right images of Fig. 9; although the LBP codes $(a, b, c, d, e, f, g, h = 0 \text{ or } 1)$ are mirrored, the neighboring relationship still remains, e.g., d is adjacent to c and e in both images.

According to the definition of uniform patterns, "an LBP is called uniform if the binary pattern contains at most two bitwise transitions from 0 to 1 or vice versa when the bit pattern is considered circular": 1) If one LBP code L is uniform and there is zero bitwise transition from 0 to 1 or 1 to 0, it means that all the bits in this LBP are 0 or 1. Thus, after mirror, for the produced LBP code L', all the bits are still 0 or 1, which is still uniform. 2) If L is uniform and there are two bitwise transitions from 0 to 1 or 1 to 0, as shown in Fig. 10, the transitions happen between d and e, and h and a [see Fig. 10 (left)]. After mirror, the neighboring relationship is unchanged; thus, the transitions are also between e and d, and a and h [see Fig. 10 (right)], and the transition times are still two, which means that the mirrored LBP is also uniform. 3) If L is nonuniform, we first assume that, after mirror, L' is uniform. We can then mirror L' again, and the obtained L'' should be equal to L. However, according to statements 1 and 2, if L' is uniform, the mirrored L'' is also uniform. However, L is nonuniform, which means $L'' <>$. It is self-contradictory. Thus, if L is nonuniform, after mirror, the obtained L' is also nonuniform. Because, in the mirror transformation, only the location of the bits changes. The value of all bits keeps the same; thus, the number of 1's is unchanged whether L is uniform or nonuniform.

Thus, the uniform patterns in a plane rotated g degrees are still uniform in a plane with $g + 180°$ and with the same number of 1's. The nonuniform patterns are still nonuniform in both planes.

We propose to make the uniform patterns with same number of 1's into one group, but they can rotate $r(r = 0, \ldots, (P_T - 1))$ bits, as shown in the left column of Fig. 11. In total, we have $P_{XY} - 1$ groups with the 1's numbered as $1, 2, 3, \ldots, P_{XY} - 1$. For the uniform patterns with 1's numbered as 0 (all zeros), P_{XY}

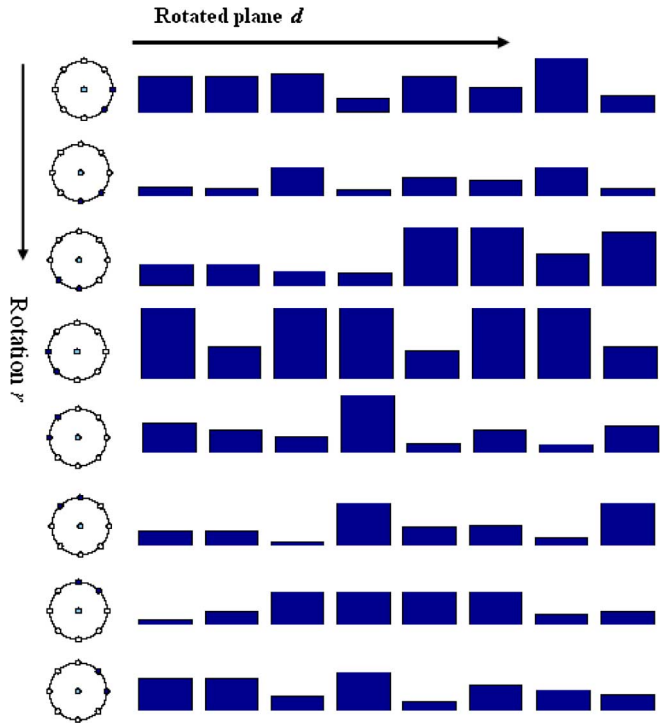

Fig. 11. Examples of the LBP with number of 1's two in different rotated motion planes with $P_{XY} = 8$ and $P_T = 8$.

(all ones), and nonuniform patterns, no matter how video sequences rotate, they remain the same. For every group, the 2-D Fourier transform is used to get the rotation-invariant descriptor, as shown in Fig. 11.

Equation (13) illustates the following computation:

$$H_2(m, u, v) = \sum_{d=0}^{P_{XY}-1} \sum_{r=0}^{P_T-1} h_2\left(U(m, r), d\right) e^{-i2\pi ud/P_{XY}} e^{-i2\pi vr/P_T} \quad (13)$$

where m $(m = 1, \ldots, P_{XY} - 1)$ is the number of 1's and u $(u = 0, \ldots, P_{XY} - 1)$ and v $(0, \ldots, P_T - 1)$ are frequencies in two directions. d is the index of the rotation degree around the T axis; r is the rotation inside the circle or the ellipse with P_T neighboring points. $U(m, r)$ is the uniform pattern with the m value of 1's, and r is its rotation index. $h_2(U(m, r), d)$ is the number of occurrences of $U(m, r)$ at plane d. Thus, $H_2(0, 0, 0)$ is the sum of all zeros in all planes, $H_2(P_{XY}, 0, 0)$ is the sum of all ones, and $H_2(P_{XY} + 1, 0, 0)$ is the sum of all nonuniform patterns in all planes, which can be used with $H_2(m, u, v)$ together to describe the DT. The difference of this descriptor from the first one is that it computes the histogram Fourier transforms from two directions, considering the frequencies not only from planes but also from pattern rotations. We call it 2-D **histogram Fourier LBP-TOP (2DHFLBP-TOP)**.

The final feature vector $V2$ is of the following form:

$$\begin{aligned} V2 = [&|H_2(1, 0, 0)|, \ldots, |H_2(1, P_{XY}/s + 1, P_T/s + 1)|, \ldots, \\ &|H_2(P_{XY} - 1, 0, 0)|, \ldots, \\ &|H_2(P_{XY} - 1, P_{XY}/s + 1, P_T/s + 1)|, \\ &|H_2(0, 0, 0)|, |H_2(P_{XY}, 0, 0)|, |H_2(P_{XY} + 1, 0, 0)|]. \end{aligned}$$

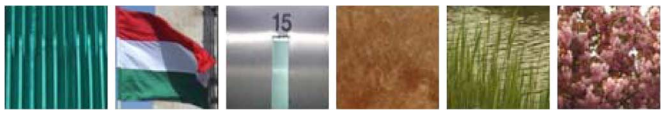

Fig. 12. DynTex database (http://projects.cwi.nl/dyntex/).

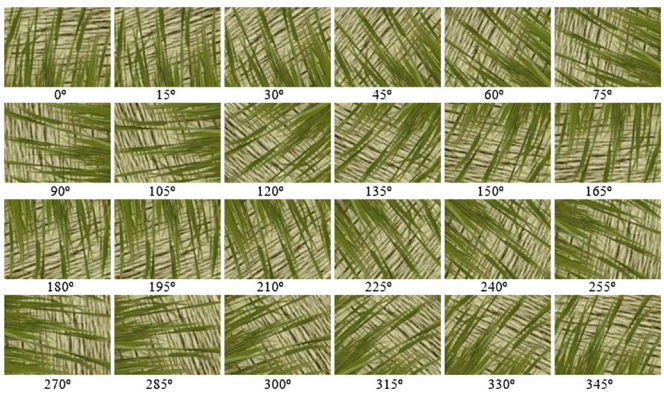

Fig. 13. Images after rotating by 15° intervals.

The total length of $V2$ is $LRI2 = (P_{XY} - 1) \times (P_{XY}/s + 2) \times (P_T/s + 2) + 3$.

VI. Experiments on Dynamic-Texture Classification

The performance of the proposed rotation-invariant video descriptors was tested on dynamic-texture classification. We carry out experiments on the DynTex database for rotation variation evaluation and the data set from [25] for view variation evaluation.

A. Experiments on Rotation Variations

DynTex, a large and varied database of DTs, which originally included 35 DTs and has been now extended to have 656 DTs, was selected for the experiments. Fig. 12 shows example DTs from this data set.

To evaluate the rotation invariance of the methods and compare with previous methods [32], we use the same setup as in [32]. The data set used in experiments includes 35 classes from the original database, and each sequence was rotated by 15° intervals, as shown in Fig. 13, obtaining 24 sequences. Every sequence was cut in length into two sequences. Thus, totally, we have 48 samples for each class. In our experiments, two sequences with 0° (no rotation) are used as training samples, and the remaining ones are test sequences. Hence, in this suite, there are 70 (35×2) training models and 1610 (35×46) testing samples.

The mean values of the rotation-invariant LBP-TOP features of the two samples without rotation are computed as the feature for the class. The testing samples are classified or verified according to their difference with respect to the class using the k NN method ($k = 1$).

Table IV demonstrates the results with different parameters of the proposed descriptors for all rotation tests. Here, we show results for both the L1 distance metric and the chi-square distance metric, and it can be seen that the L1 distance provides similar or better results than chi-square. Thus, in the following experiments of KNN classification, the reported results are for the L1 distance measurement. The first two rows are the results using the original LBP-TOP with four neighboring points. The results are very poor. Even when more neighboring points are used, such as 8 or 16, as shown in the 6th, 7th, 11th, and 12th rows, the classification rates are still less than 60%, which demonstrates that LBP-TOP is not rotation invariant. We also did the experiments using oversampled LBP-TOP, which is denoted as LBP-TOP$_{P_{XY},P_T,R_{XY},R_T}$, i.e., we extend the fixed three planes in LBP-TOP to P_{XY} planes in the spatiotemporal domain and sample the neighboring points in each plane with radii R_{XY} and R_T in the XY and T directions, respectively, for P_T points in ellipse. The uniform histograms from each plane are then concatenated together as the oversampled LBP-TOP features. The results are shown in the 3rd, 8th, and 13th rows of Table IV. It can be seen that the oversampled LBP-TOP got better results than the original LBP-TOP because oversampling can include more information. However, the accuracy is lower than that for the proposed rotation-invariant descriptors, which shows that oversampling cannot deal with rotation variations. Although it contains more information, the order of planes keeps unchanged when there are rotations that make oversampling not rotation invariant. When using four neighboring points, 2DHFLBP-TOP$_{4,4,1,1}$ obtained 82.11% with only 30 features. When using 16 neighboring points in AM planes, 1DHFLBP-TOP$_{16,16,2,2}$ and 2DHFLBP-TOP$_{16,16,2,2}$ got 98.57% and 97.33%, respectively. Both descriptors are effective for dealing with rotation variations. One may have noticed that the proposed 2DHFLBP is better than 1DHFLBP when the number of neighboring pixels is 4 but worse than 1DHFLBP for a larger number of these pixels. That is because 2DHFLBP considers only the number of 1's for the uniform patterns, i.e., in that way, only half of the rotated planes need to be used, whereas 1DHFLBP considers the uniform patterns for all planes. When the number of neighboring pixels increases, there is more information missing for 2DHFLBP compared with 1DHFLBP. However, 2DHFLBP-TOP gets a comparative accuracy to the 1DHFLBP-TOP with a much shorter feature vector and only using half of the rotated planes, which will save computation time. It is a good compromise for computational efficiency and recognition accuracy, which could make it very useful for many applications.

Fig. 14 shows the 1DHFLBP-TOP histograms and the 2DHFLBP-TOP histograms for the "square sheet" dynamic texture with 48 rotation samples. It can be seen that both descriptors have good characteristics for rotation variations.

The magnitude of LBP-HF is also extended to video description and utilized as supplemental information to the proposed rotation-invariant descriptors. The mean difference of the neighboring point's grayscale against the central pixel in each spatiotemporal plane is calculated and utilized as the threshold for getting the binary code. Results of 1DHFLBP-TOP and 2DHFLBP-TOP, which are actually the sign LBP-TOP histogram Fourier 1DHFLBP-TOP_M and 2DHFLBP-TOP_M, which are the magnitude LBP-TOP histogram Fourier, and their combination on DynTex database are demonstrated in Table V. As we can see, the magnitude information does not work as well as the sign information, which is consistent with the conclusion in [15]. The combination of the magnitude

TABLE IV
RECOGNITION RESULTS USING DIFFERENT PARAMETERS. ($u2$ DENOTES UNIFORM PATTERNS)

Features	Length	L1 Results(%)	Chi-square Results(%)
$LBP-TOP_{4,1}$	48	43.11	39.81
$LBP-TOP^{u2}_{4,1}$	45	42.98	39.32
$LBP-TOP^{u2}_{4,4,1,1}$	60	51.30	48.39
$1DHFLBP-TOP_{4,4,1,1}$	45	**79.50**	71.99
$2DHFLBP-TOP_{4,4,1,1}$	30	**82.11**	71.93
$LBP-TOP_{8,2}$	256×3	58.76	51.86
$LBP-TOP^{u2}_{8,2}$	59×3	58.63	52.86
$LBP-TOP^{u2}_{8,8,2,2}$	59×8	69.25	65.71
$1DHFLBP-TOP_{8,8,2,2}$	236	**98.07**	97.83
$2DHFLBP-TOP_{8,8,2,2}$	115	**94.41**	94.10
$LBP-TOP^{u2}_{16,1}$	243×3	52.86	47.58
$LBP-TOP^{u2}_{16,2}$	243×3	56.71	51.12
$LBP-TOP^{u2}_{16,16,2,2}$	243×16	72.05	68.32
$1DHFLBP-TOP_{16,16,2,2}$	1458	**98.57**	99.69
$2DHFLBP-TOP_{16,16,2,2}$	543	**97.33**	97.76

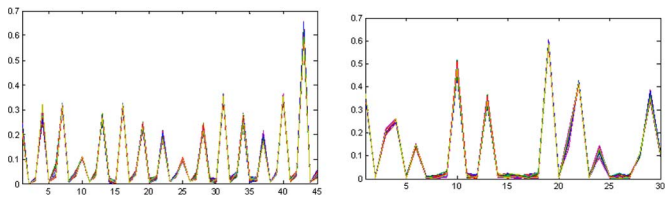

Fig. 14. (Left) 1DHFLBP-TOP histograms and (right) 2DHFLBP-TOP histograms of DT "square sheet" with 48 rotation samples.

TABLE V
RECOGNITION RESULTS USING ROTATION-INVARIANT SIGN HFLBP-TOP AND MAGNITUDE HFLBP-TOP

Features	Length	Results (%)
$1DHFLBP-TOP_{4,4,1,1}$	45	79.50
$1DHFLBP-TOP_{4,4,1,1}\_M$	45	63.60
$1DHFLBP-TOP_{4,4,1,1}\_S\_M$	45×2	**84.22**
$2DHFLBP-TOP_{4,4,1,1}$	30	82.11
$2DHFLBP-TOP_{4,4,1,1}\_M$	30	61.86
$2DHFLBP-TOP_{4,4,1,1}\_S\_M$	30×2	**86.15**
$1DHFLBP-TOP_{8,8,2,2}$	236	98.07
$1DHFLBP-TOP_{8,8,2,2}\_M$	236	86.71
$1DHFLBP-TOP_{8,8,2,2}\_S\_M$	236×2	**98.45**
$2DHFLBP-TOP_{8,8,2,2}$	115	94.41
$2DHFLBP-TOP_{8,8,2,2}\_M$	115	88.32
$2DHFLBP-TOP_{8,8,2,2}\_S\_M$	115×2	**96.27**

information together with the sign information yields much better result than using either of them solely. Particularly when the number of neighboring points and planes are fewer, as for 1DHFLBP-TOP$_{4,4,1,1}$ and 2DHFLBP-TOP$_{4,4,1,1}$, the improvement is significant, i.e., from 79.50% to 84.22% and from 82.11% to 86.15%, respectively.

Table VI lists the accuracies using different methods. The first three rows give the results using the LBP histogram Fourier transform [1], which only considers the rotation invariance in appearance. The accuracy of 40%–60% shows its ineffectiveness for rotations happening on videos. The middle four rows show the results using different versions of rotation-invariant VLBP [32], which can get quite good results using short feature vectors, e.g., 87.20% with only 26 features. However, because VLBP considers the cooccurrence of all the neighboring points in three frames of a volume at the same time, it is hard to be

TABLE VI
RESULTS USING DIFFERENT METHODS

Features	Length	Results (%)
$HFLBP_{4,1}$	12	43.60
$HFLBP_{8,2}$	31	50.37
$HFLBP_{16,2}$	93	61.93
ri #1 $VLBP_{1,4,1}$	864	75.34
ri #2 $VLBP_{2,4,1}$	4176	79.38
riu2 #2 $VLBP_{1,4,1}$	16	78.07
riunu2 #2 $VLBP_{1,4,1}$	26	87.20
$1DHFLBP-TOP_{16,16,2,2}$	1458	**98.57**
$2DHFLBP-TOP_{16,16,2,2}$	543	**97.33**

extended to use more neighboring information. Four neighboring points is almost the maximum as concluded in [31]. Comparatively, the proposed two rotation-invariant LBP-TOP descriptors inherit the advantage of the original LBP-TOP, they can be easily extended to use many more neighboring points, e.g., 16 or 24, and they obtained almost 100% accuracy DT recognition with rotations. For comparing with LBP-TOP on DTs without rotation, we carried out experiments on the DynTex database. A 92.9% accuracy is obtained with 1DHFLBP-TOP, whereas LBP-TOP obtained 93.4% accuracy with four neighboring points. We can see that, on videos without rotation, the proposed rotation-invariant descriptors give similar or slightly worse results than noninvariant LBP-TOP. More importantly, they clearly outperform LBP-TOP when there are rotations. We also computed the maximal of uniform patterns along all the rotated planes as a simple rotation-invariant descriptor for comparison. We got 70.93% (versus 82.11% for the proposed method) when using four neighboring points and a radius of 1, and 91.06% (versus 98.07%) when using eight neighboring points and a radius of 2. Thus, with this simple normalization, we can get some rotation invariance, but it works much poorer than proposed methods.

B. Experiments on View Variations

View variations are very common in DTs. The view-invariant recognition of DTs is a very challenging task. To our best knowledge, most of the proposed methods for dynamic-texture categorization validated their performance on the ordinary DT databases, without viewpoint changes, except [24] and [29]. Woolfe and Fitzgibbon addressed shift invariance [29] and Ravichandran *et al.* [24] proposed to use bag of system as the representation for dealing with viewpoint changes.

To evaluate the robustness of our proposed descriptors to view variations, we use the same data set to [24] and [25]. This data set consists of 50 classes of four video sequences each. Many previous works [4], [25] are based on the 50 class structure, and the reported results are not on the entire video sequences but on a manually extracted patch of size 48×48 [24]. In [24], the authors combine the sequences that are taken from different viewpoints and reduce the data set to a nine-class data set with the classes being boiling water (8), fire (8), flowers (12), fountains (20), plants (108), sea (12), smoke (4), water (12), and waterfall (16). The numbers in parentheses represent the number of sequences in the data set. To compare the performance of handling view variations with the methods proposed in [24] and [25], we use the following same experimental setup:

Fig. 15. Sample images from reorganized data set from [25].

1) The plant class is left out since the number of sequences of plants far outnumbered the number of sequences for the other classes; thus, the remaining eight classes are used in our experiments; 2) Four different scenarios are explored in this paper. The first set is an easy two-class problem, namely, the case of water versus fountain. The second one is a more challenging two-class problem, namely, fountain versus waterfall. The third set of experiments is on a four-class (water, fountain, waterfall, and sea) problem, and the last set is on the reorganized database with eight classes. We abbreviate these scenarios as W–F (water vs. fountain), F–WF (fountain vs. waterfall), FC (four classes), and EC (eight classes). Sample frames from the video sequences in the database are shown in Fig. 15. For every scenario, we train using 50% of the data and test using the rest.

We utilize the support vector machine (SVM) and the NN of the L1 distance (1NN) as classifiers. For the SVM, the second-degree-polynomial kernel function is used in the experiments. Fig. 16 compares our results using 1DHFLBP-TOP$_{8,8,1,1}$ and 2DHFLBP-TOP$_{8,8,1,1}$, and the results with term frequency (TF) and soft weighting (SW) from [24], and DK (DS) from [25]. TF and SW are the two kinds of representation utilized in [24] to represent the videos using the codebook. DK and DS depict the methods proposed in [25], in which a single LDS is used to model the whole video sequence and the NN and SVM classifiers are used for categorization, respectively. Fig. 16 shows the results for four scenarios using SVM and 1NN as classifiers. In all four experimental setups, the second (F–WF) and last (EC) setups are considered more challenging because, in the second scenario, the viewpoints in testing are not used in the training, which makes them totally novel. In the last scenario, we have a total of 92 video sequences with a varying number of sequences per class, and they are from various viewpoints. It is shown in Fig. 16 that our proposed descriptors obtain leading accuracy for all four scenarios. Particularly for more challenging fountain versus waterfall and all eight class problems, our results from 1DHFLBP-TOP$_{8,8,1,1}$ with the SVM are 87.5% and 86.96%, and with 1NN are 87.5% and 73.91%, respectively. These are much better than for TF (70% and 63%) and SW (76% and 80%) with the SVM [24], and DK (50% and 52%) and DS (56% and 47%) [24]. In addition, we also carried out the experiments using LBP-TOP without rotation-invariant characteristics. LBP-TOP$^{u2}_{8,1}$ obtained 75% and 71.74% with the SVM for more difficult F–WF and EC scenarios, which is inferior to either of the proposed descriptors. Rotation-invariant VLBP [32] was also implemented for comparison. For F–WF and EC scenarios, 78.15% and 80.43% are achieved with the SVM classifier, which are better than those from LBP-TOP but worse than those from the proposed descriptors. From these results and from comparison on the database with real 3-D viewpoint changes, it can be seen that, although our descriptors are not designed as view invariant, they can deal with this problem very effectively.

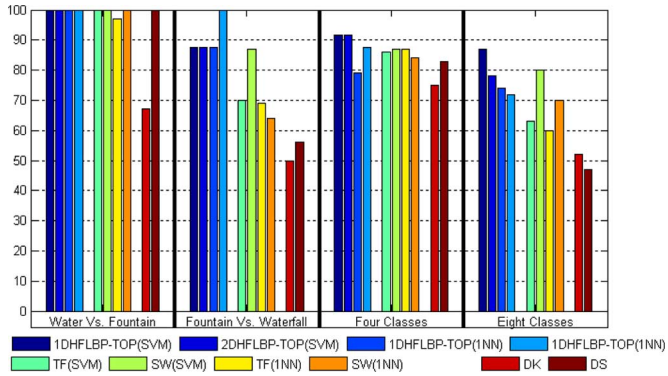

Fig. 16. Classification results for the four scenarios with different methods.

TABLE VII
RECOGNITION RESULTS ON EC PROBLEM

Features	Length	SVM (%)	NN (%)
$1DHFLBP-TOP_{4,4,1,1}$	45	65.22	59.78
$1DHFLBP-TOP_{4,4,1,1}\_S\_M$	$90(45\times 2)$	**73.91**	**77.17**
$2DHFLBP-TOP_{4,4,1,1}$	30	58.70	59.78
$2DHFLBP-TOP_{4,4,1,1}\_S\_M$	$60(30\times 2)$	**71.74**	**73.91**
$LBP-TOP^{u2}_{4,4,1,1}$	$60(15\times 4)$	**70.65**	**61.96**
$1DHFLBP-TOP_{8,8,1,1}$	236	86.96	73.91
$1DHFLBP-TOP_{8,8,1,1}\_S\_M$	$472(236\times 2)$	**86.96**	**78.26**
$2DHFLBP-TOP_{8,8,1,1}$	115	78.26	71.74
$2DHFLBP-TOP_{8,8,1,1}\_S\_M$	$230(115\times 2)$	**79.35**	**82.61**
$LBP-TOP^{u2}_{8,8,1,1}$	$472(59\times 8)$	**85.87**	**73.91**
$1DHFLBP-TOP_{16,16,2,2}$	1458	71.74	65.22
$1DHFLBP-TOP_{16,16,2,2}\_S\_M$	$2916(1458\times 2)$	**85.87**	**67.39**
$2DHFLBP-TOP_{16,16,2,2}$	543	68.48	54.35
$2DHFLBP-TOP_{16,16,2,2}\_S\_M$	$1086(543\times 2)$	**79.35**	**71.74**
$LBP-TOP^{u2}_{16,16,2,2}$	$3888(243\times 16)$	**76.09**	**67.39**

Table VII lists the results using sign LBP-TOP histogram Fourier, the combination of magnitude LBP-TOP histogram Fourier and sign LBP-TOP histogram Fourier, and oversampled LBP-TOP features for the most difficult EC problem. In this table, we can see that, when combining sign and magnitude components together, we achieve higher accuracy than using sign alone. In this database, the in-plane rotation is not the main problem but view variation. Although view changes can cause appearance differences with translation, rotation, and scaling, the local transition in appearance and motion would not change much. With oversampling, it can catch much more information about local appearance and motion transition than the original LBP-TOP providing comparable results to the proposed rotation-invariant LBP-TOP histogram Fourier descriptors but with much longer feature vectors when the number of neighboring points increases to be over eight.

The NN is one of the easiest machine learning algorithms. In the classification, it is only determined by the sample closest to the test sample. Thus, when there is big overlapping in classes, the NN works pretty well. However, it is prone to overfitting because of the highly nonlinear nature. The SVM classifier is well founded in the statistical learning theory and has been successfully applied to various object detection tasks in the computer vision. SVM finds a separating hyperplane with the maximal

margin to separate the training data in feature space. The classification of the SVM is determined by the support vectors; thus, it is usually more robust than the NN. As noticed in Table VII, when the number of planes is small, such as HFLBP-TOP$_{4,4,1,1}$, the features are not very discriminative; thus, the overlapping of classes is big, and the NN got similar or even higher results than the SVM. However, in most cases, the SVM obtained better results than the NN. In addition, the computational complexity of the SVM depends on the number of support vectors and not the dimension of feature vectors. When the dimension is very high, the SVM is computationally less expensive. However, the SVM is designed for two-class problems. If there are more than two classes, some strategy, such as reorganizing a multiclass problem into multiple two-class problems, needs to be employed. Therefore, it is generally easier to deal with multiple-class problems with the KNN than the SVM.

VII. Discussion and Conclusion

In this paper, we have proposed rotation-invariant image and video descriptors based on computing the DFT of LBP or LBP-TOP histograms, respectively.

The proposed LBP-HF features differ from the previous versions of rotation-invariant LBP since the LBP-HF features are computed from the histogram representing the whole region, i.e., the invariants are constructed from the histogram of noninvariant LBPs instead of computing invariant features independently at each pixel location.

This approach can be generalized to embed any features in the same organization of uniform patterns, such as the LBPHF_S_M shown in our experiments. By embedding different features into the LBP-HF framework, we are able to generate different rotation-invariant features and combine supplementary information together to improve the description power of the proposed method. The use of the complementary magnitude of the LBP will further improve the classification accuracy. In our experiments, LBPHF_S_M descriptors have been shown to outperform the original rotation-invariant LBP, LBP_M, LBP-HF, and LBP_S_M features for the Outex database.

Moreover, two rotation-invariant descriptors based on LBP-TOP have been developed. One is computing the 1-D histogram Fourier transform for the uniform patterns along all the rotated motion planes. The other one is computing the 2-D Fourier transform for the patterns with the same number of 1's along its rotation in bins and along all the rotated motion planes as well. Experiments on rotated DT sequences show that both descriptors achieve very promising results in recognizing video sequences with rotations. Another experiments on DTs captured from different views provided better results than the state of the art, proving the effectiveness of our approach for dealing with view variations. Thus, it can be seen that the proposed method not only works on in-plane rotations, but it can also deal with out-of-plane rotations robustly, obtaining much better results than previous methods. Moreover, the second rotation-invariant descriptors consider the mirror relations for the patterns produced from planes rotated g degrees and $g + 180^\circ$. This reduces its computational complexity to the half of the first descriptor because it only needs to compute the LBP histograms from the planes with rotation degrees less than 180°. The proposed descriptors keep the advantages of LBP-TOP, such as robustness to illumination changes and ability to describe DTs at multiple resolutions, as well as having the capability of handling rotation and view variations. They can be also extended to embed other features, such as the magnitude of LBP-TOP shown in our experiments in Section VI, which can be solely utilized or combined with the sign of LBP-TOP as supplemental information to help improve the classification accuracy. They will widen the applicability of LBP-TOP to such tasks in which the rotation invariance or the view invariance of the features is important. For example, in activity recognition, there exist rotations due to rotation of cameras or the activity itself (e.g., some hand gestures), and we believe rotation-invariant descriptor will be very useful in these problems.

References

[1] T. Ahonen, J. Matas, C. He, and M. Pietikäinen, "Rotation invariant image description with local binary pattern histogram Fourier features," in *Proc. 16th Scand. Conf. Image Anal.*, 2009, pp. 2037–2041.

[2] H. Arof and F. Deravi, "Circular neighbourhood and 1-D DFT features for texture classification and segmentation," *Proc. Inst. Elect. Eng.—Vis., Image Signal Process.*, vol. 145, no. 3, pp. 167–172, Jun. 1998.

[3] B. Caputo, E. Hayman, and P. Mallikarjuna, "Class-specific material categorisation," in *Proc. 10th IEEE Int. Conf. Comput. Vis.*, 2005, pp. 1597–1604.

[4] A. Chan and N. Vasconcelos, "Classifying video with kernel dynamic textures," in *Proc. CVPR*, 2007, pp. 1–6.

[5] A. Chan and N. Vasconcelos, "Variational layered dynamic textures," in *Proc. CVPR*, 2009, pp. 1063–1069.

[6] J. Chen, G. Zhao, and M. Pietikäinen, "Unsupervised dynamic texture segmentation using local spatiotemporal descriptors," in *Proc. ICPR*, 2008, pp. 1–4.

[7] D. Chetverikov and R. Péteri, "A brief survey of dynamic texture description and recognition," in *Proc. Int. Conf. Comput. Recog. Syst.*, 2005, pp. 17–26.

[8] T. Crivelli, P. Bouthemy, B. Cernuschi-Frias, and J. F. Yao, "Learning mixed-state Markov models for statistical motion texture tracking," in *Proc. ICCV Workshop Mach. Learn. Vis.-Based Motion Anal.*, 2009, pp. 444–451.

[9] K. J. Dana, B. Ginneken, S. K. Nayar, and J. J. Koenderink, "Reflectance and texture of real-world surfaces," *ACM Trans. Graph.*, vol. 18, no. 1, pp. 1–34, Jan. 1999.

[10] S. Fazekas and D. Chetverikov, "Analysis and performance evaluation of optical flow features for dynamic texture recognition," *Image Commun.*, vol. 22, no. 7/8, pp. 680–691, Aug. 2007.

[11] A. Fernandez, O. Ghita, E. Gonzalez, F. Bianconi, and P. F. Whelan, "Evaluation of robustness against rotation of LBP, CCR and ILBP features in granite texture classification," *Mach. Vis. Appl.*, vol. 22, no. 6, pp. 913–926, Nov. 2011.

[12] Y. Guo, G. Zhao, J. Chen, M. Pietikäinen, and Z. Xu, "Dynamic texture synthesis using a spatial temporal descriptor," in *Proc. ICIP*, 2009, pp. 2277–2280.

[13] Z. Guo, L. Zhang, D. Zhang, and S. Zhang, "Rotation invariant texture classification using adaptive LBP with directional statistical features," in *Proc. ICIP*, 2010, pp. 285–288.

[14] Z. Guo, L. Zhang, and D. Zhang, "Rotation invariant texture classification using LBP variance (LBPV) with global matching," *Pattern Recognit.*, vol. 43, no. 3, pp. 706–719, Mar. 2010.

[15] Z. Guo, L. Zhang, and D. Zhang, "A completed modeling of local binary pattern operator for texture classification," *IEEE Trans. Image Process.*, vol. 19, no. 6, pp. 1657–1663, Jun. 2010.

[16] S. He, J. J. Soraghan, B. O'Reilly, and D. Xing, "Quantitative analysis of facial paralysis using local binary patterns in biomedical videos," *IEEE Trans. Biomed. Eng.*, vol. 56, no. 7, pp. 1864–1870, Jul. 2009.

[17] V. Kellokumpu, G. Zhao, and M. Pietikäinen, "Recognition of human actions using texture descriptors," *Mach. Vis. Appl.*, vol. 22, no. 5, pp. 767–780, Sep. 2011.

[18] S. Liao, M. Law, and A. C. S. Chung, "Dominant local binary patterns for texture classification," *IEEE Trans. Image Process.*, vol. 18, no. 5, pp. 1107–1118, May 2009.
[19] Z. Lu, W. Xie, J. Pei, and J. Huang, "Dynamic texture recognition by spatiotemporal multiresolution histogram," in *Proc. IEEE Workshop Motion Video Comput.*, 2005, pp. 241–246.
[20] Y. Ma and P. Cisar, "Event detection using local binary pattern based dynamic textures," in *Proc. CVPR Workshop Visual Scene Understanding*, 2009, pp. 38–44.
[21] R. Mattivi and L. Shao, "Human action recognition using LBP-TOP as sparse spatio-temporal feature descriptor," in *Proc. CAIP*, 2009, pp. 740–747.
[22] T. Ojala, M. Pietikäinen, and T. Mäenpää, "Multiresolution gray scale and rotation invariant texture analysis with local binary patterns," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 7, pp. 971–987, Jul. 2002.
[23] T. Ojala, T. Mäenpää, M. Pietikäinen, J. Viertola, J. Kyllönen, and S. Huovinen, "Outex—New framework for empirical evaluation of texture analysis algorithms," in *Proc. 16th Int. Conf. Pattern Recog.*, 2002, vol. 1, pp. 701–706.
[24] A. Ravichandran, R. Chaudhry, and R. Vidal, "View-invariant dynamic texture tecognition using a bag of dynamical systems," in *Proc. CVPR*, 2009, pp. 1–6.
[25] P. Saisan, G. Doretto, Y. N. Wu, and S. Soatto, "Dynamic texture recognition," in *Proc. CVPR*, 2001, pp. 58–63.
[26] M. Tuceryan and A. K. Jain, "Texture analysis," in *The Handbook of Pattern Recognition and Computer Vision*, C. H. Chen, L. F. Pau, and P. S. P. Wang, Eds., 2nd ed. , Singapore: World Scientific, 1998, pp. 207–248.
[27] M. Varma and A. Zisserman, "A statistical approach to texture classification from single images," *Int. J. Comput. Vis.*, vol. 62, no. 1/2, pp. 61–81, Apr./May 2005.
[28] R. Vidal and A. Ravichandran, "Optical flow estimation & segmentation of multiple moving dynamic textures," in *Proc. CVPR*, 2005, pp. 516–521.
[29] F. Woolfe and A. Fitzgibbon, "Shift-invariant dynamic texture recognition," in *Proc. ECCV*, 2006, pp. 549–562.
[30] J. Zhang and T. Tan, "Brief review of invariant texture analysis methods," *Pattern Recognit.*, vol. 35, no. 3, pp. 735–747, Mar. 2002.
[31] G. Zhao and M. Pietikäinen, "Dynamic texture recognition using local binary patterns with an application to facial expressions," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 6, pp. 915–928, Jun. 2007.
[32] G. Zhao and M. Pietikäinen, "Improving rotation invariance of the volume local binary pattern operator," in *Proc. IAPR Conf. Mach. Vis. Appl.*, 2007, pp. 327–330.
[33] G. Zhao, M. Barnard, and M. Pietikäinen, "Lipreading with local spatiotemporal descriptors," *IEEE Trans. Multimedia*, vol. 11, no. 7, pp. 1254–1265, Nov. 2009.

Guoying Zhao received the Ph.D. degree in computer science from the Chinese Academy of Sciences, Beijing, China, in 2005.

From July 2005 to August 2010, she was a Senior Researcher with the Center for Machine Vision Research, University of Oulu, Oulu, Finland, where she has been an Adjunct Professor since September 2010. She has authored over 70 papers in journals and conferences, and has served as a reviewer for many journals and conferences. In July 2007, she gave an invited talk in the Institute of Computing Technology, Chinese Academy of Sciences. With Prof. Pietikäinen, she has lectured tutorials on local-binary-pattern-based image and video descriptors at the International Conference on Pattern Recognition in Hong Kong in August 2006 and the International Conference on Computer Vision (ICCV) in Kyoto, Japan, in September 2009. She has authored/edited three books and a special issue on the IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS-PART B: Cybernetics, and is editing another special issue on the Elsevier Journal on Image and Vision Computing. Her research interests include gait analysis, dynamic-texture recognition, facial-expression recognition, human motion analysis, and person identification.

Dr. Zhao was a cochair of the European Conference on Computer Vision 2008 Workshop on Machine Learning for Vision-based Motion Analysis (MLVMA) and the MLVMA workshop at the ICCV 2009 and the IEEE Conference on Computer Vision and Pattern Recognition 2011.

Timo Ahonen received the Ph.D. (with honors) in information engineering from the University of Oulu, Oulu, Finland, in 2009.

He visited Eidgenoessische Technische Hochschule (ETH) Zurich, Switzerland, in 2002 and the University of Maryland, College Park, in 2005. He is currently a Senior Researcher with Nokia Research Center, Palo Alto, CA, working on computational photography on mobile devices. His research interests include computational photography, computer vision, object and face recognition, and image processing.

Jiří Matas received the M.Sc. degree in cybernetics (with honors) from the Czech Technical University in Prague, Prague, Czech Republic, in 1987 and the Ph.D. degree from the University of Surrey, Surrey, U.K., in 1995.

He is with the Center for Machine Perception, Department of Cybernetics, Faculty of Electrical Engineering, Czech Technical University in Prague. He has published more than 150 papers in refereed journals and conferences. His publications have more than 4000 citations in the ISI Thomson-Reuters Science Citation Index and about 10 000 in Google scholar. His h-indexes are 21 (ISI) and 34 (Google scholar). He is a coinventor of two patents.

Dr. Matas was a recipient of the best paper prize at the British Machine Vision Conferences in 2002 and 2005 and at the Asian Conference on Computer Vision in 2007. He served in various roles at major international conferences [e.g., the International Conference on Computer Vision, the IEEE Conference on Computer Vision and Pattern Recognition, NIPS, and the European Conference on Computer Vision (ECCV)], cochairing the ECCV 2004 and the Computer Vision and Pattern Recognition 2007. He is on the editorial board of the International Journal of Computer Vision, Pattern Recognition, and the IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE since January 2009, serving as associate editor-in-chief.

Matti Pietikäinen (SM'95–F'12) received the Doctor of Science in Technology degree from the University of Oulu, Oulu, Finland, in 1982.

In 1981, he established the Machine Vision Group, University of Oulu. Currently, he is a Professor of information engineering, the Scientific Director of the Infotech Oulu Research Center, and the Director of the Center for Machine Vision Research with the University of Oulu. From 1980 to 1981 and from 1984 to 1985, he was with the Computer Vision Laboratory, University of Maryland, College Park. He has made pioneering contributions, e.g., to local binary pattern (LBP) methodology, texture-based image and video analysis, and facial-image analysis. He has authored over 260 refereed papers in international journals, books, and conference proceedings, and about 100 other publications or reports. He has authored the book entitled "Computer Vision Using Local Binary Patterns" published by Springer in 2011. His research is frequently cited, and its results are used in various applications around the world.

Dr. Pietikäinen was an associate editor of the IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE and of Pattern Recognition journals, and currently serves as an associate editor of the Image and Vision Computing journal. From 1989 to 1992, he was the president of the Pattern Recognition Society of Finland. From 1989 to 2007, he served as a member of the Governing Board of the International Association for Pattern Recognition (IAPR) and became one of the founding fellows of the IAPR in 1994. He regularly serves on program committees of the top conferences and workshops of his field. He has lectured tutorials on LBP-based image and video descriptors at the Scandinavian Conference on Image Analysis 2005, International Conference on Pattern Recognition 2006, International Conference on Computer Vision 2009, Computer Vision and Pattern Recognition 2011, and International Conference on Image Processing 2011 conferences.