Multilayer descriptors for medical image classification

Alessandra Lumini [a], Loris Nanni [b], Sheryl Brahnam [c,*]

[a] DISI, University of Bologna, Via Sacchi 3, 47521 Cesena, Italy
[b] Department of Information Engineering, Via Gradenigo 6/A, 35131 Padova, Italy
[c] Computer Information Systems, Missouri State University, 901 S. National, Springfield, MO 65804, USA

ARTICLE INFO

ABSTRACT

In this paper, we propose a new method for improving the performance of 2D descriptors by building an n-layer image using different preprocessing approaches from which multilayer descriptors are extracted and used as feature vectors for training a Support Vector Machine. The different preprocessing approaches are used to build different n-layer images ($n=3$, $n=5$, etc.). We test both color and gray-level images, two well-known texture descriptors (Local Phase Quantization and Local Binary Pattern), and three of their variants suited for n-layer images (Volume Local Phase Quantization, Local Phase Quantization Three-Orthogonal-Planes, and Volume Local Binary Patterns). Our results show that multilayers and texture descriptors can be combined to outperform the standard single-layer approaches. Experiments on 10 datasets demonstrate the generalizability of the proposed descriptors. Most of these datasets are medical, but in each case the images are very different. Two datasets are completely unrelated to medicine and are included to demonstrate the discriminative power of the proposed descriptors across very different image recognition tasks.

A MATLAB version of the complete system developed in this paper will be made available at https://www.dei.unipd.it/node/2357.

1. Introduction

The amount of multidimensional visual information (e.g., 2D images, videos, 3D surface models of objects, and 3D tomographic images) uploaded to the internet on a daily basis has increased enormously in the past few years. The video hosting website YouTube, for instance, received more than 100 h of new video every minute in 2014. In the field of medical imaging, individual radiology departments routinely produce an enormous amount of multidimensional information that is warehoused in private and public medical databases. Such large quantities of data are difficult to manually label for further access and reuse. There is, as a result, an urgent need to develop sophisticated, efficient, and accurate image classification algorithms that are able to provide best matches for specific classification tasks. Many applications require the ability of discriminating among classes of images, to distinguish, for instance, pedestrians from objects for advanced driver assistance, for disease recognition from medical images, for landmark recognition for touristic purposes, and many other applications. In these cases, computer vision plays a central role. Key to successful

classification is the ability of representing images based on visual characteristics such as texture, color, and shape [1].

Automated visual tasks such as detection, localization, categorization, and recognition by visual features are important subjects of study in computer vision and image analysis. These tasks often prove difficult, however, due to high within-class variability that can be caused by a variety of factors: noise, distortions, illumination, scale changes, occlusion, and so on. Extracting features within visual databases is one of the most important steps in the classification process as success depends on the method adopted for extracting features from a given set of images. Particularly important are invariant image descriptors since they extract information from images which is invariant to noise, illumination, distortion, etc.

One class of invariant descriptors is color composition, which in a visual scene is robust to noise, image degradations, changes in size, resolution, and orientation. Most existing systems use various color descriptors [2,3] in order to retrieve relevant samples. Unfortunately, the classification performance of color descriptors is negatively affected by their lack of discriminative power.

Descriptors based on shape are considered more generic than color and are the most widely used descriptors in many application areas such as object detection and action recognition. There are many ways to represent shapes. Some examples include axial representation [4], primitive-based representation [5], histograms

* Corresponding author. Tel.: +1 417 8739979.
E-mail address: sbrahnam@missouristate.edu (S. Brahnam).

of oriented gradients [6], contour-based representation [7], and probability density function [8]. Most shape descriptors are robust to rigid transformations, noise, occlusions, and missing data.

Another class of descriptors is texture. Texture has received considerable attention [9–11] and has proven valuable in medical imaging, image retrieval, object recognition, and industrial product identification (e.g., classifying types of marble, ceramic tiles, parquet slabs, etc.). Various methods have been introduced to analyse texture in digital images. The most well-known methods are based on statistical approaches, such as histograms of gray-level pixel values and the Gray-Level Co-occurrence (GLC) matrix [12], wavelet transforms and Gabor filters [13], and the Local Binary Pattern (LBP) operator [14,15]. The LBP operator, in particular, has proven an efficient method for describing texture in 2D. It has received much attention from the scientific community, and many powerful variants have recently been proposed [16].

Much work has also been done exploring the integration of more than one feature to improve classification accuracy [17]; however, the performance of these combinations are strictly application dependent. In some cases less efficient features can degrade more efficient features with a loss of overall system accuracy.

Most recently, there has been a move to explore color in three dimensions. In [18] the RGB space is considered as three planes in a 3D image, and a 3D version of the Local Ternary Pattern (LTP) is then proposed that extracts features directly from the 3D image, instead of from each color space (i.e., separately from R, G, and B). Moreover, they show that it is possible to generate a multilayer image starting from gray value images using a 2D circular symmetric Gaussian filter. Please note that in some works (e.g. [18]) the authors use the term "volumetric descriptors" to denote descriptors extracted from a RGB image or any kind of images represented by more than a layer. In this work we use the term multilayers to denote such a concept to avoid misunderstanding with the the notion of "volumetric descriptor" that in the context of medical image analysis refers to some measure of volume in anatomically meaningful regions on a medical image.

In this paper we start from the idea in [18] that considered the RGB image as a three-layer image and extend this representation to preprocessed images, showing that it is possible to improve the performance of single-layer descriptors by building an n-layer image ($n=3$, $n=5$, etc.) using different preprocessing approaches.

Formally speaking if a single layer image can be considered as mappings from a spatial coordinate system into a value space, i.e. a mapping from $[1,2,…,m] \times [1,2,…,l] \rightarrow \Re$ for an $m \times l$ image, a n-layer image is a mapping to a n dimensional space: $[1,2,…,m] \times [1,2,…,l] \rightarrow \Re^n$. An RGB color image is a particular case of a multilayer image where $n=3$. This definition is different from that used in the literature for existing multilayer descriptors (e.g. [18,22]) since they were proposed for dynamical textures.

The n-layer images are here described using multilayer descriptors, and these feature vectors are fed into a Support Vector Machine (SVM) [19]. We test both color and gray-level images, two well-known texture descriptors (LBP and Local Phase Quantization), and three of their variants suited for multilayer images (viz., Volume Local Phase Quantization, Local Phase Quantization Three-Orthogonal-Planes, Volume Local Binary Patterns). Our results show that multilayer and texture descriptors can be combined to outperform standard single-layer approaches. The proposed approach is applied to ten datasets to demonstrate the generalizability of this approach. In this work, we do not deal with the segmentation problem, since all the datasets considered in the experiments contain only segmented image. The segmentation of large images that can contain parts of different textures is out of the scope of this paper.

The remainder of this paper is organized as follows. In Section 2 we describe the base approaches used in our system, and in Section 3 we present our system. We conclude in Section 4 by summarizing the significance of our work and by highlighting some future directions of exploration.

2. Base methods

In this section we provide an overview of the methods used for building the ensemble of descriptors. We describe the multilayer descriptors in Section 2.1. A very brief description of the preprocessing methods employed to generate the n-layer images is available in Section 2.2.

2.1. Multilayer descriptors

In this subsection we briefly explain the methods used for building the ensemble of multilayer descriptors.

2.1.1. Volumetric local binary pattern (VLBP)

VLBP, introduced by [22], is an extension of LBP, with the notion of self-similarity, central to conventional image texture, extended to the spatiotemporal domain. VLBP deals with dynamic texture analysis on the 2D time series and not on the full 3D data; therefore, it only provides rotation invariance towards rotations around the z-axis.

VLBP extends standard LBP to handle dynamic texture analysis by considering the joint distribution v of the gray-levels of $3P+3$ pixels ($P > 1$ is the number of local neighboring points around the center in one frame):

$$V = v(g_{t_c-L,c}, g_{t_c-L,0}, …, g_{t_c-L,P-1}, g_{t_c,c}, g_{t_c,0}, …, g_{t_c,P-1}, g_{t_c+L,c}, g_{t_c} + L, 0, …, g_{t_c+L,P-1})$$

where $g_{t,p}$ $t \in [t_c-L, t_c, t_c+L], p \in [0, …, P-1]$ corresponds to the gray value of P equally spaced pixels on a circularly symmetric neighbor of the center pixel c, in the center frame t_c and its previous/posterior frames with time interval L.

To obtain the gray-scale invariance as in LBP, the gray value of the center $g_{t_c,c}$ is subtracted from the gray values of the circularly symmetric neighborhood of each frame. To achieve invariance with respect to the scaling, the resulting code is binarized by considering only the signs of the differences. Finally, a unique $VLBP_{L,P,R}$ code is obtained by binary factorization (i.e., by weighting each pixel difference by a different binomial factor). The histogram obtained from the $VLBP_{L,P,R}$ code is normalized with respect to size variations by setting the sum of its bins to unity. Rotation invariance is obtained by rotating the neighbor set in three separate frames clockwise synchronously so that a minimal value is selected. Here we have used the following parameters $R=1$; $P=4$; $L=1$.

2.1.2. Volume local phase quantization (VLPQ)

VLPQ, introduced in [23], is based on the binary encoding of the phase information of the local Fourier transform at low frequency points and is an extension to the LPQ operator [21] used for spatial texture analysis. Since dynamic texture consists of a sequence of texture in the spatio-temporal domain, the Fourier transform estimation is performed locally using Short-Term Fourier Transform (STFT). Given a sequence f(x), STFT is computed over an M-by-M-by-N neighborhood $\mathcal{N}_x$ of x (where M is the spatial size and N the size in the temporal domain):

$$F(u, x) = w_u^T f_x,$$

where $\mathbf{w_u}$ is the basis of the 3D DFT at frequency $\mathbf{u}$, and $\mathbf{f_x}$ is a vector containing all pixels from the neighborhood $\mathcal{N}_x$. The computation is efficient since STFT can be evaluated for each pixel using 1D convolutions for each dimension, due to the separability of the basis functions.

The transform matrix is obtained using the thirteen lowest non-zero frequency points (see [23] for details) and by separating the real and imaginary parts of each component. This results in a 26-by-M^2N transform matrix:

$$\mathbf{W} = \left[\text{Re}\{\mathbf{w}_{\mathbf{u}_1}\}, \text{Im}\{\mathbf{w}_{\mathbf{u}_1}\}, \ldots, \text{Re}\{\mathbf{w}_{\mathbf{u}_{13}}\}, \text{Im}\{\mathbf{w}_{\mathbf{u}_{13}}\} \right].$$

Hence, the vector form of the STFT for all frequencies $\mathbf{u}_1, \ldots, \mathbf{u}_{13}$ can be written as

$$\mathbf{F_x} = \mathbf{Wf_x}.$$

In order to reduce the length of the resulting descriptor, dimension reduction by PCA and scalar quantization to a binary value is performed. Finally, the quantized coefficients are represented as integer values, using simple binary coding, and codified as a histogram of size L (where L is the dimension retained by PCA). Here we use windows of size=3 and L=10.

2.1.3. Local phase quantization three ortogonal planes (LPQ-TOP)

LPQ-TOP, proposed by Zhao and Pietikainen [24], is an extension of LPQ that, similarly to VLBP [22], takes into consideration dynamic texture. With LPQ-TOP the basic LPQ features are extracted independently from three orthogonal planes: *XY*, *XT* and *YT* (the *XY* plane provides the spatial domain information while the other two planes provide temporal information). The phase information is computed locally in a window for every image position in the three directions. The final descriptor, LPQ-TOP$_{Wx,Wy,Wt}$, is the concatenation of the three normalized histograms obtained by accumulating the occurrence of quantized phase code in each direction (*XY*, *XT* and *YT*). W_x, W_y, W_t are the window size parameters (i.e., the dimension of the neighborhood at each pixel position for computing LPQ). These parameters are set to different values in the spatial and temporal planes. Following [24], we employed LPQ-TOP$_{3,3,3}$ and LPQ-TOP$_{5,5,5}$; the two feature vectors are then concatenated and fed into a SVM.

2.2. Preprocessing approaches

One of the aims of this work is to test the idea of improving the performance of texture descriptors by building a multilayer image using different preprocessing approaches. In this way, *multilayer* descriptors could describe the different *nlayer* images, and each feature set produced by each descriptor could then be fed into a *SVM*. In exploring this approach, we test the following preprocessing methods: decomposition by wavelets (*Wave*); Multi-Resolution by Gaussian filters (*MRG*); Gradient image (*GR*); color space transform for color images (RGB to HSV and RGB to YUV); image enhancement using exposure based subimage histogram equalization for gray-level images (*ESI*); and contrast enhancement based on layered difference representations of 2D histograms for color images (*LDR*). In some cases, when the approach (e.g., *Wave*, *MRG*, and *GR*) is applied to color images, it is applied separately to each space; thus, three processed images are obtained using these preprocessing methods. The remainder of this section describes each of aforementioned preprocessing methods in more detail.

2.2.1. Wavelet (WAV)

WAV works well in many computer vision methods that detect and recognize objects of interest. In order to use *WAV* [25] for 2D decomposition, a 2D scaling function, $\varphi(x,y)$ and three 2D wavelets functions, $\psi^i(x,y)$ are needed, where i represents the three possible intensity variations along horizontal, vertical, and diagonal edges $i = \{H, V, D\}$.

In this paper, we use the Daubechies [26] wavelet family (*Wa*) with four vanishing moments and a single scale decomposition. The four preprocessed images are thus the approximation coefficients matrix and the three details coefficients matrices

(horizontal, vertical and diagonal). The processed images are resized to the size of the original image. In the experimental section, we label each of these as follows:

- **Wave1**: the approximation coefficients matrix;
- **Wave2**: the horizontal details;
- **Wave3**: the vertical details;
- **Wave4**: the diagonal details.

2.2.2. Multi-Resolution by Gaussian Filters (MRG)

In this paper, we test the Gaussian scale-space representation, where the original image is filtered to obtain two smoothed versions. This is accomplished using a 2D symmetric Gaussian low-pass filter of size k pixels (k=3 and k=5 in this work) with standard deviation 1.

2.2.3. Gradient image (GR)

Gradient is an image operation that is commonly used to detect edges. GR calculates the magnitude of the gradients of each pixel in the x and y directions based on its neighbors. The processed image at coordinates (x,y) is given by the magnitude of the gradient vector at the same coordinates. Thus, each pixel of a gradient image measures the change in intensity in a given direction of the corresponding point in the original image.

2.2.4. Color space transform

A color space transform converts images from one color space representation to another. The input images in the tested databases are given in the RGB color space. In this paper, the original images are transformed into the following two color spaces: YUV and HSV.

2.2.5. Image enhancement (ENH)

The type of image enhancement applied depends on whether the images are colored or gray-level:

- **ESI**: For gray-level images, we apply a novel Exposure-based Sub-Image Histogram Equalization method, proposed in [27], that enhances the contrast of low exposure gray-scale images. Exposure thresholds are computed that divide the original image into subimages of different intensity levels, and the histogram is clipped using a threshold value (as an average number of gray-level occurrences) to control enhancement rate. The individual histograms of the subimages are then equalized independently, with all subimages finally integrated into one complete image that is used for analysis.
- **LDR**: For color images, we apply a novel contrast enhancement algorithm, proposed in [28], that is based on the layered difference representation of 2D histograms. The basic idea behind this method is to enhance image contrast by amplifying the gray-level differences between adjacent pixels. Enhancement is viewed as a constrained optimization problem that is based on the observation that the gray-level differences occurring more frequently in the input image should be emphasized more in the output image. This is accomplished by first taking a given RGB image and transforming it into the YUV space, where the Y band is enhanced. The YUV image is then transformed back to the RGB space.

3. Proposed approach

Below is a simple description of our proposed algorithm:

1. INPUT IMAGE
The proposed method accepts both gray-level and color images.

1. DO COLOR SPACE CONVERSION

   This step is performed only on color images and is aimed at analyzing the image from different color perspectives (see Section 2). The original RGB image is converted into HSV and YUV, thus obtaining six additional layers.

2. PREPROCESS IMAGES

   Image preprocessing techniques are usually used to improve the quality of an image before processing into an application. In this approach we use some well-known preprocessing methods to highlight distinctive aspects of the images before performing feature extraction. The preprocessing techniques are outlined in Section 2.

3. MAKE *n*-LAYER IMAGE

   In the case of a gray-level input image, the output of the preprocessing step is a 9-layer image (the original image plus eight preprocessed layers), otherwise the output is a 33-layer image (resulting from the concatenation of 24 layers produced from the preprocessing step, i.e. 8 approaches × 3 color channels, plus three layers from the original RGB image, plus six color layers from step 1).

4. EXTRACT FEATURES

   The feature extraction is performed on both the input image and the *n*-layer image:

   41. TEXTURE DESCRIPTOR EXTRACTION

   The following two texture descriptors are extracted from the gray-level image (note: RGB images are converted to gray-level images) and their processed images: canonical LBP and LPQ.

   42. *n*-LAYER DESCRIPTOR EXTRACTION

   The following descriptors (which are the *n*-layer versions of the previous ones) are extracted from the *n*-layer image (or *k*-layer images, if a reduced number of layers is considered): VLBP, VLPQ, and LPQ-TOP.

5. CLASSIFY

   Classification is performed separately for each descriptor using SVMs as the base classifier, with both linear and radial basis function kernels. For each dataset, the best kernel and the best set of parameters are chosen using a 5-fold cross-validation approach on the training data (note: we are only determining here which descriptors to combine). The SVM is implemented using the widely used tool LibSVM (available at http://www.csie.ntu.edu.tw/~cjlin/libsvm/). The features used to train and test the SVMs are normalized to [0, 1] using the training data (the testing set is always completely blind).

6. NORMALIZE AND FUSE;

Before fusion, the scores obtained by each classifier are normalized to mean 0 and standard deviation 1. The final decision is obtained by combining the resulting scores from each *n*-layer descriptor and its related texture descriptor (VLPQ+LPQ, LPQTop+LPQ, and VLBP+LBP) using the weighted sum rule with the same settings applied in all datasets (see Section 4).

Fig. 1 illustrates the steps 1–6 above. The steps of an input image vary depending on its type as indicated by the color of the arrows: gray for gray-level images and white for colored images. It should also be noted that the dimension of the layer image is different in these two cases.

4. Experimental results

4.1. Datasets

In order to generalize the proposed variants of texture descriptors, we run several comparisons on ten datasets. Most of these datasets are medical, but in each case the images are very different. Two datasets, SM and PI, are completely unrelated to medicine and are included to demonstrate the discriminative power of the proposed descriptors across very different image recognition tasks.

- **PA** [29]: this is a pap-smear cell dataset containing 917 images unevenly distributed among seven classes acquired during
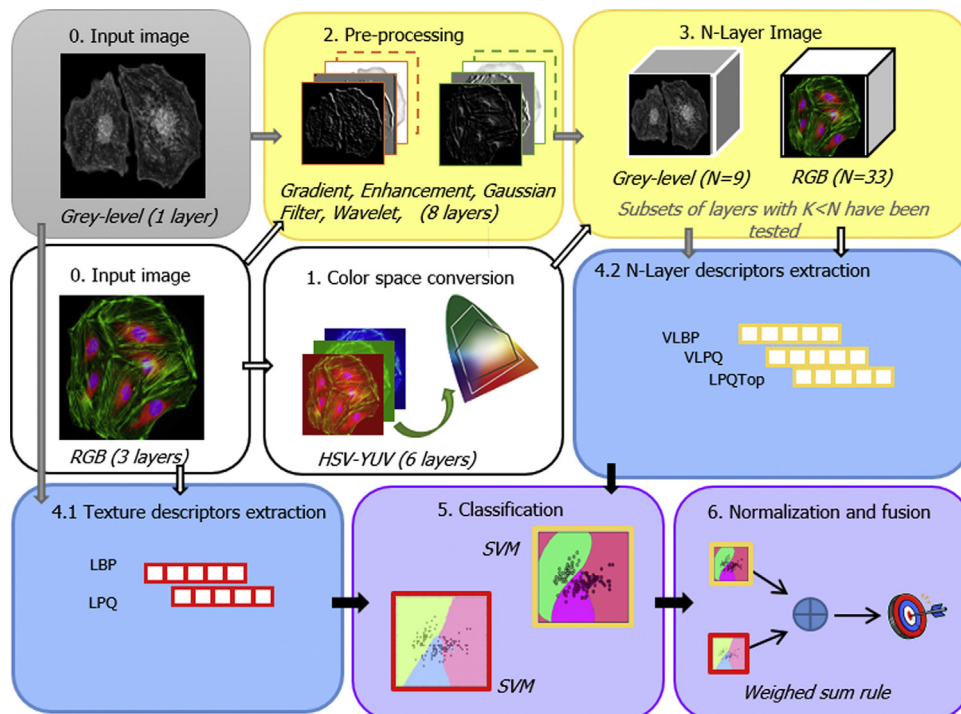

Fig. 1. Schema of the approach.

Pap-smear tests for the diagnosis of cervical cancer. The dataset is divided into 242 normal cells (superficial and intermediate squamous and columnar epithelial) and 675 abnormal cells (mild, moderate, and severe squamous non-keratinizing dysplasia and squamous cell carcinoma *in situ* intermediate).

- The *Vi* dataset in [20] contains 1500 *TEM* virus images divided into 15 classes: Denovirus, Astrovirus, CCHF, Cowpox, Dengue, Ebola, Influenza, Lassa, Marburg, Norovirus, Orf, Papilloma, Rift Valley, Rotavirus, and West Nile. Each image is 41×41 pixels and is saved as a lossless compressed 16-bit PNG. For a more detailed description of the dataset, see [20]. In our comparisons, the same 10-fold cross-validation testing protocol in [20] was applied. The dataset, masks, and indices used in the 10-fold cross-validation procedure are available at http://www.cb.uu.se/~gustaf/virustexture/index.html

- *HI* [30]: this dataset is composed of 2828 histology images captured from different organs and representing the four fundamental tissues, unevenly distributed (484 connective, 804 epithelial, 514 muscular, and 1026 nervous images). This dataset facilitates the study of different tissues and systems with specific markers for composing cells.

- *CH* [31]: this cell dataset contains 327 fluorescent microscopy images of Chinese Hamster Ovary cells that are distributed in five classes representative of different cell markers (anti-giantin, Hoechst 33258, anti-nop4, anti-tubulin, and anti-lamp2).

- *TR* [32, 33]: the LOCATE TRANSFECTED dataset contains 553 images of fluorescence-tagged or epitope-tagged protein transiently expressed in specific mouse subcellular organelles. The images are unevenly distributed in eleven classes: the same ten classes in *LE* and the additional cytoplasm class.

- *RN* this dataset contains 200 fluorescence fly cells microscopy images from *D. melanogaster* stained with *DAPI (4′, 6-diamidino-2-phenylindole)* to visualize the nucleus. Images are evenly

distributed among ten classes, each defined by the type of *RNAi* gene-knockdowns used.

- *PR* the protein dataset in [34] contains 118 DNA-binding Proteins and 231 Non-DNA-binding proteins, with texture descriptors extracted from the 2D distance matrix that represents each protein. The 2D matrix is obtained from the 3D tertiary structure of a given protein by considering only atoms that belong to the protein backbone (see [34] for details).

- *BR* the dataset in [35], containing 273 malignant and 311 benign breast cancer images.

- *SM* the dataset in [36], contains images extract from video-based smoke detection surveillance systems. The same division of the dataset into training/testing sets in [36] is used in all experiments reported in this paper.

- *PI* the dataset in [37] containing pictures extracted from digitalized pages of the Holy Bible of Borso d'Este, duke of Ferrara (Italy) from 1450 A.D. to 1471 A.D. PI is composed of 13 classes, characterized by a clear semantic meaning and significant search relevance.

A descriptive summary of each dataset along with the URL where each dataset can be downloaded is reported in Table 1. The testing protocol was the fivefold cross-validation method, except for the VI and SM datasets, where the specific protocols and testing/training sets obtained from the creators of these two datasets were used.

4.2. Experiments on 2D texture descriptors

For the performance indicator, the area under the ROC curve (AUC)[1] is used since it provides a better overview of classification results. AUC is a scalar measure that can be interpreted as the probability that the classifier will assign a higher score to a randomly picked positive sample than to a randomly picked negative

Table 1
Descriptive summary of the datasets.

Dataset	#Classes	#Samples	Sample size	URL for download
PA	2	917	Various	http://labs.fme.aegean.gr/decision/downloads
VI	15	1500	41×41	http://www.cb.uu.se/~gustaf/virustexture
CH	5	327	512×382	http://ome.grc.nia.nih.gov/iicbu2008/hela/index.html#cho
SM	2	2868	100×100	http://staff.ustc.edu.cn/~yfn/vsd.html
HI	4	2828	various	http://www.informed.unal.edu.co/histologyDS
BR	2	584	various	upon request to Geraldo Braz Junior [ge.braz@gmail.com]
PR	2	349	various	upon request to Loris Nanni [nanni@dei.unipd.it]
TR	11	553	768×512	http://locate.imb.uq.edu.au/downloads.shtml
PI	13	903	various	http://imagelab.ing.unimo.it/files/bible_dataset.zip
RN	10	200	1024×1024	http://ome.grc.nia.nih.gov/iicbu2008/rnai/index.html

Table 2
AUC obtained by LPQ texture descriptors in the tested datasets (YUV and HSV are not performed for gray scale images).

LPQ Layer	Datasets									
	PA	VI	CH	HI	BR	PR	TR	RN	SM	PI
Orig	0.9037	0.9492	0.9922	0.9136	0.9569	0.8618	0.9767	0.9527	0.999	0.9166
Enh	0.8872	**0.9498**	0.9819	0.9117	0.9528	0.8737	0.9679	0.9584	0.9989	0.9205
Gaus3	0.8993	0.946	0.992	0.9125	0.9556	0.8697	**0.9881**	0.9562	0.9995	0.9414
Gaus5	**0.9105**	0.9452	**0.9943**	**0.9159**	0.9478	0.8515	0.9875	**0.961**	**0.9995**	**0.9459**
Wave1	0.8435	0.9245	0.984	0.7159	0.8786	0.8666	0.985	0.9605	0.9912	0.9265
Wave2	0.8845	0.8218	0.9652	0.7898	0.9573	0.7814	0.9664	0.8837		
Wave3	0.8297	0.8274	0.9772	0.7507	0.9457	0.8262	0.9802	0.8867		
Wave4	0.8283	0.8108	0.9693	0.7579	0.9218	0.7961	0.9612	0.866		
Grad	0.8478	0.8829	0.9902	0.8609	**0.9636**	**0.8999**	0.9713	0.9122	0.9918	0.9057
Hsv	–	–	–	–	–	–	–	–	0.9314	0.9286
Yuv	–	–	–	–	–	–	–	–	0.9316	0.9024

sample [38]. In the multiclass problem, AUC is calculated using the one-versus-all approach (where a given class is considered "positive" and all the other classes are considered "negative"). The average AUC is reported.

The first experiment, presented in Tables 2 and 3, examines the performance of the two texture descriptors, LBP (Table 1) and LPQ (Table 2), that form the base of the variants that are studied in this work. The performance of these descriptors provide a baseline for the other approaches. In order to reduce the production of data, the performance of the texture descriptors for color images is obtained by fusing the scores of the classifiers for each color band. In other words, E$_{NH}$ denotes the fusion by sum rule of an SVM trained on the enhanced Red image + an SVM trained on the enhanced Green image + an SVM trained on the enhanced Blue image. In addition, for color images, all 12 SVMs trained with the images obtained by the wavelet preprocessing method are combined by sum rule.

The second experiment, reported in Tables 4 and 5, evaluates the performance obtained by the texture descriptors LBP and LPQ as a fusion of classifiers trained on different layers. This experiment is useful for a fair comparison with descriptors that are extracted on the same subset of K layers that are used to obtain the multilayer image. In Tables 4 and 5, the average AUC is reported.

For gray-level images, the following combination of layers are considered for fusion:

KD1: Orig+ Gaus3+ Gaus5;
KD2: Orig+ Enh +Gaus3+ Gaus5;
KD3: Orig+ Wave1+ Wave2+ Wave3+ Wave4
KD4: Orig+ Enh +Gaus3+ Gaus5;
KD5: all the preprocessed images are considered.

For color images, the following combinations are considered:

KD1: Orig+ Enh +Hsv+ Yuv;
KD2: Orig+ Enh +Hsv+ Yuv +Gaus3+ Gaus5;

KD3: Orig+Gaus3+ Gaus5+ Wave;
KD4: Orig;
KD5: all the preprocessed images are considered.

From the results reported in Tables 4 and 5, it is clear that combining different preprocessing methods boosts the performance of the descriptors extracted only from the original image. KD5 provides the best performance for both the color and gray-level images.

4.3. Experiments on n-layer descriptors

The third experiment, reported in Tables 6–8, compares the three descriptors studied in this paper: VLBP, VLPQ, and LPQTop. The experiments are related to the same subsets KD1-KD5 described above. In Tables 6–8, the average AUC is reported.

From the results reported in Tables 6–8, it is clear that multilayer descriptors can be used for extracting features from nD images built using different preprocessing approaches. KD2 obtains the best performance, on average, among the three descriptors.

4.4. Experiments on the fusion between texture and multilayer descriptors

The forth experiment, described in Table 9, evaluates the performance of a weighed fusion among a base texture descriptor and its nD version. Fusions have been performed considering the best combination of layers, both for the texture and the multilayer descriptors, which are KD5 and KD2, respectively (see the experiments reported in Sections 4.2 and 4.3). The same configuration has been maintained for all the datasets, and the average AUC is reported.

For a sake of easy comparison, some results obtained from the first and second experiments are again reported here: LPQ(O$_{RIG}$), LBP(O$_{RIG}$), LPQ(KD5), LBP(KD5) and LPQ(B$_{EST}$) and LBP(B$_{EST}$) which denote the best result among different preprocessing.

Table 3
AUC obtained by LBP texture descriptors in the tested datasets (YUV and HSV are not performed for gray scale images).

LBP	Datasets									
Layer	PA	VI	CH	HI	BR	PR	TR	RN	SM	PI
O$_{RIG}$	**0.9006**	**0.9205**	0.9943	0.8906	0.9365	0.8101	0.985	0.9473	0.9978	0.9331
E$_{NH}$	0.8991	0.9188	0.9905	0.8874	**0.9298**	0.8132	0.9812	0.933	0.9982	0.9355
G$_{AUS}$3	0.854	0.8902	0.9962	**0.8888**	0.9491	0.8019	0.9893	0.9488	0.9989	0.9394
G$_{AUS}$5	0.8418	0.886	**0.9974**	0.8872	0.932	0.7935	0.9877	0.9557	**0.999**	**0.9404**
W$_{AVE}$1	0.8139	0.8943	0.9893	0.749	0.8586	0.8091	0.9759	**0.9579**	0.9899	0.9202
W$_{AVE}$2	0.8518	0.8012	0.9543	0.7529	0.9384	0.7691	0.9764	0.8781		
W$_{AVE}$3	0.8134	0.7941	0.9737	0.7233	0.937	0.7831	0.9637	0.9121		
W$_{AVE}$4	0.7891	0.7959	0.9666	0.7202	0.9241	0.755	0.9534	0.8896		
G$_{RAD}$	0.8429	0.8622	0.9932	0.838	0.941	0.8111	0.9822	0.9558	0.9901	0.8873
H$_{SV}$	–	–	–	–	–	–	–	–	0.9245	0.9221
Y$_{UV}$	–	–	–	–	–	–	–	–	0.9389	0.8475

Table 4
AUC obtained by the fusion on several LPQ texture descriptors in the tested datasets.

LPQ	Datasets										AVG
Layer	PA	VI	CH	HI	BR	PR	TR	RN	SM	PI	
KD1	**0.9181**	0.9601	**0.995**	0.9293	0.9597	0.8907	**0.9909**	0.9637	0.9954	0.9506	0.9553
KD2	0.9152	0.9619	**0.995**	**0.9302**	0.9617	0.899	0.9899	**0.9671**	**0.9983**	0.9571	0.9575
KD3	0.9003	0.9647	0.9932	0.8826	0.9627	0.8882	0.9886	0.9359	0.9982	0.9486	0.9463
KD4	0.9152	0.9619	**0.995**	**0.9302**	0.9617	0.899	0.9899	**0.9671**	0.9978	0.9333	0.9551
KD5	0.9149	**0.9704**	0.9949	0.9256	**0.9684**	**0.9101**	0.9903	0.9598	0.9981	**0.9565**	**0.9589**

Also reported in Table 9 are the performance of two recent state-of-the-art descriptors:

- Rotation invariant co-occurrence local binary patterns (RICLBP) [41].
- Completed local binary patterns (CLBP) [40].

To statistically validate our experiments and evaluate which algorithm is better over different data sets, we can use the Wilcoxon signed rank test [39], a nonparametric test that compares the performance of classifiers by considering the number of wins and losses and the difference in performance for wins and losses on different datasets [42]. We select the Wilcoxon signed rank test because it is one of the most used comparison measures in the literature. Demšar in [39] claims that the Wilcoxon signed rank test is one of the best statistical methods for comparing distributions.

The Wilcoxon signed rank test confirms (p-value 0.1) that:

- VLBP+5 × LBP outperforms LBP(KD5);
- LPQTOP+LPQ outperforms LPQ(KD5);
- VLPQ+LPQ outperforms LPQ(KD5);
- LPQ (KD5) outperforms LPQ(ORIG);
- LBP (KD5) outperforms LBP(ORIG);
- LPQ (KD5) outperforms LPQ(BEST);
- LBP (KD5) outperforms LBP(BEST);

Even though the improvements shown in Table 9 sometimes appear modest, Wilcoxon signed rank test confirms the statistical superiority of some of the proposed approaches. Of particular interest are the following:

- Our new approach LPQ(KD5), which outperforms the 2D texture descriptors LPQ(ORIG) and LPQ(BEST);
- LBP(KD5), which likewise outperforms the 2D texture descriptors LPQ(ORIG) and LPQ(BEST);
- The combination of volumetric and standard descriptor LPQTOP+LPQ that we suggest in this paper as the approach with the best balance between computational cost and performance; it outperforms LPQ(KD5);
- VLPQ, LPQTOP, and LPQ, each of which obtains a statistically similar performance (p-value 0.1); however, their fusion outperforms all the three stand-alone approaches

Although the performance of VLBP is not impressive, LPQTOP+LPQ works well. It is important to note that our results have been obtained without any ad hoc parameter optimization for each dataset. It should also be noted that this is the first attempt to combine sets of preprocessed images to obtain a multilayer image: this idea can be exploited in the future to rank preprocessing approaches and study their correlation. In many works the aim of the research is simply to find the best preprocessing method to improve the performance, here we are interested in finding the best ensemble.

Table 5
AUC obtained by the fusion on several LBP texture descriptors in the tested datasets.

LBP	Datasets										AVG
Layer	PA	VI	CH	HI	BR	PR	TR	RN	SM	PI	
KD1	0.8898	0.933	0.9972	0.9163	0.9546	0.8477	0.9922	0.9579	0.9954	0.9506	0.9434
KD2	0.8992	0.9376	0.9975	0.917	0.9542	0.8546	0.9917	0.9574	**0.9983**	**0.9571**	0.9464
KD3	0.8888	0.9453	0.9949	0.8877	0.9634	0.8619	0.9918	0.9571	0.9982	0.9486	0.9437
KD4	0.8993	0.9376	0.9975	0.917	0.9542	0.8546	0.9918	0.9574	0.9978	0.9331	0.944
KD5	**0.9007**	**0.958**	**0.9981**	**0.9229**	**0.9669**	**0.8802**	**0.9942**	**0.9676**	0.9981	0.9565	**0.9543**

Table 6
AUC obtained by VLPQ descriptor in the tested datasets.

VLPQ	Datasets										AVG
Layer	PA	VI	CH	HI	BR	PR	TR	RN	SM	PI	
KD1	0.8847	0.9335	0.9874	**0.9154**	**0.9575**	0.8744	0.9836	0.9415	0.9734	0.9154	0.9366
KD2	0.9202	**0.9455**	**0.9968**	0.9146	0.9564	**0.8939**	0.9853	0.9504	0.9771	0.9117	**0.9451**
KD3	0.888	0.9092	0.9774	0.875	0.9627	0.8636	0.973	**0.9767**	0.9803	**0.9206**	0.9326
KD4	0.9202	**0.9455**	**0.9968**	0.9146	0.9564	0.8938	**0.9853**	0.9504	0.9651	0.8956	0.9423
KD5	**0.9029**	0.944	0.9852	0.921	0.959	0.887	0.9768	0.9685	**0.9842**	0.9185	0.9447

Table 7
AUC obtained by LBPTop descriptor in the tested datasets.

LPQTop	Datasets										AVG
Layer	PA	VI	CH	HI	BR	PR	TR	RN	SM	PI	
KD1	0.9111	0.9588	**0.9972**	**0.9288**	0.9613	**0.8986**	0.9847	0.9563	0.9925	0.9252	0.9514
KD2	0.9289	0.9632	0.9963	0.9189	0.9409	0.8949	**0.9863**	0.9492	0.9952	**0.9302**	0.9504
KD3	0.9068	0.9529	0.992	0.9078	**0.9664**	0.8833	0.968	**0.9752**	0.9929	0.9297	0.9475
KD4	0.9289	0.9632	0.9963	0.9189	0.9409	0.8949	**0.9863**	0.9492	0.9927	0.9289	0.95
KD5	**0.9327**	**0.9672**	0.9943	**0.9288**	0.9488	0.8965	0.9732	0.97	**0.9963**	0.9298	**0.9537**

Table 8
AUC obtained by VLBP descriptor in the tested datasets.

VLBP	Datasets										AVG
Layer	PA	VI	CH	HI	BR	PR	TR	RN	SM	PI	
KD1	**0.866**	0.8639	0.9609	0.7578	0.9053	0.816	**0.9777**	0.9243	0.9679	0.9116	0.8951
KD2	0.8388	**0.8735**	**0.9926**	**0.8147**	0.9052	0.8323	0.9713	**0.9333**	**0.9704**	**0.9148**	**0.9046**
KD3	0.752	0.7142	0.9314	0.6352	0.9344	0.8064	0.9746	0.8911	0.966	0.8875	0.8492
KD4	0.8388	**0.8735**	**0.9926**	**0.8147**	0.9052	0.8323	0.9713	**0.9333**	0.8949	0.8707	0.8927
KD5	0.8045	0.7919	0.9886	0.752	**0.9422**	**0.8377**	0.8744	0.932	0.9618	0.9059	0.8791

Table 9
AUC obtained by several weighed fusion methods in the tested datasets.

AUC	Datasets										AVG
Method	PA	VI	CH	HI	BR	PR	TR	RN	SM	PI	
CLBP	0.9256	0.9482	**0.999**	0.9183	0.9582	0.8669	0.9851	0.9501	0.9986	0.8852	0.9436
RICLBP	0.9182	0.9765	0.9923	0.9285	0.9263	0.8864	0.9872	0.9662	0.9981	0.9018	0.9482
LPQ (ORIG)	0.9037	0.9492	0.9922	0.9136	0.9569	0.8618	0.9767	0.9527	0.999	0.9166	0.9422
LBP (ORIG)	0.9006	0.9205	0.9943	0.8906	0.9365	0.8101	0.985	0.9473	0.9978	0.9331	0.9316
LPQ (BEST)	0.9105	0.9498	0.9943	0.9159	0.9636	0.8999	0.9881	0.961	**0.9995**	0.9459	0.9528
LBP (BEST)	0.9006	0.9205	0.9974	0.8906	0.9491	0.8132	0.9893	0.9579	0.999	0.9404	0.9358
LPQ (KD5)	0.9149	0.9704	0.9949	0.9256	0.9684	0.9101	0.9903	0.9598	0.9981	0.9565	0.9589
LBP (KD5)	0.9007	0.958	0.9981	0.9229	0.9669	0.8802	0.9942	0.9676	0.9981	0.9565	0.9543
VLPQ+LPQ	0.9221	0.9679	0.9961	0.9326	0.9671	**0.9193**	0.9891	0.9659	0.9936	0.9578	0.9611
VLPQ+5 × LPQ	0.9185	0.9717	0.9955	0.9308	**0.9685**	0.9143	0.9903	0.9632	0.9978	0.9594	0.961
LPQTOP+LPQ	**0.9372**	**0.9751**	0.9965	**0.9355**	0.9634	0.9129	0.9907	0.9647	0.9979	0.9602	**0.9634**
LPQTOP+5 × LPQ	0.9253	0.9745	0.9959	0.9325	0.9677	0.9146	0.9908	0.9636	0.9985	0.9606	0.9624
VLBP+LBP	0.8992	0.9556	0.9984	0.9137	0.9559	0.884	0.9923	0.9643	0.992	0.9593	0.9514
VLBP+5 × LBP	0.9008	0.9608	0.9982	0.925	0.9657	0.8878	**0.9943**	**0.9706**	0.9976	**0.9607**	0.9561

Table 10
Execution times in seconds on a CPU i5-3470 3.2 GHz – 8 GB Ram PC.

Step		Time
Pre-processing	ENH	0.0019
	GAUS3	0.0009
	GAUS5	0.0010
	WAVE	0.0021
	GRAD	0.0024
Feature extraction	LPQ (for each layer)	0.0062
	LBP (for each layer)	0.0059
	VLPQ	0.0557
	LPQTOP	0.1725
	VLBP	2.4975

Table 11
Descriptor size.

Descriptor	Size
LPQ (for each layer)	512
LBP (for each layer)	302
VLPQ	2048
LPQTOP	1536
VLBP	16,384

4.5. Computational cost and execution times

As far as computational effort is concerned, we report in this section the execution time required for the main steps of the

method: image preprocessing and feature extraction. The results reported in Table 10 have been obtained considering a sample gray scale image of 100×100 pixels.

The reported execution time (in seconds) has been obtained without parallelizing the MATLAB code on a PC (CPU i5-3470 3.2 GHz – 8 GB Ram). Almost all the approaches require computational times that are acceptable for most of applications; the only exception is VLBP, which is very computationally expensive. Using a multicore computer can offset some of the computational time considerations as the different preprocessing and feature extraction methods in our proposed approach can be parallelized.

Finally, in Table 11 the length of each descriptor is reported: the size of the texture descriptors is related to a single-layer image, while for nD descriptors the size is invariant with respect to the number of layers.

5. Conclusions

The aim of this work is to show the feasibility of building an n-layer image using different preprocessing approaches. These images are then described using multilayer descriptors, which form feature vectors that are used to train a set of SVMs. For assessing the performance of this approach, we run tests on several datasets, on both color and gray-level images, using both LPQ and LBP texture descriptors, and three multilayer descriptors based on LPQ and LBP: viz. VLBQ, VLBP, and VLB-TOP. Our results show that n-layer and single-layer descriptors can be combined for outperforming the standard single-layer approaches. This conclusion is confirmed using the Wilcoxon signed rank test with a p-value 0.1.

Since the performance of n-layer descriptors is strictly related to the order of the considered layers, we plan in the future to study how this order influence performance with the aim of designing a

---

[1] AUC is implemented as in dd_tools 0.95 davidt@ph.tn.tudelft.nl.

selection/ranking approach for selecting/ranking the layers most useful for classification purposes. Another feasible way to improve classification accuracy is to cluster layers into several groups, extract descriptors from each group, and then exploit fusions to improve classification performance.

Conflict of interest statement

None declared.

References

[1] R. Datta, et al., Image retrieval: ideas, influences, and trends of the new age, ACM Comput. Surv. 40 (2) (2008) 1–60.
[2] M.J. Swain, D.H. Ballard, Indexing via color histograms, in: Proceedings of the IEEE International Conference on Computer Vision, Osaka, Japan, 1990, pp. 390–393.
[3] J. Huang, S.R. Kumar, M. Mitra, Combining supervised learning with color correlograms for content based image retrieval, in: Proceedings of the 5th ACM Multimedia Conference, 1997, pp. 325–334.
[4] K. Siddiqi, S.M. Pizer, Medial Representations: Mathematics, Algorithms and Applications, Springer, Berlin, 2008.
[5] P.J. Flynn, A. Jain, CAD-based computer vision: from CAD models to relational graphs, IEEE Trans. Pattern Anal. Mach. Intell. 13 (1991) 114–132.
[6] S. Maji, A.C. Berg, J. Malik, Classification using intersection kernel support vector machines is efficient, in: Proceedings of the 2008 Conference on Computer Vision and Pattern Recognition (CVPR), Anchorage, Alaska, 2008, pp. 1–8.
[7] J. Shotton, A. Blake, R. Cipolla, Multiscale categorical object recognition using contour fragments, IEEE Trans. Pattern Anal. Mach. Intell. 30 (2008) 1270–1281.
[8] L. Ding, M. Belkin, Component based shape retrieval using differential profiles, in: Proceedings of the First ACM International Conference on Multimedia Information Retrieval, Vancouver, Canada, 2008, pp. 216–222.
[9] M. Varma, A. Zisserman, A statistical approach to texture classification from single images, Int. J. Comput. Vis. 62 (2005) 61–81.
[10] M. Tuceryan, A.K. Jain, Texture analysis, in: C.H. Chen, L.F. Pau, P.S.P. Wang (Eds.), The Handbook of Pattern Recognition and Computer Vision, World Scientific Publishing Co., River Edge, NJ, USA, 1998, pp. 207–248.
[11] R.C. Gonzalez, R.E. Woods, Digital Image Processing, 2nd edition, Addison-Wesley Longman Publishing Co., Inc., Boston, 2001.
[12] R.M. Haralick, K. Shanmugam, I. Dinstein, Textural features for image classification, IEEE Trans. Syst. Man Cybern. 3 (6) (1973) 610–621.
[13] B. Manjunath, W. Ma, Texture features for browsing and retrieval of image data, IEEE Trans. Pattern Anal. Mach. Intell. 18 (8) (1996) 837–842.
[14] T. Ojala, M. Pietikainen, D. Harwood, Performance evaluation of texture measures with classification based on kullback discrimination of distributions, in: Proceedings of the 12th IAPR International Conference, 1994.
[15] T. Ojala, M. Pietikäinen, D. Harwood, A comparative study of texture measures with classification based on featured distribution, Pattern Recognit. Lett. 29 (1) (1996) 51–59.
[16] 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.
[17] Q. Iqbal, J.K. Aggarwal, Feature integration, multi-image queries and relevance feedback in image retrieval, in: Proceedings of the 6th International Conference on Visual Information Systems (VISUAL), Miami, Florida, 2003, pp. 467–474.
[18] S. Murala, Q.M.J. Wu, Spherical symmetric 3D local ternary patterns for natural, texture and biomedical image indexing and retrieval, Neurocomputing 149 (Part C) (2015) 1502–1514.
[19] V. Vapnik, A. Lerner, Pattern recognition using generalized portrait method, Autom. Remote Control 24 (1963) 774–780.
[20] G. Kylberg, M. Uppström, I.-M. Sintorn, Virus texture analysis using local binary patterns and radial density profiles, in: S. Martin, S.-W. Kim (Eds.), Proceedings of the 18th Iberoamerican Congress on Pattern Recognition (CIARP), 2011, pp. 573–580.
[21] V. Ojansivu, J. Heikkila Blur, Insensitive texture classification using local phase quantization, in: Proceedings of the International Conference on Image and Signal Processing (ICISP), 2008, pp. 236–243.
[22] 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.
[23] J. Päivärinta, E. Rahtu, J. Heikkilä, Volume local phase quantization for blur-insensitive dynamic texture classification, in: Anders Heyden, Fredrik Kahl (Eds.), Image Analysis, Springer Berlin, Heidelberg, 2011, pp. 360–369.
[24] B. Jiang, M.F. Valstar, M. Pantic, Action unit detection using sparse appearance descriptors in space-time video volumes, in: Proceedings of the 2011 IEEE International Conference on Automatic Face and Gesture Recognition and Workshops (FG 2011), 2011.
[25] S. Mallat, A theory for multiresolution signal decomposition, IEEE Trans. Pattern Anal. Mach. Intell. 11 (1989) 674–693.
[26] I. Daubechies, Orthonormal bases of compactly supported wavelets, Commun. Pure Appl. Math. 41 (1988) 909–996.
[27] K. Singh, R. Kapoor, Image enhancement using exposure based sub image histogram equalization, Pattern Recognit. Lett. 36 (2014) 10–14.
[28] C. Lee, C. Lee, C.-S. Kim, Contrast enahancement based on layered difference representation of 2D histograms, IEEE Trans. Image Process. 22 (12) (2013) 5372–5384.
[29] J. Jantzen, et al., Pap-smear benchmark data for pattern classification, in: Karl Lieven (Ed.), Nature Inspired Smart Information Systems (NiSIS), Albufeira, Portugal, 2005, pp. 1–9.
[30] A. Cruz-Roa, J.C. Caicedo, F.A. González, Visual pattern mining in histology image collections using bag of features, Artif. Intell. Med. (52) (2011) 91–106.
[31] M.V. Boland, M.K. Markey, R.F. Murphy, Automated recognition of patterns characteristic of subcellular structures in fluorescence microscopy images, Cytopathology 33 (3) (1998) 366–375.
[32] N. Hamilton, et al., Fast automated cell phenotype classification, BMC Bioinform. (2007) 8–110.
[33] J.L. Fink, et al., LOCATE: a mouse protein subcellular localization database, Nucleic Acids Res. 34 (2006) 213–217.
[34] L. Nanni, et al., Protein classification using texture descriptors extracted from the protein backbone image, J. Theor. Biol. 3 (7) (2010) 1024–1032.
[35] G.B. Junior, et al., Classification of breast tissues using Moran's index and Geary's coefficient as texture signatures and SVM, Comput. Biol. Med. 39 (12) (2009) 1063–1072.
[36] F. Yuan, Video-based smoke detection with histogram sequence of LBP and LBPV pyramids, Fire Saf. J. 46 (3) (2011) 132–139.
[37] D. Borghesani, C. Grana, R. Cucchiara, Miniature illustrations retrieval and innovative interaction for digital illuminated manuscripts in multimedia systems, Multimed. Syst. 20 (2014) 65–79.
[38] T. Fawcett, ROC Graphs: Notes and Practical Considerations for Researchers, HP Laboratories, Palo Alto, USA, 2004.
[39] J. Demšar, Statistical comparisons of classifiers over multiple data sets, J. Mach. Learn. Res. 7 (2006) 1–30.
[40] Z. Guo, L. Zhang, D. Zhang, A completed modeling of local binary pattern operator for texture classification, IEEE TIP 19 (6) (2010) 1657–1663.
[41] R. Nosaka, K. Fukui, HEp-2 cell classification using rotation invariant co-occurrence among local binary patterns, Pattern Recognit. Bioinform. 47 (7) (2014) 2428–2436.
[42] O. Irsoy, O.T. Yildiz, E. Alpaydin, Design and analysis of classifier learning experiments in bioinformatics: survey and case studies, IEEE/ACM Trans. Comput. Biol. Bioinform. 9 (6) (2012) 1663–1675.