Rotation Invariant Static And Dynamic Texture Classification With Local Binary Count (LBC)

ABSTRACT

Texture classification is one of the four problem domains in the field of texture analysis. Local Binary Count (LBC) is a static local descriptor that can enhance the performance of rotation invariant texture classification. It extracts the local binary grayscale difference information and abandons the microstructure information. A variant of LBC named Completed LBC (CLBC) is used for static texture classification. It has two additional operators namely magnitude and centre along with LBC. For dynamic textures, two descriptors have been proposed namely Volume LBC (VLBC) and LBC-TOP (LBC from Three Orthogonal Planes). Experiments were conducted using OUTEX database for Static texture classification and DYNTEX database for Dynamic texture classification. The experimental results show that proposed descriptors can provide better classification accuracy with reduced computational complexity and time and also effectively deal with rotation variations of dynamic textures than the earlier approaches based on Local Binary Pattern (LBP).

Keywords: LBC, completedLBC, VLBC, LBC-TOP, χ² statistics

1. INTRODUCTION

Texture analysis is a basic vision problem, with application in many areas, e.g., object recognition, remote sensing and content-based image retrieval. Texture analysis can be divided into four problem domains as follow: texture classification, texture segmentation, texture synthesis and shape from 3D.

The main goal of texture classification is to assign an unknown sample texture into a set of known texture classes. In many practical applications, textures are captured in arbitrary orientations and scale. So far, many approaches have been proposed to achieve rotation invariance for texture classification. In statistical methods, texture is generally described by the statistics of selected features, e.g., invariant histogram, texture elements and micro-structures. These include early approaches such as Co-occurrence matrices, Fourier descriptors, descriptors based on Hough transform. In model based methods, texture is usually presented as a probability model or a linear combination of a set of basic functions. It includes autoregressive model, hidden Markov model, four tap wavelet filter coefficients. But they are not robust to variance in illumination.

In [1], Ojala et al proposed an efficient method, namely Local Binary Pattern (LBP), for rotation invariant texture classification. The algorithm of LBP contains two main steps, i.e., thresholding step and encoding step. In the thresholding step, the values of neighbor pixels are turned to binary values (0 or 1) by comparing them with the central pixel. Obviously, the local binary grayscale difference information is extracted in the thresholding step. In the encoding step, the binary numbers are encoded to characterize a structural pattern, and then the code is transformed into decimal number. Aiming at achieving rotation invariance, Ojala proposed rotation invariant uniform LBP (LBP^uni), in which only rotation invariant uniform local binary patterns were selected. It was believed that LBP is an excellent measure of the spatial structure of local image texture since it can effectively detect micro-structures (e.g., edges, lines, spots) information. After that, a lot of variants of the LBP for rotation invariant texture classification have been proposed. For example, Heikkila et al [2] proposed center-symmetric LBP (CS-LBP) by comparing center-symmetric pairs of pixels instead of comparing neighbors with central pixels. Liao et al [3] presented Dominant LBP (DLBP), in which dominant patterns were experimentally chosen from all rotation invariant patterns. Others tried to further explore the contrast information. For example, Tan and Triggs [4] proposed the method of Local Ternary Pattern (LTP), which extends original LBP to 3-valued codes. Guo et al [5] proposed the completed LBP (CLBP) by combining the conventional LBP with the measures of local intensity difference and central gray level. LBP encoding process is used in all of these variants mentioned above because it is believed that structural patterns characterized by the binary codes are more important for rotation invariant texture recognition while local binary grayscale difference information is considered to be merely a supplement of micro-structures. Experimental results illustrate that the most discriminative information of local texture for rotation invariant texture classification is not the ‘micro-structures’ information but the local binary grayscale difference information. Thus it can be eliminated by using a local operator that discards the structural information from LBP operator, which is known as Local Binary Count (LBC). A completed LBC (CLBC) similar to CLBP can achieve comparable accurate classification rates. In addition, CLBC allows slight computational savings in the process of training and classification.

The rest of this paper is organized as follows: Section II presents the static texture descriptors. Section III presents the proposed dynamic texture descriptors. Section VI
describes the classifier. Experimental results are presented in Section V and Section VI concludes the paper with some conclusive remarks.

II. STATIC TEXTURE DESCRIPTORS

A. Local Binary Count (LBC)

In the original LBP and its variants, each pixel in the local neighbor set is turned to binary form by comparing it with the central pixel. Then these binary values are encoded to form the local binary patterns. In the proposed LBC, we only count the number of value 1’s in the binary neighbor sets instead of encoding them. The working principle of LBC is illustrated in Fig. 1. The number of value 1’s is 4 in the binary neighbor set, thus the LBC code of the central pixel is also 4.

As a result, we can define the computing process for the LBC as follows:

\[ LBC_{P,R} = \sum_{p=1}^{P-1} s(g_p - g_c) \]  

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

where \( g_c \) represents the gray value of the center pixel and \( g_p \) (\( p = 0, 1, \ldots, P-1 \)) denotes the gray value of the neighbor pixel on the circle of radius R and P denotes the total number of neighbors. The main difference between the LBP and the LBC is that the LBP is to use the binary number to encode local patterns while the LBC merely counts the number of value 1’s in local neighbor set. But their meanings are very different. Usually, the LBP is to focus on the local structural information characterized by various patterns, while the LBC is only involved in the fact that how many pixels have comparatively higher gray level than the central one in local area. In other words, the LBP can extract the local structure information, while the LBC is merely to focus on the local binary grayscale difference information. Macroscopic textures can be regarded as the repeats for a large number of local microscopic patterns. Thus, the statistics of the selected local microscopic patterns can characterize the whole texture. But the “micro-structure” is quite different from macroscopic textural structure.

B. Completed Local Binary Count (CLBC)

Completed Local Binary Count (CLBC) can extract completed local textural information. It contains three operators:

- CLBC-Sign (CLBC_S),
- CLBC-Magnitude (CLBC_M) and
- CLBC-Center (CLBC_C).

Generally, the CLBC_S equals to the original LBC described above in Eqn. (1). In order to code the CLBC_M in a consistent format with that of the CLBC_S, the CLBC_M can be defined as:

\[ CLBC_{M,P,R} = \sum_{p=1}^{P-1} s(m_p - c), \]  

where \( g_c, g_p, \) and \( s(x) \) are defined as in Eqn. (1) and \( c \) denotes the mean value of \( m_p \) in the whole image. The CLBC_M counts how many neighbors have comparatively much higher intensity than the central pixel. Thus it is used to extract additional information of the local intensity differences.
In [5], it has been proven that the center pixel can be used to express the local gray level in the image. Thus the CLBC_C can be defined identical to the CLBP_C as follows:

$$\text{CLBC}_C(p, R) = s (g_c - c_I)$$  \hspace{1cm} (3)

where threshold $c_I$ is set as the average gray value of the whole image. The different operators can be combined jointly and hybridly. In the first way, joint histograms (3D histograms) are taken and in the second way the 2D histograms are concatenated.

### III. DYNAMIC Texture DESCRIPTORS

#### A. Volume LBC (VLBC)

VLBC is computed from three frames of the video sequence: the current frame or the frame of the pixel for which the code is being computed and the previous and posterior neighbouring frames with time interval $L$.

#### B. LBC-TOP

The LBC is computed from three orthogonal planes i.e., XY, XT, YT planes. The histograms obtained are concatenated into a single histogram.

![Fig. 3 Computation of LBC-TOP](image)

### VI. CLASSIFIER

#### A. Dissimilarity Measure - $\chi^2$ Statistics

In this paper, we utilized the $\chi^2$ statistics as the dissimilarity between two histograms. The $\chi^2$ statistics is a bin-by-bin distance, which means only the pairs of bins that have the same index are matched. If $H = \{h_i\}$ and $K = \{k_i\}$ (i = 1, 2...B) denote two histograms, then $\chi^2$ statistics can be calculated as follows:

$$d_{\chi^2}(H, K) = \sum_{i=1}^{B} \frac{(h_i - k_i)^2}{h_i + k_i}$$  \hspace{1cm} (4)

$\chi^2$ metric is computed for the histograms of the resultant test and train images or sequences. A simple multi-resolution framework can be used to improve the classification accuracy.

### V. EXPERIMENTS AND DISCUSSIONS

Experiments can be carried on three large and representative databases for static texture classification: the Outex database [6], CURET database [7] and UIUC database [8]. For static texture, we have carried out experiments on Outex database. We have used Dyntex database for dynamic texture classification.

#### A. Experimental setup for static texture classification

We used the Outex test suite Outex_TC_0010 (TC10) which contains 24 classes of texture images captured under three illuminations “inca” and nine rotation angles ($0^\circ$, $5^\circ$, $10^\circ$, $15^\circ$, $30^\circ$, $45^\circ$, $60^\circ$, $75^\circ$, and $90^\circ$). There are twenty $128 \times 128$ images for each rotation angle. The $24 \times 20$ images of rotation angle $0^\circ$ were adopted as the training data. The other 8 rotation angles were used for test. Firstly, the CLBC_S and the CLBP_S, the CLBC_M and the CLBP_m...
achieve similar accurate classification rates respectively. The results of CLBC_S, CLBC_M, CLBC_C operators are shown in the following figure.

![Image of input image texture and results of CLBC_S, CLBC_M, CLBC_C operators]

Fig. 4. From top left: Input image texture (Outex TC_00010), result of CLBC_S, CLBC_M, CLBC_C operators

Secondly, better classification rates than the ones obtained by LTP, LBP/VAR and DLBP can be achieved by combining ‘Magnitude’ with ‘Sign’ jointly or hybridly. In 2D-joint way, the CLBC_S_M and the CLBP_S_M can get similar classification rates. In the 3D-joint way, the CLBP_S/M/C and the CLBC_S/M/C achieve much better results than the other methods. The CLBP_S/M/C is slightly better than the CLBC_S/M/C. By applying the multi-scale scheme, some better results could be obtained. The following Table I lists the experimental results of different methods on TC10Database.

<table>
<thead>
<tr>
<th>Classification Rate (%)</th>
<th>P=8</th>
<th>P=16</th>
<th>P=8</th>
<th>P=16</th>
</tr>
</thead>
<tbody>
<tr>
<td>CLBC_S</td>
<td>82.94</td>
<td>87.68</td>
<td>82.34</td>
<td>88.67</td>
</tr>
<tr>
<td>CLBC_M</td>
<td>78.95</td>
<td>84.42</td>
<td>84.92</td>
<td>92.44</td>
</tr>
<tr>
<td>CLBC_S/M/C</td>
<td>93.75</td>
<td>92.36</td>
<td>92.29</td>
<td>95.91</td>
</tr>
<tr>
<td>CLBC_M/C</td>
<td>89.50</td>
<td>95.98</td>
<td>96.30</td>
<td>96.92</td>
</tr>
<tr>
<td>CLBC_S/C</td>
<td>94.89</td>
<td>95.75</td>
<td>93.80</td>
<td>95.26</td>
</tr>
<tr>
<td>CLBC_M_S/C</td>
<td>95.26</td>
<td>96.53</td>
<td>96.48</td>
<td>97.57</td>
</tr>
<tr>
<td>CLBC_S_M</td>
<td>90.72</td>
<td>94.66</td>
<td>93.72</td>
<td>96.27</td>
</tr>
<tr>
<td>CLBC_S/M</td>
<td>95.23</td>
<td>96.77</td>
<td>98.09</td>
<td>98.09</td>
</tr>
<tr>
<td>CLBC_S/M/C</td>
<td>97.16</td>
<td>97.89</td>
<td>98.56</td>
<td>98.54</td>
</tr>
</tbody>
</table>

TABLE I: CLASSIFICATION RATES (%) ON TC10 DATASETS

B. Experimental setup for dynamic texture classification

Dyntex database is used for dynamic texture classification. The experimental setup used is a combination of the ones that were used in [9] and [10]. The setup in [10] cannot be used to evaluate the rotation invariance of the proposed descriptors. The new experimental setup is created using the new Dyntex database because the size of the DTs has been greatly reduced when compared to the original database. It is created from 18 videos belonging to seven classes (i.e., seas, calm water, fountains, vegetation, trees, flowers and traffic). The sequences were resized to 128 x 128 window size as used in static texture classification. Each sequence was divided into eight non-overlapping subsets half in X, Y and T. The segmented sequences were rotated through four angles (0°, 90°, 180°, 270°). Thus a total of 576 (18 x 8 x 4) sequences were used as training samples. The sequences that were cut only in time direction were used as tests, accounting to 36 (18 x 2) test models.

![Table II: Classification rates (%) on Dyntex database]

<table>
<thead>
<tr>
<th>Classification Rate (%) for</th>
<th>Classification Rate (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>VLBC_{1,1,1}</td>
<td>91.6667</td>
</tr>
<tr>
<td>VLBC_{2,1,1}</td>
<td>90.2778</td>
</tr>
<tr>
<td>VLBC_{1,2,1}</td>
<td>92.0139</td>
</tr>
<tr>
<td>VLBC_{2,2,1}</td>
<td>88.3681</td>
</tr>
<tr>
<td>VLBC_{1,2,1}</td>
<td>76.2153</td>
</tr>
<tr>
<td>LBCTOP_{6,8,1,1,1}</td>
<td>72.7431</td>
</tr>
<tr>
<td>LBCTOP_{6,1,1,3,3,1}</td>
<td>79.5139</td>
</tr>
<tr>
<td>LBCTOP_{8,8,1,1,1,1}</td>
<td>77.7778</td>
</tr>
<tr>
<td>LBCTOP_{16,8,1,1,1,1}</td>
<td>81.7708</td>
</tr>
</tbody>
</table>

TABLE II: CLASSIFICATION RATES (%) ON DYNTEX DATABASE

VI. CONCLUSION

This paper presented two descriptors namely Volume LBC(VLBC) and LBC-TOP (LBC from three Orthogonal Planes) for dynamic texture and A variant of LBC named...
Completed LBC (CLBC) is used for static texture classification. The proposed variants of LBC can achieve comparable accurate classification rates with slight computational savings in the process of training and classification.

REFERENCES


Author Details:

1. Ms. A. Shakin Banu, Associate Professor/ECE, K.L.N College Of Information Technology, Sivagangai.
2. Ms. B. Vijayalakshmi (Research Scholar), Associate Professor/ECE, K.L.N. College of Information Technology, Sivagangai.