Pattern Recognition Letters 65 (2015) 8–14

Contents lists available at ScienceDirect

Pattern Recognition Letters

journal homepage: www.elsevier.com/locate/patrec

Dominant local binary patterns for texture classification: Labelled or unlabelled?☆

Francesco Bianconi[a,*], Elena González[b], Antonio Fernández[b]

[a] Università degli Studi di Perugia, Department of Engineering, Via G. Duranti 93, Perugia, 06125, Italy
[b] Universidade de Vigo, Department of Engineering Design, Vigo, 36310, Spain

ARTICLE INFO

Article history:
Received 17 October 2014
Available online 9 July 2015

Keywords:
Texture classification
Feature selection
Dominant local binary patterns

ABSTRACT

This paper investigates the problem of learning sets of discriminative patterns from local binary patterns (LBP). Such patterns are usually referred to as 'dominant local binary patterns' (DLBP). The strategies to obtain the dominant patterns may either keep knowledge of the patterns labels or discard it. It is the aim of this work to determine which is the best option. To this end the paper studies the effectiveness of different strategies in terms of accuracy, data compression ratio and time complexity. The results show that DLBP provides a significant compression rate with only a slight accuracy decrease with respect to LBP, and that retaining information about the patterns' labels improves the discrimination capability of DLBP. Theoretical analysis of time complexity revealed that the gain/loss provided by DLBP vs. LBP depends on the classification strategy: we show that, asymptotically, there is in principle no advantage when classification is based on computationally-cheap methods (such as nearest neighbour and nearest mean classifiers), because in this case determining the dominant patterns is computationally more expensive than classifying using the whole feature vector; by contrast, pattern selection can be beneficial with more complex classifiers such as support vector machines.

1. Introduction

LBP is a very popular approach to texture analysis with applications in a wide range of areas such as, among others, surface inspection, face recognition, biometrics and medical image analysis [2]. The method is much appreciated for its many desirable properties, such as ease of implementation, invariance to illumination changes, limited computational demand and high descriptive performance – especially when the level of noise is low [14]. LBP considers, as image features, the occurrence probability of the binary patterns that can be generated from an image patch of predefined shape and size when thresholded at the value of the central pixel. It is well known that the resulting probability distribution tends to be highly uneven: some patterns tend to occur much more frequently than others [24]. Many researchers have been concerned with the problem of reducing the dimensionality of LBP by determining the subsets of patterns that convey the largest amount of information. A common approach consists of reducing the number of features by using some *a priori* rules [20]: Ojala et al. for instance proposed to cluster patterns into rotationally-equivalent classes, an approach which generates the well-known family of rotation invariant descriptors (LBP^{ri}). They also suggested that further reduction could be obtained by considering the so called 'uniform patterns' (LBP^{riu2}), namely those patterns that have at most two bitwise transitions [24]. Experiments have shown that uniform patterns are the most common in natural images [24], a finding which was later on explained on a theoretical basis [1].

As an alternative, Liao et al. [17] and, more recently, Nanni et al. [22] and Guo et al. [12], proposed *a posteriori* strategies in which the patterns to retain are learnt from some training data. Liao et al. [17] for instance suggested to retain, as features, the probability of occurrence of the smallest set of patterns that, in any given image, represent a certain percent – 80% in their implementation – of the total population. The resulting dominant local binary patterns (DLBP) bear no information about the patterns' labels [17]; instead, they consider the relative patterns' frequency only. As a consequence this scheme does not guarantee that the *i*th element of the feature vector extracted from an image $\mathbf{I}_1$ and the *i*th element of the feature vector extracted from an image $\mathbf{I}_2$ refer to the same pattern. For this reason we refer to such selection strategy as an *unlabelled* model. A natural question arises whether comparing the probability of occurrence of different patterns makes sense altogether [6]. In [17] the authors affirm that omitting the pattern type information is not harmful; in

☆ This paper has been recommended for acceptance by A. Heyden.

[*] Corresponding author. Tel.: +39 75 5853703; fax: +39 75 5853703.
E-mail address: bianco@ieee.org (F. Bianconi).

http://dx.doi.org/10.1016/j.patrec.2015.06.025
0167-8655/© 2015 Elsevier B.V. All rights reserved.

this paper we endorse a diametric opposite view: that neglecting information about the patterns type has negative effects on the discrimination capability of the method. Our main claim is that feature selection schemes that keep knowledge of the patterns' type outperform the unlabelled approach. We refer to such reduction schemes as *labelled* methods.

In the remainder of the paper, after recalling the basics of LBP in Section 2, we discuss the unlabelled (Section 3) and labelled (Section 4) approach for determining dominant local binary patterns and perform an experimental comparison in Section 5. The results presented in Section 6 show that in no case the unlabelled model is superior to the labelled counterparts. We also evaluate the compression ratio that can be obtained with the various methods and study the effect of the different feature reduction schemes on the overall computing time. Section 7 concludes the paper with some final considerations.

2. Brief overview of LBP

The LBP operator characterizes images through the probability of occurrence of certain binary patterns that a neighbourhood of predefined shape and size can generate [24]. The typical configuration consists of a central pixel plus a set of peripheral points evenly spaced along a circle (see Fig. 1) – but other arrangements have been proposed as well [21]. The intensity values of those points that do not coincide with image pixels are estimated through interpolation. Such neighbourhoods are conventionally indicated in the form (m, R), where m represents the number of peripheral points and R the radius of the circle.

For each position of the neighbourhood a corresponding binary pattern is obtained by thresholding the intensity values of the peripheral points at the value of the central pixel. Each binary patterns is then assigned a unique label in the following way:

$$\text{LBP}_{m,R} = \sum_{i=0}^{m-1} 2^i \xi (I_i - I_c) \tag{1}$$

where ξ is the binary thresholding function (Eq. 2).

$$\xi(x) = \begin{cases} 1, & \text{if } x \geq 0 \\ 0, & \text{if } x < 0 \end{cases} \tag{2}$$

As a result, the $\text{LBP}_{m,\,R}$ operator produces 2^m different binary patterns. Theoretically, when the input image rotates by angular steps of $\pm 2\pi/m$ radians, the binary sequence $\{\xi(I_i - I_c)\}$, $i \in \{1, \ldots, m-1\}$ circularly shifts by one position to the left or to the right. To make the descriptor invariant against rotation, one can consider equivalent all the patterns that can be transformed into one another by a rotation of multiples of $\pm 2\pi/m$ radians. This approach gives rise to the rotation invariant operator, usually referred to as $\text{LBP}^{ri}_{m,R}$. The number of rotationally-equivalent classes for a given m can be determined through group theory, as detailed in Ref. [8]. Table 1 shows the number of features generated by the $\text{LBP}_{m,R}$ and $\text{LBP}^{ri}_{m,R}$ operators for different values of m. Clearly the dimension of the descriptors grows quickly as m increases.

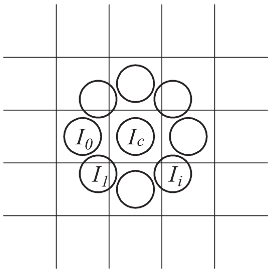


Fig. 1. Circular neighbourhood.

Table 1
Number of local binary patterns.

m	N	
	$\text{LBP}_{m,R}$	$\text{LBP}^{ri}_{m,R}$
4	16	6
8	256	36
16	65,536	4116
24	16,777,216	699,252

High dimensional data are in general difficult to handle due to the 'curse of dimensionality' [7]. Moreover, both experimental and theoretical studies have suggested that the probability of occurrence of local binary patterns may vary greatly from one pattern to another, and that certain patterns very seldom occur in practice [1,24]. As a result, some of them are likely to produce only noisy and irrelevant features that may mislead the classification [12]. The problem of determining the set of 'most discriminative' patterns is therefore a very actual and interesting one both from a theoretical and practical standopint.

3. Dominant local binary patterns: the unlabelled model (DLBP)

As we mentioned in Section 1, the unlabelled approach discards any information about the patterns' labels. The method consists of sorting the LBP histogram of each image in descending order and retaining a certain number of bins. Given a set of train images, the number of bins to retain is computed by determining, for each train image, the cardinality of the smallest set of patterns that accounts for a given fraction of the total occurrence probability and by averaging this value over the whole train set. Each histogram is sorted independently of the others in this scheme, therefore any information about the patterns' type is lost: the resulting DLBP features will only contain information about the patterns' frequencies. This strategy is based on the assumption that it is the relative probability distribution what really matters, not the occurrence probability of each specific pattern [17]. As for the fraction of the total occurrence to retain, throughout the paper we maintain the settings proposed in the above-cited reference, where the authors recommend the value 0.8. From a computational standpoint, the algorithm is dominated by the ordering of each vector of the train set, therefore executes in $\mathcal{O}(MN\log N)$ time, where M is the number of train patterns and N the dimension of the original descriptor.

4. Dominant local binary patterns: the labelled model

As opposed to the unlabelled model, the labelled model keeps knowledge of the patterns' labels. Different implementations of this approach have been proposed in the literature: we briefly recall them in the following subsections.

4.1. Labelled dominant local binary patterns (L-DLBP)

Labelled dominant local binary patterns have been described by Fu et al. [10] and, more recently, by González et al. [11]. In this scheme the original LBP histograms of the train images are first averaged column-wise (feature-by-feature) and the resulting vector (average patterns' frequencies) is sorted in descending order. Then the labels of the smallest set of co-occurrences that sum at least 0.8 are retained; the others are discarded. The labels this way obtained constitute the set of dominant patterns; the feature vector of any image is represented by the probabilities of occurrence of these patterns. From a computational standpoint the approach involves computing the average of each feature over the train set and sorting the result, which gives a time complexity of $\mathcal{O}(\max[MN, N\log N])$.

4.2. Highest-variance dominant local binary patterns (HV-DLBP)

Highest-variance dominant local binary patterns, introduced by Nanni et al. [22], retain the local binary patterns with the highest variance. The first step of this procedure therefore consists of computing the column-wise (feature-by-feature) variance of the pattern histograms of the train images; the resulting vector is then sorted in descending order. The dominant patterns are represented by the labels of the smallest set of patterns that accounts for at least 0.8 of the total variance. The computational complexity is the same as L-DLBP.

4.3. Highest-rank dominant local binary patterns (HR-DLBP)

Highest-rank dominant local binary patterns, recently proposed by Doshi and Schaefer [6], select the dominant patterns through a preliminary rank transform of the pattern histograms. In the first step each pattern histogram of the train images is sorted and ranked: the most frequent pattern in each image is assigned the highest rank; the least frequent is assigned rank one. The ranks are averaged and the resulting vector (average rank) is sorted in descending order. The number of patterns to retain is the cardinality of the minimum set that accounts for at least 0.8 of the sum of the average rank. The method requires sorting each train patterns and averaging the resulting ranks, therefore executes in $\mathcal{O}(MN\log N)$ time.

4.4. Discriminative features (DF-DLBP)

Guo et al. [12] discriminative features represent a more involved learning framework for determining dominant patterns. This three-layer approach works as follows. The first step (referred to as 'layer 1' in Ref. [12]) involves sorting each pattern histogram of the train set and retaining the smallest set of labels that account for a given fraction (0.8 in this case) of the total probability. This operation that can be carried out in $\mathcal{O}(MN\log N)$ time. The second step ('layer 2') consists of intersecting, for each class, the sets of labels returned by each train pattern belonging to that class. Layer 2 gives, as a result, the dominant pattern set of each class. Assuming that the train patterns are equally distributed among the classes, so that there are M/C train patterns per class, we perform this operation in $\mathcal{O}(2N(M-C))$ time, where C is the number of classes. In the third step ('layer 3') the dominant patterns set is obtained as the union of dominant patterns of each class. The complexity of this operation is $\mathcal{O}(2N(C-1))$ in the worst case. Asymptotically, the execution is dominated by layer 1, therefore the time complexity of the whole method is $\mathcal{O}(MN\log N)$.

Differently from the other methods, this approach guarantees that any class in the training set is adequately represented within the selected patterns. A potential disadvantage could be the high number of features that the method produces, which in principle gets higher when the number of classes grows.

5. Experiments

We conducted a supervised image classification experiment to evaluate the effectiveness of the different strategies for obtaining dominant local binary patterns presented in Sections 3 and 4. The overall objective was to determine the effect of including/omitting information about the patterns' labels on the classification accuracy. We also investigate the compression ratio that can be achieved with the various strategies, as well as the average computing time of the different methods. As a baseline for comparison, we considered the original, full-dimensional, $\mathrm{LBP}^{ri}_{8,1}$ and $\mathrm{LBP}^{ri}_{16,2}$ descriptors [24]. For calibration purposes we also included the results obtained with the $\mathrm{LTP}^{riu2}_{8,1}$ and $\mathrm{LTP}^{riu2}_{16,2}$ operators [27]. In all the experiments the threshold value for LTP was set to $\tau = 3$ as suggested in Ref. [23].

5.1. Datasets

We considered eight datasets containing different types of texture images. Dataset one contains a selection of 80 texture images from the ALOT database [3]. Dataset two contains 13 texture classes from Brodatz (hardware-rotated digital images have been captured in our laboratory directly from the original book.) Dataset three includes all the 25 classes of the Kylberg–Sintorn rotation database. Dataset four is the whole Mondial Marmi database (v 1.1), which comprises 12 classes of granite textures. Dataset five covers a selection of 45 texture classes from Outex. Dataset six is composed of a set of texture images obtained from vectorial pictures: in this case the vectorial images were first rotated, then raster-scanned to get the rotated texture images. A detailed description of datasets from one to six is also available in [11]. Dataset seven is a two-class database containing a subset of images from the Pap-smear benchmark [13,23]: to have an equal number of the two classes (i.e.: normal vs. abnormal) we selected 68 image samples from each of the three normal classes (i.e.: superficial squamous epithelial, intermediate squamous epithelial and columnar epithelial) and 51 from each of four the abnormal classes (i.e.: light dysplastic, moderately dysplastic, severely dysplastic and carcinoma *in situ*). Finally, dataset eight is made up of a selection of images from the 2D HeLa database [25]. In this case for each of the 10 classes of the database – which represent sub-cellular organelles such as nuclei, endoplasmic reticulum, giantin, cis/medial Golgi, cis Golgi, lysosomes, mitochondria, nucleoli, actin, endosomes, and tubulin – we obtained 20 images samples for each class by manually selecting representative textured regions from as many images of the corresponding organelles. Summary data and sample images of each dataset are reported in Table 2.

5.2. Classification and accuracy estimation

We employed four different classification strategies: (1) nearest-neighbour (1-NN) rule with L_2 distance; (2) nearest mean classifier (NMC) with L_2 distance; (3) naïve Bayes (NB) classifier with multinomial distribution and (4) support vector machine (SVM) with radial basis kernel. Based on theoretical considerations on the spread of the input data (see Ref. [5]), we used a fixed value of $2\sqrt{(m-1)/m}$ for SVM parameter C, where m is the number of features, and estimated γ trough 10-fold validation from the training data over the discrete set of values $\{2^{-9}, 2^{-7}, \ldots, 2^{9}\}$.

Accuracy estimation was based on 100-fold split-half validation with stratified sampling. For each subdivision into train and test set the classifier was trained with the images of the 0°-group and tested with the images of each of the θ°-group (including the 0°-group), being θ one of the rotation angles available in each dataset (see Table 2). The accuracy for each rotation angle was the percentage of test images correctly classified. These values were averaged over the rotation angles of each dataset to give a global accuracy measure. The resulting figures are reported in Tables 3 and 4.

5.3. Implementation, execution and reproducible research

All the algorithms discussed in this work have been implemented in MATLAB® and executed on a laptop PC with 8 Gb RAM powered by INTEL® CORE™ i5 and WINDOWS™ 7 Professional. Extraction of LBP features was based on the routines provided by the Center for Machine Vision Research at the University of Oulu, Oulu, Finland [15]. The classification routines were based on PRTools v5 [26] and libsvm [18] for the 1-NN and SVM classifiers, respectively; on MATLAB®'s built-in functions for the naïve Bayes and NMC. For reproducible research purposes, all the data required to replicate the experiments (i.e.: source code, images and subdivisions into train and validation sets) are available online (Ref. [16])[1].

[1] To access the page: user = `dominant`, password = `patterns`

Table 2
Image datasets used in the experiments: summary table.

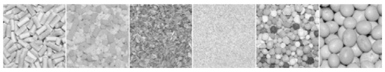
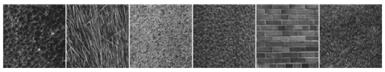
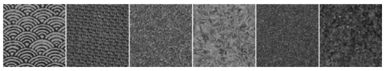
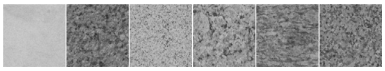
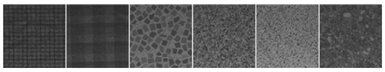
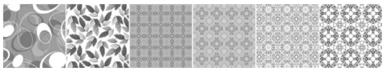
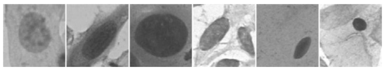
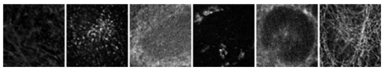
No.	Source	Rotation angles	No. of classes	Samples/ class	Image resolution	Sample images
1	ALOT	0°, 60°, 120°, 180°	80	16	181 × 181	
2	Brodatz	0°, 10°, 20°, 30°, 40°, 50°, 60°, 70°, 80°, 90°	13	16	205 × 205	
3	Kylberg–Sintorn	0°, 40°, 80°, 120°, 160°, 200°, 240°, 280°, 320°	25	16	512 × 512	
4	Mondial-Marmi	0°, 5°, 10°, 15°, 30°, 45°, 60°, 75°, 90°	12	16	272 × 272	
5	Outex	0°, 5°, 10°, 15°, 30°, 45°, 60°, 75°, 90°	45	20	128 × 128	
6	Vectorial	0°, 10°, 20°, 30°, 40°, 50°, 60°, 70°, 80°, 90°	20	16	225 × 225	
7	Pap smear	0°	2	204	Variable	
8	2D HeLa	0°	10	20	Variable	

Table 3
Classification results (1-NN and SVM classifier).

Classifier	1-NN (L_2)								SVM							
Dataset	1	2	3	4	5	6	7	8	1	2	3	4	5	6	7	8
Baseline																
$\mathrm{LBP}^{ri}_{8,1}$	*70.6*	*83.4*	*89.9*	*82.5*	*74.4*	*70.5*	*75.5*	*60.1*	*69.6*	*83.6*	*89.3*	*76.5*	*77.5*	*66.2*	*82.4*	*63.6*
Unlabelled methods																
$\mathrm{DLBP}^{ri}_{8,1}$	58.1	81.9	83.4	80.9	67.5	55.7	72.7	50.7	54.9	81.9	81.4	74.6	69.7	53.4	77.7	53.1
Labelled methods																
$\mathrm{LBP}^{riu2}_{8,1}$	**70.0**	**84.6**	**89.5**	**82.4**	**74.4**	**71.9**	**75.2**	59.1	**69.0**	**84.5**	**88.9**	**76.4**	**77.4**	**70.3**	**81.9**	62.1
$\mathrm{L\text{-}DLBP}^{ri}_{8,1}$	66.4	82.8	89.0	81.5	72.7	54.8	73.2	**60.0**	64.9	82.8	88.1	75.5	75.6	52.7	81.6	**63.7**
$\mathrm{HV\text{-}DLBP}^{ri}_{8,1}$	67.9	83.4	87.5	80.2	74.2	65.7	74.2	**60.0**	66.2	83.3	86.5	74.5	77.1	61.3	81.6	63.5
$\mathrm{HR\text{-}LBP}^{ri}_{8,1}$	66.4	80.9	89.0	81.5	72.6	54.8	73.2	**60.0**	64.9	80.9	88.1	75.6	75.5	52.8	81.6	63.3
$\mathrm{DF\text{-}DLBP}^{ri}_{8,1}$	69.4	81.7	**89.5**	81.5	74.2	61.4	65.0	**60.0**	68.1	81.7	88.8	75.6	77.1	59.2	75.2	63.2
$\mathrm{LTP}^{ri}_{8,1}$	72.4	91.8	91.5	86.1	76.5	77.8	81.2	62.0	74.0	91.2	91.2	85.2	77.5	78.7	84.2	65.1
$\mathrm{LTP}^{riu2}_{8,1}$	72.8	92.0	91.5	86.1	77.4	79.1	81.8	61.0	74.3	91.3	90.9	85.2	78.5	80.2	85.0	63.4
Baseline																
$\mathrm{LBP}^{ri}_{16,2}$	*74.9*	*95.0*	*86.6*	*87.3*	*79.5*	*75.4*	75.7	65.2	*72.7*	*93.4*	*85.3*	*78.9*	*83.0*	*75.0*	*80.5*	*71.0*
Unlabelled methods																
$\mathrm{DLBP}^{ri}_{16,2}$	64.9	90.5	82.7	85.6	70.2	66.6	72.2	56.0	56.4	89.4	75.4	75.6	71.5	61.9	74.5	56.5
Labelled methods																
$\mathrm{LBP}^{riu2}_{16,2}$	73.5	**95.0**	85.9	85.4	78.4	68.1	74.1	64.0	71.2	93.2	84.2	77.3	81.3	69.7	79.6	67.9
$\mathrm{L\text{-}DLBP}^{ri}_{16,2}$	74.4	**95.0**	86.5	86.9	79.1	69.4	75.5	66.5	71.9	**93.4**	85.0	78.6	82.5	66.5	80.5	70.5
$\mathrm{HV\text{-}DLBP}^{ri}_{16,2}$	**74.8**	**95.0**	86.5	**87.2**	**79.5**	**74.8**	**75.6**	65.1	**72.5**	**93.4**	**85.2**	**78.8**	**82.9**	**73.6**	**80.6**	**70.8**
$\mathrm{HR\text{-}DLBP}^{ri}_{16,2}$	74.3	**95.0**	86.5	86.9	79.1	69.4	75.5	66.4	71.8	**93.4**	85.0	78.5	82.5	66.5	80.5	70.6
$\mathrm{DF\text{-}DLBP}^{ri}_{16,2}$	**74.8**	**95.0**	**86.6**	86.9	79.4	73.4	71.7	**66.6**	72.4	**93.4**	**85.2**	78.6	82.8	71.7	76.6	70.4
$\mathrm{LTP}^{ri}_{16,2}$	82.0	97.3	94.7	93.9	86.5	80.8	80.5	65.5	83.7	96.6	94.8	92.5	88.5	81.4	82.9	71.1
$\mathrm{LTP}^{riu2}_{16,2}$	81.0	97.5	93.1	92.4	86.3	83.4	82.1	64.3	82.0	96.5	91.9	92.3	88.1	82.7	84.6	68.3

6. Results

In this section we summarise and discuss the results of the experiments with respect to classification accuracy, compression ratio and computing time.

6.1. Classification accuracy

The average accuracy values for each datasets are reported in Tables 3 and 4. In boldface we have highlighted the best result achieved with dominant local binary patterns. The results

Table 4
Classification results (naïve Bayes and NMC classifier).

Classifier	Naïve Bayes								NMC (L_2)							
Dataset	1	2	3	4	5	6	7	8	1	2	3	4	5	6	7	8
Baseline																
$\mathrm{LBP}^{ri}_{8,1}$	*67.0*	*81.7*	*87.2*	*80.4*	*74.2*	*75.9*	*68.2*	*51.7*	*57.6*	*77.3*	*81.6*	*76.1*	*70.7*	*59.8*	*57.7*	*50.7*
Unlabelled methods																
$\mathrm{DLBP}^{ri}_{8,1}$	49.8	75.5	79.0	78.2	64.2	45.4	56.7	38.4	46.8	75.5	73.3	75.2	62.9	52.4	54.2	44.7
Labelled methods																
$\mathrm{LBP}^{riu2}_{8,1}$	60.5	**82.4**	84.9	77.7	72.0	**72.7**	58.7	50.3	56.4	**77.9**	81.0	**75.9**	69.8	**60.1**	**57.1**	49.8
$\mathrm{L\text{-}DLBP}^{ri}_{8,1}$	59.2	75.8	**87.2**	**78.6**	70.5	43.9	**59.2**	**50.9**	55.3	76.6	80.2	**75.9**	69.1	52.0	57.0	**50.6**
$\mathrm{HV\text{-}DLBP}^{ri}_{8,1}$	59.5	79.7	84.4	77.1	**73.5**	56.6	58.3	**50.9**	55.6	77.3	79.2	74.9	**70.5**	57.8	56.7	**50.6**
$\mathrm{HR\text{-}DLBP}^{ri}_{8,1}$	59.2	77.0	**87.2**	**78.6**	69.2	43.9	**59.2**	**50.9**	55.3	75.1	80.2	**75.9**	69.1	52.0	57.0	**50.6**
$\mathrm{DF\text{-}DLBP}^{ri}_{8,1}$	**61.0**	74.0	**87.2**	**78.6**	**73.5**	55.0	58.0	50.8	**56.8**	76.0	**81.3**	**75.9**	**70.5**	57.0	54.3	**50.6**
$\mathrm{LTP}^{ri}_{8,1}$	67.1	84.4	88.4	81.5	61.1	80.2	70.0	52.8	57.7	79.5	86.3	81.1	53.2	63.4	62.6	52.7
$\mathrm{LTP}^{riu2}_{8,1}$	65.4	85.3	88.0	80.6	60.0	79.6	68.9	51.4	58.1	79.7	86.9	80.9	53.7	64.5	63.6	51.2
Baseline																
$\mathrm{LBP}^{ri}_{16,2}$	*72.6*	*93.8*	*86.7*	*84.8*	*80.8*	*84.0*	*73.1*	*56.4*	*58.5*	*91.0*	*79.5*	*79.5*	*78.1*	*61.5*	*68.3*	*56.4*
Unlabelled methods																
$\mathrm{DLBP}^{ri}_{16,2}$	55.2	90.7	76.8	79.3	71.1	54.1	64.7	48.8	48.5	89.2	71.9	77.9	68.7	55.5	64.8	47.8
Labelled methods																
$\mathrm{LBP}^{riu2}_{16,2}$	60.7	91.9	80.1	79.2	78.5	74.0	67.2	58.3	54.4	90.2	75.0	74.6	73.2	**61.2**	63.5	52.9
$\mathrm{L\text{-}DLBP}^{ri}_{16,2}$	67.8	**92.8**	81.7	82.5	81.3	59.6	71.5	61.5	58.1	**91.1**	79.3	79.2	77.7	59.3	**68.3**	56.3
$\mathrm{HV\text{-}DLBP}^{ri}_{16,2}$	**71.7**	92.2	84.5	**86.0**	**83.7**	**75.4**	**72.8**	**62.0**	**58.4**	91.0	**79.4**	**79.4**	**78.0**	**61.2**	**68.3**	**56.4**
$\mathrm{HR\text{-}DLBP}^{ri}_{16,2}$	66.3	**92.8**	81.3	82.2	81.2	59.6	71.6	61.1	58.0	**91.1**	79.3	79.2	77.7	59.3	**68.3**	56.3
$\mathrm{DF\text{-}DLBP}^{ri}_{16,2}$	71.3	92.6	**84.9**	83.1	83.6	71.5	69.4	60.0	**58.4**	91.0	**79.4**	79.3	77.9	60.6	67.4	56.2
$\mathrm{LTP}^{ri}_{16,2}$	80.5	94.4	93.7	91.9	80.3	84.6	81.8	55.2	69.6	91.3	92.1	89.4	68.6	68.1	69.1	57.7
$\mathrm{LTP}^{riu2}_{16,2}$	76.0	94.2	92.0	88.3	78.7	84.7	80.8	58.7	66.9	91.1	89.0	84.8	72.1	72.0	80.0	52.7

Table 5
Compression ratio.

	Dataset							
	1	2	3	4	5	6	7	8
$\mathrm{LBP}^{riu2}_{8,1}$	4:1	4:1	4:1	**4:1**	4:1	4:1	4:1	**4:1**
$\mathrm{DLBP}^{ri}_{8,1}$	**5:1**	**7:1**	**5:1**	**4:1**	**5:1**	**18:1**	5:1	**4:1**
$\mathrm{L\text{-}DLBP}^{ri}_{8,1}$	**5:1**	**7:1**	**5:1**	**4:1**	**5:1**	**18:1**	5:1	**4:1**
$\mathrm{HR\text{-}DLBP}^{ri}_{8,1}$	**5:1**	**7:1**	**5:1**	**4:1**	**5:1**	**18:1**	5:1	**4:1**
$\mathrm{HV\text{-}DLBP}^{ri}_{8,1}$	4:1	5:1	4:1	**4:1**	4:1	9:1	4:1	**4:1**
$\mathrm{DF\text{-}DLBP}^{ri}_{8,1}$	4:1	**7:1**	4:1	**4:1**	4:1	12:1	**7:1**	**4:1**
$\mathrm{LBP}^{riu2}_{16,2}$	**229:1**	**229:1**	**229:1**	**229:1**	**229:1**	229:1	229:1	229:1
$\mathrm{DLBP}^{ri}_{16,2}$	96:1	274:1	100:1	56:1	66:1	**1029:1**	58:1	33:1
$\mathrm{L\text{-}DLBP}^{ri}_{16,2}$	96:1	274:1	100:1	56:1	66:1	**1029:1**	53:1	29:1
$\mathrm{HR\text{-}DLBP}^{ri}_{16,2}$	96:1	274:1	100:1	56:1	66:1	**1029:1**	53:1	29:1
$\mathrm{HV\text{-}DLBP}^{ri}_{16,2}$	25:1	96:1	45:1	22:1	20:1	317:1	12:1	5:1
$\mathrm{DF\text{-}DLBP}^{ri}_{16,2}$	29:1	137:1	40:1	57:1	30:1	343:1	**294:1**	**51:1**

clearly show that in no case DLPB emerged out as the best option for determining the dominant patterns: the labelled approach always performed better than the unlabelled one. Noteworthily, the decrease in accuracy that one may come into when using unlabelled dominant patterns can be rather conspicuous, and in more than one case even in excess of 10 percentage points. We also observe that dominant local binary patterns – no matter the strategy adopted to determine them – in general performed worse than the original LBP, though in most cases the difference was very slight. In the (8, 1) configuration it is worth mentioning the very good performance of LBP^{riu2} which provided the best results in most of the datasets and in some cases it was even better than the original LBP. In the (16, 2) configuration, however, the accuracy of uniform patterns declined and it was surpassed by the other methods, among which HV-DLBP stands out as the best in the majority of the classification problems.

6.2. Compression ratio

Table 5 reports the compression ratio (rounded to the nearest integer) achieved by each feature reduction scheme in each dataset. This is the proportion between the number of patterns at the baseline (those produced by $\mathrm{LBP}^{ri}_{8,1}$ and $\mathrm{LBP}^{ri}_{16,2}$) and the number of dominant patterns generated by the various methods. For each dataset the highest values are indicated in boldface. The values are in general quite high, ranging from 4:1 to 18:1 in the (8, 1) configuration and from 20:1 to 1029:1 in the (16, 2) configuration. Note that the compression ratio provided by the '*riu*2' model is established a priori, hence is the same for all datasets. On average we can see that DF-DLBP and HV-DLBP provided lower compression ratios than the other methods. As for the former, the outcome is in perfect agreement with the theoretical considerations reported in Section 4.4; as for the latter, the result suggests that the variance of the patterns' frequency is distributed more evenly among the patterns than the frequency.

Table 6
Average computing time (feature selection + classification, seconds per problem; 1-NN and SVM classifier).

Classifier	1-NN (L_2)								SVM							
Dataset	1	2	3	4	5	6	7	8	1	2	3	4	5	6	7	8
$\text{LBP}^{ri}_{8,1}$	0.53	**0.08**	**0.12**	**0.08**	**0.28**	0.11	0.09	0.10	27.21	0.68	2.45	0.58	8.93	1.64	0.38	0.50
$\text{LBP}^{riu2}_{8,1}$	0.53	**0.08**	**0.12**	**0.08**	**0.28**	0.11	**0.07**	**0.10**	27.42	0.64	**2.03**	**0.49**	**6.89**	**1.30**	0.28	0.41
$\text{DLBP}^{ri}_{8,1}$	**0.52**	**0.08**	**0.12**	**0.08**	**0.28**	0.11	**0.07**	**0.10**	22.48	0.61	2.45	0.62	8.31	1.71	**0.27**	**0.38**
$\text{L-DLBP}^{ri}_{8,1}$	**0.52**	0.09	**0.12**	**0.08**	**0.28**	**0.10**	**0.07**	**0.10**	25.29	0.60	2.32	0.54	7.24	1.90	**0.27**	**0.38**
$\text{HR-DLBP}^{ri}_{8,1}$	0.54	**0.08**	**0.12**	**0.08**	**0.28**	0.11	**0.07**	**0.10**	**24.71**	0.60	2.35	0.55	7.74	1.63	**0.27**	0.37
$\text{HV-DLBP}^{ri}_{8,1}$	0.53	0.14	**0.12**	0.07	0.28	**0.10**	**0.07**	**0.10**	25.21	0.59	2.32	0.57	7.22	1.95	0.28	**0.38**
$\text{DF-DLBP}^{ri}_{8,1}$	0.61	0.11	0.15	0.09	0.35	0.11	0.11	0.13	25.53	**0.57**	2.31	0.56	7.26	1.72	**0.27**	0.43
$\text{LTP}^{ri}_{8,1}$	0.73	0.10	0.17	0.10	0.38	0.14	0.07	0.09	110.53	2.82	10.68	2.34	35.10	6.70	0.81	1.78
$\text{LTP}^{riu2}_{8,1}$	0.72	0.11	0.18	0.11	0.39	0.15	0.08	0.10	110.61	2.74	10.70	2.43	33.51	5.83	0.52	1.52
$\text{LBP}^{ri}_{16,2}$	1.04	0.14	0.24	0.13	0.60	0.19	0.20	0.15	272.11	5.01	29.61	6.85	107.44	8.39	21.55	8.01
$\text{LBP}^{riu2}_{16,2}$	**0.53**	**0.08**	**0.12**	**0.07**	**0.27**	**0.10**	**0.07**	**0.10**	**23.75**	**0.61**	2.28	**0.55**	**7.50**	**1.55**	**0.31**	**0.42**
$\text{DLBP}^{ri}_{16,2}$	1.16	0.18	0.33	0.19	0.70	0.27	0.29	0.22	25.61	0.64	**2.27**	0.75	9.24	1.77	0.63	0.88
$\text{L-DLBP}^{ri}_{16,2}$	0.75	0.10	0.16	0.09	0.39	0.14	0.11	0.13	27.10	0.65	2.40	0.67	9.26	1.57	0.52	0.94
$\text{HR-DLBP}^{ri}_{16,2}$	1.24	0.18	0.32	0.19	0.70	0.27	0.28	0.20	29.49	0.70	2.59	0.75	9.80	1.75	0.63	1.01
$\text{HV-DLBP}^{ri}_{16,2}$	0.63	0.11	0.17	0.10	0.42	0.15	0.13	0.14	34.57	0.70	2.59	0.82	14.47	1.62	1.63	3.25
$\text{DF-DLBP}^{ri}_{16,2}$	1.11	0.18	0.30	0.18	0.67	0.26	0.27	0.21	34.44	0.70	2.93	0.72	11.45	1.74	0.43	0.75
$\text{LTP}^{ri}_{16,2}$	1.57	0.20	0.35	0.20	0.92	0.32	0.31	0.19	514.91	11.04	65.81	15.67	198.90	21.44	36.09	15.40
$\text{LTP}^{riu2}_{16,2}$	0.73	0.11	0.18	0.10	0.38	0.15	0.07	0.09	27.78	0.68	2.60	0.72	10.19	1.53	0.55	0.48

Table 7
Average computing time (feature selection + classification, seconds per problem; naïve Bayes and NMC classifier).

Classifier	Naïve Bayes								NMC							
Dataset	1	2	3	4	5	6	7	8	1	2	3	4	5	6	7	8
$\text{LBP}^{ri}_{8,1}$	0.05	0.02	0.02	**0.01**	0.03	0.02	**0.01**	**0.01**	**0.01**	$\ast$	$\ast$	$\ast$	**0.01**	$\ast$	$\ast$	$\ast$
$\text{LBP}^{riu2}_{8,1}$	**0.03**	**0.01**	0.02	**0.01**	0.02	**0.01**	**0.01**	**0.01**	**0.01**	$\ast$	0.01	$\ast$	**0.01**	$\ast$	$\ast$	$\ast$
$\text{DLBP}^{ri}_{8,1}$	**0.03**	0.01	**0.01**	**0.01**	0.02	**0.01**	**0.01**	**0.01**	**0.01**	$\ast$	0.01	$\ast$	**0.01**	$\ast$	$\ast$	0.01
$\text{L-DLBP}^{ri}_{8,1}$	**0.03**	0.01	**0.01**	**0.01**	0.02	**0.01**	**0.01**	**0.01**	**0.01**	$\ast$	0.01	$\ast$	**0.01**	$\ast$	$\ast$	$\ast$
$\text{HR-DLBP}^{ri}_{8,1}$	0.04	0.01	**0.01**	**0.01**	0.02	0.01	**0.01**	**0.01**	**0.01**	$\ast$	0.01	$\ast$	**0.01**	$\ast$	0.01	$\ast$
$\text{HV-DLBP}^{ri}_{8,1}$	**0.03**	0.01	0.02	**0.01**	0.02	0.01	**0.01**	**0.01**	**0.01**	$\ast$	0.01	$\ast$	**0.01**	$\ast$	$\ast$	$\ast$
$\text{DF-DLBP}^{ri}_{8,1}$	0.14	0.01	0.05	0.02	0.10	0.01	0.02	**0.01**	0.06	0.01	0.03	0.01	0.04	0.01	0.02	0.01
$\text{LTP}^{ri}_{8,1}$	0.06	0.01	0.02	**0.01**	0.04	0.02	0.02	**0.01**	**0.01**	$\ast$	$\ast$	$\ast$	**0.01**	$\ast$	$\ast$	$\ast$
$\text{LTP}^{riu2}_{8,1}$	**0.03**	**0.01**	**0.01**	**0.01**	0.02	**0.01**	**0.01**	**0.01**	**0.01**	$\ast$	**0.01**	$\ast$	**0.01**	$\ast$	$\ast$	$\ast$
$\text{LBP}^{ri}_{16,2}$	2.03	0.27	0.47	0.26	1.00	0.38	0.21	0.21	0.45	**0.03**	0.06	0.02	0.22	0.05	0.04	0.03
$\text{LBP}^{riu2}_{16,2}$	0.04	**0.01**	**0.02**	**0.01**	**0.02**	**0.02**	**0.01**	**0.01**	**0.01**	$\ast$	$\ast$	$\ast$	**0.01**	$\ast$	$\ast$	$\ast$
$\text{DLBP}^{ri}_{16,2}$	0.79	0.07	0.24	0.13	0.50	0.11	0.25	0.13	0.63	0.07	0.15	0.08	0.38	0.11	0.15	0.07
$\text{L-DLBP}^{ri}_{16,2}$	0.25	0.02	0.07	0.04	0.17	0.03	0.07	0.03	0.14	0.01	0.03	0.01	0.09	0.02	0.03	0.02
$\text{HR-DLBP}^{ri}_{16,2}$	0.79	0.07	0.26	0.14	0.51	0.12	0.26	0.13	0.68	0.07	0.16	0.08	0.38	0.12	0.15	0.07
$\text{HV-DLBP}^{ri}_{16,2}$	0.34	0.04	0.09	0.05	0.23	0.06	0.09	0.08	0.17	0.02	0.04	0.02	0.11	0.03	0.04	0.02
$\text{DF-DLBP}^{ri}_{16,2}$	0.73	0.12	0.24	0.13	0.48	0.18	0.14	0.13	0.60	0.06	0.14	0.07	0.36	0.10	0.14	0.07
$\text{LTP}^{ri}_{16,2}$	4.20	0.45	0.89	0.42	2.02	0.69	0.34	0.38	0.93	0.05	0.13	0.05	0.43	0.10	0.09	0.04
$\text{LTP}^{riu2}_{16,2}$	0.04	**0.01**	0.02	**0.01**	0.03	0.02	**0.01**	**0.01**	**0.01**	$\ast$	0.01	$\ast$	**0.01**	0.01	$\ast$	$\ast$

NOTE: Values lower than 0.01 s have been indicated with symbol '$\ast$'

With respect to the relation between accuracy and compression ratio it is first of all important to notice that the number of retained features depends a great deal on the dataset: more complex textures are likely to require a higher number of features for a correct description. The results in fact show that the highest compression ratio was obtained with dataset six, which is composed of artificial textures mainly made up by simple geometric primitives. Tables 3–5 also suggest there is a direct trade-off – more noticeable with the (16, 2) configuration – between performance and compression: HV-DLBP provided the best performance yet the worst compression; the reverse occurred with LBP^{riu2}. The user interested in high accuracy with fairly good compression (at least much better compression than LBP^{ri}) would choose HV-DLBP, whereas LBP^{riu2} would be a good candidate for the best compression.

6.3. Computing time

In Section 4 we have shown that the time complexity of the feature reduction schemes based on training can be, in the various cases, either $\mathcal{O}(MN\log N)$ or $\mathcal{O}(\max[MN, N\log N])$. On the other hand, the time complexity of the nearest neighbour, nearest mean and naïve Bayes classifier, can be estimated (assuming linear scan) as $\mathcal{O}(NM)$, $\mathcal{O}(NC)$ and $\mathcal{O}(NC)$, respectively; whereas that of SVM is $\mathcal{O}\left(\max[M, N] \times \min[M, N]^2\right)$ time [4]. As a result, the whole process (feature selection + classification) is in principle dominated by the feature selection step when classification is based on any of the nearest neighbour, nearest mean and naïve Bayes classifier; and by the classification step when SVM is used. Consequently, there is in theory no advantage in performing feature selection with the former group of classifiers, since feature selection is computationally more demanding than classification itself; by contrast, feature selection can beneficial with SVM. In practice things can be slightly different due to diverse implementations and the presence of overhead. In the case of the naïve Bayes classifier, for instance, the multinomial implementation chosen in our experiments can increase the computational cost of the method.

The results show that the original LBP in the (8, 1) configuration is approximately as fast as the reduced-dimension versions with all the classifiers tested (see Tables 6 and 7). In the (16, 2) configuration no clear trend emerges with the 1-NN classifier, there is however a significant reduction in the computing time when classification is performed through SVM. In this case the reduced-dimension version can be ten times faster than the original LBP.

7. Conclusions

In this paper we investigated the problem of determining sets of discriminative patterns from local binary patterns. The strategies to obtain dominant patterns can be divided in two groups: those that keep knowledge of the patterns labels (labelled methods) and those that discard such knowledge (unlabelled methods). It was the aim of this paper to answer a very specific question: which is the best option? The result of our study is clear-cut: in none of the datasets considered in the experiments did the unlabelled strategy outperform the labelled approach. It seems therefore that neglecting information about the patterns labels has negative effects on the discrimination capability of the method. The reader who is interested in utilising dominant local binary patterns can clearly see that keeping data labels can improve performance. Moreover, and differently from what other authors suggested [17], we found that dominant local binary patterns – no matter which strategy we adopt to determine them – were in most cases less accurate than the original LBP, though the difference was slight in some cases.

We also investigated the potential advantages that dominant local binary patterns can provide in terms of compression ratio and reduction of computing time. The dimensionality reduction can be conspicuous, as shown in Table 5. In our experiments the number of features to retain was determined indirectly by assigning the threshold value of a given parameter, such as, depending on the method, the percentage of the variance (HV-DLBP), of the average rank (HR-DLBP) or of the population (DLBP and L-DLBP) accounted for by the selected patterns. As suggested by other authors [17] we set this threshold at 0.8, a seemingly reasonable value also supported by empirical laws such as the 80/20 rule [19]. The effect that changing this parameter may have on accuracy and compression ratio is interesting subject for future studies.

Regarding computing time, the main result is that the overall gain (or loss) depends on the classification strategy. The use of dominant patterns theoretically provides no advantage when classification is based on computationally cheap classifiers, such as nearest neighbour, nearest mean and naïve Bayes: asymptotically the original LBP should perform, at least in principle, faster, in these cases. The experimental results clearly confirmed this assumption for the nearest neighbour and nearest mean classifiers, whereas showed that some increase in speed can be obtained with the naïve Bayes, most probably because reducing the number of features in this case also reduces the overhead related to building the multinomial distribution (cf. Section 6.3). By contrast, just the reverse occurs with SVM: here, due to the intrinsically higher complexity of the classifier, the overall computing time benefits from a reduction of the space dimension.

In summary, the 'take-home' message of this study is that the use of dominant binary patterns requires particular care. With respect to accuracy our results indicate that keeping track of the patterns labels improves the discrimination capability of DLBP. Moreover, in our experiments the baseline descriptors (i.e.: $\mathrm{LBP}^{ri}_{8,1}$ and $\mathrm{LBP}^{ri}_{16,2}$) outperformed their reduced-dimension counterparts in most cases. As regards to speed, we have pointed out that using dominant binary patterns is not necessarily advantageous, because the process of selecting the dominant patterns has itself a computational cost, which may not be paid back in the classification stage. A natural question therefore arises whether and when searching for dominant patterns makes sense altogether. The answer depends a great deal on the specific application. With some datasets (e.g. datasets two and three) the use of simple classifiers such as 1-NN and NMC plus the baseline descriptors proved faster and at least as good as the other methods: in such situations we could see little reason for using dominant patterns. By contrast, when more complex and computationally-demanding methods (such as SVM) turn out to be more accurate (as it happens, for instance, with dataset seven), the reduction in dimensionality can significantly reduce the whole processing time: in similar cases the use of dominant binary patterns can be a viable approach.

Calibration against local ternary patterns (LTP) showed in general a better performance of this method versus LBP, as one would expect. An interesting subject for future studies would be the extension of the experiments to a larger set of methods of the class Histograms of Equivalent Patterns [9], LTP included.

References

[1] F. Bianconi, A. Fernández, On the occurrence probability of local binary patterns: a theoretical study., J. Math. Imaging Vis. 40 (3) (2011) 259–268.

[2] S. Brahnam, L.C. Jain, L. Nanni, A. Lumini (Eds.), *Local binary patterns: new variants and applications*, vol. 506 of *Studies in Computational Intelligence*, Springer, 2014.

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

[4] O. Chapelle, Training a support vector machine in the primal, Neural Comput. 19 (2007) 1155–1178.

[5] V. Cherkassky, Y. Ma, Practical selection of SVM parameters and noise estimation for SVM regression, Neural Netw. 17 (1) (2004) 113–126.

[6] N.P. Doshi, G. Schaefer, Dominant multi-dimensional local binary patterns, in: Proceedings of the IEEE International Conference on Signal Processing, Communications and Computing, ICSPCC 2013, Kunming, China, 2013.

[7] R.O. Duda, P.E. Hart, D.G. Stork, Pattern Classification, second edition, Wiley-Interscience, 2001.

[8] A. Fernández, O. Ghita, E. González, F. Bianconi, P.F. Whelan, Evaluation of robustness against rotation of LBP, CCR and ILBP features in granite texture classification, Mach. Vis. Appl. 22 (6) (2011) 913–926.

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

[10] X. Fu, M. Shi, H. Wei, H. Chen, Fabric defect detection based on adaptive local binary patterns, in: Proceedings of the IEEE International Conference on Robotics and Biomimetics (ROBIO 2009), 2009, pp. 1336–1340.

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

[12] Y. Guo, G. Zhao, M. Pietikäinen, Discriminant features for texture description, Pattern Recognit. 45 (10) (2012) 3825–3843.

[13] J. Jantzen, J. Noras, G. Dounias, B. Bjerregaard, Pap-smear benchmark data for pattern classification, in: Proceedings of NiSIS 2005: Nature inspired Smart Information Systems (NiSIS), Albufeira, Portugal, 2005.

[14] G. Kylberg, I.-M. Sintorn, Evaluation of noise robustness for local binary pattern descriptors in texture classification, EURASIP J. Image Vid. Process. 2013 (17) (2013).

[15] A general local binary pattern (LBP) implementation for MATLAB®, 2013. Available online at http://www.cse.oulu.fi/CMV/Downloads/LBPMatlab (last accessed 11.03.14).

[16] LDLBP, Code, data and results related to this paper, 2015, Available online at http://webs.uvigo.es/antfdez/downloads.html.

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

[18] LIBSVM – A Library for Support Vector Machines, 2014. Available online at http://www.csie.ntu.edu.tw/~cjlin/libsvm/ (last accessed 11.03.15).

[19] W. Lidwell, K. Holden, J. Butler, Universal Principles of Design, Rockport Publishers, Beverly (USA), 2003.

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

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

[22] L. Nanni, S. Brahnam, A. Lumini, Selecting the best performing rotation invariant patterns in local binary/ternary patterns, in: Proceedings of the 2010 International Conference on Image Processing, Computer Vision, and Pattern Recognition (IPCV'10), Las Vegas, USA, 2010b, pp. 369–375.

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

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

[25] National Institute on Ageing, 2D HeLa dataset. Available online at http://ome.grc.nia.nih.gov/iicbu2008/hela/index.html (last accessed 18.03.15).

[26] PRTools: A Matlab toolbox for pattern recognition, 2014, Available online at http://prtools.org/ (last accessed 11.03.14).

[27] X. Tan, B. Triggs, Enhanced local texture feature sets for face recognition under difficult lighting conditions, in: Analysis and Modelling of Faces and Gestures, in: Lecture Notes in Computer Science, vol. 4778, Springer, 2007, pp. 168–182.