Local binary features for texture classification: Taxonomy and experimental study

CrossMark

Li Liu [a,*], Paul Fieguth [b], Yulan Guo [c], Xiaogang Wang [d], Matti Pietikäinen [e]

[a] *College of Information System and Management, National University of Defense Technology, 109 Deya Road, Changsha, Hunan410073, China*
[b] *Department of Systems Design Engineering, University of Waterloo, Waterloo, Ontario, Canada N2L 3G1*
[c] *College of Electronic Science and Engineering, National University of Defense Technology, Changsha, Hunan 410073, China*
[d] *Department of Electronic Engineering, The Chinese University of Hong Kong, Hong Kong, China*
[e] *The Center for Machine Vision Research, University of Oulu, Oulu 90014, Finland*

ARTICLE INFO

ABSTRACT

Local Binary Patterns (LBP) have emerged as one of the most prominent and widely studied local texture descriptors. Truly a large number of LBP variants has been proposed, to the point that it can become overwhelming to grasp their respective strengths and weaknesses, and there is a need for a comprehensive study regarding the prominent LBP-related strategies. New types of descriptors based on multistage convolutional networks and deep learning have also emerged. In different papers the performance comparison of the proposed methods to earlier approaches is mainly done with some well-known texture datasets, with differing classifiers and testing protocols, and often not using the best sets of parameter values and multiple scales for the comparative methods. Very important aspects such as computational complexity and effects of poor image quality are often neglected.

In this paper, we provide a systematic review of current LBP variants and propose a taxonomy to more clearly group the prominent alternatives. Merits and demerits of the various LBP features and their underlying connections are also analyzed. We perform a large scale performance evaluation for texture classification, empirically assessing forty texture features including thirty two recent most promising LBP variants and eight non-LBP descriptors based on deep convolutional networks on thirteen widely-used texture datasets. The experiments are designed to measure their robustness against different classification challenges, including changes in rotation, scale, illumination, viewpoint, number of classes, different types of image degradation, and computational complexity. The best overall performance is obtained for the Median Robust Extended Local Binary Pattern (MRELBP) feature. For textures with very large appearance variations, Fisher vector pooling of deep Convolutional Neural Networks is clearly the best, but at the cost of very high computational complexity. The sensitivity to image degradations and computational complexity are among the key problems for most of the methods considered.

© 2016 Elsevier Ltd. All rights reserved.

1. Introduction

1.1. Motivations and scope

Texture is a fundamental characteristic of the appearance of virtually all natural surfaces, is ubiquitous in natural images, and is a key component of many computer vision systems. Texture classification, as one of the major problems in texture analysis, has received considerable attention due to its value both in understanding how texture recognition works in humans as well as in the important role it plays in the fields of computer vision and pattern recognition, including biomedical image analysis, industrial inspection, analysis of satellite or aerial imagery, document image analysis, face analysis and biometrics, object recognition, and content based image retrieval.

A good texture feature, a crucial component of texture classification, is expected to achieve two competing goals: low computational complexity, to enable the classification task to run in real-time; and capturing the most representative texture information of a texture class, such that different texture classes can be distinguished despite the presence of various imaging distortions (including illumination, rotation, view point, scaling, occlusion, nonrigid deformations and noise). In spite of almost 50 years

of research and development in this field, most proposed texture feature extraction methods have not, however, been capable of performing at a level sufficient for real world textures.

Of the many methods, Local Binary Pattern (LBP) methods [1] have emerged as one of the most prominent and widely-studied classes of texture features, such that a vast number of LBP variants has been proposed for a diverse range of problems including texture classification [2–4], dynamic texture recognition [5], image matching [6], visual inspection [7], image retrieval [8], biomedical image analysis [9,10], face image analysis [11], motion and activity analysis [5,12], object detection [13,14], and background substraction [15]. There is, however, no comprehensive study concerning the connections between LBP methods and a corresponding comprehensive experimental evaluation. This paper intends to present such a comparative study.

Texture analysis is itself a complex problem because of the large number of natural texture classes and the associated dynamics within a class (intra-class variations), such as variations in periodicity, directionality and randomness, and the external dynamics due to changes in the imaging conditions including variations in illumination, rotation, view point, scaling, occlusion and noise. However, despite this complexity, most existing LBP variants have been evaluated only on small texture datasets with a relatively small number of texture classes, such as certain popular benchmark Outex test suites [2]. Experimental results based on datasets with small intraclass variations can be misleading; there are more challenging texture datasets with many texture classes or large intraclass variations, such as UIUC [16], UMD [17], CUReT [18] and KTHTIPS2b [19], ALOT [20] and Outex_TC40 [21], however, the performance of many LBP variants in these more challenging datasets is unknown. There is therefore significant value in performing a large scale empirical study on challenging texture datasets to demonstrate the strengths and weakness of LBP variants.

Many LBP variants have been proposed in the literature, usually in conjunction with their intended applications. Some literature surveys on LBP exist [1,22], but they miss recent variants and do not include experimental evaluations. To our knowledge, there are four papers [8,22–24] in the literature devoted to surveying LBP, however all four papers are limited in the number of LBP variants reviewed, in the extent of experimental evaluation, and a common disadvantage of these works is that they did not provide a systematic survey or grouping of existing LBP techniques:

1. The work of Fernandez et al. [23] seeks to build a general framework for texture analysis which the authors refer to as the histograms of equivalent patterns (HEP), which includes only a subset of LBP features. Most of the LBP features are implemented and experimentally evaluated in a very basic form, such as a 3×3 neighborhood. Furthermore, although the authors used eleven texture datasets, with the exception of UIUC all of the other datasets have very small intraclass variations, are small, and have appeared only in [23], greatly limiting the usefulness of the results.
2. Huang et al. [22] presented a review of the LBP variants in the application area of facial image analysis. However, half of the work is devoted to feature selection after LBP, and there is no experimental study of the LBP methods themselves.
3. The aim of the task-specific work of Nanni et al. [24] is to examine the performance of LBP features in a fairly specific and narrow application, that of classifying cell and tissue images, and then to try to find a fusion approach which works well for classification. The paper [24] is a short paper, not a survey.
4. Doshi and Schaefer [8] conducted a comparative study of LBP variants for the problem of texture retrieval, using a texture dataset which has a large number of a texture classes, but only limited intraclass variations. They concluded that almost all LBP methods significantly outperform other texture methods including cooccurrence and Gabor features. The paper [8] is a four-page conference paper, not a survey.

To the best of our knowledge there is no large scale experimental evaluation available in the literature studying LBP variants and their invariance to rotation, illumination and scale, robustness to noise and computational complexity.

1.2. Contributions

In recent years, LBP and its variants have led to significant progress in applying texture methods to different applications; due to this progress the division between texture descriptors and more generic image or video descriptors has been disappearing. We note that no extensive evaluation of various LBP variants has been performed, and the primary aim of this paper is to provide such a comparative study. In this paper, we will present a comprehensive survey of LBP features for 2D textures, including recent variations and applications to problems in image processing and computer vision. We undertake an experimental comparative study of the classification performance of LBP features applied to large scale benchmark texture databases. The major contributions of our work are summarized as follows:

1. We will provide an LBP taxonomy, grouping the LBP features into categories to offer an appreciation of the diverse approaches, an overview particularly helpful to new researchers seeking a high-level perspective.
2. We will undertake a comprehensive study concerning the motivations, characteristics of, and connections between LBP variants.
3. For experimental evaluation, we restrict ourselves to the original application of LBP: texture classification. We provide a comprehensive experimental study, evaluating the multi-scale versions of LBP features and their robustness to rotation, illumination, view point, scale and noise changes and their computational complexity.
4. We will study the performance of *forty* texture features including *thirty two* representative LBP features and *eight* recent deep convolutional network based features, using *thirteen* texture databases with varying levels of difficulty, as listed in Table 2. To our best knowledge, we are the first to perform such a large scale experimental evaluation, with most previously reported work considering datasets of only a small or medium number of different classes. In addition to experimenting with smaller, well-established datasets, we also compare the chosen LBP features on a very large dataset with about 300 texture classes. Robustness will be examined against noise types including Gaussian noise, image blur, salt and pepper, and random pixel corruption.

We believe the experimental work presented in this paper has significant insights to offer to the research community.

2. A taxonomy of LBP methods

Generally, most existing LBP-like feature extraction methods can be decomposed into the steps summarized in Fig. 1, where the actual sequence of steps taken clearly depends on the particular LBP method. As illustrated in Fig. 1, the original LBP approach had no preprocessing step and the LBP feature is extracted directly from the image. The local neighborhood structure is simply a set of pixels taken from a square neighborhood of 3×3 pixels, compared (thresholded) against the value of the central pixel, from which a
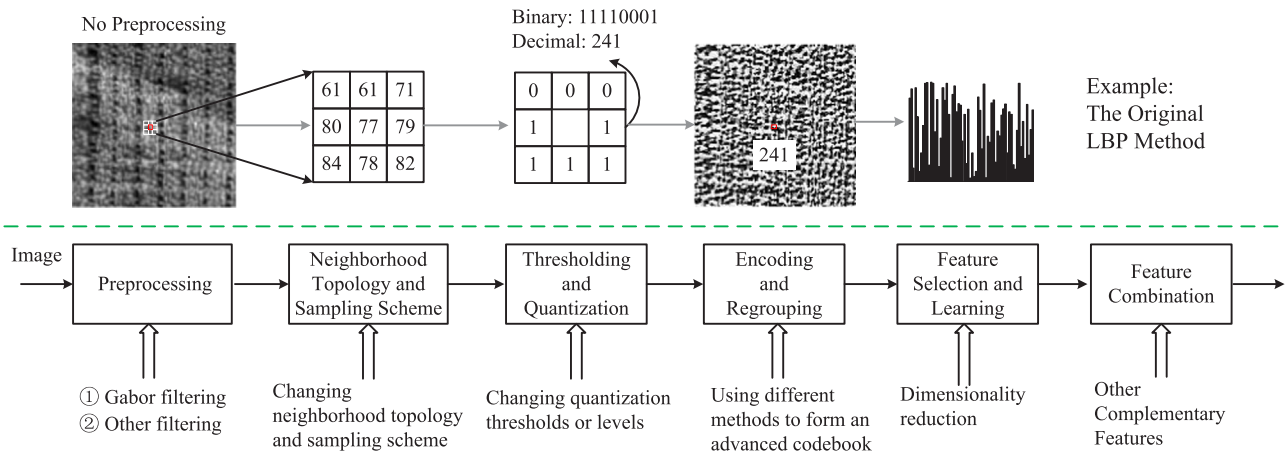
Fig. 1. The standard pipeline used to extract LBP like features.

binary vector of 8 bits is extracted.

Although the original LBP method had significant limitations, because of its efficiency and flexibility the overall LBP-like philosophy has proven very popular, and a great many extensions and modifications have been proposed to improve robustness and discriminative power. Because of the overwhelming proliferation of methods, some sort of organizational structure is needed. That is, in order to support an informed comparison of existing LBP methods, we develop in this section a taxonomy for LBP variants.

Of the steps in Fig. 1, *Preprocessing* and *Feature selection and learning* are problem-specific, in that they depend strongly on the statistics or characteristics of the problem at hand, and do not so much lead to novel LBP strategies. The six classes of our taxonomy are thus based on the observation that LBP variants generally perform some subset of the following six steps:

1. *Traditional LBP*: The classic, fundamental LBP approach.
2. *Neighborhood topology and sampling*: Pixels or pixel patterns with some neighborhood topology, sampled to form local feature vectors.
3. *Thresholding and quantization*: Binarization, or possibly quantization to multiple levels, on the basis of some threshold.
4. *Encoding and regrouping*: A definition of groups of patterns, to be combined, to improve distinctiveness.
5. *Combining complementary features*: A current trend in local image and video descriptors is to combine multiple complementary LBP-like descriptors, or to combine LBP-like with non-LBP.
6. *Methods inspired by LBP*: Further LBP-related strategies.

In this section different LBP variants are classified into such categories that describe their roles in feature extraction. Some of the variants could belong to more than one category, but in such cases only the most obvious category was chosen. A summary of the variants with method name abbreviations is presented in Tables 8 and 9. The purpose of this taxonomy is to reveal the underlying relationship among different LBP variants and to quickly understand the big picture. Our goal is to enable the reader to understand the context of a particular methodology's development, to comprehend the advantages and disadvantages, and as an organizational basis for the experimental tests.

2.1. Class 1: traditional LBP

The original LBP firstly appeared in [25] in 1994 and then in [26]. It was in 2002, with the introduction of the $LBP_{r,p}^{ri}$, $LBP_{r,p}^{u2}$ and $LBP_{r,p}^{riu2}$ descriptors [2], that LBP began to attract broad interest of the research community. The *original LBP* characterizes the spatial structure of a local image texture pattern by thresholding a 3×3 square neighborhood with the value of the center pixel and considering only the sign information to form a local binary pattern.

As illustrated in Fig. 2, given an image pixel c, the LBP pattern at c is computed by comparing the pixel's gray value g_c with the gray values of its p neighbors $\{g_n\}_{n=0}^{p-1}$ that are evenly distributed in angle on a circle of radius r centered at c:

$$\text{LBP}_{r,p}(c) = \sum_{i=0}^{p-1} s(g_i - g_c)2^i, \quad s(x) = \begin{cases} 1 & x \geq 0 \\ 0 & x < 0 \end{cases} \tag{1}$$

In practice the neighboring pixels are sampled on a circle, such that the gray values of neighbors which do not fall exactly in the center of pixels are estimated by interpolation. The cooccurrence of the comparison results is recorded in $\text{LBP}_{r,p}(c)$ by a unique string of binary numbers, where the sign function $s()$ ensures that the LBP code is invariant against any monotonic transformation of image brightness.

Given an $N*M$ texture image, a LBP pattern $\text{LBP}_{r,p}(c)$ can be the computed at each pixel c, such that a textured image can be characterized by the *distribution* of LBP patterns, representing a whole image by a LBP histogram vector $\underline{h}$:

$$\underline{h}(k) = \sum_{i=1}^{N} \sum_{j=1}^{M} \delta(\text{LBP}_{r,p}(i,j) - k) \tag{2}$$

where $0 \leq k < d = 2^p$ is the number of LBP patterns. By altering r and p, one can compute LBP features for any quantization of the angular space and for any spatial resolution.

The distinctive advantages of LBP are its ease of implementation, invariance to monotonic illumination changes, and low computational complexity.

Despite these merits, the original LBP has significant disadvantages:

(1) Producing rather long histograms, overwhelmingly large even for small neighborhoods, leading to decreased distinctiveness and large storage requirements.
(2) Capturing only the very local structure of the texture and failing to detect large-scale textural structures.
(3) Being sensitive to image rotation.
(4) Being highly sensitive to noise: the slightest fluctuation above or below the value of the central pixel is treated the same way as a major contrast.
(5) Losing local textural information due to the use of hard, fixed and coarse quantization and only the signs of differences of neighboring pixels are utilized.
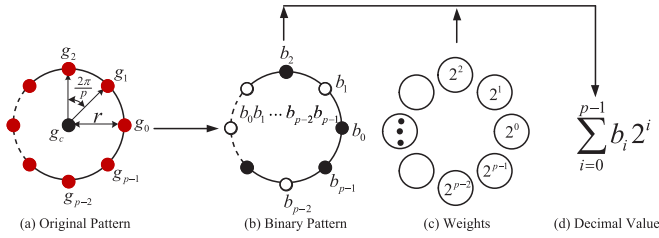
Fig. 2. (a) A typical (r,p) neighborhood type used to derive a LBP like operator: central pixel g_c and its p circularly and evenly spaced neighbors $g_0, \ldots, g_{p-1}$ on a circle of radius r.

To address these limitations, the traditional fundamental LBP strategy led to three early generalizations:

1. *Rotation invariant LBP*: The original LBP descriptor is not rotationally invariant, a serious limitation for many real-world applications. A rotation invariant version $LBP_{r,p}^{ri}$ of $LBP_{r,p}$ was obtained by grouping together those LBPs that are actually rotated versions of the same pattern, introduced by Pietikäinen et al. in [27]. Formally, the $LBP_{r,p}^{ri}$ is defined as

$$LBP_{r,p}^{ri} = \min\{ROR(LBP_{r,p}, i) | i = 0, 1, \ldots, p - 1\} \qquad (3)$$

where function $ROR(x, i)$ performs a circular i-step bit-wise right shift on the pattern binary string x, i times. Keeping only those rotationally-unique patterns leads to a significant reduction in feature dimensionality, however the number of codes in $LBP_{r,p}^{ri}$ still increases rapidly with p.

2. *Uniform LBP*: Ojala et al. in [2] observed that some LBP patterns occur more frequently than others, therefore the *uniform* LBP $LBP_{r,p}^{u2}$ preserves only the uniform patterns and groups all information contained in the nonuniform patterns. In particular, the uniformity measure

$$U(LBP_{r,p}) = \sum_{i=1}^{p} |s(g_{mod(i,p)} - g_c) - s(g_{i-1} - g_c)|. \qquad (4)$$

counts the number of 0/1 or 1/0 transitions between successive bits in the circular representation of the pattern binary code. All patterns with $U > 2$ are called nonuniform patterns and are classified under a single group, thus the 2^p original $LBP_{r,p}$ patterns are classified into $p(p-1) + 3$ different groups, leading to significant dimensionality reduction.

3. *Rotation invariant uniform LBP*: In order to obtain improved rotation invariance and to further reduce the feature dimensionality, building on $LBP_{r,p}^{ri}$ and $LBP_{r,p}^{u2}$, Ojala et al. [2] proposed the rotation invariant uniform LBP descriptor

$$LBP_{r,p}^{riu2} = \begin{cases} \sum_{i=0}^{p-1} s(g_i - g_c), & \text{if } U(LBP_{r,p}) \leq 2 \\ p + 1, & \text{otherwise} \end{cases} \qquad (5)$$

where the uniformity measure U is defined in (4). $LBP_{r,p}^{riu2}$ classifies all 2^p LBPs into $p + 2$ distinct groups, leading to a significantly lower feature dimensionality.

Discussion

$LBP_{r,p}^{ri}$ has all the shortcomings of the original LBP, except for having some level of rotation invariance. Unfortunately, the $LBP_{r,p}^{ri}$ descriptor was found to give poor performance for rotation invariant texture classification [27,2].

In comparison with $LBP_{r,p}$ and $LBP_{r,p}^{ri}$, $LBP_{r,p}^{u2}$ only avoids the disadvantage of high feature dimensionality. Although uniform patterns are beneficial in practice, their use is still heuristic: the

experimental observation that the uniform patterns appear to be fundamental properties of local image textures and have dominating proportions has been challenged by researchers [3,28–30], who argue that uniform LBPs extracted from some images containing high curvature edges, crossing boundaries or corners are not necessarily the dominant patterns.

The many extensions and modifications of LBP, discussed in the following sections, build upon the fundamental LBP strategy, seeking to increase robustness, to improve discriminative power, and to avoid the disadvantages of the traditional LBP methods.

2.2. Class 2: neighborhood topology and sampling

The traditional LBP method identifies a neighborhood as a set of pixels on a circular ring. To be sure, a great many neighborhood topologies can and have been defined, where Fig. 5 offers a visual illustration of the bewildering variety. We subdivide the topologies into three subcategories:

1. Fig. 5 (*b–e*): anisotropic information.
2. Fig. 5(*f–l*): local differences or magnitudes.
3. Fig. 5(*m–x*): micro- and macro-structures.

2.2.1. Exploiting anisotropic information

A circular neighborhood is important, particularly for rotation invariance, however there are some applications, such as face recognition, where rotation invariance is not required and anisotropic information may have value.

The most basic anisotropic topologies are circular-like, such as elliptical neighborhoods in Elliptical Binary Patterns (EBP) [31], also generalized in [9] to parabolic, hyperbolic, and spiral neighborhood topologies.

Very different topologies were introduced in the Local Line Binary Pattern (LLBP) [32], which uses lines in vertical and horizontal directions for LBP computations, further generalized in the Local Quantized Pattern (LQP) [33,34], which makes use of vector quantization and lookup tables to allow local pattern features to have a larger or deeper neighborhood and more quantization levels. By allowing larger local neighborhoods, LQP is expected to provide an increase in discriminative power. However, the number of different LBP patterns increases exponentially with the size of the spatial support of the pattern and the number of quantization levels.

2.2.2. Local differences or magnitudes

The standard LBP encoding rule thresholds all neighboring pixel values against the single, central pixel. This thresholding offers an encoding of a neighborhood with regard to its central value, but all relations *between* pixels in the neighborhood are lost. Therefore a class of methods has been proposed to generate LBP codes based on simple local differences or magnitudes among neighbors.

Most fundamental was the idea [35] to decompose local differences into two complementary components: the signs and the magnitudes, corresponding to operators CLBP_S and CLBP_M, respectively. CLBP_S is therefore the same as the original LBP, and CLBP_M offers local contrast (variance) information.

Local difference information was proposed in [36], measuring differences radially (RDLBP) and angularly (ADLBP), as illustrated in Fig. 5(h, i). Related to ADLBP in taking angular differences are the CSLBP [6,37] of Fig. 5(g), the BGC [38] of Fig. 5(l), tLBP [13] of Fig. 5(k), and the directional LBP (dLBP) of Fig. 5(j). The Local Neighboring Intensity Relationship Pattern (LNIRP) [39] is similar, but is based on second-order derivatives in the circular direction.

Although the literature gives the impression that the angular-LBP family has been studied more intensively than radial

differences, from [36] the RDLBP is far more effective than ADLBP for texture classification.

Two further related methods include the Geometric LBP (GLBP) [40] and the Local Derivative Patterns (LDP) [41]. Rather than dense local neighborhoods, the GLBP explores intensity changes on oriented neighborhoods, meaning a geometry of points on circles with different radii; the neighborhood topology of multi-scale RDLBP and multi-orientation GLBP are nearly the same, the difference lies in the subset of radial differences they choose to generate an LBP code. The LDP is a general framework to encode directional pattern features based on local derivative variations, such that the (n)th-order LDP compares the $(n-1)$th-order *directional* derivative at the center pixel with those at neighboring pixels. LBP is essentially the first-order special case of LDP. The third-order LBP was found to be most effective in the context of recognition experiments.

2.2.3. Patch-based LBP methods

The traditional LBP methods and many variants are criticized for encoding only local microtexture and being unable to capture nonlocal macrotexture that may be dominant in some cases. Patch-based LBP variants aim to alleviate this problem by integrating over larger areas.

There are, to be sure, many mechanisms for introducing non-locality, including the Local Binary Pattern Filtering (LBPF) [42], Multiscale Block LBP (MBLBP) [43], Three Patch LBP (TPLBP) [44], Four Patch LBP (FPLBP) [44], Pixel to Patch (PTP) [39], and Patterns of Oriented Edge Magnitudes (POEM) [45], Binary Rotation Invariant and Noise Tolerant (BRINT) [30], and very recently Median Robust Extended LBP (MRELBP) [46]. These methods are all related, with variations on the shapes of the patches (rectangular, square or pixel arc), filtering (raw pixels or filtered values), the nature of the central comparison (single pixel, patch mean, or patch median), whether one or multiple rings of patches are used, and whether directional or gradient information is captured.

LBP has also led to local binary feature descriptors designed for image matching, including BRIEF [47], BRISK [48] and FREAK [49], shown in Fig. 5(v), (w) and (x) respectively. BRIEF uses random pairwise graylevel pixel or Gaussian smoothed pixel comparisons to produce 128, 256 or 512 bit binary patch descriptors for key-point matching. These binary descriptors provide a comparable matching performance with the widely-used region descriptors such as SIFT and SURF, but have very short extraction times and very low memory requirements, especially suitable for emerging applications using mobile devices with limited computational capabilities. Comparative evaluations of these descriptors can be found in [50].

2.3. Class 3: thresholding and quantization

A serious drawback of the original LBP operator is that the thresholding operation directly compares pixel values, making it highly sensitive to noise. Researchers have proposed many LBP variants by changing thresholding schemes to gain noise robustness and discrimination power, falling into three categories:

1. Changes in thresholding.
2. Changes in number of quantization levels.
3. Preserving additional information.

2.3.1. Changes in thresholding

Instead of using only the gray value of the center pixel for thresholding, many other thresholds have been considered, including local means (NILBP [36], ILBP [51]), local median (MBP [52]), and one or more shifted thresholds (TMLBP [15], UTGLBP [53]).

Of these methods, ILBP and MBP also code the value of the center pixel, resulting in a doubling in the number of LBP bins. Similarly, the UTGLBP applies a union of multiple TMLBP features computed with different thresholds, thus able to capture a distribution of behavior, but at a cost of a significant increase in feature dimensionality.

2.3.2. Changes in number of quantization levels

The pioneering approach here is the Texture Spectrum (TS) [54–56], which actually predates LBP. TS uses an additional parameter, τ, which defines a tolerance for similarity between different gray intensities, allowing for robustness to noise, however at $3^8 = 6561$ bins, TS has a far higher dimensionality than $2^8 = 256$ of LBP.

Motivated by TS and LBP, [57] introduced a split coding scheme in order to reduce the dimensionality and proposed LTP. The pair of thresholds leads to an important strength: LTP is capable of encoding pixel similarity modulo noise using the simple rule that any two pixels within some range of intensity are considered similar.

Subsequent to [57] many LTP variants were proposed in the literature. In [9] a quinary code was proposed, with two thresholds further split into four binary codes. In [4] the SILBP was proposed to deal with gray scale intensity changes in complex backgrounds, but again at high feature dimensionality. In a related strategy, some researchers proposed to encode small pixel difference as an uncertain bit first and then to determine its value based on the other bits of the LBP code, such as the SoftLBP [58], Fuzzy LBP (FLBP) [59] and Noise Resistant LBP (NRLBP) [60].

2.3.3. Preserving additional information

FLBP and NRLBP allow multiple LBP patterns to be generated at one pixel position, but at considerably increased computational cost. As a result, the NRLBP [60] was proposed as a lower-complexity improvement on FLBP. In NRLBP, only the uniform patterns are with a simplified contribution weight, which allows the complex method of FLBP to be replaced with a lookup table. Similarly the Robust LBP (RLBP) [61], similar to FLBP, is based directly on the original LBP, inspecting consecutive three-bit substrings in non-uniform patterns, but at a risk of mapping natural nonuniform patterns to uniform ones.

Extended LBP (ELBP) presented in [62,63] generates multiple LBP codes at a pixel position in a different way. The ELBP operator not only encodes the sign information between the central pixel and its neighbors, but also encodes the magnitudes using some additional binary units in an interesting way. The Improved Local Ternary Pattern (ILTP) [10] combines ILBP and LTP in that the LTP code is split into positive and negative ILBP codes.

2.4. Class 4: encoding and regrouping

The original LBP operator produces rather long histograms of 2^P distinct patterns, overwhelmingly large even for small neighborhoods, leading to poor discriminative power and large storage requirements. It is clear that not all local patterns are meaningful for modeling the characteristics of textures or other images, thus researchers have attempted to obtain more discriminative, robust and compact features by identifying the most informative pattern groups according some criteria:

1. Heuristic groupings.
2. Cooccurrence groupings.
3. Learning strategies.

2.4.1. Heuristically grouping LBPs into classes

The goal here is to attempt to develop a different grouping

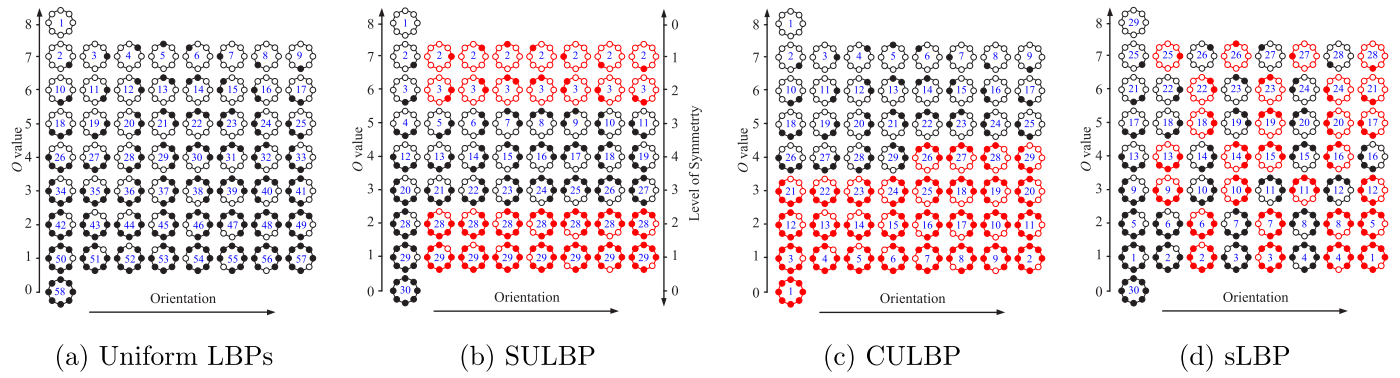(a) Uniform LBPs   (b) SULBP   (c) CULBP   (d) sLBP

Fig. 3. Examples illustrating four closely-related LBP variants. Empty and filled circles correspond to bit values of 0 and 1, respectively. The number in the center of each pattern is just a category label, with red patterns indicating the collecting of multiple patterns into one group. Thus the uniform LBP has 58 patterns (a), 30 groups in SULBP (b), 29 different groups in CRLBP (c), and 30 different groups in semantic LBP (d). (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

scheme to regroup all LBP patterns to improve on the traditional descriptors. Figs. 3 and 4 visually contrast the grouping behavior for a number of methods.

SULBP [64], CULBP [65,66] and sLBP [67], all illustrated in Fig. 3, share the common idea in regrouping the uniform LBPs to obtain lower feature dimensionality and higher discriminative ability than $LBP_{r,p}^{u2}$. Thus SULBP asserts that uniform patterns representing "edge" and "corner" occur more frequently in face images than those representing "line end" and are more discriminative, CULBP[1] is designed to be robust to inverted changes of the background and foreground intensities [65,66] for object detection, and sLBP [67] groups the uniform patterns with similar arch length (number of consecutive 1s) and close orientations together.

NTLBP [29] , SLBP [68], LBC [69], NLBP [70], RSCILBP [71], NELBP [28], HLBP [72] and scLBP_S [73] are closely related LBP variants which are designed to make better use of the nonuniform patterns instead of discarding them, with illustrations given in Fig. 4. The groups in scLBP_S are highly similar to those in $LBP_{r,p}^{ri}$ [73]. Similar to scLBP_S, scLBP_M, which is derived from CLBP_M [35], was also proposed in [73].

2.4.2. Cooccurrence of LBPs

The use of cooccurrences of LBP (CoLBP) patterns is borrowed from Gray Level Cooccurrence Matrices (GLCM) [74]. The general idea is to consider the joint probability of pairs of LBPs at certain relative displacements. Applying GLCM on $LBP_{r,p}^{u2}$ and $LBP_{r,p}^{riu2}$ results in UU-CoLBP and RURU-CoLBP [75,76] respectively.

Traditional methods for making the GLCM matrix rotation invariant can be applied to CoLBP, such as summing multiple CoLBP over orientation. However, there is an essential difference between applying GLCM to image gray values and to LBP codes; in particular, unlike a gray value, an LBP pattern has spatial structure and orientation. Researchers have thus proposed methods to achieve rotation invariance either at the local feature level or at the cooccurrence level, or both. The key for the computation of rotation invariant at the cooccurrence level is to construct a rotation invariant coordinate system for each pair of LBP patterns.

PRICoLBP [75,76] was proposed to achieve global rotation invariance based on RUU-RICoLBP by sampling the neighboring point along the unit gradient direction or the unit normal direction at the center point. The method was further extended PRICoLBP$_g$ to multi-scale and multi-orientation analysis by sampling multiple neighboring points along the gradient direction and

norm direction at the center pixel, but resulting in a fairly high dimension feature. Instead of considering cooccurrences of LBPs at different locations, some strategies seek to encode the pairwise information of LBPs at the same location but from different scales, such as MDLBP [77], MSJLBP [78] and HMSLBP [79].

2.4.3. Learning discriminative LBPs

With similar motivations as the heuristically designed LBP variants (Section 2.4.1), researchers seek to learn the most reliable and discriminative dominant patterns; representative work along this line includes DLBP [3], disLBP [80], LDRICoLBP [81] and ABLBP [82].

DLBP [3] makes use of the most frequently occurring LBP patterns and discards those rarely occurring ones by examining the occurrence frequencies of the rotation invariant LBP groups. disLBP [80] improves on DLBP by considering the intra-class similarity and inter-class distance during learning. Following DLBP and disLBP, LDRICoLBP [81] learns cooccurrencing LBP patterns by using similar learning schemes. ABLBP [82] uses AdaBoost to select the most discriminative LBP pattern groups.

2.4.4. Discussion

The dimensionality of the LBP variants in this subcategory is compared in Table 1. All heuristically designed methods (except for RSCILBP) (Section 2.4.1), have fairly low feature dimensionality, which is advantageous when generalizing to multiscale analysis and does not increase the computational complexity over traditional LBP methods. In contrast, the cooccurrence LBP methods attempt to increase discriminative power by including spatial cooccurrence information. However, their feature dimensionality is increased considerably, limiting the number of neighbors to be at most eight for multiscale analysis. The learning based methods in Section 2.4.3 have moderate feature dimensionality, however they require a pre-training step.

2.5. Class 5: combining with complementary features

There has been quite a few new effective local image and video descriptors proposed by combining the strengths of complementary descriptors. Methods along this thread can be further divided into three subcategories: (1) Preprocess the input image prior to LBP type feature extraction, mostly using filtering techniques; (2) Combining multiple LBP type descriptors to obtain a more powerful one; (3) First extracting LBP type features and other non-LBP type features in the parallel way, and then fusing two kinds of features:

---

[1] In their original work [65,66], CULBP is referred to as NonRedundant LBP (NRLBP). However, there is another method which is also abbreviated as NRLBP.
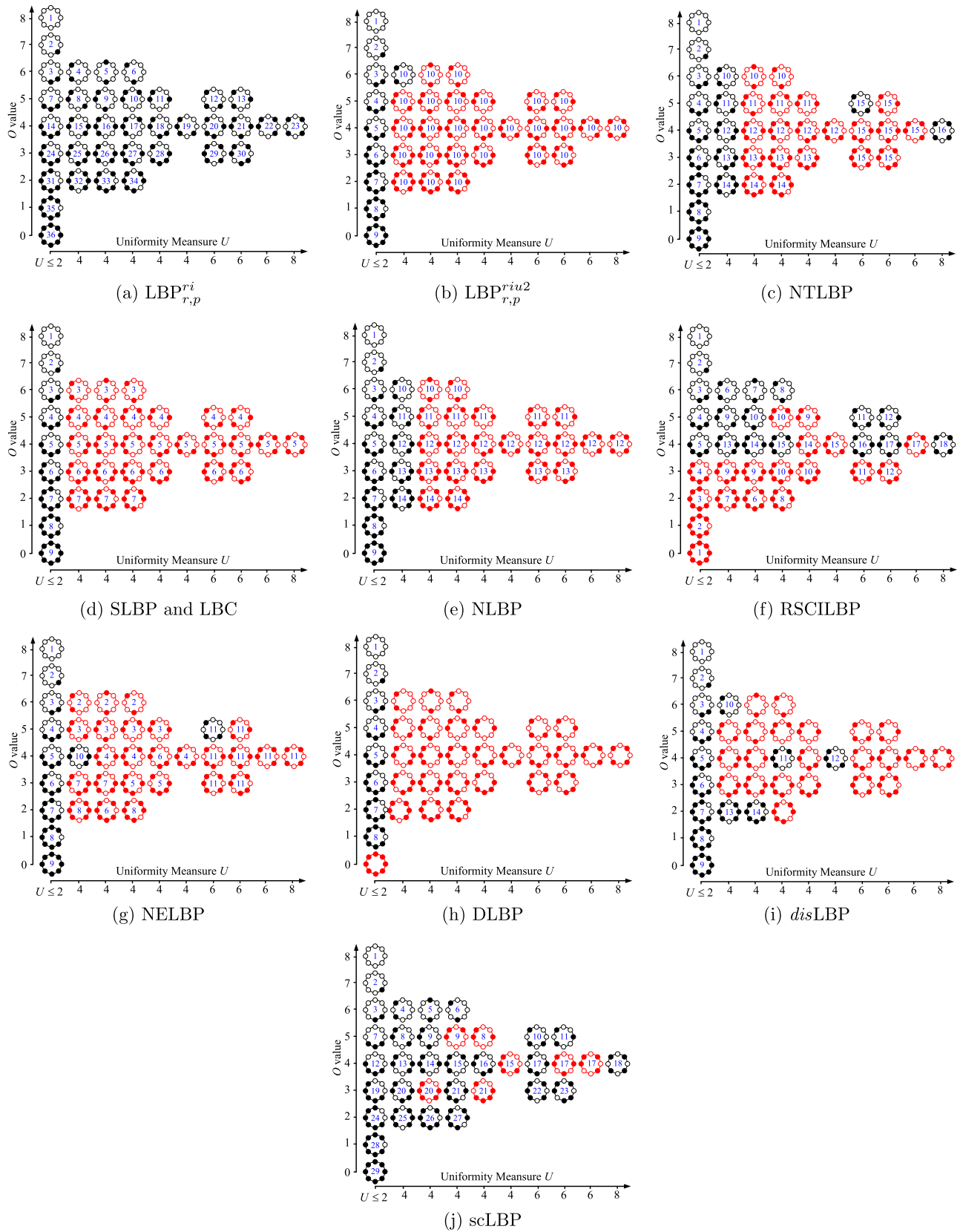
Fig. 4. A continuation from Fig. 3, with shading indicating bit value and numbers indicating category labels, comparing nine strategies in space of uniformity U (4) and O value denotes the number of 1s in a pattern. Empty and filled circles correspond to bit values of 0 and 1, respectively. For example, the non-uniform LBP codes, with a uniformity $U > 2$, may be preserved (a), lumped into a single category (b), divided into a few categories (e), or not be preserved at all (h).
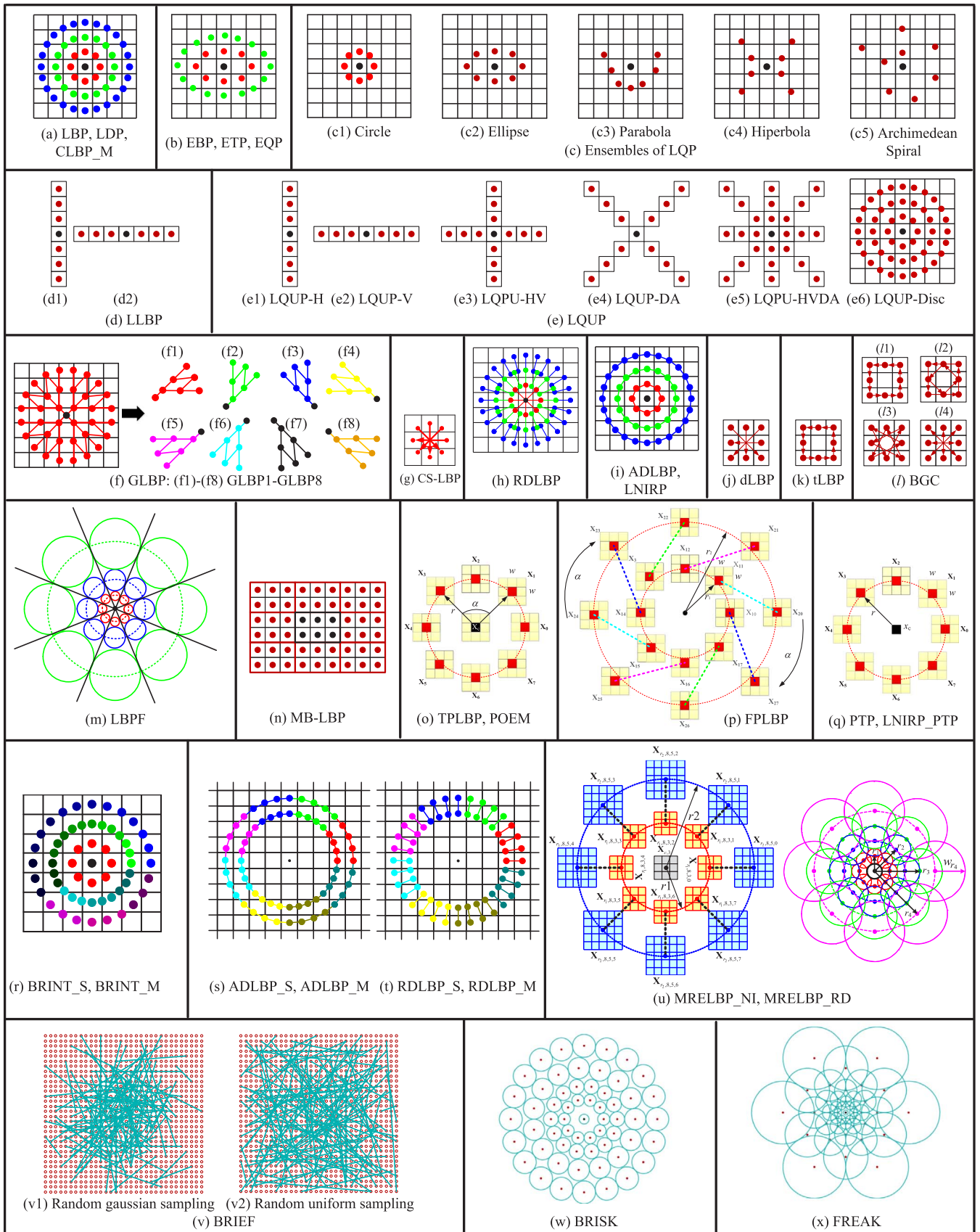
Fig. 5. Neighborhood topology used by the LBP variants in Class 2 — Neighborhood Topology and Feature category.

Table 1
Dimensionality comparison for LBP variants in Class 4.

Parameters	(r,p)		
Methods	(1,8)	(2,16)	(3,24)
$LBP_{r,p}$	256	65536	2^{24}
$LBP_{r,p}^{ri}$	36	4116	699,252
$LBP_{r,p}^{u2}$	59	243	555
$LBP_{r,p}^{riu2}$	10	18	26
SULBP	30	62	94
CULBP	29	121	277
sLBP	30	30	30
HLBP	58	242	554
NELBP	11	23	34
NLBP	14	30	46
RSCILBP	18	1224	
SLBP	9	17	25
LBC	9	17	25
NTLBP	16	36	56
CoLBP	2^{16}	2^{32}	2^{48}
RICoLBP	2^{16}	2^{32}	2^{48}
UUCoLBP	59*59	243*243	555*555
RURUCoLBP	10*10	18*18	26*26
RUUCoLBP	59*10	243*18	555*26
PRICoLBP	59*10	243*18	555*26
MSJLBP	59*10	243*18	555*26
MDLBP	10*18*26		
DLBP	7	53	516
disLBP	14	228	960

1. Preprocessing.
2. Multiple LBP-like codes.
3. Combining LBP and other features.

2.5.1. Preprocessing

The most widely used preprocessing method before LBP computation is Gabor filtering. This is because Gabor filtering and LBP provide complementary information: LBP captures small and fine details, while Gabor filters encode appearance information over a broader range of scales. The pioneering work is LGBP [83]. It filters an image with multi-scale and multi-orientation Gabor filters and then extracts LBP features from Gabor magnitude images. Disadvantages of LGBP includes its high dimensionality and the extra computation burden introduced by filtering. Following LGBP, Gabor preprocessing has also been coupled with other LBP variants to further boost recognition performance, such as LDP [41] and LQP [33,34].

Zhang et al. [84] proposed LEP for texture classification, where multi-scale and multi-orientation Gaussian-like second-order derivative filters are used to filter the original image. Different from LGBP where each filter response map is converted to one LBP feature map, the filter responses from the same scale but across all orientations are used to generate a LBP type feature map. LEP encodes the relationship among different feature channels using N-nary coding scheme instead of binary coding scheme. A downside of LEP is that it requires pretraining. Similar to LEP, BSIF [85] describes each pixel's neighborhood by a binary code which is obtained by first convolving the image with a set of linear filters and then binarizing the filter responses. The bits in the code string correspond to binarized responses of different filters. However, BSIF learns the filters by utilizing statistics of natural images instead of a manually predefined set of filters. Qian et al. [86] proposed first to transform an image into spatial pyramid domain and then to compute the traditional LBP descriptors, and named their method as Pyramid LBP (PLBP). Galoogahi and Sim [87] introduced LRBP where radon transform is used to preprocess an image before LBP computation.

Li et al. [88] proposed an LBP variant named Scale and Rotation Invariant SubUniform LBP ($LBP_{r,p}^{sri-su2}$) for texture classification. The radius for computation of an local binary pattern at each pixel is defined as the characteristic scale of the local patch centered at that pixel. Different pixels have different characteristic scales, resulting in local scale invariance. The subuniform pattern groups defined by Li et al. [88] are essentially the same as the rotation variant uniform groups in $LBP_{r,p}^{u2}$. After obtaining the subuniform histograms, a circular shift LBP histogram is computed to obtain rotation invariance. Note that the circular shifting of the subuniform histograms is actually the same as the glob al matching scheme proposed in [89], which was applied in $LBPV_{r,p}^{u2}GM_{PD2}$. Li et al. [88] showed a good discrimination capability of $LBP_{r,p}^{sri-su2}$ for texture classification. Davarzani et al. [90] proposed Weighted Rotation and Scale Invariant LBP (WRSI LBP) to address rotation and scale variations in texture classification. To achieve rotation invariance, dominant orientation needs to be estimated for each pixel in a image. To achieve scale invariance, an approach similar to the one used in $LBP_{r,p}^{sri-su2}$ [88] is used. Furthermore, Davarzani et al. [90] used the minimum magnitude of local differences as an adaptive weight to adjust the contribution of an LBP code in histogram calculation, resulting WLBP operator. One downside of $LBP_{r,p}^{sri-su2}$ and WRSI LBP is that characteristic scale and dominant orientation estimation is computationally expensive and unreliable.

2.5.2. Combining multiple LBP-like codes

Guo et al. [35] was the first to combine multiple LBP type features[2] and proposed CLBP to combine CLBP_S, CLBP_M and CLBP_C. The image local differences between a center pixel and its neighbors are decomposed into two complementary components: the signs and the magnitudes and thus two operators CLBP_S and CLBP_M were proposed. The magnitude component provides an effective alternative for the complementary contrast measure of LBP. As well, the center pixels representing the image gray level were also regarded to have discriminative information and they are converted into a binary code by global thresholding, resulting in the CLBP_C feature. CLBP is actually motivated by the earlier ELBP method [63]. The CMLBP [91] is closely related to the CLBP descriptor. Unlike CLBP where two separate LBP type patterns CLBP_S and CLBP_M were proposed, CMLBP used two bits for each neighbor in order to represent the sign as well as the magnitude information, resulting a $2p$-bit code to the center pixel with p neighbors. In order to reduce feature dimensionality, the CMLBP pattern is split into two subpatterns.

The CLBC approach [69] is highly similar to CLBP, but with a different binary patterns regrouping scheme (i.e. the scheme shown in Fig. 4(d)). Guo et al. [80] combined CLBP and disLBP to introduce disCLBP. Later on, among their individual LBP like descriptors including NILBP, RDLBP, ADLBP and CILBP (CILBP is the same as CLBP_C) which have also been discussed in Section 2.3, Liu et al. [36] found that CINIRD, which is the joint distribution of NILBP, RDLBP and CILBP, turned out to be very discriminative.

CLBP, CLBC and CINIRD are highly sensitive to image noise, high in dimensionality and being too local. To overcome these shortcomings, Liu et al. recently proposed BRINT [30] and MRELBP [46]. BRINT combines three individual descriptors BRINT_S, BRINT_M and BRINT_C. Unlike CLBP and CINIRD where only rotation invariant uniform patterns are considered, BRINT uses all the rotation invariant patterns to avoid the risk of the uniform patterns not taking the dominant proportion. In BRINT, pixels are sampled in a circular neighborhood, but keeping the number of bins in a

---

[2] We would like to point out that all individual LBP descriptors have been discussed in Section 2.2.2, where readers can find more details.

single-scale LBP histogram constant and small, such that arbitrarily large circular neighborhoods can be sampled and compactly encoded over a number of scales. BRINT can extract features from a number of scales with low feature dimensionality. Moreover, BRINT was showed to be robust to noise.

In order to encapsulate microtexture and macrotexture information at the same time and further to enhance the discriminative power and noise robustness, Liu et al. [46] presented MRELBP to combine MRELBP_NI, MRELBP_RD and MRELBP_CI. Although MRELBP was derived based on CINIRD, it outperformed CINIRD significantly, especially in random noise corrupted situations. Moreover, MRELBP has much lower feature dimensionality than other methods in this subcategory. Ryu [73] developed a method to represent a local binary pattern as a feature vector in Euclidean space by sorting numbers of consecutive patterns. Ryu et al. [73] applied the idea to CLBP_S and CLBP_M, resulting three new LBP type features scLBP_S, scLBP_M+ and scLBP_M− (CLBP_M was separated into two features) which was concatenated into a single feature vector named scLBP. Subsequently, Ryu [73] introduced dictionary learning of scLBP based on kd-tree which separates data with a space partitioning strategy. Guo et al. [92] proposed SSLBP with the aim of addressing scale variation for texture classification. In SSLBP, a scale space of a texture image is firstly derived by a Gaussian filter; Then a histogram of prelearned dominant binary patterns is built for each image in the scale space; Finally, for each pattern, the maximal frequency among different scales is considered as the scale invariant feature. Essentially, SSLBP combines the ideas of DLBP [3] and CLBP (CLBP_CS and CLBP_CM) [35] to learn the dominant binary patterns.

2.5.3. Combining LBP and other features

From the very beginning the original LBP operator was designed as a complementary measure of local image contrast, and the joint histogram of the two complementary features, namely LBP/VAR, was proposed for rotation invariant texture classification [2]. However, the value of VAR is continuous so that a quantization step is needed to calculate the histogram.

Ahonen et al. [93] proposed LBP Histogram Fourier features (LBPHF) to achieve rotation invariance globally by combining LBP and Discrete Fourier transform (DFT). LBPHF is derived by first computing a uniform LBP histogram over the whole image, and then constructing rotationally invariant features from the DFT transform of the histogram. Later in [94], LBPHF is combined with the CLBP_S and CLBP_M descriptors [35] to further improve its distinctiveness.

Guo et al. [89] proposed LBPV to incorporate the local contrast information into the LBP histogram by utilizing the variance VAR as a locally adaptive weight to adjust the contribution of each LBP code. LBPV avoids the pretraining step for VAR quantization used in [2]. Likewise, Guo et al. [89] also obtained rotation invariance globally. They estimate the principal orientation of an image from the uniform LBP histogram and then use the estimated principle orientation to align the LBPV features. This raises two issues. For one, it is unstable and inaccurate to estimate the dominant orientation of a texture image since lots of texture images do not have an obvious principle orientation. For another, their proposed global matching procedure have to handle high dimensional feature. Arguing that local variance feature VAR is an isotropic measurement and fails to capture orientation information in textures, Guo et al. developed ALBP [95] by incorporating the directional statistical features (i.e. the mean and standard deviation of the local absolute differences) for rotation invariant texture classification. In addition, the least square estimation is used to adaptively minimize the local difference for more stable directional statistical features.

Similar to LBPV [89], DRLBP [14] is proposed to combine LBP and gradient magnitude information. Instead of considering the LBP code frequencies, the pixel gradient magnitude is used as a weight to be assigned to each code which is then voted into the bin that represents the code. DRLBP is formulated by concatenating two histogram vectors: the difference histogram which is the absolute difference between the bins of an LBP code and its complement and the sum histogram which is the sum of the bins of an LBP code and its complement. It is hoped that DRLBP can solve the problem of discrimination between a bright object against a dark background and vice versa inherent in LBP. Satpathy et al. [14] also extend the idea of DRLBP to LTP and proposed DRLTP approach.

In addition to applying LBP to Gabor filtered face images [83], the fuse of LBP variants and Gabor features has also been explored, with applications in texture classification [3] and face recognition [57]. Khellah [96] proposed DNS and combined it with LBP for texture classification, claiming noise robustness. Wang et al. [97] combined Histogram of Gradients (HOG) with LBP, performing very well in human detection with partial occlusion handling. In addition to HOG and LBP, Hussain and Triggs [98] used LTP. Klare and Jain [99] exploited the combination of LBP and SIFT for heterogeneous face recognition. Chan et al. [100] fused LBP and LPQ for face recognition with robustness to image degradation caused by illumination variations and blurring. Combining ideas from Haar and LBP features have given excellent results in accurate and illumination invariant face detection [101].

2.6. Class 6: other methods inspired by LBP

LBP has inspired the development of related local image descriptors, including LPQ [102,103], WLD [104], HGPP [105], LHS [106], LFD [107,108] and DFD [109].

LPQ [102] is generated by quantizing the Fourier transform phase in local neighborhoods, such that histograms of LPQ labels computed within local regions are used as a texture descriptor similar to LBP. LPQ is, by design, tolerant to most common types of image blurs. A rotation invariant extension of LPQ (denoted as RILPQ) has also been presented [103]. WLD [104] is designed based on the fact that human perception of a pattern depends not only on the change of a stimulus but also on the original intensity of the stimulus. WLD [104] is a 2D histogram of differential excitation and gradient orientation, where the differential excitation reflects the ratio between the relative intensity differences of a center pixel against its neighbors and the intensity of the center pixel itself. WLD demonstrated good results on texture classification and face detection.

In HGPP [105], an image is first convolved with 40 Gabor filters at five different scales and eight orientations. Then the Gabor phase information is encoded for 90 "images" (five real Global Gabor Phase Patterns (GGPPs), five imaginary GGPPs, 40 real Local Gabor Phase Patterns (LGPPs), and 40 imaginary LGPPs), with the same size as the original image. Due to the richer information from this additional Gabor filtering stage, HGPP improve face recognition performance when compared with the original LBP method. However, the Gabor filtering step black introduces a heavy computational burden. It is not suitable for real-time applications due to its extremely high dimensionality.

Lategahn et al. [110] applied a filter bank to transform a texture image into multidimensional filter response space and subsequently estimated the joint probability density functions of the filter responses by Gaussian mixture models (GMM). Two types of filter banks, i.e. the oriented difference filters of the LBP method [111] and wavelet frame transform filter bank, were considered. This method avoids the crude quantization errors of LBP. Motivated by the idea of Fisher Vector [112], Sharma et al. [106]

proposed Local Higher-Order Statistics (LHS) of oriented difference filter responses of the LBP method [111] for image description. Like the method proposed by [110], the LHS requires neither any user specified quantization of the feature space nor any heuristics for discarding low occupancy volumes of the space. Experiments with texture and face databases demonstrate good performance.

Motivated by LBP, Maani et al. [107,108] proposed Local Frequency Descriptor (LFD) for texture classification. The LFD descriptor [107,108] is based on the local frequency components that are computed by applying 1D Fourier transform on the neighboring pixels on a circle centered at each pixel. The low frequency components are kept since they are the major constituents of the circular neighbors and can effectively represent textures. Then three sets of features are extracted from the low frequency components, two based on the phase and one based on the magnitude. LFD is invariant to rotation and linear changes of illumination. Maani et al. [113] also extend the idea of LFD and proposed Robust Edge Aware Descriptor (READ) for image matching. The recent DFD descriptor learns the most discriminating local features, computing the difference vectors between the center patch and each of its neighboring patches to form a pixel difference matrix, which is then projected and re-grouped to form the discriminant pattern vector. A standard bag of words model [16,114] is applied for face representation. The DFD descriptor performs well in face identification and verification, however it involves a number of parameters, has high dimensionality, a time consuming training process, and high training data needs.

3. Experiments

3.1. Evaluated LBP variants

Because no extensive evaluation has been performed on a great number of LBP variants, this paper aims to provide such a comparative study. We evaluated the 30 LBP methods shown in Table 7 (also bolded in Tables 8 and 9) on 13 datasets. We use the original source code if it is publicly available, and for the remainder we have developed our own implementation. To ensure fair comparisons, the parameters of each LBP method are fixed across all the datasets, since it is difficult and undesirable to tune the parameters of each LBP method for each evaluation. In most cases we use the default parameters suggested in the original papers.

For most of the tested methods, multiscale variations had been proposed in the original work, but usually limited to three scales. Since the spatial support of a texture descriptor influences its classification performance, for fair comparison we implemented multiscale and rotational invariant formulations of *each* LBP method up to nine scales, following the multiscale analysis approach proposed by Ojala et al. [2], representing a texture image by concatenating histograms from multiple scales.

3.2. Evaluated non-LBP deep learning approaches

FV-CNN: Deep convolutional neural networks (CNN) have demonstrated their power as a universal representation for recognition. However, global CNN activations lack geometric invariance, which limits their robustness for recognizing highly variable images. Cimpoi et al. [115] propose an effective texture descriptor FV-CNN, obtained by first extracting CNN features from convolutional layers for an texture image at multiple scale levels, and then performing orderless Fisher Vector pooling of these features.

ScatNet: Despite significant progress, there is still little insight into the internal operation and behavior of deep CNN models. Arguably one instance that has led to a clear mathematical

justification is the multistage architectures of ConvNet [115,116], and specifically in the wavelet convolutional scattering network (ScatNet) [117,118] where the convolutional filters are predefined as wavelets, hence no learning process is needed. ScatNet has been extended to achieve rotation and scale invariance [119].

PCANet and RandNet: Motivated by ScatNet, Chan et al. [120] proposed a simple deep learning network, PCANet, based on cascaded/multistage principal component analysis (PCA), binary hashing, and histogram pooling. The authors also introduced RandNet, a simple variation of PCANet, which shares the same topology as PCANet, but in which the cascaded filters are randomly selected, not learned.

3.3. Image data and experimental setup

We conducted experiments on the *thirteen* texture datasets shown in Table 2, including three very large datasets with 294 texture classes. These datasets are derived from the six most commonly used texture sources: Outex [21], CUReT [18], Brodatz [121], UIUC [16], UMD [17] and KTHTIPS2b [19]. The experimental setup on the three test suites **Outex_TC10**, **Outex_TC12_000** and **Outex_TC12_001**, which were designated by Ojala et al. [2] for rotation and illumination invariant texture classification, was kept exactly the same as [2].

For large-scale texture classification, we created **Outex_TC40_A**, **Outex_TC40_B** and **Outex_TC40_C** following a similar way in Ojala et al. [2]. Each dataset contains 294 texture classes, with training data acquired under illuminant "inca" and rotations 0°, 30°, 45° and 60°, and tested with rotations 5°, 10°, 15°, 75° and 90°. The test images in **A** are from illumination "inca", the same as the training images, and thus simpler than datasets **B** and **C**, with testing data from illumination types "Horizon" and "TL84", respectively.

For **CUReT**, we use the same subset of images as in [18,122]. For **Brodatz** [121] we use the same dataset as [16,123,68]. The **BrodatzRot** dataset is generated from Brodatz by rotating each sample at a random angle, helping to test rotation invariance. The challenging **UIUC** dataset [16] contains images with strong scale, rotation and viewpoint changes in uncontrolled illumination environment. The **UMD** dataset [17] is similar to UIUC with higher resolution images but exhibits less nonrigid deformations and stronger illumination changes. Each image in the **KTHTIPS2b** [19] is obtained with 3 viewing angles, 4 illuminants, and 9 different scales. The **ALOT** dataset [20] consists of 250 classes each of which has100 images. ALOT is challenging as it represents a significantly larger number of classes (250) compared to UIUC and UMD (25) and has very strong illumination change (8 levels of illumination). The viewpoint change is however less dramatic compared to UIUC and UMD.

To evaluate the robustness with respect to random noise, we considered Gaussian noise, image blurring, salt-and-pepper noise, and random pixel corruption, the same noise types tested in [124]. We use only the noise-free texture images for training and test on the noisy data, as summarized in Table 2. The test suites are based on Outex_TC11n and Outex_TC23n, which have 24 and 68 texture classes, respectively. The noise parameters include Gaussian noise standard deviation σ, Gaussian blur standard deviation σ, Salt-and-Pepper noise density ρ, and pixel corruption density v.

We resize images in UIUC, UMD and ALOT to obtain lower resolution, as shown in Table 2. Each texture sample is preprocessed, normalized to zero mean and unit standard deviation. For CUReT Brodatz, BrodatzRot, UIUC, UMD and ALOT, half of the class samples were selected at random for training and the remaining half for testing, and all results are reported over 100 random partitionings of training and testing sets. For KTHTIPS2b, we follow the training and testing scheme of [125]: training on three samples

Table 2

Summary of texture datasets used in our experiments. $\Theta_1 = \{5°, 10°, 15°, 30°, 45°, 60°, 75°, 90°\}$, $\Theta_2 = \{0°, 5°, 10°, 15°, 30°, 45°, 60°, 75°, 90°\}$.

Texture dataset	Texture classes	Sample size (pixels)	Samples per class	# Training per class	# Testing per class	Samples in total	Train/Test Predefined?	Challenges
Outex_TC10	24	128 × 128	180	20	160	4320	Yes	Rotation changes (0° angle for training and angles in Θ_1 for testing)
Outex_TC12_000	24	128 × 128	200	20	180	4800	Yes	Illumination variations, rotation changes
Outex_TC12_001	24	128 × 128	200	20	180	4800	Yes	(0° angle for training and angles in Θ_2 for testing)
CUReT	61	200 × 200	46	46	92	5612	No	Illumination changes, small rotations, specularities, shadowing, pose changes
Brodatz	111	215 × 215	9	3	6	999	No	Lack of intraclass variations, medium number of classes
BrodatzRot	111	128 × 128	9	3	6	999	No	Rotation changes, lack of intraclass variations, medium number of classes
UIUC	25	320 × 240	40	20	20	1000	No	Strong scale, rotation and viewpoint changes, nonrigid deformations
UMD	25	320 × 240	40	20	20	1000	No	Small illumination changes, strong scale, rotation and view-point changes
KTH-TIPS2b	11	200 × 200	432	324	108	4752	Yes	Illumination changes, small rotations, small pose changes, scale changes
ALOT	250	384 × 256	100	50	50	25,000	No	Strong illumination changes, large number of classes, rotation changes
Outex_TC40_A	294	128 × 128	180	80	100	52,920	Yes	Rotation changes, large number of classes
Outex_TC40_B	294	128 × 128	180	80	100	52,920	Yes	Illumination changes, rotation changes, large number of classes
Outex_TC40_C	294	128 × 128	180	80	100	52,920	Yes	Illumination changes, rotation changes, large number of classes

Datasets for noise robustness evaluation

Texture dataset	Texture classes	Sample size (pixels)	# Images per class	# Training images in total	# Testing images in total	Challenges
Outex_TC11n	24	128 × 128	20	480 (20*24)	480 (20*24)	Training: illuminants (inca), Rotations (0°)
Outex_TC23n	68	128 × 128	20	1360 (20*68)	1360 (20*68)	Testing: Training images injected with Gaussian Noise
Outex_TC11b	24	128 × 128	20	480 (20*24)	480 (20*24)	Training: illuminants (inca), Rotations (0°)
Outex_TC23b	68	128 × 128	20	1360 (20*68)	1360 (20*68)	Testing: Training images blurred by Gaussian PSF
Outex_TC11s	24	128 × 128	20	480 (20*24)	480 (20*24)	Training: illuminants (inca), Rotations (0°)
Outex_TC23s	68	128 × 128	20	1360 (20*68)	1360 (20*68)	Testing: Training images injected with Salt-and-Pepper
Outex_TC11c	24	128 × 128	20	480 (20*24)	480 (20*24)	Training: illuminants (inca), Rotations (0°)
Outex_TC23c	68	128 × 128	20	1360 (20*68)	1360 (20*68)	Testing: Training images with Random Pixel Corruption

and testing on the remainder. There have been some proposals to use more sophisticated classifiers, such as, support vector machines (SVM), SVM ensembles, decision trees, random forests, for categorizing material surfaces based upon the observed textural features. However, in this work, our focus was on the distinctiveness and robustness of various LBP variants, rather than on the impact of the classifier. Therefore, we limit our study to using the nearest neighbor classifier (NNC) and keep the other components as similar as possible. For ScatNet, we used the same feature presented in [117]. For PCANet and RandNet, we used the parameter settings suggested for texture classification in [120].

3.4. Results

3.4.1. MultiScale analysis

In the milestone work by Ojala et al. [2], it showed that multi-scale $LBP_{r,p}^{riu2}$ generally improved the performance. They also found that rotation invariance is hampered by the crude quantization of the angular space provided by the neighbor set of eight pixels ($p=8$) on radius r, and suggested to use a larger p since the quantization of the angular space is defined by $360°/p$. In the traditional $LBP_{r,p}^{riu2}$, two factors have to be considered for the selection of p. First, there is an upper limit for the number of nonredundant sampling points in the neighborhood to avoid oversampling. Second, an efficient implementation with a lookup table of 2^p entries sets a practical upper limit for p. Therefore, Ojala et al. [2] explored p values up to 24, which requires a lookup table that can be easily managed. Regarding multi-scale analysis, Ojala et al. [2] explored three scales by setting (r,p) as (1,8), (2,16) and (3,24) for realizing the LBP operators, which has been followed by many latter researchers.

However, there is no guarantee that the combination of the three scales (*i.e.* (1,8), (2,16) and (3,24) with spatial support area of size 7×7) produces the optimal operator for texture classification. Features calculated in a local 7×7 neighborhood cannot capture large-scale structures that may be the dominant features of some textures. The operator is not very robust against local changes in the texture, caused for example by varying viewpoints or illumination directions. It is interesting to explore a larger spatial support area for various LBP variants beyond three scales. Therefore, in our experiments, we implemented up to nine scales for most of the LBP variants if possible. The setting of the (r,p) parameters, with computational efficiency reasons taken into consideration, are summarized in Table 7. Most LBP variants adopt "Scheme 1". The reasons for some methods using a different scheme for (r,p) are mainly due to computational expense. For example, in the case of FLBP and NRLBP, the number of neighboring pixels p is constantly set to 8 for any radius r since FLBP is very time consuming at the feature extraction stage even with $p=8$ and NRLBP requires a memory consuming lookup table of size 3^p.

Table 3 presents the results for all LBP methods as a function of neighborhood size. In Table 3, there is a general trend that the classification performance increases with the neighborhood size (*i.e.* the number of scales) and then levels off, except the BSIF method whose performance decreases with the neighborhood size. It is apparent that integrating LBP features at different scales leads to improvement of performance. For most LBP methods, the best performance is achieved with a neighborhood size larger than 7×7. Keep in mind that in the LBP literature, usually only three scales with neighborhood size 7×7 is considered. For some methods such as LBP^{ri}, $FLBP^{riu2}$ and $NRLBP^{riu2}$, the performance first increases and then decreases with neighborhood size. This indicates that the performance is hampered by the crude quantization of the angular space provided by the neighbor set of eight pixels ($p=8$) on large radius. We would like to mention that, the

comparative study done by Fernández et al. [127] with only a neighborhood size 3×3 can be somewhat misleading. For example, LTP^{riu2} and $LoQP^{riu2}$ outperform LBP^{riu2} significantly for neighborhood size 3×3 on Outex_TC10, but the performance of the three methods are very close for a neighborhood size 19×19. In summary, the results in Table 3 clearly indicate the necessity of using larger spatial support area for LBP feature extraction. It is a disadvantage for the LBP variant (such as LBP^{ri}, $FLBP^{riu2}$ and $NRLBP^{riu2}$) that cannot effectively capture the texture information contained in distant neighborhoods.

In the C5 category (Combining LBP with other complementary features), generally we can still observe performance improvement with the increase of number of scales used. However, the improvement is less significant compared with the single type LBP methods (*i.e.* methods in C1 and C2). This is more noticeable on the Outex_TC12 dataset. Note that in the C5 category, the CINIRD method is an exception, since its highest score is achieved with three scales. Based on the results in Table 3, in our following experiments we use the neighborhood size which gives the average highest score for each LBP method.

In summary, we can observe a general trend of performance increase with neighborhood size, with most LBP methods achieving a best performance beyond three scales, clearly indicating the necessity of using larger areas of spatial support for LBP feature extraction.

3.4.2. Global view of the obtained results

The main results for the selected LBP variants and recent deep convolutional network based features are summarized in Table 4, including a comprehensive evaluation of forty features (thirty two LBP type features and eight non-LBP features) on thirteen popular datasets with varying difficulty, computation complexity comparison (including feature extraction time and feature dimensionality), with detailed noise robustness evaluation presented in Tables 5 and 6.

Based on the results in Part I of Table 4, we can make the following observations:

- The highest classification accuracies are given by the methods in C5 (combining LBP with other complementary features) family, but they have as their main drawback that they have higher dimensionality and require more computation time in general.
- The best method in terms of accuracy is SSLBP [92] which gives top 5 classification accuracies on *eight* datasets.[3] The second best method is MRELBP [46] which give top 5 classification accuracies on *seven* datasets. Then come methods CLBP [35] and CINIRD [36], which give top 5 classification accuracies on *six* datasets.
- The best method considering the trade off between classification accuracy and computational complexity is MRELBP [46]. The feature dimensionality of SSLBP is much higher than that of MRELBP. Although the feature extraction in SSLBP is faster than MRELBP, SSLBP requires a fairly slow extra training stage. It is noteworthy that SSLBP (91.31%) significantly outperformed the rest methods on the large scale database Outex_TC40_BC. The reason may be that it learns dominant patterns across multiscale Gaussian scale space. Parallel to MRELBP in [46], GRELBP (Gaussian RELBP) was also proposed. GRELBP achieves 86.11% on Outex_TC40_BC, much higher than MRELBP. Moreover, SSLBP

---

[3] For each dataset, we identify the highest five classification scores. For each method, the number of datasets in which the method achieves a top 5 performance is used as a measure for overall classification performance and stability in terms accuracy.

Table 3

Classification results (%) for various LBP variants on the Outex_TC10 and Outex_TC12 (Outex_TC12_000 and Outex_TC12_001) test suites as a function of neighborhood size (the number scales used for multiscale analysis). For each method, the highest classification accuracies are highlighted in bold for each dataset. LEP filtering support is 65×65. Results (○) are not provided for efficiency reasons.

No.	Method	Outex_TC10 (Rotation invariance)									Outex_TC12 (Illumination and rotation invariance)								
		3×3	5×5	7×7	9×9	11×11	13×13	15×15	17×17	19×19	3×3	5×5	7×7	9×9	11×11	13×13	15×15	17×17	19×19
C1	LBP^{riu2} [2]	84.71	93.44	97.21	98.91	99.01	99.38	99.56	99.66	**99.69**	64.97	82.07	86.79	89.64	89.12	89.72	90.81	91.39	**92.14**
C1	LBP^{ri} [2]	78.91	83.96	**86.69**	86.15	83.46	80.57	78.91	77.47	77.11	70.98	80.60	**83.68**	81.97	80.57	79.19	78.53	78.36	78.06
C3	$ILBP^{riu2}$ [51]	86.72	96.38	98.18	99.04	99.48	99.61	**99.66**	99.64	99.66	74.92	89.85	92.14	91.97	92.28	92.35	92.51	92.89	**93.34**
C3	MBP^{riu2} [52]	80.21	87.40	89.92	92.47	94.24	94.90	95.16	95.21	**95.29**	63.18	73.01	79.71	83.66	84.57	85.09	85.69	86.22	**86.69**
C3	LTP^{riu2} [57]	92.94	97.14	98.54	99.32	99.53	99.74	99.84	99.84	**99.92**	73.59	86.46	90.88	92.08	92.35	92.78	93.25	93.77	**94.28**
C3	$LoQP^{riu2}$ [9]	93.98	97.21	98.44	99.14	99.38	99.53	99.61	99.61	**99.64**	74.95	86.68	91.16	92.95	93.67	94.04	94.48	94.72	**94.91**
C3	$FLBP^{riu2}$ [59]	89.27	94.38	97.53	98.33	**98.78**	97.89	94.84	92.79	90.86	65.59	81.39	88.60	92.29	**92.48**	91.85	89.44	88.02	87.41
C3	$NRLBP^{riu2}$ [60]	89.79	93.78	96.67	97.01	**98.07**	97.81	95.60	95.05	93.44	71.35	83.00	87.05	88.92	89.57	**90.20**	88.78	87.48	86.76
C3	$RLBP^{riu2}$ [61]	84.11	93.49	97.19	99.04	99.17	99.35	99.58	**99.71**	99.66	66.45	83.45	88.01	90.73	89.92	91.02	92.21	92.89	**93.53**
C4	EXLBP [28]	83.52	93.88	97.08	98.70	98.88	98.93	99.48	99.53	**99.64**	69.02	85.34	88.72	89.91	89.59	90.10	91.30	92.15	**93.55**
C4	NTLBP [29]	84.24	91.88	96.15	98.10	98.88	99.19	**99.35**	99.32	99.24	67.06	82.21	88.28	91.61	92.71	93.63	94.88	**95.27**	95.23
C4	$MDLBP^{riu2}$ [77]	–	–	–	–	**99.22**	–	–	–	–	–	–	–	–	**95.64**	–	–	–	–
C4	$PRICoLBP_g$ [75]	–	–	–	–	–	94.48	–	–	–	–	–	–	–	–	92.53	–	–	–
C4	MSJLBP [78]	–	–	96.67	–	–	–	–	–	–	–	–	95.47	–	–	–	–	–	–
C4	DLBP [3]	72.29	88.85	94.79	97.01	97.71	99.17	99.27	99.32	**99.45**	45.06	74.11	81.24	85.53	87.12	90.13	91.18	91.60	**91.97**
C5	disCLBP [80]	89.30	97.47	98.93	99.79	**99.95**	○	○	○	○	75.22	89.80	94.40	96.00	**96.10**	○	○	○	○
C5	LEP [84]	–	–	–	–	–	–	–	–	81.90	–	–	–	–	–	–	–	–	81.46
C5	CLBP [35]	96.72	98.67	99.35	99.45	99.51	99.51	99.51	99.53	**99.58**	91.54	94.48	95.67	**95.78**	95.49	95.39	95.43	95.43	95.42
C5	CINIRD [36]	96.41	99.38	99.66	**99.71**	99.71	99.66	99.64	99.56	99.53	92.08	97.37	**97.57**	97.08	96.52	96.10	96.06	96.05	96.03
C5	BRINT [30]	91.88	96.95	98.52	99.04	99.32	99.32	99.30	**99.40**	99.35	87.48	94.29	96.28	97.16	97.29	97.53	97.71	97.96	**98.13**
C5	MRELBP [46]	–	98.44	–	99.69	–	99.79	–	**99.82**	–	–	96.24	–	99.03	–	99.56	–	**99.57**	–
C5	CLBC [69]	97.45	98.72	99.19	**99.40**	99.40	○	○	○	○	91.56	93.98	95.17	**95.75**	95.61	○	○	○	○
C5	LBP^{riu2}/VAR [2]	96.67	98.02	98.52	**98.78**	98.70	98.72	98.59	98.57	98.52	78.63	85.03	87.72	89.19	89.72	90.03	90.88	91.24	**91.53**
C5	$LBPV^{riu2}$ [89]	91.30	94.35	97.24	98.49	98.93	99.22	**99.27**	99.14	99.11	76.88	86.76	92.72	93.34	93.92	93.81	93.92	**94.03**	94.00
C5	$LBPV^{u2}_{r,p}GM_{PD2}$ [89]	73.07	90.63	96.69	98.20	99.01	99.22	99.27	99.22	**99.32**	74.07	89.59	94.16	95.80	**96.83**	96.79	96.48	96.30	95.91
C5	LBPHF [93]	83.52	92.37	96.59	98.49	99.22	99.38	99.45	99.64	**99.69**	77.58	89.39	92.55	93.74	93.53	93.43	94.27	94.69	**95.01**
C5	CLBPHF [94]	87.42	94.61	98.20	99.01	99.56	99.69	**99.71**	99.71	99.69	78.39	90.29	93.34	94.10	94.07	94.07	94.39	94.61	**94.80**
C5	LBPD [126]	–	–	98.78	–	–	–	–	–	–	–	–	96.67	–	–	–	–	–	–
C5	SSLBP [92]	–	–	–	–	–	–	–	–	99.82	–	–	–	–	–	–	–	–	99.36
C6	RILPQ [103]	–	–	–	–	–	99.58	–	–	–	–	–	–	–	–	97.43	–	–	–
C6	BSIF [85]	–	57.68	52.68	52.97	54.06	56.17	54.19	53.91	–	–	**63.06**	58.98	56.33	58.76	59.38	56.86	58.66	–
C5	LFD [108]	–	–	–	–	–	–	–	–	99.38	–	–	–	–	–	–	–	–	98.46

Table 4
Performance comparison for various LBP variants tested on a number of texture datasets in terms of classification scores (%) and computational complexity (including feature extraction time and feature dimensionality). For each dataset in Part I, the top five classification accuracies are highlighted in bold and the highest score is highlighted in bold and shadow. For each method, the total number of top five classification accuracies in bold are given in the third column from right. For each dataset in Part II, the two highest scores are highlighted in bold. Results ("No") are not provided for efficiency reasons. TC10 (Outex_TC10), TC12 (Outex_TC12) TC40_A and TC40_BC represents Outex_TC40_A and Outex_TC40_BC respectively. The results for Outex_TC12 is the average scores obtained on Outex_TC12_000 and Outex_TC12_001. The results for Outex_TC40_BC is the average scores obtained on Outex_TC40_B and Outex_TC40_C. In the "Time" column, methods labeled with ($\star$) are those need extra computation time at the feature extraction stage. The ($\diamond$) label in LBPD means that although LBPD has low feature dimensionality, it is time consuming in the classification stage since it requires an affine invariant metric in the NNC classification. BRot and KT2b represent BrodatzRot and KTHTIPS2b respectively.

No.	Method # Classes	TC10 (24)	TC12 (24)	CUReT (61)	Brodatz (111)	BRot (111)	UMD (25)	UIUC (25)	KT2b (11)	ALOT (250)	TC40_A (294)	TC40_BC (294)	# Bold	Feat. ext. Time (ms)	Feat. Dim
Part I: Evaluation the performance of representative LBP methods															
C1	LBP^{riu2} [2]	**99.69**	92.14	97.03	90.70	79.22	96.15	88.36	62.69	94.15	94.83	71.72	1	87.2	210
C1	LBP^{ri} [2]	86.69	83.68	95.38	89.93	71.73	94.99	79.03	62.48	93.29	85.84	54.75	0	47.5	108
C3	ILBP^{riu2} [51]	99.66	93.34	94.66	91.66	82.27	96.54	92.84	61.93	95.71	95.15	69.48	0	90.8	420
C3	MBP^{riu2} [52]	95.29	86.69	92.09	87.25	74.57	92.41	80.89	61.49	88.23	84.90	45.46	0	215.6	420
C3	LTP^{riu2} [57]	**99.92**	94.28	96.33	92.41	**83.51**	96.66	93.27	63.45	94.60	96.85	69.14	2	231.8	420
C3	LoQP^{riu2} [9]	99.64	94.91	95.73	91.02	**83.13**	97.36	93.20	63.74	94.31	**96.89**	64.02	2	360.6	840
C3	FLBP^{riu2} [59]	98.78	92.48	95.38	89.02	78.72	94.89	84.03	62.50	No	No	No	0	13,517.4	50
C3	NRLBP^{riu2} [60]	98.07	89.57	94.00	87.42	75.77	93.32	81.10	58.61	87.86	89.93	61.34	0	356.9	50
C3	RLBP^{riu2} [61]	99.66	93.53	**97.20**	91.09	79.59	95.50	88.29	61.20	94.23	94.98	73.75	1	488.6	210
C4	EXLBP [28]	99.64	93.55	96.85	90.19	80.08	95.55	88.29	62.39	95.20	95.39	**74.87**	1	91.3	273
C4	NTLBP [29]	99.32	95.27	96.11	89.31	80.25	95.72	88.13	61.30	94.47	91.70	69.49	0	332.3	388
C4	MDLBP^{riu2} [77]	99.22	95.64	96.92	**93.40**	82.31	97.26	88.05	**66.52**	95.81	89.97	66.99	2	26.3	1000
C4	PRICoLBP$_g$ [75]	94.48	92.53	96.25	92.94	77.00	95.69	80.38	61.17	94.38	89.56	64.16	0	380.4	3540
C4	MSJLBP [78]	96.67	95.47	**97.20**	92.94	79.11	96.53	83.00	65.51	95.65	88.59	60.09	1	854.6	3540
C4	DLBP [3]	99.45	91.97	94.38	88.73	75.04	93.58	83.71	61.72	No	90.63	67.54	0	($\star$)565.3	14,150
C5	disCLBP [80]	**99.95**	96.10	96.98	**93.18**	**83.77**	97.53	94.24	63.83	95.01	**97.54**	74.00	4	($\star$)585.8	7796
C5	LEP [84]	81.90	81.46	88.31	82.64	61.41	91.75	81.80	63.13	89.67	74.97	56.07	0	($\star$)1088.9	520
C5	CLBP [35]	99.45	95.78	**97.33**	92.34	**84.35**	**98.62**	**95.75**	64.18	**96.74**	**96.98**	65.49	6	127.9	3552
C5	CINIRD [36]	99.66	**97.57**	96.60	**93.24**	**85.92**	**98.93**	**94.61**	64.84	**97.21**	96.18	67.70	6	114.6	2200
C5	BRINT [30]	99.35	**98.13**	97.02	90.83	78.77	97.44	93.30	**66.67**	96.13	96.24	**81.85**	3	248.8	1296
C5	MRELBP [46]	**99.82**	**99.58**	97.10	90.86	81.92	98.66	**94.73**	**68.98**	**97.28**	96.20	**78.97**	7	416.6	800
C5	CLBC [69]	99.40	95.75	96.86	90.24	82.17	**98.41**	**94.37**	58.27	96.46	**97.30**	71.57	4	202.9	4168
C5	LBP^{riu2}/VAR [2]	98.52	91.53	96.04	88.71	80.56	95.88	84.44	58.52	93.62	96.47	74.26	0	($\star$)288.8	3360
C5	LBPV^{riu2} [89]	99.27	93.92	95.85	87.63	75.89	93.79	81.98	59.03	91.87	92.88	73.20	0	350.7	158
C5	LBPV$^{u2}_{r,p}$GM$_{PD2}$ [89]	99.22	96.79	95.69	89.07	72.28	93.80	80.74	59.43	No	89.73	74.16	0	350.7	10,034
C5	LBPHF [94]	**99.69**	95.01	**97.51**	92.56	82.22	96.78	89.58	**67.51**	94.91	95.56	71.48	3	99.6	2290
C5	CLBPHF [93]	**99.69**	94.80	97.05	91.95	82.07	97.24	92.55	**68.10**	96.30	96.42	69.63	2	256.2	4580
C5	LBPD [126]	98.78	96.67	94.23	89.74	74.79	92.99	90.98	63.47	92.82	89.96	60.60	0	($\diamond$)54.2	289
C5	SSLBP [92]	**99.82**	**99.36**	**98.79**	89.94	80.03	**98.4)**	**95.40**	65.57	**96.68**	**98.33**	**91.31**	8	($\star$)180.10	2400
C6	RILPQ [103]	99.58	97.43	92.15	91.37	79.59	97.49	91.17	58.75	94.85	90.76	69.33	0	44.8	256
C6	BSIF [85]	57.68	63.06	96.83	**94.48**	48.76	96.10	73.39	63.74	93.09	75.44	60.49	1	55.6	1024
C6	LFD [108]	99.38	**98.46**	97.39	**94.92**	71.65	92.89	87.28	56.65	89.11	88.85	**74.40**	3	289.88	264
Part II: Comparing with deep convolutional network based approaches															
1	MRELBP (SVM) [46]	**99.97**	**99.77**	**99.02**	93.12	85.06	99.36	96.88	77.91	99.08	**97.15**	**77.79**	5	416.6	800
2	ScatNet (PCA) [118]	**99.69**	**99.06**	**99.66**	84.46	75.08	98.40	96.15	68.92	98.03	**94.07**	**77.93**	5	10,883.7	596
3	ScatNet (NNC) [118]	98.59	98.10	95.51	83.03	73.72	93.36	88.64	63.66	85.27	87.55	72.45	0	10,883.7	596
4	PCANet [120] (NNC)	39.87	45.53	92.03	90.89	37.21	90.50	57.70	59.43	88.35	59.49	44.39	0	($\star$)711.8	2048
5	PCANet^{riu2} [120] (NNC)	35.36	40.88	81.48	85.76	29.96	85.67	49.80	52.15	79.77	33.25	21.80	0	($\star$)725.6	80
6	RandNet [120] (NNC)	47.43	52.45	90.87	91.14	40.84	90.87	56.57	60.67	86.94	65.28	42.55	0	711.8	2048
7	RandNet^{riu2} [120] (NNC)	43.54	45.70	80.46	85.59	30.78	87.40	48.20	56.90	73.51	45.14	25.96	0	725.6	80
8	FV-AlexNet (SVM) [115]	67.3	72.3	98.4	98.2	83.1	**99.7**	99.1	77.9	99.1	90.4	51.8	1	($\star$)238.6	32,768
9	FV-VGGM (SVM) [115]	72.8	77.5	98.7	**98.6**	**88.2**	**99.9**	**99.7**	**79.9**	**99.4**	92.6	56.8	7	($\star$)358.8	65,536
10	FV-VGGVD (SVM) [115]	80.0	82.3	**99.0**	**98.7**	**92.1**	**99.9**	**99.8**	**88.2**	**99.5**	93.7	71.6	8	($\star$)2655.4	65,536

needs enough data for training in order to obtain good performance, and lack of enough training data could decrease its performance, such as Brodatz.

- Among the individual (not combined) features, the best performing LBP variant in terms of accuracy is LTP, then followed by LoQP and MDLBP. However, the disadvantage of LTP and LoQP being no longer strictly invariant to gray level transformations is more noticeable when dealing with large scale datasets with illumination changes.
- Considering the classification accuracies of the traditional LBP^{riu2} as baseline, MRELBP is the only LBP variant that outperforms LBP^{riu2} consistently on all the datasets, even though most of the LBP variants claim to address various deficiencies of the traditional LBP^{riu2} descriptor. SSLBP, disCLBP and LBPHF

outperform LBP^{riu2} on all except one datasets.
- Generally, LBP^{ri}, MBP, LEP and BSIF produce the worst overall classification accuracy.
- Some methods could be very time consuming when the dataset scales up. This is the case for FLBP (time consuming at feature extraction stage), DLBP and LBPV$^{u2}_{r,p}$GM$_{PD2}$ (high feature dimensionality).
- According to the overall performance of LBP^{riu2}/VAR, LBPV^{riu2} and LBPV$^{u2}_{r,p}$GM$_{PD2}$, combining LBP^{riu2} with local variance measure is deleterious to classification performance since these combined methods are usually inferior to the single LBP^{riu2} method.
- The idea to combine multiple complementary LBP type features has shown to improve discriminative power in general, which is

Table 5
Classification scores (%) for various methods on Outex_TC11n, Outex_TC11b, Outex_TC23n and Outex_TC23b. The testing images in Outex_TC11n and Outex_TC23n were added with additive Gaussian noise with standard deviation $\sigma = 5$. For Outex_TC11b and Outex_TC23b, the testing images were artificially blurred using Gaussian PSF with standard deviation σ. "C7" represents the class of deep convolutional network based features.

Robust to		Gaussian noise		Gaussian blur							
Dataset		Outex_TC11n	Outex_TC23n	Outex_TC11b				Outex_TC23b			
No.	Method	$\sigma = 5$	$\sigma = 5$	$\sigma = 0.5$	$\sigma = 0.75$	$\sigma = 1$	$\sigma = 1.25$	$\sigma = 0.5$	$\sigma = 0.75$	$\sigma = 1$	$\sigma = 1.25$
C1	LBP^{riu2} [2]	17.7	8.4	94.2	46.5	24.6	12.7	72.4	30.3	16.6	9.7
C1	LBP^{ri} [2]	16.0	7.9	86.9	44.6	26.0	18.1	57.7	28.3	16.0	9.4
C3	ILBP^{riu2} [51]	17.5	10.4	97.3	59.8	29.4	20.4	81.7	43.2	25.1	16.7
C3	MBP^{riu2} [52]	12.1	5.2	85.4	29.0	18.5	11.9	58.7	22.5	13.5	10.6
C3	LTP^{riu2} [57]	13.7	7.7	96.9	58.3	27.3	13.7	77.3	43.1	24.3	13.3
C3	LoQP^{riu2} [9]	14.8	9.3	98.3	62.9	34.2	24.8	82.1	48.9	29.8	16.8
C3	FLBP^{riu2} [59]	17.5	6.3	91.3	39.4	18.1	9.6	62.0	30.5	17.6	8.5
C3	NRLBP^{riu2} [60]	21.7	9.1	93.3	46.0	20.0	9.2	63.2	36.3	20.3	8.8
C3	RLBP^{riu2} [61]	22.1	11.9	95.0	49.8	28.7	16.5	75.4	33.2	18.4	10.7
C4	EXLBP [28]	19.2	10.3	94.0	47.7	28.3	17.1	73.3	32.0	17.8	10.5
C4	NTLBP [29]	24.0	9.0	96.3	49.0	33.1	19.4	80.1	35.7	21.7	14.1
C4	MDLBP^{riu2} [77]	12.5	6.1	100.0	60.2	36.9	23.8	95.7	35.1	20.6	12.2
C4	PRICoLBP$_g$ [75]	15.4	5.6	98.1	50.0	26.5	14.4	81.1	32.5	19.6	11.3
C4	MSJLBP [78]	17.7	4.9	96.0	46.0	26.0	11.9	74.9	28.9	14.8	8.9
C4	DLBP [3]	14.8	8.2	90.4	43.5	21.9	13.1	67.7	31.3	16.5	8.7
C5	disCLBP [80]	25.2	12.3	100.0	70.2	39.4	20.8	95.6	51.0	27.1	14.1
C5	LEP [84]	**91.9**	**76.8**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	99.8
C5	CLBP [35]	11.9	5.6	98.8	74.8	49.6	23.1	86.6	55.4	36.1	21.2
C5	CINIRD [36]	9.4	3.3	98.3	71.5	38.5	21.5	86.2	39.9	19.7	11.0
C5	BRINT [30]	*61.9*	*27.4*	*100.0*	*97.1*	*80.4*	*44.6*	*100.0*	*79.5*	*59.1*	*39.1*
C5	MRELBP [46]	**91.5**	**79.2**	**100.0**	**100.0**	**93.8**	**75.4**	**99.9**	**97.9**	**85.8**	**61.8**
C5	CLBC [69]	26.7	15.4	100.0	87.5	59.6	40.8	94.7	67.1	46.3	31.9
C5	LBP^{riu2}/VAR [2]	27.5	17.5	100.0	73.3	35.8	25.4	92.6	59.0	33.8	22.6
C5	LBPV^{riu2} [89]	27.1	15.4	96.9	52.1	22.3	17.1	73.9	34.3	15.6	8.3
C5	LBPV$^{u2}_{r,p}$GM$_{PD2}$ [89]	40.2	28.7	100.0	79.2	31.5	20.8	98.1	58.0	29.8	17.2
C5	LBPHF [93]	24.4	18.2	99.4	70.2	34.0	20.6	87.0	53.8	29.8	17.3
C5	CLBPHF [94]	20.6	17.5	99.6	81.3	47.9	29.4	85.4	59.2	39.1	25.1
C5	LBPD [126]	24.6	14.8	99.4	85.8	65.2	45.4	87.7	56.0	40.2	30.6
C5	SSLBP [92]	**97.1**	**91.5**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**90.6**
C6	RILPQ [103]	*82.9*	*56.5*	*100.0*	*99.2*	*76.7*	*45.8*	*100.0*	*76.0*	*53.9*	*37.2*
C6	BSIF [85]	45.4	27.1	100.0	87.7	64.8	30.6	100.0	59.8	39.1	25.0
C6	LFD [108]	73.1	45.5	100.0	74.6	26.7	17.5	92.6	47.8	19.3	10.1
C7	ScatNet (PCA) [118]	60.2	31.3	100.0	94.8	80.0	64.6	97.7	72.4	53.0	41.1
C7	ScatNet (NNC) [118]	77.1	45.3	100.0	91.7	68.5	40.2	92.7	60.4	41.9	24.0
C7	PCANet [120]	74.0	50.7	100.0	100.0	86.0	56.9	100.0	99.2	51.9	31.0
C7	PCANet^{riu2} [120]	62.7	43.9	100.0	88.8	52.5	32.5	100.0	64.6	36.8	25.7
C7	RandNet [120]	15.3	6.2	100.0	78.1	56.5	37.4	96.2	40.4	27.7	19.4
C7	RandNet^{riu2} [120]	14.8	5.9	97.8	64.2	42.1	33.3	81.1	37.2	20.6	18.9
C7	FV-AlexNet (SVM) [115]	81.5	46.0	100.0	98.8	87.7	60.4	97.1	82.8	63.6	43.4
C7	FV-VGGM (SVM) [115]	81.5	43.9	100.0	99.0	87.3	60.8	96.5	87.7	65.7	42.4
C7	FV-VGGVD (SVM) [115]	93.1	71.5	100.0	100.0	96.5	89.8	99.6	94.1	83.1	71.8

evidenced by the good performance of methods including SSLBP, MRELBP, CLBP and CINIRD.

3.4.3. Rotation and gray scale invariance

Outex_TC10 and BrodatzRot are two datasets designed for the evaluation of rotation invariance properties. In Outex_TC10, training is done with just one rotation angle and testing with samples of other eight different rotation angles, which allows a more conclusive analysis of the rotation invariance. Each image in BrodatzRot has a random rotation angle. The Brodatz dataset has no rotation variations. From the results on the three datasets in Table 4, we can observe that BSIF gives the best result on Brodatz, but is significantly inferior to all other methods on Outex_TC10 and BrodatzRot. This shows that BSIF feature is discriminative but lack of rotation invariance. LEP and LBP^{ri} are lack of distinctiveness as well as rotation invariance. Although PRICoLBP and MSJLBP are designed to encode rotation invariance at the cooccurrence level, they produce worse performance on Outex_TC10 and BrodatzRot

than LBP^{riu2}. Note that, unlike most other LBP variants, LBPV$^{u2}_{r,p}$GM$_{PD2}$, LBPHF and CLBPHF are designed to achieve rotation invariance globally rather than locally, we can observe that the global invariance scheme used in LBPHF and CLBPHF is more effective than that developed in LBPV$^{u2}_{r,p}$GM$_{PD2}$.

In terms of rotation and gray scale invariance, SSLBP, MRELBP and BRINT have to be emphasized by comparing their results on Outex_TC12 and Outex_TC40_BC. The performance of CINIRD and CLBP on small Outex_TC12 dataset is fairly close to that of MRELBP and SSLBP, but much worse on large Outex_TC40_BC dataset (even worse than that of LBP^{riu2}). LBP^{riu2}/VAR and LBPV^{riu2} show the opposite behavior, performing well on large dataset though not so well on small dataset.

3.4.4. Results on datasets with complex variations

From the results achieved on CUReT, UMD and UIUC in Table 4, we can see that the UIUC dataset is much more challenging for most of the LBP methods tested. Eighteen methods produce a

Table 6

Classification scores (%) for Outex_TC11s, Outex_TC23s, Outex_TC11c and Outex_TC23c. For Outex_TC11s and Outex_TC23s, the testing images were corrupted with random salt and pepper noise with densities ρ. For Outex_TC11c and Outex_TC23s, we corrupted certain percentage of randomly chosen pixels from each of the images, replacing their values with independent and identically distributed samples from a uniform distribution. The corrupted pixels are randomly chosen for each test image, and the locations are unknown to the algorithm. We vary the percentage of corrupted pixels υ in the range {5%, 10%, 20%, 30%, 40%}.

Robustness to	Salt and pepper noise										Random corrupted pixels									
Dataset	Outex_TC11s (24 classes)					Outex_TC23s (68 classes)					Outex_TC11c (24 classes)					Outex_TC23c (68 classes)				
Noise parameter	Noise density ρ					Noise density ρ					Percentage of corrupted pixels υ					Percentage of corrupted pixels υ				
No. Method	5%	15%	30%	40%	50%	5%	15%	30%	40%	50%	5%	10%	20%	30%	40%	5%	10%	20%	30%	40%
C1 LBP$riu2$ [2]	31.7	4.2	4.2	4.4	4.2	11.8	1.5	1.5	1.5	1.5	51.5	8.3	4.2	4.2	4.2	21.3	6.0	1.5	1.5	1.5
C1 LBPri [2]	47.1	10.0	4.2	4.2	4.2	26.5	4.7	2.2	1.5	1.5	60.0	32.5	8.3	8.3	8.1	33.9	12.5	5.7	4.4	1.4
C3 ILBP$riu2$ [51]	27.3	4.2	4.2	4.2	4.2	10.7	2.1	1.5	1.5	1.5	56.7	21.0	8.5	5.2	4.2	21.8	6.9	4.2	3.4	2.9
C3 MBP$riu2$ [52]	31.0	8.3	4.2	4.2	4.2	17.0	2.5	1.5	1.5	1.5	45.0	18.8	8.1	4.2	4.2	23.5	8.0	2.6	1.6	1.5
C3 LTP$riu2$ [57]	9.0	6.3	8.3	8.3	8.3	4.3	3.5	2.9	2.9	2.9	60.0	23.3	4.4	4.2	4.2	21.1	6.6	2.9	2.6	2.9
C3 LoQP$riu2$ [9]	12.1	8.1	6.3	10.4	8.3	5.4	5.8	3.7	3.6	2.9	59.0	24.6	7.5	4.2	4.2	21.3	8.0	3.7	2.9	2.9
C3 FLBP$riu2$ [59]	7.3	4.2	4.2	4.2	4.2	3.7	1.5	1.5	1.5	1.5	63.5	16.2	8.3	8.3	8.3	24.1	5.0	2.9	2.9	4.0
C3 NRLBP$riu2$ [60]	8.8	8.1	8.3	4.2	5.0	2.1	2.9	2.6	1.5	1.5	72.5	41.3	21.9	12.7	7.7	25.1	10.1	5.3	2.6	1.5
C3 RLBP$riu2$ [61]	39.2	4.2	4.2	4.2	4.2	18.5	1.5	1.5	1.5	1.5	64.0	16.7	4.2	4.2	4.2	30.1	10.1	1.5	1.5	1.5
C4 EXLBP [28]	27.3	4.2	4.2	4.2	4.2	12.2	1.5	1.5	1.5	1.5	51.5	11.3	4.2	4.2	4.2	25.3	5.6	1.5	1.5	1.5
C4 NTLBP [29]	74.4	22.1	4.8	5.0	6.3	40.5	4.7	3.8	2.6	2.7	82.5	45.6	11.9	4.2	4.2	49.8	22.9	3.7	1.5	1.5
C4 MDLBP$riu2$ [77]	71.9	13.5	8.3	4.2	4.2	38.2	3.7	2.9	2.5	1.9	94.6	35.4	8.5	8.3	8.3	66.0	14.3	3.7	2.9	2.6
C4 PRICoLBP$_g$ [75]	9.6	5.2	4.2	4.2	4.2	4.2	2.1	1.5	1.5	1.5	31.7	10.0	4.2	4.2	4.2	9.0	3.0	1.5	1.5	1.5
C4 MSJLBP [78]	14.2	8.3	4.4	4.2	4.2	7.1	3.5	1.5	1.5	1.5	32.3	16.7	7.5	4.2	4.2	14.4	5.6	2.7	1.7	1.2
C4 DLBP [3]	29.8	5.4	4.2	4.2	4.2	16.5	4.9	1.5	1.5	1.5	45.2	11.0	4.2	4.2	4.2	21.8	6.3	1.5	1.5	1.5
C5 disCLBP [80]	11.0	8.3	4.2	4.2	4.2	5.1	4.4	1.5	1.5	1.5	57.5	24.6	4.2	4.2	4.2	21.5	5.8	2.6	1.5	1.5
C5 LEP [84]	14.0	5.0	4.6	4.0	4.2	10.1	1.8	1.7	1.4	1.5	86.5	64.0	24.2	12.3	7.1	65.8	28.4	5.6	2.7	1.5
C5 CLBP [35]	17.3	8.3	4.2	4.2	4.2	7.6	2.9	1.5	1.6	1.5	61.9	26.5	4.2	4.2	4.2	28.7	5.7	2.9	3.0	2.9
C5 CINIRD [36]	40.4	4.6	4.2	4.2	4.2	16.2	1.5	1.5	1.5	1.5	60.6	31.9	13.1	7.1	4.2	25.4	11.3	4.4	2.9	2.9
C5 BRINT [30]	30.8	7.1	6.0	4.4	4.2	15.9	1.5	1.5	1.3	1.5	89.0	53.5	17.5	7.3	4.2	62.1	20.4	1.6	1.5	1.5
C5 MRELBP [46]	**100.0**	**100.0**	**100.0**	**85.8**	**50.2**	**100.0**	**99.9**	**94.0**	**54.6**	**19.2**	**100.0**	**100.0**	**100.0**	**99.6**	**90.6**	**99.6**	**99.2**	**96.9**	**89.8**	**57.5**
C5 CLBC [69]	6.9	6.5	4.8	4.2	4.2	8.8	0.4	1.5	1.5	1.5	70.6	39.4	12.5	6.7	4.2	38.3	10.0	0.7	1.5	1.6
C5 LBP$riu2$/VAR [2]	4.4	4.0	4.2	4.2	4.2	3.1	1.8	1.5	2.1	1.5	49.2	13.3	4.2	4.2	4.2	24.4	2.9	1.5	1.5	1.5
C5 LBPV$riu2$ [89]	4.6	4.2	4.2	4.2	4.2	2.8	1.5	1.5	1.5	1.5	17.7	5.0	4.2	4.2	4.2	4.0	1.5	1.5	1.5	1.5
C5 LBPV$u2$GMPD2 [89]	4.2	4.2	4.2	4.2	4.2	2.9	2.0	1.5	1.5	1.5	41.5	4.4	4.2	4.2	4.2	9.0	3.0	1.5	1.5	1.5
C5 LBPHF [93]	18.3	4.2	4.2	4.2	5.2	5.2	1.5	1.5	1.4	1.6	55.6	9.8	4.2	4.2	4.2	24.2	3.4	1.5	1.7	1.0
C5 CLBPHF [94]	14.4	4.2	4.2	4.2	4.2	6.0	2.9	1.6	1.4	1.5	50.0	11.9	4.2	4.2	4.2	23.2	3.4	1.5	0.3	1.5
C5 LBPD [126]	25.2	8.3	4.2	4.2	4.2	10.3	2.9	1.5	1.5	0.1	32.3	18.1	7.1	4.2	4.2	12.5	6.5	2.6	1.5	1.5
C5 SSLBP [92]	29.0	9.6	4.2	4.2	4.2	24.5	2.8	1.5	1.5	1.5	94.4	75.8	40.4	13.5	7.9	78.1	56.3	12.3	3.5	1.3
C6 RILPQ [103]	15.0	4.2	4.2	4.2	4.2	3.2	1.5	1.5	1.5	1.5	62.7	37.7	11.7	7.5	5.4	27.9	8.7	2.6	2.0	1.6
C6 BSIF [85]	18.8	4.2	4.2	4.2	4.2	10.7	1.5	2.1	1.8	1.5	52.7	18.5	9.6	8.3	5.2	20.7	9.5	2.4	1.5	1.5
C6 LFD [108]	4.6	4.2	4.2	4.2	4.2	1.5	1.5	1.8	1.8	1.5	54.0	24.2	6.0	4.2	4.2	14.0	5.1	3.0	2.7	2.6
C7 ScatNet (PCA) [118]	4.6	4.2	4.2	4.2	4.2	1.4	1.5	1.5	1.5	1.5	29.2	12.1	5.2	4.2	4.2	3.5	1.5	1.5	1.5	1.5
C7 ScatNet (NNC) [118]	4.2	4.2	4.2	4.2	4.2	1.5	1.5	1.5	1.5	1.5	56.0	9.8	4.2	4.2	4.2	14.6	3.2	2.9	1.7	1.5
C7 PCANet [120]	14.6	6.7	4.4	4.4	4.8	1.6	1.5	1.5	1.5	1.5	70.6	32.5	11.9	10.4	7.7	19.1	5.1	1.5	1.5	1.5
C7 PCANet^{riu2} [120]	4.8	4.4	4.6	4.6	4.0	2.4	1.5	1.5	1.5	1.5	49.2	20.2	6.5	5.4	4.8	19.9	6.2	2.6	1.4	1.5
C7 RandNet [120]	4.8	4.2	4.2	4.2	4.2	1.5	1.5	1.5	1.5	1.4	10.7	4.9	4.2	4.2	4.2	3.4	1.7	1.5	1.5	1.5
C7 RandNet^{riu2} [120]	4.2	4.2	4.2	4.2	4.3	1.5	1.5	1.5	1.5	1.5	11.5	4.2	4.0	4.7	5.5	2.6	1.7	1.5	1.6	1.5
C7 FV-AlexNet (SVM) [115]	10.4	6.7	6.6	4.2	4.2	2.8	5.0	4.3	3.1	1.5	44.8	29.2	11.2	9.0	6.0	13.5	7.0	8.6	4.1	2.8
C7 FV-VGGM (SVM) [115]	6.2	5.6	3.1	4.2	3.8	2.	1.5	3.5	1.8	2.1	34.4	15.8	12.1	9.2	13.3	10.4	7.8	4.9	5.2	5.0
C7 FV-VGGVD (SVM) [115]	21.0	12.1	6.0	6.5	4.2	10.3	5.2	2.3	1.5	1.8	63.5	51.5	23.1	11.7	10.0	34.3	19.1	9.5	4.4	2.8

classification accuracy below 90% on UIUC. This may be caused by the nonrigid deformations in UIUC. Most of the evaluated methods present clear difference between the results on CUReT and UMD and those on UIUC. However, this is not the case for LTP, LoQP, disCLBP, CLBP, CINIRD, BRINT, MRELBP and CLBC. This clearly shows that these methods are capable of handling affine transformations. For KTHTIPS2b, we can see that MRELBP is the best performing method, and followed by CLBPHF and LBPHF.

3.4.5. Dealing with large-scale datasets

There is a significant performance drop for all the evaluated methods when dealing with large scale dataset (Outex_TC40_BC) with both rotation and illumination variations, except SSLBP [92]. However, when classifying large scale dataset with rotation variations but without illumination variations (i.e. Outex_TC40_A), the performance decrease is much less significant. This is also the case for ALOT. Keep in mind that we are only using NNC classifier, and performance could be further improved with a more sophisticated classifier like SVM. We can observe that the best scores on Outex_TC40_A (with 294 classes) and ALOT (with 250 classes) are achieved by disCLBP and MRELBP, respectively, with their corresponding scores being 97.54% and 97.28%. Surprisingly, SSLBP gives a fairly high classification score of 91.31% on Outex_TC40_BC, significantly higher than the rest. The reason may be that it learns dominant patterns across multi-scale Gaussian scale space. Parallel to MRELBP in [46], GRELBP (Gaussian RELBP) was also proposed. GRELBP achieves 86.11% on Outex_TC40_BC, much higher than MRELBP. It can be inferred that SSLBP has a good generalization ability. One downside is that it requires enough data for training. GRELBP [46] is also a good candidate.

We use LBP^{riu2} baseline to compare the results achieved on the small dataset Outex_TC12 with those on the large dataset Outex_TC40_BC (both datasets have rotation and illumination variations). Some methods present clear performance drop when dealing with datasets with a large number of texture classes, such as LoQP, MSJLBP, CLBP, CINIRD and LBPD. Notice that these methods outperform LBP^{riu2} on small Outex_TC12, and the combination of LBP^{riu2} and VAR gives better performance in the large dataset. Besides, only ten out of the 32 LBP variants perform better than the traditional LBP^{riu2} on the more difficult Outex_TC40_BC dataset and only six LBP variants give consistently better results on all the three large scale datasets, though many of the LBP variants claim to address various deficiencies of the traditional LBP^{riu2} feature.

3.4.6. Noise robustness

In their original papers, half of the selected LBP methods claimed to be noise robust alternatives to the original LBP. The results for evaluation of noise robustness against several noise types are shown in Tables 5 and 6. In both tables, the best performing methods are shown in bold. The results in both tables are consistently strong: the MRELBP has exceptional noise tolerance that could not be matched by any of the other 31 LBP features. It is very clear that the nonlinear, robust behavior of the local medians leads MRELBP to be the clear winner in noise robustness, particularly in the cases of salt and pepper noise and random pixel corruption. The classification results are particularly impressive keeping in mind that the training images were all noise free, which makes the problem very difficult. As shown in Table 4, MRELBP gives impressive performance on all of the thirteen datasets (without noise). SSLBP [92] demonstrates high robustness to random Gaussian noise and image blur due to the multi-scale Gaussian pre-filtering step, but fails in other noise situations. It is noteworthy that SSLBP performs generally the best in noise free situations. In contrast, although CLBP and CINIRD produce overall good performance on all noise-free datasets, their performance in

noisy situations is relatively poor, especially when the noise level is high.

From Table 5 alone, we can observe that the best performing method is LEP, followed by SSLBP and MRELBP. The performance of LEP, SSLBP and MRELBP for random Gaussian noise is very close. However, it must be remembered that the performance of LEP in normal (noise free) situations is fairly poor. RILPQ and BRINT also demonstrate some level of noise robustness. Note that, although RILPQ is particularly designed to address image blur, it is outperformed by LEP, SSLBP and MRELBP.

Table 6 presents the results for salt and pepper noise and random pixel corruption respectively. As the noise level increases, it is expected that the performance of all LBP methods level out to the accuracy of a random classification, i.e., a classification accuracy of (1/number of classes). We can observe from Table 6 that MRELBP stands out as the clear winner despite of 31 competitors, performing very well (above 90%) up to 30 percent random corruption in both noise situations. There are difficult noise levels where MRELBP can still offer strong performance, but where not a single other method delivers acceptable results. The performance of all other methods drops quickly with increasing levels of noise and soon reach the accuracy of a random guess.

It can also be observed from Table 6 that, for low levels of salt and pepper noise, NTLBP and MDLBP outperform the others, which show similar poor performance. Whereas for low levels of random pixel corruption, SSLBP, NTLBP, MDLBP, LEP and BRINT achieve relatively better performance.

3.4.7. Comparison of LBP methods and non-LBP methods

The results can be found in Tables 4–6. The most robust method is MRELBP [46] which gives the best overall performance, considering the trade off between classification accuracy, computational complexity and robustness to several types of noise. In normal situations, SSLBP and MRELBP even perform generally better than the recent well-known deep convolutional networks based approach—ScatNet [119]. Keep in mind that the expensive computational cost of ScatNet is a severe drawback. The MRELBP benefits from its sampling scheme the spatial domain spanned by which is much larger than by many other LBP variants. This is likely to result in better discrimination capability. More importantly, instead of applying the standard thresholding to the raw pixel values, MRELBP applies it to the local medians, which works surprisingly robustly.

For the noise-free results of Table 4, we can clearly observe the best performing methods as SSLBP [92], MRELBP [46], CLBP [35], CINIRD [36], ScatNet (PCA) [117,118] and FV-CNN [115]. Among these six methods, clearly the feature extraction time of ScatNet is much more longer than others and represents a significant drawback. The feature dimensionality of CLBP, ELBP and SSLBP are relatively high, with the FV-CNN at an extremely high feature dimension. A serious shortcoming of PCANet and RandNet is their lack of rotation invariance.

If the textures have very large within-class appearance variations, due to view and scale variations and combined texture categories as in KTHTIPS2b, then the FV-VGGVD clearly performs the best. Noticeably MRELBP also gives impressive results on KTHTIPS2b. Nevertheless, from the Outex results it can be observed that FV-CNN is relatively weak on rotation invariance, despite FV-CNN methods using data augmentation to explore multi-scale information. Moreover, FV-CNN is computationally expensive, making it unfeasible to run in real-time embedded systems with low-power constraints.

Finally, from Table 4, the best scores on datasets Outex_TC10, Outex_TC12 and CUReT are 99.95%, 99.58% and 99.66, nearly perfect scores even with simple NNC classification. Especially for Outex_TC10, fifteen methods give scores higher than 99%, leaving

Table 7

Summary of various LBP methods used in our experimental study. Different schemes for parameters (r,p) are defined. Sch 1: (1,8), (2,16), $(r, 24)$ for $3 \leq r \leq 9$; Sch 2: $(r, 8)$, $r = 1, …, 9$; Sch 3: (1,8), $(r, 24)$ for $2 \leq r \leq 9$; Sch 4: (2,8); Sch 5: (1,8), (3,8) and (5,8); Sch 6: $(r, 8)$, $r = 2, 4, 6, 8$. Sch 7: (1,8), (2,16), (3,24), (4,28), (5,32), (6,36), (7,40), (8,44), (9,48); Sch 8: (3,24), (9,24). "Partial" in the "Noise Robust?" column means "robust to random Gaussian white noise and blur but highly sensitive to salt and pepper and random pixel corruption". Those with ($\star$) in the "Optimal Operator Size" column represent the size of the receptive field, meaning much larger input image size is required. In the "Relative Performance" column, we consider the classification performance of LBP as baseline and use ★ and ☆ to represent better and worse than baseline respectively. The "Parameters Setting" column lists default parameters (Notations appeared in the original work) for each method, as suggested in the original publications. Readers are referred to the original papers to understand the parameter notations.

No.	Method	(r,p)	Encoding scheme	Needs training?	Optimal operator size	Feature extraction	Feature dimension	Noise robust?	Rotation invariant?	Monotonic illumination invariant?	Relative performance	Involved parameters setting	Distance metric	Code avail.
C1	LBP^{riu2}	Sch 1	riu2		19×19	Very fast	210	No	Yes	Yes	Baseline	None	Chi Square	√
C1	LBP^{ri}	Sch 2	ri		7×7	Very fast	108	No	Yes	Yes	☆☆☆	None	Chi Square	√
C3	ILBP^{riu2}	Sch 1	riu2		19×19	Very fast	420	No	Yes	Yes	Similar	None	Chi Square	
C3	MBP^{riu2}	Sch 1	riu2		19×19	Fast	420	No	Yes	No	☆☆	None	Chi Square	
C3	LTP^{riu2}	Sch 1	riu2		19×19	Fast	420	No	Yes	No	★	$t = 0.3$	Chi Square	
C3	LoQP^{riu2}	Sch 1	riu2		19×19	Fast	840	No	Yes	No	★	$\tau_1 = 0.2$, $\tau_2 = 0.4$	Chi Square	
C3	FLBP^{riu2}	Sch 2	riu2		11×11	Very slow	50	No	Yes	No	☆☆	$T = 0.3$	Chi Square	
C3	NRLBP^{riu2}	Sch 2	riu2		11×11	Fast	50	No	Yes	No	☆☆	$t = 0.3$		
C3	RLBP^{riu2}	Sch 1	riu2		19×19	Moderate	210	No	Yes	Yes	Similar	None	Chi Square	
C4	NELBP	Sch 1	Reported		19×19	Very fast	273	No	Yes	Yes	Similar	None	Chi Square	
C4	NTLBP	Sch 1	Reported		17×17	Fast	388	No	Yes	Yes	☆	$k=1$ $(r=1)$, $k=3$ $(r=2)$, $k=4$ $(r > 2)$	Chi Square	
C4	MDLBP^{riu2}	Sch 5	riu2		11×11	Very fast	1000	No	Yes	Yes	★	None	Chi Square	
C4	PRICoLBP$_g$	Sch 4	Reported		13×13	Fast	3540	No	Somewhat	Yes	☆	$(a_i, b_i) = \{(2, 0), (3, 0), (6, 0), (0, 2), (0, 3), (0, 6)\}$	Chi Square	√
C4	MSJLBP	Sch 5	Reported		7×7	Moderate	3540	No	Somewhat	Yes	☆	None	Chi Square	
C4	DLBP	Sch 1	Reported	√	19×19	Moderate	14,150	No	Yes	Yes	☆☆	$K_{80\%}$	Chi Square	
C5	disCLBP	Sch 1	Reported	√	11×11	Moderate	7796	No	Yes	Yes	★	DomPattern Ratio $n = 90\%$	Chi Square	√
C5	LEP	PreFiltering	ri	√	32×32	Fast	520	Partial	No	No	☆☆	LEP$^{hb}_{6,3}$, $L=4$	Chi Square	
C5	CLBP	Sch 1	riu2		9×9	Fast	3552	No	Yes	Yes	★★	CLBP_C/S/M	Chi Square	√
C5	CINIRD	Sch 1	riu2		7×7	Fast	2200	No	Yes	Yes	★★	LBP_CI/NI/RD	Chi Square	√
C5	BRINT	Sch 3	ri		19×19	Fast	1296	Partial	Yes	Yes	★	None	Chi Square	√
C5	MRELBP	Sch 6	riu2		17×17	Fast	800	Yes	Yes	Yes	★★★	$\omega_c = 3$, $r \in \{2, 4, 6, 8\}$	Chi Square	√
												$\omega_r \in \{3, 5, 7, 9\}$	Chi Square	√
C5	LBP^{riu2}/VAR	Sch 1	riu2	√	19×19	Moderate	3360	No	Yes	Yes	★	VAR bins 16	Chi	

Table 7 (continued)

No.	Method	(r,p)	Encoding scheme	Needs training?	Optimal operator size	Feature extraction	Feature dimension	Noise robust?	Rotation invariant?	Monotonic illumination invariant?	Relative performance	Involved parameters setting	Distance metric	Code avail.
C5	$LBPV_{r,p}^{riu2}$	Sch 1	riu2		15 × 15	Moderate	158	No	Yes	Yes	☆	None	Square Chi Square	
C5	$LBPV_{r,p}^{u2}GM_{PD2}$	Sch 1	Reported		13 × 13	Moderate	10034	No	Yes	Yes	☆☆	None	Chi Square	
C5	CLBPHF	Sch 1	u2		19 × 19	Fast	4580	Partial	Yes	Yes	★★	CLBPHF_S_M	L1	√
C5	LBPD	PreFiltering	Reported		7 × 7	Fast	289	Partial	Yes	Yes	☆	Section 5.B in [126]	Affine Invariant	√
C5	SSLBP	Sch 8	Reported	√	19 × 19	Fast	2400	No	Yes	Yes	★★★	$\sigma = 2^{0.25}$, $L=4$, $K=600$	Chi Square	√
C6	RILPQ	PreFiltering	Reported		13 × 13	Fast	256	Partial	Yes	Yes	★	Section 4 in [103]	Chi Square	√
C6	BSIF	PreFiltering	lbp		5 × 5	Very fast	1024	Partial	No	No	☆☆☆	FilterSize 5 × 5, 10 bits	Chi Square	√
C6	LFD	Sch 7	N/A		19 × 19	fast	264	Partial	Yes	Yes	☆☆	Section 2.E in [108], HLO^{PH} radius 51; HLO^{PHD} radius 5	L1	√
C7	PCANet	Multistage filtering, binarizing	Reported	√	5 × 5	Moderate	2048	Partial	No	No	☆☆☆	$k_1 = k_2 = 5$	Chi Square	√
C7	$PCANet^{riu2}$		riu2	√	5 × 5	Moderate	80	Partial	No	No	☆☆☆	$L_1 = L_2 = 8$		
C7	RandNet		Reported		5 × 5	Moderate	2048	No	No	No	☆☆☆	B=ImageSize		
C7	$RandNet^{riu2}$		riu2		5 × 5	Moderate	80	No	No	No	☆☆☆			
C7	ScatNet		N/A		32 × 32	Very slow	596	Partial	Yes	Yes	★★	$m = 2, J = 5, K = 8$	Euclidean	√
C7	AlexNet+FV	Repeating filtering, nonlinear, pooling	N/A	√	163 × 163(★)	Moderate	32768	Partial	No	No	★★★			
C7	VGG-M+FV		N/A	√	139 × 139(★)	Moderate	65536	Partial	No	No	★★★	# VisWords K=64	Linear SVM	√
C7	VGG-VD+FV		N/A	√	252 × 252(★)	Slow	65536	Partial	No	No	★★★			

Table 8
A taxonomy of different LBP variants (Part I). The methods are grouped to contrast their roles in feature extraction. Methods selected for our experimental test are shown in bold face.

Class	Category	Complete name	Abbr. name	Applications	Year	Reference
Class C1	**Traditional LBP**	Local Binary Pattern	LBP	Texture Classification	1996	[26]
		Uniform Local Binary Pattern	$LBP_{r,p}^{u2}$	Texture Classification	2002	[2]
		Rotation Invariant Local Binary Pattern	$LBP_{r,p}^{ri}$	Texture Classification	2000	[27]
		Rotation Invariant Uniform Local Binary Pattern	$LBP_{r,p}^{riu2}$	Texture Classification	2002	[2]
Class C2	**Neighborhood topology and sampling**	Exploiting Anisotropic Information				
		Elongated local Binary Pattern	EBP	Face Recognition	2007	[31]
		Elongated Local Ternary Pattern	ETP	Medial Image Analysis	2010	[9]
		Elongated Local Quinary Pattern	EQP	Medial Image Analysis	2010	[9]
		Ensembles of Local Quinary Pattern	ENS-LoQP	Medial Image Analysis	2010	[9]
		Local Line Binary Pattern	LLBP	Face Recognition	2009	[32]
		Local Quantized Pattern	LQP	Visual Recognition	2010	[33,34]
		LBP variants based on local differences or magnitudes				
		Completed Local Binary Pattern_Magnitude	CLBP_M	Texture Classification	2010	[35]
		Geometric Local Binary Pattern	GLBP	Texture Analysis	2010	[40]
		Center Symmetric Local Binary Pattern	CS-LBP	Image Matching	2009	[37,6]
		Radial Difference Local Binary Pattern	RD-LBP	Texture Classification	2012	[36]
		Angular Difference Local Binary Pattern	AD-LBP	Texture Classification	2012	[36]
		Local Neighborhood Intensity Relationship Pattern	LNIRP	Texture Classification	2013	[39]
		Transition Local Binary Pattern	tLBP	Face Recognition	2010	[13]
		Direction coded Local Binary Pattern	dLBP	Face Detection	2010	[13]
		Binary Gradient Contour	BGC	Texture Classification	2011	[38]
		Local Derivative Pattern	LDP	Face Recognition	2010	[41]
		Patch-based LBP Methods				
		Local Binary Pattern Filtering	LBPF	Texture Classification	2003	[42]
		Multi-Block Local Binary Pattern	MBLBP	Face Recognition	2007	[43]
		Three Patch Local Binary Pattern	TPLBP	Face Recognition	2011	[44]
		Patterns of Oriented Edge Magnitudes	POEM	Face Recognition	2012	[45]
		Four Patch Local Binary Pattern	FPLBP	Face Recognition	2011	[44]
		Binary Rotation Invariant and Noise Tolerant_Sign	BRINT_S	Texture Classification	2013	[30]
		Binary Rotation Invariant and Noise Tolerant_Magnitude	BRINT_M	Texture Classification	2013	[30]
		LBP_Pixel To Patch	LBP_PTP	Texture Classification	2013	[39]
		LNIRP_Pixel To Patch	LNIRP_PTP	Texture Classification	2013	[39]
		Radial Difference LBP_Sign	RDLBP_S	Face Recognition	2014	[128]
		Radial Difference LBP_Magnitude	RDLBP_M	Face Recognition	2014	[128]
		Angular Difference LBP_Sign	ADLBP_S	Face Recognition	2014	[128]
		Angular Difference LBP_Magnitude	ADLBP_M	Face Recognition	2014	[128]
		Median Robust Extended LBP_Neigborhood Intensity	MRELBP_NI	Texture Classification	2016	[46]
		Binary Robust Independent Elementary Features	BRIEF	Image Matching	2012	[47]
		Binary Robust Invariant Scalable Keypoints	BRISK	LIP Detection	2011	[48]
		Fast REtinA Keypoint	FREAK	LIP Detection	2012	[49]
		Local Difference Binary	LDB	Image Matching	2013	[129]
Class C3	**Thresholding and Quantization**	Neighborhood Intensity Local Binary Pattern	NI-LBP	Texture Classification	2012	[36]
		Improved Local Binary Pattern	ILBP	Face Recognition	2004	[51]
		Texture Spectrum	TS	Texture Classification	1991	[54–56]
		Uniformly-sampled Thresholds for Generalized LBP	UTGLBP	Texture Classification	2013	[53]
		Threshold Modified Local Binary Pattern	TMLBP	Background Subtraction	2006	[15]
		Scale Invariant Local Ternary Pattern	SILTP	Background Subtraction	2010	[4]
		Local Tetra Patterns	LTrP	Image Retrieval	2012	[130]
		Centralized Local Binary Pattern	CBP	Face Expression Analysis	2008	[131]
		Median Local Binary Pattern	MBP	Texture Classification	2007	[52]
		Local Ternary Pattern	LTP	Face Recognition	2007	[57]
		Local Quinary Pattern	LoQP	Texture Classification	2010	[9]
		Bayesian Local Binary Pattern	BLBP	Texture Classification	2008	[132]
		Soft Local Binary Pattern	SoftLBP	Texture Classification	2008	[58]
		Fuzzy Local Binary Pattern	FLBP	Texture Classification	2008	[59]
		Noise Resistant Local Binary Pattern	NRLBP	Texture Classification	2013	[60]
		Robust Local Binary Pattern	RLBP	Texture Classification	2013	[61]
		Extended Local Binary Pattern	ELBP	Face Recognition	2006	[62,63]
		Improved Local Ternary Pattern	ILTP	Medial Image Analysis	2010	[10]

essentially no room for improvement. Because of that saturation, and because most LBP variants have not been evaluated in recognizing a large number of texture classes, we prepared the new Outex_TC40 benchmark test suite with 294 texture classes, where the results are significantly more spread out.

3.4.8. Computational complexity comparison

The feature extraction time and the feature dimensionality (as shown in Table 4) are two factors to determine the computational cost of the LBP methods. The computation time listed in Table 4 is the average time spent by each method to generate its multi-scale feature. All the methods were implemented with MATLAB 2010b and the experiments were conducted on a laptop with 2.9 GHz Intel Duo CPU and 16 GB RAM. We first measured the feature extraction time costed on each image with a size of 128×128, and the average time on 480 individual images are then used to measure the computational time for each method. Note that, the

Table 9
A taxonomy of different LBP variants (Part II).

Class	Category	Complete name	Abbr. name	Applications	Year	Reference
Class C4	**Encoding and regrouping**	Heuristically Grouping LBPs into Classes				
		Symmetric Uniform Local Binary Pattern	SULBP	Face Recognition	2005	[64]
		Complementary Uniform Local Binary Pattern	CULBP	Object Detection	2010	[65,66]
		semantic Local Binary Pattern	sLBP	Human Detection	2008	[67]
		Hamming Local Binary Pattern	HLBP	Face Recognition	2007	[72]
		Novel Extended Local Binary Pattern	NELBP	Texture Classification	2008	[28]
		Number Local Binary Pattern	NLBP	Texture Classification	2011	[70]
		Rotation, Symmetry and Complement Invariant LBP	RSCILBP	Texture Classification	2011	[71]
		Sorted Local Binary Pattern	SLBP	Texture Classification	2012	[68]
		Local Binary Count	LBC	Texture Classification	2012	[69]
		Noise Tolerant Local Binary Pattern	NTLBP	Texture Classification	2012	[29]
		sorted **c**onsecutive Local Binary Pattern_Sign	scLBP_S	Texture Classification	2015	[73]
		Cooccurrence of LBPs				
		Cooccurrence of Adjacent Local Binary Pattern	CoALBP	Texture Classification	2012	[133]
		Rotation Invariant Cooccurrence of adjacent LBP	RICoLBP	Texture Classification	2012	[134]
		Multi-dimensional Local Binary Pattern	MDLBP	Texture Classification	2012	[77]
		Hierarchical MultiScale Local Binary Pattern	HMSLBP	Face Recognition	2010	[79]
		Pairwise Rotation Invariant Co-occurrence LBP	PRICoLBP	Texture Classification	2012	[76,75]
		Cooccurrence of Uniform LBP	UUCoLBP	Texture Classification	2012	[76,75]
		Cooccurrence of Rotation invariant Uniform LBP	RURUCoLBP	Texture Classification	2012	[76,75]
		Multiscale Joint encoding of Local Binary Patterns	MSJLBP	Texture Classification	2013	[78]
		Learning Discriminative LBPs				
		Dominant Local Binary Pattern	DLBP	Texture Classification	2009	[3]
		Discriminative Local Binary Pattern	disLBP	Texture Classification	2012	[80]
		Learning discriminative LBP patterns with AdaBoost	ABLBP	Face Expression Analysis	2008	[82]
		Labeled Dominant RICoLBP	LDRICoLBP	Texture Classification	2012	[81]
		Median Robust Extended LBP_Radial Difference	MRELBP_RD	Texture Classification	2016	[46]
		Scale Selective Local Binary Pattern_Center and Sign	SSLBP_CS	Texture Classification	2016	[92]
Class C5	**Combining with complementary features**	Preprocessing with filtering				
		Local Gabor Binary Pattern	LGBP	Face Recognition	2005	[83]
		Gabor Local Quantized Pattern	GLQP	Face Recognition	2012	[34]
		Gabor Local Derivative Pattern	GLDP	Face Recognition	2012	[41]
		Pyramid Local Binary Pattern	PLBP	Texture Classification	2011	[86]
		Scale and Rotation Invariant SubUniform LBP	$LBP_{r,p}^{sri-su2}$	Texture Classification	2012	[88]
		Weighted Rotation and Scale Invariant LBP	WRSI_LBP	Texture Classification	2015	[90]
		Local Energy Pattern	LEP	Texture Classification	2013	[84]
		Binarized Statistical Image Features	BSIF	Face Recognition	2012	[85]
		Local Radon Binary Pattern	LRBP	Face Recognition	2012	[87]
		Combining multiple LBP type features				
		CLBP_C, **CLBP_S and CLBP_M**	CLBP	Texture Classification	2010	[35]
		Compound Local Binary Pattern	CMLBP	Texture Classification	2010	[91]
		CILBP, NILBP and RDLBP	CINIRD	Texture Classification	2012	[135,36]
		BRINT_C, **BRINT_S and BRINT_M**	BRINT	Texture Classification	2013	[30]
		CLBC_C, **CLBC_S and CLBC_M**	CLBC	Texture Classification	2012	[69]
		MRELBP_CI, MRELBP_NI and MRELBP_RD	MRELBP	Texture Classification	2016	[46]
		scLBP_S, scLBP_M $\pm$ and CLBP_C with kd-tree	scLBP	Texture Classification	2015	[73]
		SSLBP_CS and SSLBP_CM	SSLBP	Texture Classification	2016	[92]
		disCLBP_S and disCLBP_M	disCLBP(S+M)	Texture Classification	2012	[80]
		Combining LBP with other features				
		LBP and VAR	LBP/VAR	Texture Classification	2002	[2]
		Local Binary Pattern Variance	LBPV	Texture Classification	2010	[89]
		Adaptive Local Binary Pattern	ALBP	Texture Classification	2010	[95]
		Discriminative Robust Local Binary Pattern	DRLBP	Object Recognition	2014	[14]
		Discriminative Robust Local Ternary Pattern	DRLTP	Object Recognition	2014	[14]
		LBP and Fourier Features	LBPHF	Texture Classification	2009	[93]
		CLBP and Fourier Features	LBPHF_S_M	Texture Classification	2012	[94]
		DLBP and Gabor Filters	DLBP_Gabor	Texture Classification	2009	[3]
		LTP and Gabor	LTP and Gabor	Face Recognition	2011	[57]
		Local Binary Pattern Difference	LBPD	Texture Classification	2012	[126]
		LBP and Dominant Neighborhood Structure	LBP and DNS	Texture Classification	2011	[96]
		LBP and SIFT	LBP and SIFT	Face Recognition	2012	[99]
		LBP and Local Phase Quantization	LBP and LPQ	Texture Classification	2012	[100]
		Histogram of Gradients-Local Binary Pattern	HOG-LBP	Human Detection	2009	[97]
		Haar Local Binary Pattern	HLBP	Face Detection	2008	[136]
Class C6	Methods inspired by LBP	**Local Phase Quantization**	LPQ	Texture Classification	2008	[102]
		Local Frequency Descriptor	LFD	Texture Recognition	2013	[107,108]
		Weber Law Descriptor	WLD	Texture and Face Recognition	2009	[104]

Table 9 (*continued*)

Class	Category	Complete name	Abbr. name	Applications	Year	Reference
		Histogram of Gabor Phase Patterns	HGPP	Texture and Face Recognition	2007	[105]
		Local Higher-Order Statistics	LHS	Texture and Face Recognition	2012	[106]
		Discriminant Face Descriptor	DFD	Face Recognition	2014	[109]

reported time does not include the training time for those methods labeled with (⋆) in Table 4. The reported feature dimensionality is the final dimensionality of each method given to the NNC classifier.

FLBP and ScatNet are the most computationally expensive methods in feature extraction, followed by FV-VGGVD. The time costs of feature extraction for FLBP and ScatNet are 125 times and 155 times that of LBP^{riu2}, respectively. Compared with LBP^{riu2}, most of the remaining methods do not introduce much computation overhead at the feature extraction stage. In terms of feature dimensionality, FV-CNN is extreme. DLBP and LBPV$_{r,p}^{u2}$GM$_{PD2}$ are high in dimensionality, with the dimensionality of *dis*CLBP, CLBPHF, CLBP, PRICoLBP, MSJLBP, SSLBP, PCANet and RandNet also relatively high.

Finally, we provide a table (Table 7) to summarize the properties of all evaluated methods including recommended operator size, feature dimensionality, classification performance, parameters setting, robustness to image variations, tolerance of image noise and computational complexity.

4. Conclusions

This paper presents a review on LBP variants proposed in the literature. A comprehensive study on the motivations and characteristics of various LBP variants and their connections is provided. Based on their relations, a taxonomy for the LBP variants is proposed to classify these methods according to their roles in feature extraction. Merits and demerits of the LBP method categories are analyzed. Even more importantly, a total of 40 methods including 32 LBP variants and 8 recent deep convolutional network based features were applied to thirteen datasets, designed to test and stress an exceptional range of class types, image sizes and disturbance invariance. Computational complexity comparison is also provided for all 40 evaluated features.

We have seen how various LBP variants yield different results for different datasets. The large-scale performance evaluation have demonstrated the strengths and limitations of each LBP variants, facilitating better understanding of the state of the art LBP approaches and providing guidelines for practitioners to decide which features to use according to their requirements. In addition to the classification accuracy issue are the issues of noise robustness and computational complexity. If we take the classification accuracy, noise robustness and computational complexity into consideration, several remarks and guidelines can be obtained based on the overall performance from the experiments:

- One of our main findings is the best overall performance is obtained for the MRELBP when distinctiveness, robustness and computational complexity are all taken into consideration. If robustness to noise situations such as salt and pepper or random pixel corruption is not necessary, SSLBP is also a good choice.
- Both micro- and macro-structures are important for texture description, since most LBP variants achieve their best performance beyond three scales (operator size larger than 7×7). In future

work, researchers should take this into consideration for fair comparison. Furthermore, researchers should use a large spatial support when designing new features, such as SSLBP and MRELBP.
- An LBP variant is more robust to Gaussian white noise and blur when a pre-filtering step is involved, however it does not necessarily guarantee good discriminability (*e.g.* LEP) and robustness to other noise types (*e.g.* salt and pepper).
- Combination of multiple complementary texture descriptors turns out to be more powerful than single texture descriptor.
- Excellent results are obtained with FV-CNN for most test sets, but lack some robustness to noise and rotations. The role of Fisher Vector is very important and should be considered also with LBP methods in future studies.
- It is possible that a classic CNN network could learn how to explore the properties of texture images more efficiently when trained on a very large texture dataset (similar to ImageNet). Unfortunately, to the best of our knowledge, such a database does not exist. Therefore, this comes to the truly important question: What makes a good large scale texture dataset? This deserves more future attention.
- Based on our study, the work on CNNs for texture recognition mainly focuses on the domain transferability of CNNs. For texture, it is possible that simple networks (*e.g.* ScatNet) might be enough to achieve similar or better results on texture datasets. Instead of devoting to design more and more complex networks, we feel that designing simple and efficient networks is important for problems such as mobile computing. Therefore, in the future, it is an interesting research topic to combine LBP variants with deep convolutional networks or even as building blocks of novel cascaded deep convolutional network architectures.

The experiments reported in this paper have taken months of computer time on dozens of powerful modern workstations. The time consumption is an indication of the problem with several of the approaches—they are very computationally complex. It must be remembered that one of the most attracting advantages of LBP is its low computational complexity. Therefore future research of developing more powerful LBP variant should inherit this advantage. Our work strongly suggests that future research work should be performed in this area, and more efficient and effective methods are required to handle large scale dataset acquired in the wild. We note that the above conclusions may not directly apply to applications such as texture segmentation, face detection, face recognition, tracking or pedestrian detection. Perhaps some LBP variants should be considered in problems where small operator sizes should be used.

Acknowledgments

References

[1] M. Pietikäinen, A. Hadid, G. Zhao, T. Ahonen, Computer Vision Using Local Binary Patterns, Springer, London, UK, 2011.

[2] T. Ojala, M. Pietikäinen, T. Maenpää, Multiresolution gray-scale and rotation invariant texture classification with local binary patterns, IEEE Trans. Pattern Anal. Mach. Intell. 24 (7) (2002) 971–987.

[3] S. Liao, M. Law, A. Chung, Dominant local binary patterns for texture classification, IEEE Trans. Image Process. 18 (5) (2009) 1107–1118.

[4] S. Liao, G. Zhao, V. Kellokumpu, M. Pietikäinen, S. Li, Modeling pixel process with scale invariant local patterns for background subtraction in complex scenes, in: IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), 2010, pp. 1301–1306.

[5] G. Zhao, M. Pietikäinen, Dynamic texture recognition using local binary patterns with an application to facial expressions, IEEE Trans. Pattern Anal. Mach. Intell. 29 (6) (2007) 915–928.

[6] M. Heikkilä, M. Pietikäinen, C. Schmid, Description of interest regions with local binary patterns, Pattern Recognit. 42 (3) (2009) 425–436.

[7] O. Silven, M. Niskanen, H. Kauppinen, Wood inspection with nonsupervised clustering, Mach. Vis. Appl. 13 (2003) 275–285.

[8] N. Doshi, G. Schaefer, A comprehensive benchmark of local binary pattern algorithms for texture retrieval, in: International Conference on Pattern Recognition (ICPR), 2012, pp. 2760–2763.

[9] L. Nanni, A. Lumini, S. Brahnam, Local binary patterns variants as texture descriptors for medical image analysis, Artif. Intell. Med. 49 (2) (2010) 117–125.

[10] L. Nanni, S. Brahnam, A. Lumini, A local approach based on a local binary patterns variant texture descriptor for classifying pain states, Expert Syst. Appl. 37 (12) (2010) 7888–7894.

[11] T. Ahonen, A. Hadid, M. Pietikäinen, Face description with local binary patterns: application to face recognition, IEEE Trans. Pattern Anal. Mach. Intell. 28 (12) (2006) 2037–2041.

[12] V. Kellokumpu, G. Zhao, M. Pietikäinen, Human activity recognition using a dynamic texture based method, in: British Machine Vision Conference, 2008.

[13] J. Trefny, J. Matas, Extended set of local binary patterns for rapid object detection, in: Computer Vision Winter Workshop 2010, 2010.

[14] A. Satpathy, X. Jiang, H. Eng, Lbp based edge texture features for object recognition, IEEE Trans. Image Process. 23 (5) (2014) 1953–1964.

[15] M. Heikkilä, M. Pietikäinen, A texture-based method for modeling the background and detecting moving objects, IEEE Trans. Pattern Anal. Mach. Intell. 28 (4) (2006) 657–662.

[16] S. Lazebnik, C. Schmid, J. Ponce, A sparse texture representation using local affine regions, IEEE Trans. Pattern Anal. Mach. Intell. 27 (8) (2005) 1265–1278.

[17] Y. Xu, X. Yang, H. Ling, H. Ji, A new texture descriptor using multifractal analysis in multiorientation wavelet pyramid, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2010, pp. 161–168.

[18] M. Varma, A. Zisserman, A statistical approach to material classification using image patches, IEEE Trans. Pattern Anal. Mach. Intell. 31 (11) (2009) 2032–2047.

[19] P. Mallikarjuna, M. Fritz, A. Targhi, E. Hayman, B. Caputo, J.-O. Eklundh, The kth-tips and kth-tips2 databases, ⟨http://www.nada.kth.se/cvap/databases/kth-tips/⟩.

[20] G. Burghouts, J. Geusebroek, Material specific adaptation of color invariant features, Pattern Recognit. Lett. 30 (3) (2009) 306–313.

[21] T. Ojala, T. Mäenpää, M. Pietikänen, J.K.J. Viertola, S. Huovinen, Outex—new framework for empirical evaluation of texture analysis algorithms, in: Proceedings of 16th International Conference on Pattern Recognition, 2002, pp. 701–706.

[22] D. Huang, C. Shan, M. Ardabilian, Y. Wang, L. Chen, Local binary patterns and its application to facial image analysis: a survey, IEEE Trans. Syst. Man Cybern.—Part C: Appl. Rev. 41 (6) (2011) 765–781.

[23] A. Fernández, M. Álvarez, F. Bianconi, Texture description through histograms of equivalent patterns, J. Math. Imaging Vis. 45 (1) (2013) 76–102.

[24] L. Nanni, A. Lumini, S. Brahnam, Survey on lbp based texture descriptors for image classification, Expert Syst. Appl. 39 (3) (2012) 3634–3641.

[25] T. Ojala, M. Pietikäinen, D. Harwood, Performance evaluation of texture measures with classification based on Kullback discrimination of distributions, in: International Conference on Pattern Recognition, IEEE, Jerusalem, Israel, 1994, pp. 582–585.

[26] T. Ojala, M. Pietikäinen, D. Harwood, A comparative study of texture measures with classification based on feature distributions, Pattern Recognit. 29 (1) (1996) 51–59.

[27] M. Pietikäinen, T. Ojala, Z. Xu, Rotation-invariant texture classification using feature distributions, Pattern Recognit. 33 (1) (2000) 43–52.

[28] H. Zhou, R. Wang, C. Wang, A novel extended local-binary-pattern operator for texture analysis, Inf. Sci. 178 (22) (2008) 4314–4325.

[29] A. Fathi, A. Naghsh-Nilchi, Noise tolerant local binary pattern operator for efficient texture analysis, Pattern Recognit. Lett. 33 (9) (2012) 1093–1100.

[30] L. Liu, Y. Long, P. Fieguth, S. Lao, G. Zhao, Brint: binary rotation invariant and noise tolerant texture classification, IEEE Trans. Image Process. 23 (7) (2014) 3071–3084.

[31] S. Liao, A.C.S. Chung, Face recognition by using elongated local binary patterns with average maximum distance gradient magnitude, in: Proceedings of Asian Conference on Computer Vision (ACCV), 2007, pp. 672–679.

[32] A. Petpon, S. Srisuk, Face recognition with local line binary pattern, in: International Conference on Image and Graphics, 2009, pp. 533–539.

[33] S. ul Hussain, B. Triggs, Visual recognition using local quantized patterns, in: European Conference on Computer Vision (ECCV), 2012, pp. 716–729.

[34] S. ul Hussain, T. Napoléon, F. Jurie, Face recognition using local quantized patterns, in: British Machine Vision Conference, 2012.

[35] Z. Guo, L. Zhang, D. Zhang, A completed modeling of local binary pattern operator for texture classification, IEEE Trans. Image Process. 9 (16) (2010) 1657–1663.

[36] L. Liu, L. Zhao, Y. Long, G. Kuang, P. Fieguth, Extended local binary patterns for texture classification, Image Vis. Comput. 30 (2) (2012) 86–99.

[37] M. Heikkilä, M. Pietikäinen, C. Schmid, Description of interest regions with center symmetric local binary patterns, in: Indian Conference on Computer Vision, Graphics and Image Processing, 2006, pp. 58–69.

[38] A. Fernándeza, M. Álvarez, F. Bianconib, Image classification with binary gradient contours, Opt. Lasers Eng. 49 (9-10) (2011) 1177–1184.

[39] K. Wang, C. Bichot, C. Zhu, B. Li, Pixel to patch sampling structure and local neighboring intensity relationship patterns for texture classification, IEEE Signal Process. Lett. 20 (9) (2013) 853–856.

[40] S.O. Vargas, F. Rooms, W. Philips, Geometric local binary pattern, a new approach to analyse texture in images, in: International Conference on Topology and its Applications, 2010, pp. 179–181.

[41] B. Zhang, Y. Gao, S. Zhao, J. Liu, Local derivative pattern versus local binary pattern: face recognition with high-order local pattern descriptor, IEEE Trans. Image Process. 19 (2) (2010) 533–544.

[42] T. Mäenpää, M. Pietikäinen, Multi-scale binary patterns for texture analysis, in: Lecture Notes in Computer Science (Image Analysis), vol. 2749, 2003, pp. 885–892.

[43] L. Zhang, R. Chu, S. Xiang, S. Liao, S. Li, Face detection based on multi-block lbp representation, in: S. Lee, S. Li (Eds.), Advances in Biometrics, Lecture Notes in Computer Science, vol. 4642, 2007, pp. 11–18.

[44] L. Wolf, T. Hassner, Y. Taigman, Effective unconstrained face recognition by combining multiple descriptors and learned background statistics, IEEE Trans. Pattern Anal. Mach. Intell. 33 (10) (2011) 1978–1990.

[45] N.-S. Vu, A. Caplier, Enhanced patterns of oriented edge magnitudes for face recognition and image matching, IEEE Trans. Image Process. 21 (3) (2012) 1352–1365.

[46] L. Liu, S. Lao, P. Fieguth, Y. Guo, X. Wang, M. Pietikainen, Median robust extended local binary pattern for texture classification, IEEE Trans. Image Process. 25 (3) (2016) 1368–1381.

[47] M. Calonder, V. Lepetit, M. Ozuysal, T. Trzcinski, C. Strecha, P. Fua, Brief: computing a local binary descriptor very fast, IEEE Trans. Pattern Anal. Mach. Intell. 34 (7) (2012) 1281–1298.

[48] S. Leutenegger, M. Chli, R. Siegwart, Brisk: Binary robust invariant scalable keypoints, in: International Conference on Computer Vision (ICCV), 2011, pp. 2548–2555.

[49] A. Alahi, R. Ortiz, P. Vandergheynst, Freak: fast retina keypoint, in: Computer Vision and Pattern Recognition (CVPR), IEEE, Rhode Island, America, 2012, pp. 510–517.

[50] J. Heinly, E. Dunn, J. Frahm, Comparative evaluation of binary features, in: European Conference on Computer Vision (ECCV), 2012, pp. 759–773.

[51] H. Jin, Q. Liu, H. Lu, X. Tong, Face detection using improved lbp under Bayesian framework, in: Proceedings of the 3rd International Conference on Image and Graphics, 2004, pp. 306–309.

[52] A. Hafiane, G. Seetharaman, B. Zavidovique, Median binary pattern for textures classification, in: Proceedings of the 4th International Conference on Image Analysis and Recognition, 2007, pp. 387–398.

[53] M. Ghahramani, G. Zhao, M. Pietikäinen, Incorporating texture intensity information into lbp-based operators, in: J.-K. Kämäräinen, M. Koskela (Eds.), Image Analysis, Lecture Notes in Computer Science, vol. 7944, 2013, pp. 66–75.

[54] D. He, L. Wang, Texture unit, texture spectrum, and texture analysis, IEEE Trans. Geosci. Remote Sens. 28 (4) (1990) 509–512.

[55] D. He, L. Wang, Texture features based on texture spectrum, Pattern Recognit. 24 (5) (1991) 391–399.

[56] D. He, L. Wang, Unsupervised textural classification of images using the texture spectrum, Pattern Recognit. 25 (3) (1992) 247–255.

[57] X. Tan, B. Triggs, Enhanced local texture feature sets for face recognition under difficult lighting conditions, IEEE Trans. Image Process. 19 (6) (2010) 1635–1650.

[58] T. Ahonen, M. Pietikäinen, Soft histograms for local binary patterns, in: Finnish Signal Processing Symposium, IEEE, Oulu Finland, 2007, pp. 645–649.

[59] D. Iakovidis, E. Keramidas, D. Maroulis, Fuzzy local binary patterns for ultrasound texture characterization, in: A. Campilho, M. Kamel (Eds.), Image Analysis and Recognition, Lecture Notes in Computer Science, vol. 5112, Springer, Berlin, Heidelberg, 2008, pp. 750–759.

[60] J. Ren, X. Jiang, J. Yuan, Noise-resistant local binary pattern with an embedded error-correction mechanism, IEEE Trans. Image Process. 22 (10) (2013) 4049–4060.

[61] J. Chen, V. Kellokumpu, G. Zhao, M. Pietikänen, Rlbp: robust local binary pattern, in: British Vision Conference on Computer Vision (BMVC), 2013.

[62] Y. Huang, Y. Wang, T. Tan, Combining statistics of geometrical and correlative features for 3d face recognition, in: Proceedings of British Machine Vision Conference (BMVC), 2006, pp. 879–888.

[63] D. Huang, Y. Wang, Y. Wang, A robust method for near infrared face recognition based on extended local binary pattern, in: Advances in Visual

Computing, Lecture Notes in Computer Science, vol. 4842, 2007, pp. 437–446.

[64] O. Lahdenoja, M. Laiho, A. Paasio, Reducing the feature vector length in local binary pattern based face recognition, in: IEEE International Conference on Image Processing (ICIP), 2005, pp. 914–917.

[65] D. Nguyen, Z. Zong, P. Ogunbona, W. Li, Object detection using non-redundant local binary patterns, in: International Conference on Image Processing (ICIP), 2010, pp. 4609–4612.

[66] D. Nguyen, P. Ogunbona, W. Li, A novel shape-based non-redundant local binary pattern descriptor for object detection, Pattern Recognit. 46 (5) (2013) 1485–1500.

[67] Y. Mu, S. Yan, Y. Liu, T. Huang, B. Zhou, Discriminative local binary patterns for human detection in personal album, in: IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), 2008, pp. 1–8.

[68] L. Liu, P. Fieguth, G. Kuang, D. Clausi, Sorted random projections for robust rotation invariant texture classification, Pattern Recognit. 45 (6) (2012) 2405–2418.

[69] Y. Zhao, D. Huang, W. Jia, Completed local binary count for rotation invariant texture classification, IEEE Trans. Image Process. 21 (10) (2012) 4492–4497.

[70] Y. Ma, Number local binary pattern: an extended local binary pattern, in: International Conference on Wavelet Analysis and Pattern Recognition (ICWAPR), 2011, pp. 272–275.

[71] S. Orjuela, R. Quinones, B. Ortiz-Jaramillo, Improving textures discrimination in the local binary patterns technique by using symmetry and group theory, in: International Conference on Digital Signal Processing, 2011, pp. 1–6.

[72] H. Yang, Y. Wang, A lbp-based face recognition method with hamming distance constraint, in: Proc. Int. Conf. Image Graph., IEEE, Sichuan, China, 2007, pp. 645–649.

[73] J. Ryu, S. Hong, H. Yang, Sorted consecutive local binary pattern for texture classification, IEEE Trans. Image Process. 24 (7) (2015) 2254–2265.

[74] R. Haralick, K. Shanmugam, I. Dinstein, Textural features for image classification, IEEE Trans. Syst. Man Cybern. 3 (6) (1973) 610–621.

[75] X. Qi, R. Xiao, C. Li, Y. Qiao, J. Guo, X. Tang, Pairwise rotation invariant cooccurrence local binary pattern, IEEE Trans. Pattern Anal. Mach. Intell. 36 (11) (2014) 2199–2213.

[76] X. Qi, R. Xiao, J. Guo, L. Zhang, Pairwise rotation invariant cooccurrence local binary pattern, in: European Conference on Computer Vision (ECCV), 2012, pp. 158–171.

[77] G. Schaefer, N. Doshi, Multidimensional local binary pattern descriptors for improved texture analysis, in: International Conference on Pattern Recognition (ICPR), 2012, pp. 2500–2503.

[78] X. Qi, Y. Qiao, C. Li, J.J. Guo, Multiscale joint encoding of local binary patterns for texture and material classification, in: Proceedings of British Machine Vision Conference (BMVC), 2013.

[79] Z. Guo, L. Zhang, D. Zhang, X. Mou, Hierarchical multiscale lbp for face and palmprint recognition, in: IEEE International Conference on Image Processing (ICIP), 2010, pp. 4521–4524.

[80] Y. Guo, G. Zhao, M. Pietikäinen, Discriminative features for texture description, Pattern Recognit. 45 (10) (2012) 3834–3843.

[81] E. González, A. Fernández, F. Bianconi, General framework for rotation invariant texture classification through cooccurrence of patterns, J. Math. Imaging Vis. 50 (3) (2014) 300–313.

[82] C. Shan, T. Gritti, Learning discriminative lbp-histogram bins for facial expression recognition, in: Proceedings of British Machine Vision Conference (BMVC), 2008.

[83] W. Zhang, S. Shan, W. Gao, X. Chen, H. Zhang, Local Gabor binary pattern histogram sequence (lgbphs): a novel non-statistical model for face representation and recognition, in: IEEE International Conference on Computer Vision (ICCV), 2005, pp. 786–791.

[84] J. Zhang, J. Liang, H. Zhao, Local energy pattern for texture classification using self-adaptive quantization thresholds, IEEE Trans. Image Process. 22 (1) (2013) 31–42.

[85] J. Kannala, E. Rahtu, Bsif: binarized statistical image features, in: International Conference on Pattern Recognition (ICPR), 2012, pp. 1363–1366.

[86] X. Qian, X. Hua, P. Chen, L. Ke, Plbp: an effective local binary patterns texture descriptor with pyramid representation, Pattern Recognit. 44 (10–11) (2011) 2502–2515.

[87] H. Galoogahi, T. Sim, Face sketch recognition by local radon binary pattern: Lrbp, in: IEEE International Conference on Image Processing (ICIP), 2012, pp. 1837–1840.

[88] Z. Li, G. Liu, Y. Yang, J. You, Scale and rotation-invariant local binary pattern using scale-adaptive texton and subuniform-based circular shift, IEEE Trans. Image Process. 21 (4) (2012) 2130–2140.

[89] Z. Guo, L. Zhang, D. Zhang, Rotation invariant texture classification using lbp variance (lbpv) with global matching, Pattern Recognit. 43 (3) (2010) 706–719.

[90] R. Davarzani, S. Mozaffari, K. Yaghmaie, Scale and rotation invariant texture description with improved local binary pattern features, Signal Process. 111 (2015) 274–293.

[91] F. Ahmed, E. Hossain, A. Bari, A. Shihavuddin, Compound local binary pattern (clbp) for robust facial expression recognition, in: IEEE International Symposium on Computational Intelligence and Informatics, 2011, pp. 391–395.

[92] Z. Guo, X. Wang, J. Zhou, J. You, Robust texture image representation by scale selective local binary patterns, IEEE Trans. Image Process. 25 (2) (2016) 687–699.

[93] T. Ahonen, J. Matas, C. He, M. Pietikäinen, Rotation invariant image

[94] Y. Zhao, T. Ahonen, J. Matas, M. Pietikäinen, Rotation invariant image and video description with local binary pattern features, IEEE Trans. Image Process. 21 (4) (2012) 1465–1477.

[95] Z. Guo, L. Zhang, D. Zhang, S. Zhang, Rotation invariant texture classification using adaptive lbp with directional statistical features, in: International Conference on Image Processing (ICIP), 2010, pp. 285–288.

[96] F. Khellah, Texture classification using dominant neighborhood structure, IEEE Trans. Image Process. 20 (11) (2011) 3270–3279.

[97] X. Wang, T. Han, S. Yan, An hog-lbp human detector with partial occlusion handling, in: IEEE International Conference on Computer Vision (ICCV), 2009, pp. 32–39.

[98] S. Hussain, B. Triggs, Feature sets and dimensionality reduction for visual object detection, in: British Machine Vision Conference (BMVC), 2010, pp. 112.1–10.

[99] B. Klare, A. Jain, Heterogeneous face recognition using kernel prototype similarities, IEEE Trans. Pattern Anal. Mach. Intell. 35 (6) (2013) 1410–1422.

[100] C. Chan, M.A. Tahir, J. Kittler, M. Pietikäinen, Multiscale local phase quantization for robust component-based face recognition using kernel fusion of multiple descriptors, IEEE Trans. Pattern Anal. Mach. Intell. 35 (5) (2013) 1164–1177.

[101] A. Roy, S. Marcel, Haar local binary pattern feature for fast illumination invariant face detection, in: Proceedings of British Machine Vision Conference (BMVC), 2009.

[102] V. Ojansivu, J. Heikkilä, Blur insensitive texture classification using local phase quantization, in: A. Elmoataz, O. Lezoray, F. Nouboud, D. Mammass (Eds.), Image and Signal Processing, Lecture Notes in Computer Science, vol. 5099, 2008, pp. 236–243.

[103] V. Ojansivu, E. Rahtu, J. Heikkilä, Rotation invariant local phase quantization for blur insensitive texture analysis, in: IEEE International Conference on Pattern Recognition (ICPR), 2008, pp. 1–4.

[104] J. Chen, S. Shan, C. He, G. Zhao, M. Pietikainen, X. Chen, W. Gao, Wld: a robust local image descriptor, IEEE Trans. Pattern Anal. Mach. Intell. 32 (9) (2010) 1705–1720.

[105] B. Zhang, S. Shan, X. Chen, W. Gao, Histogram of Gabor phase patterns (hgpp): a novel object representation approach for face recognition, IEEE Trans. Image Process. 16 (1) (2007) 57–68.

[106] G. Sharma, S. ul Hussain, F. Jurie, Local higher-order statistics (lhs) for texture categorization and facial analysis, in: European Conference on Computer Vision (ECCV), 2012, pp. 1–12.

[107] R. Maani, S. Kalra, Y. Yang, Noise robust rotation invariant features for texture classification, Pattern Recognit. 46 (8) (2013) 2103–2116.

[108] R. Maani, S. Kalra, Y. Yang, Rotation invariant local frequency descriptors for texture classification, IEEE Trans. Image Process. 22 (6) (2013) 2409–2419.

[109] Z. Lei, M. Pietikäinen, S. Li, Learning discriminant face descriptor, IEEE Trans. Pattern Anal. Mach. Intell. 36 (2) (2014) 289–302.

[110] H. Lategahn, S. Gross, T. Stehle, T. Aach, Texture classification by modeling joint distributions of local patterns with Gaussian mixtures, IEEE Trans. Image Process. 19 (6) (2010) 1548–1557.

[111] T. Ahonen, M. Pietikainen, Image description using joint distribution of filter bank responses, Pattern Recognit. Lett. 30 (4) (2009) 368–376.

[112] J. Sanchez, F. Perronnin, T. Mensink, J. Verbeek, Image classification with the Fisher vector: theory and practice, Int. J. Comput. Vis. 105 (3) (2013) 222–245.

[113] R. Maani, S. Kalra, Y. Yang, Robust edge aware descriptor for image matching, in: Asian Conference on Computer Vision, 2014, pp. 553–568.

[114] L. Liu, P. Fieguth, Texture classification from random features, IEEE Trans. Pattern Anal. Mach. Intell. 34 (3) (2012) 574–586.

[115] M. Cimpoi, S. Maji, A. Vedaldi, Deep filter banks for texture recognition and segmentation, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015, pp. 3828–3836.

[116] A. Krizhevsky, S. Ilya, G. Hinton, Imagenet classification with deep convolutional neural networks, in: Advances in Neural Information Processing Systems, 2012, pp. 1097–1105.

[117] L. Sifre, S. Mallat, Combined scattering for rotation invariant texture analysis, in: Proceedings of European Symposium on Artificial Neural Networks, 2012.

[118] J. Bruna, S. Mallat, Invariant scattering convolution networks, IEEE Trans. Pattern Anal. Mach. Intell. 35 (8) (2013) 1872–1886.

[119] L. Sifre, S. Mallat, Rotation, scaling and deformation invariant scattering for texture discrimination, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2013, pp. 1233–1240.

[120] T. Chan, K. Jia, S. Gao, J. Lu, Z. Zeng, Y. Ma, PCANet: a simple deep learning baseline for image classification?, IEEE Trans. Image Process. 24 (12) (2015) 5017–5032.

[121] P. Brodatz, Textures: A Photographic Album for Artists and Designers, Dover Publications, New York, USA, 1966.

[122] L. Liu, P. Fieguth, Texture classification from random features, IEEE Trans. Pattern Anal. Mach. Intell. 34 (3) (2012) 574–586.

[123] J. Zhang, M. Marszalek, S. Lazebnik, C. Schmid, Local features and kernels for classification of texture and object categories: a comprehensive study, Int. J. Comput. Vis. 73 (2) (2007) 213–238.

[124] J. Wright, A. Yang, A. Ganesh, S. Sastry, Y. Ma, Robust face recognition via sparse representation, IEEE Trans. Pattern Anal. Mach. Intell. 31 (2) (2009) 210–227.

[125] B. Caputo, E. Hayman, P. Mallikarjuna, Class-specific material categorization,

in: International Conference on Computer Vision (ICCV), 2005, pp. 1597–1604.

[126] X. Hong, G. Zhao, M. Pietikainen, X. Chen, Combining lbp difference and feature correlation for texture description, IEEE Trans. Image Process. 23 (6) (2014) 2557–2568.

[127] A. Fernández, M. Álvarez, F. Bianconi, Texture description through histograms of equivalent patterns, J. Math. Imaging Vis. 45 (1) (2013) 76–102.

[128] L. Liu, P. Fieguth, G. Zhao, M. Pietikainen, Extended local binary pattern fusion for face recognition, in: IEEE International Conference on Image Processing (ICIP), 2014, pp. 718–722.

[129] X. Yang, K. Cheng, Local difference binary for ultra-fast and distinctive feature description, IEEE Trans. Pattern Anal. Mach. Intell. 35 (1) (2014) 188–194.

[130] S. Murala, R. Maheshwari, R. Balasubramanian, Local tetra patterns: a new feature descriptor for content-based image retrieval, IEEE Trans. Image Process. 21 (5) (2012) 2874–2886.

[131] X. Fu, W. Wei, Centralized binary patterns embedded with image Euclidean distance for facial expression recognition, in: International Conference on

Natural Computation, vol. 4, 2008, pp. 115–119.

[132] H. Chu, T. Ahonen, M. Pietikainen, A Bayesian local binary pattern texture descriptor, in: International Conference on Pattern Recognition (ICPR), 2008, pp. 1–4.

[133] R. Nosaka, Y. Ohkawa, K. Fukui, Feature extraction based on cooccurrence of adjacent local binary patterns, in: Advances in Image and Video Technology, 2012, pp. 82–91.

[134] R. Nosaka, C. Suryanto, K. Fukui, Rotation invariant cooccurrence among adjacent lbps, in: Asian Conference Computer Vision Workshops, 2013, pp. 15–25.

[135] L. Liu, P. Fieguth, G. Kuang, Generalized local binary patterns for texture classification, in: Proceedings of British Machine Vision Conference (BMVC), 2011.

[136] S. Yan, S. Shan, X. Chen, W. Gao, Locally assembled binary (lab) feature with feature-centric cascade for fast and accurate face detection, in: IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), 2008, pp. 1–7.

Li Liu received the B.S. degree in Communication Engineering, the M.S. degree in Photogrammetry and Remote Sensing and the Ph.D. degree in Information and Communication Engineering from the National University of Defense Technology, Changsha, China, in 2003, 2005 and 2012, respectively. She joined the faculty at the National University of Defense Technology in 2012, where she is currently an associate professor in School of Information System and Management. During her Ph.D. study, she spent two years and three months as a Visiting Student at the University of Waterloo, Canada, from 2008 to 2010. From March 2015 to January 2016, she visited the Department of Electronic Engineering at the Chinese University of Hong Kong. Dr. Liu served as a Co-Chair of the International Workshop on Robust local descriptors for computer vision at ACCV2014 and the International Workshop on Robust Features for computer vision at CVPR2016. Her current research interests include computer vision, texture analysis, pattern recognition and video analysis and retrieval.

Paul Fieguth received the B.A.Sc. degree from the University of Waterloo, Ontario, Canada, in 1991 and the Ph.D. degree from the Massachusetts Institute of Technology, Cambridge, in 1995, both degrees in Electrical Engineering. He joined the faculty at the University of Waterloo in 1996, where he is currently a Professor in Systems Design Engineering. He has held visiting appointments at the University of Heidelberg in Germany, at INRIA/Sophia in France, at the Cambridge Research Laboratory in Boston, at Oxford University and the Rutherford Appleton Laboratory in England, and with postdoctoral positions in Computer Science at the University of Toronto and in Information and Decision Systems at MIT. His research interests include statistical signal and image processing, hierarchical algorithms, data fusion, and the interdisciplinary applications of such methods, particularly to remote sensing.

Yulan Guo received his B.Eng. in Communication Engineering from National University of Defense Technology (NUDT) in 2008. He is currently working towards his Ph.D. degree at NUDT. He has been a visiting Ph.D. student at the University of Western Australia from November 2011 to November 2013. His research interests include 3D object recognition, 3D face recognition, 3D modeling, pattern recognition, and signal processing.

Xiaogang Wang received the B.S. degree from the University of Science and Technology of China in 2001, the M.S. degree from the Chinese University of Hong Kong in 2003, and the Ph.D. degree from the Computer Science and Artificial Intelligence Laboratory at the Massachusetts Institute of Technology in 2009. He is currently an Assistant Professor in the Department of Electronic Engineering at The Chinese University of Hong Kong. His research interests include computer vision and machine learning.

Matti Pietikäinen received his Doctor of Science in Technology degree from the University of Oulu, Finland. He is currently a professor, Scientific Director of Infotech Oulu and Director of Center for Machine Vision Research at the University of Oulu. From 1980 to 1981 and from 1984 to 1985, he visited the Computer Vision Laboratory at the University of Maryland. 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 285 refereed papers in international journals, books, and conferences. He was an Associate Editor of IEEE Transactions on Pattern Analysis and Machine Intelligence and Pattern Recognition journals, and currently serves as an Associate Editor of Image and Vision Computing journal. He was a President of the Pattern Recognition Society of Finland from 1989 to 1992. From 1989 to 2007 he served as a Member of the Governing Board of International Association for Pattern Recognition (IAPR), and became one of the founding fellows of IAPR in 1994. He is a IEEE Fellow for contributions to texture and facial image analysis for machine vision.