Available online at www.sciencedirect.com
SciVerse ScienceDirect
Procedia Engineering 15 (2011) 1771 – 1779
Procedia Engineering
www.elsevier.com/locate/procedia

Advanced in Control Engineering and Information Science

Directional Gaussian Filter-based LBP Descriptor For Textural Image Classification

Xiangping Sun, Jin Wang[a], Ronghua Chen, Lingxue Kong, Mary F.H. She, a*

The Institute for Technology Research and Innovation, Deakin University, Waurn Ponds, Vic, 3216, Australia

Abstract

As an important area in computer vision, textural image classification has been intensely investigated in the last decades. Among all existing methods, a recently developed image descriptor - local binary pattern (LBP) - has received tremendous attention because of its simplicity and robustness for representing textures. However, the selection of local binary patterns has been difficult, when the number of the patterns is very large and not all of them are discriminative in terms of representing textures. In this paper, we develop a new LBP operator (LBP_ex) for discriminative local binary patterns selection, and propose a directional Gaussian filter-based LBP_ex descriptor for textural image classification through two major steps: 1) using a bank of directional Gaussian filters to retrieve the anisotropic information in the textural images; 2) combining the LBP_ex histograms calculated from both original images and filtered images to form feature vectors that represent isotropic and anisotropic properties of the texture images in attempt to further improve the classification accuracy. We experimentally evaluate the performance of the method through comparing with four existing state-of-the-art LBP algorithms on the same database OUTex, and the results demonstrate that the features represented by the new LBP_ex descriptor are more discriminative, leading to a superior performance.

Selection and/or peer-review under responsibility of [CEIS 2011]

Keywords: local binary pattern; rotation invariant; non-uniform; isotropic; anisotropic

* Corresponding author.
E-mail address: jay.wangjin@gmail.com.

doi:10.1016/j.proeng.2011.08.330

1. Introduction

As an important area in computer vision, textural image classification has been intensely investigated during the last several decades. It plays a pivotal role in many areas including medical imaging, remote sensing, food industry, and content-based image retrieval (CBIR).

Existing methods for textural image classification could be mainly divided into four categories: statistical, structural, model-based, and transform-based methods. In recent ten years, statistical methods have gone through flourishing developments, and the representative approaches include filter bank-based methods [1-3] that use the response vector of filter banks as texture feature, patch-based algorithms [4, 5] that utilize raw pixel values in image patches for classification, bags-of-key points [6-9] where features are retrieved from the detected key points or interest areas. Besides these well-developed but complex algorithms, an image descriptor called local binary patterns (LBP) was also developed [10, 11]. LBP uses binary patterns to describe local neighborhoods of each pixel in an image, and then counts the occurrence of binary patterns to form LBP histogram as image features. Because of the simplicity and robustness of LBP for representing different images, it has received tremendous attention and been used in many different areas [12-14].

LBP is a type of higher order statistical feature which was first described by Ojala et al. [10, 15]. Since its first introduction, the selection of discriminative binary patterns for describing the images has always been a challenge. In the detailed study of LBP, Ojala et al. defined a set of rotation invariant and "uniform" binary patterns to represent a textural image (LBP^{riu2}). Liao et al. [16] believed that the uniform LBPs were not the dominating patterns (i.e., patterns of the largest proportions in an image) in some textures with irregular edges and shapes, and proposed a dominant local binary patterns (DLBP) algorithm which used first 80% most frequently occurred binary patterns as features for texture classification. Guo et al. [17] defined global dominant LBPs based on Fisher separation criteria (FSC) in three steps: firstly, find a dominant LBP set of each training image using a similar method to Liao's [16], but the threshold was set to 90%; secondly, find a dominant LBP set of each class by intersecting the dominant LBP sets of all images belonging to that class; finally, the global dominant types were formed by merging dominant types among different classes. The method is denoted as FBL-LBP.

DLBP and FLB-LBP are both dedicated to select a set of dominant binary patterns to represent the textures, with the difference that one is fixed in number, and the other is fixed in types. Although some good results were reported, there are still doubts about whether patterns which account for a small proportion (i.e., the "non-dominant" LBPs) are useless at all for differentiating one texture from the other. Furthermore, though LBP^{riu2} has been proved to be a good descriptor by considering all binary patterns, a considerable part of the discriminative power inside the LBP^{riu2} is lost as all "non-uniform" patterns are treated as one pattern. It is more desirable to preserve the discriminative power in the LBPs in order to yield high classification accuracy in most applications. In addition, He et al. [18] showed in their work that the conventional LBP methods only consider the isotropic micro structures of images, and that a higher classification accuracy could be achieved by using one Gaussian filter and four anisotropic filters on an image pyramid to extract the isotropic and anisotropic macro structures. Varma and Zisserman [3] used 38 filters to retrieve the isotropic and anisotropic feature of the original image which also showed greater performance, where the anisotropic information is presented by an edge and a bar filter at 6 orientations and 3 scales each.

In the current work, we propose a new LBP operator (denoted as LBP_ex) that can maintain the discriminative power of the LBPs by considering the "non-uniform" LBPs in more details regardless of the occurring frequency of the patterns by categorizing all LBPs into several groups based on their U values and the number of bit "1" in the binary patterns. Subsequently, we use a bank of directional Gaussian filters to retrieve the anisotropic information in textures, and combine the LBP_ex histograms calculated from original and the filtered images to form feature vectors that represent isotropic and anisotropic properties of texture images for classification. The rest of the paper is organized as follows. A brief introduction of the LBP algorithm will be given in Section 2. In Section 3 we will present the directional Gaussian filter-based LBP_ex algorithm. Experimental evaluation results obtained in this work were compared with a few existing state-of-the-art algorithms for image classification on some popular datasets (Section 4). Section 5 briefly concludes the major contributions of the work. (10 pt) Here introduce the paper, and put a nomenclature if necessary, in a box with the same font size as the rest of the paper. The paragraphs continue from here and are only separated by headings, subheadings, images and formulae. The section headings are arranged by numbers, bold and 10 pt. Here follows further instructions for authors.

2. Local Binary Pattern

In this section, original LBP algorithm will be described and its advantages and drawbacks will be evaluated in this section. Especially, the representativeness of LBP^{riu2} will be analyzed, which is the main target we propose to improve in terms of representing textures for better classification performance.

2.1. The LBP methods

LBP is a texture operator which encodes the pixel intensities in a local neighborhood into a set of binary patterns. Considering a circularly symmetric neighborhood with radius R where P pixels are equally located on the circle, the joint distribution of these P pixels with the centre pixel C could be denoted as:

$$T = t(g_C, g_0, ..., g_{P-1}) = t(g_C, g_0 - g_C, ..., g_{P-1} - g_C) \tag{1}$$

where $\{g_0, \ldots, g_{P-1}\}$ and g_C are the gray-levels of the P pixels and the centre pixel respectively. Suppose that the centre pixel is independent from the P pixels, then (1) could be rewrite as:

$$T = t(g_C)t(g_0 - g_C, ..., g_{P-1} - g_C) \tag{2}$$

Supposing that the distribution of the centre pixel $t(g_C)$ is unrelated to the local textural image representation, (2) could be simplified as:

$$T = t(g_0 - g_C, ..., g_{P-1} - g_C) \tag{3}$$

LBP is proposed by just considering the sign of the gray-level differences in (3), which could be transformed to:

$$T = t(s(g_0 - g_C), ..., s(g_{P-1} - g_C)) \tag{4}$$

where

$$s(x) = \begin{cases} 1, & x \geq 0 \\ 0, & x < 0 \end{cases}$$

Then the neighborhood distribution could be translated to a binary pattern $\{s(g_0-g_C), \ldots, s(g_{P-1}-g_C)\}$. Assigning a binomial factor for each bit, the binary pattern could be encoded as:

$$LBP_{P,R} = \sum_{p=0}^{P-1} s(g_p - g_C)2^p \tag{5}$$

The number of binary pattern types of $LBP_{P,R}$ is 2^P.

The coordinates of the P pixels are calculated by $(x_C - R\cos(2\pi p/P), y_C - R\sin(2\pi p/P))$. If the calculated coordinate do not locate on a existing pixel, interpolation will be needed to get the gray-level value of the point. Three neighborhoods, (P, R) = (8, 1), (16, 2), (24, 3) are mostly considered.

2.2. Representativeness analysis of LBPs

The feature vector will be very large if all LBP types are used. For example, when P = 8, the number will be 256, and when P = 16, it will reach 65536! As not all types are discriminative, Ojala et al. [11] introduced a LBP type selection algorithm based on the measurement of the uniformity of a binary pattern (denoted as U("pattern"), see (7)), which calculated the number of spatial transitions (bitwise 0/1 changes) in the pattern. If the U value of a pattern is less than 2, it is "uniform", otherwise it is "non-uniform". Then a LBP operator was defined as:

$$LBP_{P,R}^{riu2} = \begin{cases} \sum_{p=0}^{P-1} s(g_p - g_C), & \text{if } U(LBP_{P,R}) \le 2 \\ P+1, & \text{otherwise}, \end{cases} \tag{6}$$

$$U(LBP_{P,R}) = \left| s(g_p - g_C) - s(g_{P-1} - g_C) \right| + \sum_{p=0}^{P-1} \left| s(g_p - g_C) - s(g_{p-1} - g_C) \right| \tag{7}$$

Through this operator, the number of the generated patterns is largely reduced from 2^P to P+2, without losing discriminative power compared to the original LBP operator. However, since all LBPs with U values greater than 2 ("non-uniform") are categorized into one class, the individual discriminating ability of them is damaged. And also those "non-uniform" LBPs contain a large part in the whole images, especially when the radius of the neighborhood increases (see Fig. 1). So it is highly possible that the LBP feature will be more discriminative if the "non-uniform" patterns are decomposed into some representative structures.

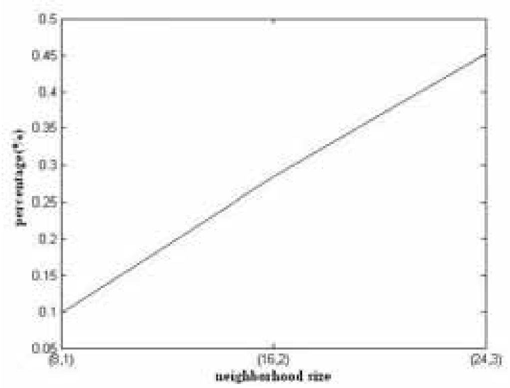


Fig. 1. The percentage of the "non-uniform" LBPs in textures in database OUTex on average

3. Directional Gaussian Filter-based LBP_ex

3.1. LBP_ex operator

As seen from the development of LBP, two most important characters of a local binary pattern are the number of bit “1” in it and the U value. LBP^{riu2} is defined on the basis of these two quantities, but it does not discriminate between the patterns with U value greater than 2, since Ojala et al. [11] thought that the relative proportion of “non-uniform” patterns of all patterns accumulated into a histogram is too small. However, we have shown in section 2.2 that the “non-uniform” LBPs actually take a large part, especially when the neighborhood size increases. Thus it is necessary to extend the “non-uniform” local binary patterns to contain more details.

We propose the new LBP operator defined as:

$$LBP\_ex_{P,R} = \begin{cases} 0, & N(LBP_{P,R}) = 0 \\ \sum_{i=0}^{N(LBP_{P,R})-1} M(i) + \dfrac{U(LBP_{P,R})}{2} - 1, & N(LBP_{P,R}) > 1 \\ \sum_{i=0}^{P-1} M(i), & N(LBP_{P,R}) = P \end{cases} \tag{8}$$

where $N(LBP_{P,R})$ denotes the number of bit “1” in the pattern $LBP_{P,R}$, and M(i) is the number of U values in all patterns with *i* bit “1”s (see (9)). The number of LBP types created by the $LBP\_ex_{P,R}$ operator is $N^2/4+2$.

$$M(i) = \begin{cases} 1, & i = 0 \\ i, & i > 0 \text{ and } i \le P/2 \\ P - i, & i > P/2 \text{ and } i \le P \end{cases} \tag{9}$$

Considering that (P, R) = (8, 1), the LBP_ex operator divides all rotation invariant local binary patterns into 18 groups, which does not only preserve the original 9 “uniform” LBPs, and but also extends the “non-uniform” LBPs to 9 classes (Fig. 2(a)). Every blue rectangle denotes one LBP class with the same number of bit “1” and U value, and it could be seen that some local binary patterns in each blue rectangle happen to be very similar in structure. Some of these patterns are shown in Fig. 2(b). We just show the structure of the patterns with number of bit “1” less than P/2, since the patterns are centre symmetric with respect to number of bit “1” in them.

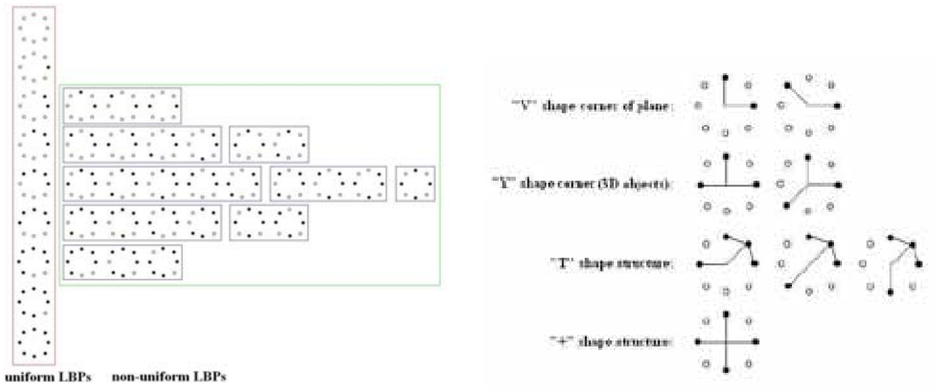
3.2. Directional Gaussian filter-based feature extraction

The directional Gaussian filter is a kind of elliptical filters, which cause greater blurring along the long axis of the ellipse (the direction of the Gaussian filter), thus could be used to smooth images whilst retaining the edge details (edge in the same direction of the Gaussian filter). A typical directional Gaussian function is defined as:

$$dG(x,y,\theta) = \frac{1}{2\sigma_x\sigma_y}\exp\left(\frac{1}{2}\left(\frac{x'^2}{\sigma_x^2} + \frac{y'^2}{\sigma_y^2}\right)\right) \tag{10}$$

where

$$x' = x\cos(\theta) + y\sin(\theta)$$
$$y' = y\cos(\theta) - x\sin(\theta)$$



(a) (b)

Fig. 2. (a) The categorization of the rotation invariant local binary patterns with parameters (P, R) = (8, 1) by the LBP_ex operator. Black points represent bit "1", white points for bit "0". (b) The structure representation of part of the local binary patterns.

We selected a bank of directional Gaussian filters F(3,6) with three pairs of sigma {(1.5, 0.25), (3.0, 1.0), (4.5, 2.25)} and six directions {0, π/6, π/3, π/2, 2π/3, 5π/6} (Fig. 3).

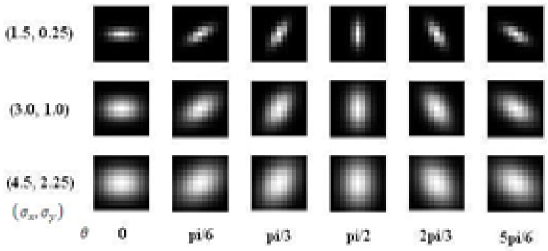


Fig. 3. The directional Gaussian filter bank

Define an original image as I. Apply the 18 filters on it and then down-sample the filtered image by 2 to get 18 subimages, denoted as $I_{l,d}$ (l=1,2,3, d=1,2,3,4,5,6).

$$I_{l,d} = (I * F(l,d)) \downarrow 2 \tag{11}$$

where * is the convolution operation, $\downarrow 2$ means downing sample by 2.

Calculate the LBP_ex histograms from I and subimages $I_{l,d}$, and denote them as $H_{0,0}$ and $H_{l,d}$ respectively.

The similarity distance between a model M and a sample S is computed by the formula:

$$SD(S,M) = D\left(H_{0,0}^{M}, H_{0,0}^{S}\right) + \sum_{i=1}^{3} D_{mm}\left(H_i^{M}, H_i^{S}\right) \tag{12}$$

$$\begin{cases} D_{min}(H_l^M, H_l^S) = \frac{1}{6}\sum_{d=1}^{6} D(H^M_{l,mod(d+k-1,6)+1}, H^S_{l,d}) \\ k = \arg\min_j \left(\sum_{l=1}^{3}\sum_{d=1}^{6} D\left(H^M_{l,mod(d+j-1,6)+1}, H^S_{l,d}\right)\right), j = 0,1,2,3,4,5 \end{cases}$$

where $D(H^M_{l,d}, H^S_{l,d})$ calculates the chi-square distance between $H^M_{l,d}$ and $H^S_{l,d}$. The chi-square distance between two vectors V and W is defined as:

$$D(V,W) = \sum_i \frac{(V(i) - W(i))^2}{|V(i) + W(i)|} \tag{13}$$

4. Experimental Results

We experimentally evaluated the textural image classification performance based on the proposed LBP_ex descriptor by using the images in the database of OUTex [19]. Two experiments were conducted, with the intention to evaluate the LBP_ex and LBP^{riu2} descriptors on representing the original textural images on the one hand, and on the other hand compare our proposed filter-based LBP_ex method with some other state-of-the-art algorithms.

OUTex: this database contains 16 ready-made test suites with different image conditions for texture classification. We choose two most popular test suites of it in our experiments- OUTex_TC_00010 and OUTex_TC_00012. These two datasets have the same 24 textures (Fig. 4), and each texture is captured in 9 rotation angles {0°, 5°, 10°, 15°, 30°, 45°, 60°, 75°, 90°}. In each angle, there are 20 sample images. In OUTex_TC_00012 all textures are imaged under three illuminants (‘inca’, ‘tl84’, ‘horizon’) respectively, while in OUTex_TC_00010 only the ‘inca’ illuminant is used. In the experiments, we select 480 images from OUTex_TC_00010 for training (20 images of angle 0° for each texture), and use the rest images in OUTex_TC_00010 and all images in OUTex_TC_00012 for test.

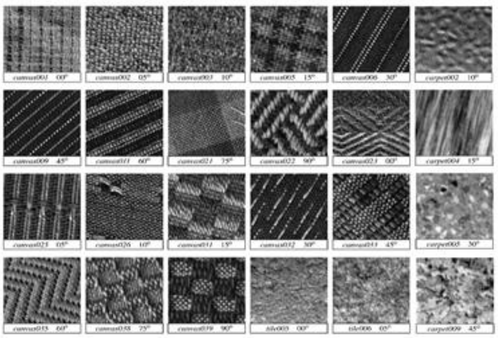


Fig. 4. Textures in OUTex_TC_00010 and OUTex_TC_00012

4.1. Experiment 1

We first use the LBP_ex and LBP^{riu2} methods to classify textures in the two OUTex datasets. We used nearest neighbor (NN) as the classifier (same classifier in experiment 2), and recorded the percentage of accurately classified test images (i.e., classification accuracy) for comparison. Table 1 shows the comparison results of these two methods, and it can be seen that LBP_ex is superior to LBP^{riu2} in all cases.

Table 1. Classification accuracy (%) of the LBP_ex and LBP^{riu2} methods for OUTex

	LBP^{riu2}			**LBP_ex**		
	(8,1)	**(16,2)**	**(24,3)**	**(8,1)**	**(16,2)**	**(24,3)**
TC_00010	**84.89**	**89.24**	**95.17**	**86.72**	**93.41**	**95.18**
TC_00012 (‘tl84’)	**65.90**	**82.29**	**85.05**	**70.09**	**84.86**	**87.41**
TC_00012 (‘horizon’)	**63.75**	**75.14**	**80.81**	**67.50**	**79.72**	**83.59**

4.2. Experiment 2

The proposed directional Gaussian filter-based LBP_ex method (denoted as filter-LBP_ex) was compared with four popular LBP algorithms, $LBP^{riu2}/VAR_{P,R}$ [11], DLBP+NGF [16], FBL-LBP [17], and MS-LBP^{riu2} [18] for the textural image classification tasks in database OUTex. The highest accuracies obtained by these five methods are depicted in Table II.

Table 2. Comparison of the classification accuracy (%) of different methods

	OUTex		
	TC_00010	**TC_00012 (‘tl84’)**	**TC_00012 (‘horizon’)**
$LBP^{riu2}/VAR_{P,R}$	**97.8**	**87.4**	**87.0**
DLBP+NGF	**99.1**	**93.2**	**90.4**
FBL-LBP	**98.31**	**93.68**	**89.56**
MS-LBP^{riu2}	**99.30**	**98.26**	**97.08**
filter-LBP_ex	**99.90**	**97.69**	**98.03**

From Table 2, we could see that except in dataset TC_00012 (‘tl84’) in which classification accuracy of our method is slightly lower than the MS-LBP^{riu2} method, the filter-LBP_ex is superior to all other methods on other two datasets. It can be seen that the performance of MS-LBP^{riu2} is close to our method, and these two methods both get their highest classification accuracy in scale (16, 2). Overall, our method produces more accurate results than MS-LBP^{riu2}. The results would be ascribed to three attributes in our method. Firstly our method considers retrieving the anisotropic information in 6 directions, while MS-LBP^{riu2} uses only 4 directions. Secondly, we used directional Gaussian filters to lower the dimension of the original image while retaining the edge information, unlike the pyramid construction process in MS-LBP^{riu2} where the edges might be blurred. Thirdly, the LBP_ex operator captures more microstructures from images, which would be more discriminative.

5. Conclusions

In this paper we developed a new LBP operator (LBP_ex) to calculate the LBP patterns coupled with a bank of directional Gaussian filters to retrieve isotropic and anisotropic LBP features from original and filtered images for texture classification, respectively. The experimental results demonstrate that the proposed method is superior to other state-of-the-art LBP algorithms in classification of textural images in the database OUTex. In the future, we will work on developing more discriminative image descriptors and applying our method to other image classification tasks (object and scene classification).

References

[1] T. Leung and J. Malik, “Representing and recognizing the visual appearance of materials using three-dimensional textons,” International Journal of Computer Vision, 2001. 43(1): p. 29-44.

[2] O. G. Cula and K. J. Dana, “3D texture recognition using bidirectional feature histograms,” International Journal of Computer Vision, 2004. 59(1): p. 33-60.

[3] M. Varma and A. Zisserman, “A statistical approach to texture classification from single images,” International Journal of Computer Vision, 2005. 62(1): p. 61-81.

[4] M. Varma and A. Zisserman, “A Statistical Approach to Material Classification Using Image Patch Exemplars,” IEEE Transactions on Pattern Analysis and Machine Intelligence, 2009. 31(11): p. 2032-2047.

[5] J. Xie, L. Zhang, J. You, and D. Zhang, “Texture classification via patch-based sparse texton learning,” in 2010 17th IEEE International Conference on Image Processing (ICIP). 2010.

[6] G. Csurka, C. R. Dance, L. Fan, J. Willamowski, C. Bray, “Visual categorization with bags of keypoints,” in Workshop on Statistical Learning in Computer Vision, ECCV. 2004.

[7] K. Mikolajczyk and C. Schmid, “A performance evaluation of local descriptors,” IEEE Transactions on Pattern Analysis and Machine Intelligence, 2005. 27(10): p. 1615-1630.

[8] S. Lazebnik, C. Schmid, and J. Ponce, “A sparse texture representation using local affine regions,” IEEE Transactions on Pattern Analysis and Machine Intelligence, 2005. 27(8): p. 1265-1278.

[9] J. Zhang, M. Marszalek, S. Lazebnik, and C. Schmid., “Local features and kernels for classification of texture and object categories: A comprehensive study,” International Journal of Computer Vision, 2007. 73(2): p. 213-238.

[10] T. Ojala, M. Pietikäinen, and D. Harwood, “A comparative study of texture measures with classification based on featured distributions,” Pattern Recognition, 1996. 29(1): p. 51-59.

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

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

[13] G. Zhao and M. Pietikainen, “Dynamic texture recognition using local binary patterns with an application to facial expressions,” IEEE Transactions on Pattern Analysis and Machine Intelligence, 2007. 29(6): p. 915-928.

[14] L. Yeffet and L. Wolf, “Local trinary patterns for human action recognition,” in 2009 IEEE 12th International Conference on Computer Vision. 2009.

[15] T. Ojala, M. Pietikainen, and D. Harwood, “Performance evaluation of texture measures with classification based on Kullback discrimination of distributions,” in Pattern Recognition, 1994. Vol. 1 - Conference A: Proceedings of the 12th IAPR International Conference on Computer Vision & Image Processing. 1994.

[16] S. Liao, M. W. K. Law, and A. C. S. Chung, “Dominant local binary patterns for texture classification,” IEEE Transactions on Image Processing, 2009. 18(5): p. 1107-1118.

[17] Y. Guo, G. Zhao, M. Pietikäinen, and Z. Xu, “Descriptor learning based on Fisher separation criterion for texture classification,” in Computer Vision – ACCV 2010, R. Kimmel, R. Klette, and A. Sugimoto, Editors. 2010, Springer Berlin / Heidelberg. p. 185-198.

[18] Y. He, N. Sang, and C. Gao, “Pyramid-based multi-structure local binary pattern for texture classification,” in Computer Vision – ACCV 2010, R. Kimmel, R. Klette, and A. Sugimoto, Editors. 2010, Springer Berlin / Heidelberg. p. 133-144.

[19] T. Ojala, T. Mäenpää, M. Pietikäinen, J. Viertola, J. Kyllönen, and S. Huovinen, “Outex - New framework for empirical evaluation of texture analysis algorithms,” in Proceedings of the 16 th International Conference on Pattern Recognition (ICPR'02) Volume 1 - Volume 1. 2002, IEEE Computer Society. p. 10701.