Local Binary Patterns as Texture Descriptors for User Attitude Recognition

Mohamed Eisa, A. ElGamal, R. Ghoneim and A. Bahey

Department of computer science
Mansoura University, 35516
Egypt

Abstract
Texture plays an important role in numerous computer vision applications. Many methods for describing and analyzing of textured surfaces have been proposed. Variations in the appearance of texture caused by changing illumination and imaging conditions, for example, set high requirements on different analysis methods. In addition, real-world applications tend to produce a great deal of complex texture data to be processed that should be handled effectively in order to be exploited.
A local binary pattern (LBP) operator offers an efficient way of analyzing textures. It has a simple theory and combines properties of structural and statistical texture analysis methods. LBP is invariant against monotonic gray-scale variations and has also extensions to rotation invariant texture analysis. Analysis of real-world texture data is typically very laborious and time consuming. Often there is no ground truth or other prior knowledge of the data available, and important properties of the textures must be learned from the images. This is a very challenging task in texture analysis.
Keywords:
Texture descriptors, image analysis, local binary pattern, image matching, content based image retrieval, user attitude

1. Introduction

Texture analysis plays an important role in computer vision and image processing. Numerous algorithms of textural features extraction have been presented during the past decades [1], which can be divided mainly into statistical approaches and structural approaches. The former, including co-occurrence matrix, wavelet transform, and Gabor filter, are the most commonly used in practice. More recently, the local-binary-pattern (LBP) operator [2], [3], [4] has received considerable attention and has been used in applications such as face recognition, industrial visual inspection, segmentation of remote-sensing images, and classification of real outdoor images. LBP operator is a statistical texture descriptor in terms of the characteristics of the local structure. LBP provides a unified description including both statistical and structural characteristics of a texture patch, so that it is more powerful for texture analysis.

Local image feature detection and description have received a lot of attention in recent years. The basic idea is to first detect interest regions that are covariant to a class of transformations [5]. Then, for each detected region, an invariant descriptor is built. Once we have the descriptors computed, we can match interest regions between images. This approach has many advantages. For example, local features can be made very tolerant to illumination changes, perspective distortions, image blur, image zoom, and so on. The approach is also very robust to occlusion. Local features have performed very well in many computer vision applications, such as image retrieval, wide baseline matching, object recognition, texture recognition, and robot localization [6].
Many existing texture operators have not been used for describing interest regions so far. One reason might be that, by using these methods, usually a large number of dimensions is required to build are liable descriptor. The local binary pattern (LBP) texture operator has been highly successful for various computer vision problems such as face recognition, background subtraction, and recognition of 3D textured surfaces, but it has not been used for describing interest regions so far. The LBP has properties that favor its usage in interest region description such as tolerance against illumination changes and computational simplicity. Drawbacks are that the operator produces a rather long histogram and is not too robust on flat image areas. To address these problems, a LBP-based texture feature, denoted as center-symmetric local binary pattern (CS-LBP) that is more suitable for the given problem is proposed [7].
Since the SIFT and other distribution-based descriptors similar to it have shown state-of-the-art performance in different problems, we decided to focus on this approach. We were especially interested to see if the gradient orientation and magnitude-based feature used in the SIFT algorithm could be replaced by a different feature that offers better or comparable performance. In this paper, an interest region descriptor, denoted as CS-LBP descriptor, that combines the good properties of the SIFT and LBP is introduced. This is achieved by adopting the SIFT descriptor and using the novel CS-LBP feature instead of

Manuscript received June 5, 2010
Manuscript revised June 20, 2010

original gradient feature. The new feature allows simplifications of several steps of the algorithm which makes the resulting descriptor computationally simpler than SIFT. It also appears to be more robust to illumination changes than the SIFT descriptor [8].

2. Texture feature extraction

There are number of ways to carry out the actual feature extraction, and typically methods are divided into categories such as statistical, geometrical, model based, and signal processing [9]. Different features work better with a certain type of textured images than with others, and often an empirical evaluation is required to find the most effective features.

2.1 Statistical texture features

A simple texture measure is to construct a histogram of gray-level pixel values and use it as a feature vector. Hadjidemetriou et al. [10] filtered the original image with several Gaussian filters, and constructed multiresolution histogram features, achieving reasonably good accuracy with low computing costs. Co-occurrence matrices give information about patterning of the texture, and it is possible to calculate textural properties from them. These features are sensitive to illumination variations, but have been very popular in different texture analysis applications. Another simple texture statistic is to calculate the autocorrelation function of the image. This can be used to assess the amount of regularity, as well as the coarseness of the texture present in the image [9]. Yet another example for detecting texture properties statistically is the use of gray-level run length features to detect texture properties [11]. They are easy to extract, but their performance has been found to be quite poor.

2.2 Structural and geometrical texture features

Structural and geometrical features are usually more stable in overall illumination changes than statistical features, but rely strongly on primitive detection. After primitives are found, one can calculate the statistical properties of the primitives, construct placement rules for them, or otherwise utilize textons found. Often structural based methods are justified using psychophysical studies of the texture perception: a human could discrimate textures with different texton elements. [12] used basic mathematical morphology operations to detect texture primitives. Watershed segmentation based on mathematical morphology is also applied for detecting textons from images. Another structural based method given by [12] filters the image with Laplacian of Gaussian (LoG) masks at different scales and tries to extract blobs in the image.

2.3 Signal-processing based texture features

Signal processing based texture analysis approaches usually filter the image with specific filters and utilize the filter responses to create texture features. In textured surfaces, different frequencies have their own textural properties. Both spatial and frequency domain approaches can be used for filtering images and capturing relevant information. In spatial analysis images are usually convolved via certain masks. In frequency domain analysis, the Fourier transform is applied to the image, and features are calculated. Power spectrum features have been applied, for example, to describe the roughness of texture. It is very popular to process a texture image with Gabor filtering. Gabor wavelet features proposed by [13] are among the most popular texture measures at the moment.

2.4 Local texture features and texton statistics

Detection of local spatial patterning of the image can be used to create texture features. [14] proposed method based on center-symmetric local 3×3 neighborhoods and calculated auto-correlation, covariance, and variance measures from these local regions. [4] proposed signed gray-level differences, which are based on the absolute gray-level difference method, but consider also the signs of the differences.

[15] proposed a method based on relational kernel functions (RKF) to create texture features using circular local neighborhoods. The use of texton statistics in texture analysis has became very popular in viewpoint based recognition. The idea is to build a universal texton library and represent textures using, for example, the frequency histogram of different textons.

[9] also used filtering based approach for building a texton library. In their approach, the image was filtered with filters at multiple orientations, but only the orientation with a maximum response was considered. They also used k-means clustering for building the texton library. Instead of using filter banks, [9] constructed the texton library using pure pixel values of local image patches and achieved very good classification accuracy.

2.5 Invariance in texture features

Variations in illumination, scale, and rotation cause changes in texture appearance, and several proposals for invariant texture measures have been given. Gray-scale changes caused by illumination variations are often handled with global normalization of the input image. [10] used histogram equalization to normalize images before feature extraction.

Recently, local methods have become very popular in computer vision. The idea is to first detect from the images the interest regions that are covariant to a class of transformations, such as rotation or scale (detected regions are adapted to a viewpoint so that they correspond the same region in the original image). After detection, a descriptor is calculated for each region. There are several ways of detecting the interest regions from images. [14] made an extensive comparison of different methods. A large number of different local descriptors have also been proposed. A recent comparative study [15] indicated that SIFT based [16] descriptors perform best. Local approaches have been widely applied in different computer vision applications, such as object recognition.

3. General Frameworks

Because images are represented as vectors in CBIR system, classification can be an effective factor for improving the retrieval effectiveness. Images that are similar to each other are grouped together into one class. The advantage of grouping is that it brings very similar images together. Although the automated clustering is helpful, the refinement of a human expert to the resulting clustering groups for relevance is necessary specially, in the human attitude reporting system. The proposed user attitude reporting framework presented in this paper follows the following algorithm.

<u>Offline Phase</u>

For special user in an organization (such as student in a classroom)

Obtain set of digital images and construct a digital image database
Apply Invariant feature to obtain the intermediate images
Extract the Local Binary Pattern vectors
Get the LBPs database

For every user attitude case
Enter well defined query image
Construct intermediate query image
Extract the Local Binary Pattern feature vector for the query Image
Check the similarity of the query among the feature database

IF the query image is within specified relevance measure
THEN
{**Make** grouping class
Index the images in the class
Assure the relevance by a human expert according to facial expression}
End IF

Next user attitude
Next user

<u>Online Phase</u>

For specific user
Observe specific user by means of video camera
Make segmentation of the images
Apply Invariant feature to get intermediate images
Extract the Local Binary Pattern feature vector for the observed image
Compare the query image LBP with the LBPs database
Find the maximum similarity ratio and determine its code
Decode the determined code
Report the user attitude
End

The user attitude cases (actions) considered for recognition may be:

- displeasure, or sadness.
- anger, or fear
- anxious feelings, nervousness

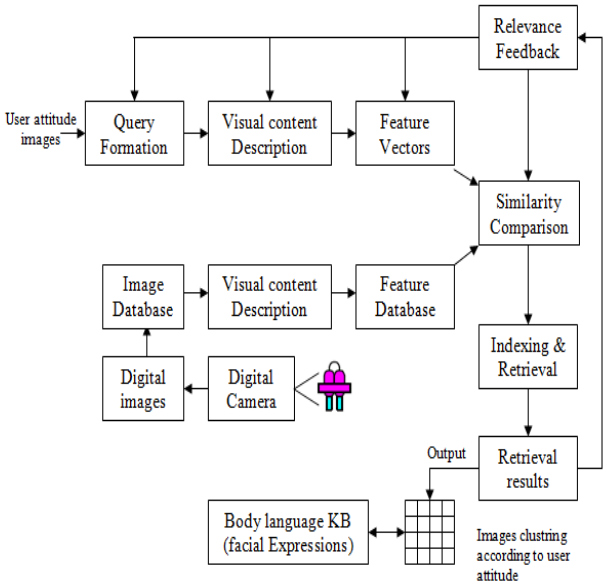

Fig. 1. Offline phase: Image classification and LBP modification

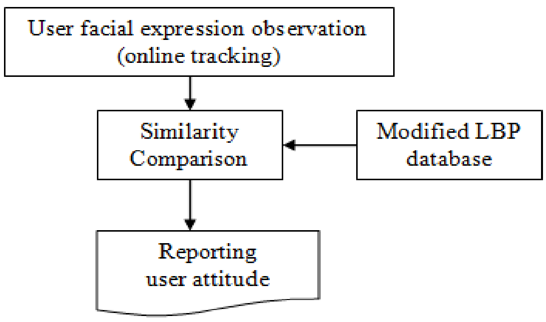

Fig. 2. Online phase: User tracking, LBP comparing and user attitude reporting

4. SIFT and LBP methods

Before presenting in detail the CS-LBP operator and the CS-LBP descriptor, we give a brief review of the SIFT and LBP methods that form the basis for our work.

4.1. SIFT descriptor

The SIFT descriptor is a 3D histogram of gradient locations and orientations. Location is quantized into a 4 × 4 location grid and the gradient angle is quantized into eight orientations, resulting in a 128-dimensional descriptor. First, the gradient magnitudes and orientations are computed within the interest region. The gradient magnitudes are then weighted with a Gaussian window overlaid over the region. To avoid boundary effects in the presence of small shifts of the interest region, a trilinear interpolation is used to distribute the value of each gradient sample into adjacent histogram bins. The final descriptor is obtained by concatenating the orientation histograms over all bins. To reduce the effects of illumination change the descriptor is first normalized to unit length. Then, the influence of large gradient magnitudes is reduced by thresholding the descriptor entries, and renormalizing to unit length.

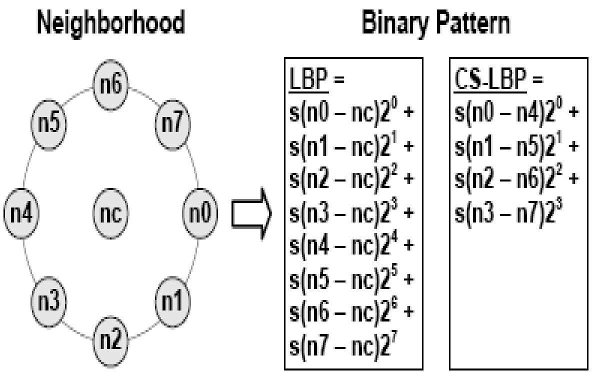

Fig. 3. LBP and CS-LBP features for a neighborhood of 8 pixels.

4.2. LBP operator

The local binary pattern (LBP) is a powerful illumination invariant texture primitive. The histogram of the binary patterns computed over a region is used for texture description. The operator describes each pixel by the relative gray-levels of its neighboring pixels; see Fig. 3 for an illustration with 8 neighbors. If the gray-level of the neighboring pixel is higher or equal, the value is set to one, otherwise to zero. The descriptor describes the result over the neighborhood as a binary number (binary pattern):

$$\mathrm{LBP}_{R,N}(x,y)=\sum_{i=0}^{N-1} s(n_i-n_c)2^i,\quad s(x)=\begin{cases}1, & x\geq 0,\\ 0 & \text{otherwise}\end{cases} \qquad (1)$$

where n_c corresponds to the gray-level of the center pixel of a local neighborhood and n_i to the gray-levels of N equally spaced pixels on a circle of radius R. Since correlation between pixels decreases with distance, a lot of the texture information can be obtained from local neighborhoods. Thus, the radius R is usually kept small. In practice, Eq. (1) means that the signs of the differences in a neighborhood are interpreted as an N-bit binary number, resulting in 2^N distinct values for the binary pattern. From above, it's easy to figure out that the LBP has several properties that favor its usage in interest region description. The features have proven to be robust against illumination changes, they are very fast to compute, and do not require many parameters to be set [17].

5. Center-symmetric local binary patterns

The LBP operator produces rather long histograms and is therefore difficult to use in the context of a region descriptor. To address the problem we modified the scheme of how to compare the pixels in the neighborhood. Instead of comparing each pixel with the center pixel, we compare center-symmetric pairs of pixels as illustrated in Fig. 3. This halves the number of comparisons for the same number of neighbors. We can see that for eight neighbors, LBP produces 256 (28) different binary patterns, whereas for CS-LBP this number is only 16 (24). Furthermore, robustness on flat image regions is obtained by thresholding the gray-level differences with a small value T as proposed in [18]:

$$CS-LBP_{R,N,T}(x,y)=\sum_{i=0}^{(N/2)-1} s\left(n_i-n_{i+(N/2)}\right)2^i, \qquad (2)$$

where n_i and $n_{i+(N/2)}$ correspond to the gray values of center- symmetric pairs of pixels of N equally spaced pixels on a circle of radius R. It should be noticed that the CS-LBP is closely related to gradient operator, because like some gradient operators it considers gray-level differences between pairs of opposite pixels in a neighborhood. Since the focus of this paper is in the region description we do not present any operator level comparison between the LBP and CS-LBP. Instead, the comparison is done in the context of the region descriptor in Section 7. In addition to our work, there also exists other ways to reduce the histogram size of the LBP. Maybe the best known method is to use uniform patterns proposed in [17]. For N neighbors we have

$N(N-1)+2$ different uniform patterns. This makes 58 patterns for eight neighbors.

6. CS-LBP descriptor

In the following, we present CS-LBP descriptor in detail. The input for the descriptor is a normalized interest region. The process is depicted in Fig. 4. The region detection and normalization steps are described in experimental results. In our experiments, the region size after normalization is fixed to 41×41 pixels and the pixel values lie between 0 and 1.

6.1. Feature extraction with CS-LBPs

We extract a feature for each pixel of the input region by using the CS-LBP operator. The operator has three parameters: radius R, number of neighboring pixels N, and threshold on the gray level difference T. Our experiments have shown that good values for these parameters are in general {1, 2} for R, {6, 8} for N, values for these parameters are in general {1, 2} for R, {6, 8} for N, and {0,...,0.02} for T.

6.2. Feature weighting

A weight is associated with each pixel of the input region based on the used feature. A comparison of three different weighting strategies, namely uniform, Gaussian-weighted gradient magnitude (SIFT), and Gaussian, showed that simple uniform weighting is the most suitable choice for the CS-LBP feature.

6.3. Descriptor construction

In order to incorporate spatial information into the descriptor, the input region is divided into cells with a location grid. We tried two different grids, namely Cartesian and log-polar, and found that the Cartesian one gives better performance. In the experiments presented in this paper, we use either a 3×3 (9 cells) or 4×4 (16 cells) Cartesian grid. For each cell a CS-LBP histogram is built. Thus, the resulting descriptor is a 3D histogram of CS-LBP feature locations and values. As explained earlier, the number of different feature values $(2^{N/2})$ depends on the neighborhood size (N) of the chosen CS-LBP operator. In order to avoid boundary effects in which the descriptor abruptly changes as a feature shifts from one cell to another, bilinear interpolation over x and y dimensions is used to share the weight of the feature between four nearest cells. The share for a cell is determined by the bilinear interpolation weights.

6.4. Descriptor normalization

The final descriptor is built by concatenating the feature histograms computed for the cells to form a $M \text{ X } M \text{ X } 2^{N/2}$-dimensional vector, where the M and N are the grid size and CS-LBP neighborhood size, respectively. For $(M=3, N=6)$, $(M=3, N=8)$, $(M=4, N=6)$, and $(M=4, N=8)$ the lengths of the CS-LBP descriptors are 72, 144, 128, and 256, respectively.

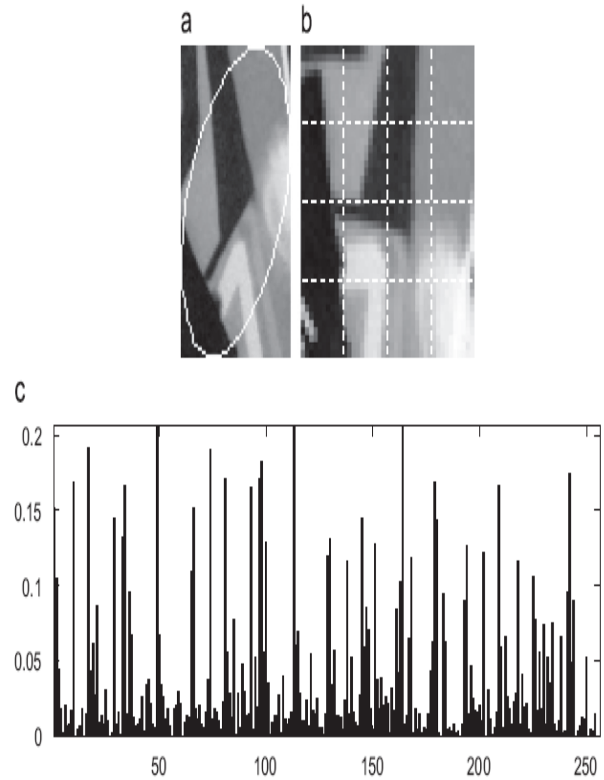

Fig. 4. The CS-LBP descriptor (a) An elliptical image region detected by Hessian-Affine detector. (b) The region with Cartesian location grid after affine normalization. (c) The resulting CS-LBP descriptor computed for the normalized region.

The descriptor is then normalized to unit length. The influence of very large descriptor elements is reduced by thresholding each element to be no larger than a threshold. This means that the distribution of features has greater emphasis than individual large values.

7. Experimental Results

We have conducted classification and retrieval tests on indoor images. The real world images consist of a set of nearly 1200 photograph images that are generated by using a digital video camera.

The nonlinear monomial kernel function, is used to extract the color features from the three channels of RGB color space and the resulting image is called the intermediate image. The color space of the intermediate image is discretized, such that there are only 32 distinct color bins for each channel of RGB color space.

In the experiments, the normalized region size is fixed to 41×41 pixels. For region detection, normalization, and computing SIFT descriptors we use the software routines provided by the matching protocol explained in [19].

We proposed some combinations of descriptors in order to point out that the use of some LBP variants can be very useful for obtained a good descriptor that works very well in different datasets.

The offline stage begins with introducing 3 query images representing anxiety, sad and without concentration modes for a female student. Their codes are 10, 11, and 12 respectively. Figures 5, 6 and 7 show the most similar 9 images. In these figures one can find another student's images, so the expert drops these images from comparison. Tables 1 and 2 show the offline resulted mode indices and similarity ratios for the two students (female and male) in the most similar database images. Table 3 shows the student report in a small interval.

Fig. 5. Student in anxiety mode

Fig. 6. Student in sad mode

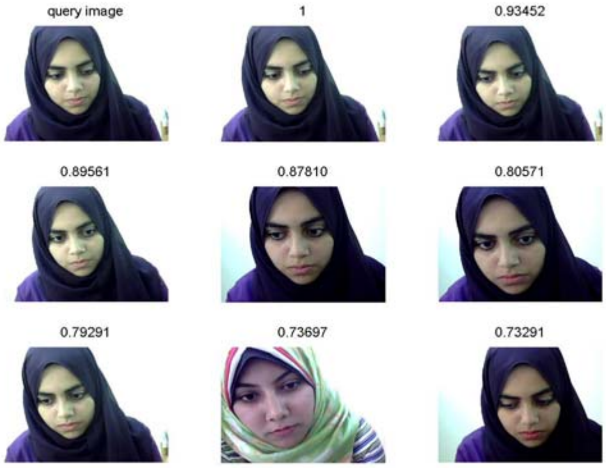

Fig. 7. Student in 'without concentration' mode

Table 1 (similarity measure for the female student)

Q1 anxiety			Q2 Sad			Q3without concentration		
Index	Mode index	Similarity ratio	Index	Mode index	Similarity ratio	Index	Mode index	Similarity ratio
120	10	1	135	11	1	158	12	1
124	10	0.89904	50	11	0.8132	159	12	0.7833
123	10	0.8956	40	11	0.8021	157	12	0.7829
119	10	0.88782	136	11	0.6455	152	12	0.7750
122	10	0.8795	51	11	0.5742	154	12	0.7747
130	10	0.8279	50	11	0.57	153	12	0.7560
131	10	0.8220	39	11	0.5690	151	12	0.7340
141	10	0.8210	49	11	0.5569	151	12	0.7289
126	10	0.8045	28	11	0.5560	150	12	0.7287
134	10	0.8032	38	11	0.5429	156	12	0.6366

Table 2 (similarity measure for the male student)

Q1 anxiety			Q2 Sad			Q3without concentration		
Index	Mode index	Similarity ratio	Index	Mode index	Similarity ratio	Index	Mode index	Similarity ratio
411	15	1	209	16	1	824	17	1
417	15	0.89904	230	16	0.8079	866	17	0.7743
415	15	0.8956	243	16	0.8034	803	17	0.7639
433	15	0.88782	205	16	0.7922	813	17	0.7345
425	15	0.8795	214	16	0.7869	7	17	0.6934
437	15	0.8279	246	16	0.7852	25	17	0.6764
468	15	0.8220	210	16	0.7765	929	17	0.6750
420	15	0.8210	13	16	0.7750	947	17	0.6709
538	15	0.8045	8	16	0.7449	918	17	0.6708
493	15	0.8032	703	16	0.7439	891	17	0.6439

Table 3 (Online student report)

	Duration													
Female	W	A	W	S	S	W	A	W	S	A	W	W	W	A
Male	S	W	W	A	A	A	W	S	W	S	A	A	S	W
	A = Anxiety S = Sad W = Without concentrate													

8. Conclusion

The proposed CS-LBP descriptor combines the strengths of two well- known methods, the SIFT descriptor and the LBP texture operator. It uses a SIFT-like grid and replaces SIFTs gradient features with a LBP-based feature, i.e. CS-LBP which was also introduced in this paper. The CS-LBP feature has many properties that make it well suited for this task, namely a relatively short feature histogram, tolerance to illumination changes, and computational simplicity. Furthermore, it does not require many parameters to be set. The performance of the CS-LBP descriptor was compared to that of the SIFT descriptor in the contexts of matching and object category classification.

The similarity ratios detected from the well defined user attitude images play major part in categorizing the database images. Each image in the database will be similar to some cases of the user attitude images. The online user tracking produces images which may be similar to some other images in the database. Defining user attitude depends on the maximum similarity ratio attached to the image in the offline phase. This framework tries to solve and follow up complicated physiological problem.

References

[1] T. Ahonen, A. Hadid, M. Pietikainen, “Face description with local binary patterns: application to face recognition” IEEE Transactions on Pattern Analysis and Machine Intelligence 28 (12) (2006) 2037–2041.

[2] L. H. He, C.R. Zou, L. Zhao, D. Hu “An enhanced LBP feature based on facial expression recognition” in Proceedings of the 2005 IEEE Engineering in Medicine and Biology 27th Annual Conference, Shanghai, 2005, pp. 3300–3303.

[3] O. Lahdenoja, M. Laiho, A. Paasio “Reducing the feature vector length local binary pattern based face recognition” in: ICIP 2005, IEEE International Conference on Publication, vol. 2, 2005, pp. II-914–II-917.

[4] T. Ojala, M. Pietikainen, T. Maenpaa “Multiresolution gray-scale and rotation invariant texture classification with local binary patterns” IEEE Transactions on Pattern Analysis and Machine Intelligence 24 (7) (2002) 971–987.

[5] M. Heikkila, M. Pietikainen, C. Schmid “Description of interest regions with center-symmetric local binary patterns” in 5th Indian Conference on Computer Vision, Graphics and Image Processing, vol. 4338, 2006, pp. 58–69.

[6] L. Nanni, A. Lumini, “Local binary patterns for a hybrid fingerprint matcher” Pattern Recognition 41 (2008), pp 3461-3466.

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

[8] M. Heikkila, M. Pietikainen, C. Schmid “Description of interest regions with local binary patterns” Pattern Recognition 42 (2009), pp 425-436.

[9] M. Varma, A. Zisserman “A Statistical Approach to Texture Classification from Single Images’ International Journal of Computer Vision, 2005.

[10] E. Hadjidemetriou, M. D Grossberg, & S. K. Nayar “Multiresolution histograms and their use for recognition” IEEE Transactions on Pattern Analysis and Machine Intelligence, July 2004, pp 831–847, ISSN: 0162-8828.

[11] H. Zhou, R. Wang, C. Wang “A novel extended local binary pattern operator for texture analysis“ Information Sciences 178 (2008), pp 4314-4325.

[12] A. Lucieer, A. Stein, P. Fisher “Multivariate texture-based segmentation of remotely sensed imagery for extraction of objects and their uncertainty” International Journal of Remote Sensing 26 (14) (2005) 2917–2936.

[13] T. Maenpaa “The Local Binary Pattern Approach to Texture Analysis-Extensions and Applications” Dr. tech. dissertation, University of Oulu (2003).

[14] K. Mikolajczyk, C. Schmid “A Performance Evaluation of Local Descriptors” IEEE Trans. on Pattern Analysis & Machine Intell., VOL. 27, NO. 10, Oct. 2005.

[15] M. A. Garcia, D. Balu “Supervised texture classification by integration of multiple texture methods and evaluation windows” Image and Vision Computing 25 (2007) 1091–1106.

[16] G. Lowe David "Distinctive image features from scale-invariant keypoints" International Journal of Computer Vision, 60, 2 (2004), pp. 91-110.

[17] L. Nanni, A. Lumini, S. Brahnam “Local binary patterns for medical image analysis“ Artificial Intelligence in Medicine, 2010.

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

[19] http://www.robots.ox.ac.uk/～vgg/research/affine/.