Improving Rotation Invariance of the Volume Local Binary Pattern Operator

Guoying Zhao
Machine Vision Group, Infotech Oulu and Department of Electrical and Information Engineering, P. O. Box 4500 FI-90014 University of Oulu, Finland
gyzhao@ee.oulu.fi

Matti Pietikäinen
Machine Vision Group, Infotech Oulu and Department of Electrical and Information Engineering, P. O. Box 4500 FI-90014 University of Oulu, Finland
mkp@ee.oulu.fi

Abstract

Dynamic texture is an extension of texture to the temporal domain. Recently, a powerful method for dynamic texture recognition based on volume local binary patterns (VLBP) was proposed. In this paper, we investigate improvements of the original VLBP operator. A proof on the relation of the two rotation invariant VLBP patterns is given. Methods for obtaining rotation invariance are experimentally evaluated with differently rotated dynamic texture sequences. An approach for combining uniform and non-uniform local binary patterns is also proposed. Experimental results on DynTex database show the effectiveness of the approach, especially its robustness to rotation variations and effective compression of feature vectors.

1. Introduction

Dynamic textures (DT) are image sequences of moving scenes [1], such as a flowing river, drifting smoke, waving foliage, etc. Description and recognition of DTs is needed, for example, in video retrieval systems, which have attracted growing attention. Because of their unknown spatial and temporal extend, the recognition of DTs is a challenging problem compared with the static case [2].

Chetverikov and Péteri made a survey on approaches for dynamic texture description and recognition [3], and in [4] some popular methods [5-14] were also recently reviewed. Rotation invariant and computationally effective representation of DT is still a problem, and better solutions are needed.

Recently, a theoretically and computationally simple yet very powerful approach was proposed, in which dynamic textures are modeled with volume local binary patterns (VLBP) [15]. The local binary pattern (LBP) [16] histogram model developed for ordinary textures was extended to a volume model. The sequence is thought as a 3D volume in $X-Y-T$ space, and the volume LBP is defined for the sequence. The texture features extracted in a small local neighborhood of the volume combine the motion and appearance together. This approach is robust with respect to illumination changes due to the gray-scale invariance of the LBP operator. Two kinds of rotation invariant descriptors have been proposed in [15] and [4], respectively. To reduce the lengths of the feature vectors effectively, the uniform patterns as defined for LBP in [16] were exploited in these methods. However, the non-uniform patterns are not as negligible in the 3D case as those in the ordinary LBP. In this paper, we give a proof on the relation of the two proposed rotation invariant VLBP descriptors and evaluate the robustness of different versions to rotation variations. This kind of evaluation with rotated image sequences has not been usually done for dynamic texture descriptors. We also investigate combined use of non-uniform and uniform patterns in order to improve the recognition accuracy. Experiments on dynamic texture sequences with and without rotation are carried out to evaluate the performance.

2 Original VLBP

The basic VLBP operator was introduced as a measure for dynamic texture recognition in [15]. Dynamic texture V in a local neighborhood of a monochrome dynamic texture sequence is defined as the joint distribution v of the gray levels of $3P+2(P>1)$ image pixels. P is the number of local neighboring points around center pixel in one frame.

$$V = v(s(g_{t_c-L,c} - g_{t_c,c}), s(g_{t_c-L,0} - g_{t_c,c}), .., \\ s(g_{t_c-L,P-1} - g_{t_c,c}), \mathrm{s}(g_{t_c,0} - g_{t_c,c}), ..., s(g_{t_c,P-1} - g_{t_c,c}), \\ s(g_{t_c+L,0} - g_{t_c,c}), ..., s(g_{t_c+L,P-1} - g_{t_c,c}), \mathrm{s}(g_{t_c+L,c} - g_{t_c,c})).$$

where gray value $g_{t_c,c}$ corresponds to the gray value of the center pixel of the local volume neighborhood, $g_{t_c-L,c}$ and $g_{t_c+L,c}$ correspond to the gray value of center pixel in the previous and posterior neighboring frame with time interval L ; $g_{t,p}(t=t_c-L, t_c, t_c+L; p=0,...,P-1)$ correspond to the gray values of P equally spaced pixels on a circle of radius $R(R>0)$ in image t, which form a circularly symmetric neighbor set. $s(x)=\begin{cases}1, x\geq 0\\ 0, x<0\end{cases}$. Invariance with respect to the scaling of the gray scale is achieved by considering just the signs of the differences instead of their exact values.

To deal with rotation variations, the original version of rotation invariant VLBP [15] computes the rotation invariant LBP from each frame, and then combines them. But it does not consider the synchronization of the three frames. In real rotation, the three frames rotate simultaneously and rotate same angles. This method is referred as ri#1 in the paper.

Another way for rotation invariance proposed in [4] is obtained by rotating the neighboring set in separate three frames clockwise, and this happens synchronously so that a minimal value is selected as the VLBP rotation invariant code. We call it ri#2.

The pattern set of ri#1 is a subset of ri#2. Here we give the proof.

Proof: Assume A and B are sets of rotation invariant patters r1#1 and r1#2, respectively.

1) $\forall a \in A$, a can be expressed as (a_{posC} , a_{posN} , a_{curN} , a_{preN} , a_{preC}), while a_{preC} and a_{posC} represent the binary values of the center pixels in previous and posterior frames. a_{preN} , a_{curN} , a_{posN} are rotation invariant LBP codes of neighboring points in previous, current and posterior frames, respectively, so they are minimal in their separation of rotation operations [15]. Then we perform ri#2 for a to get $b = (b_{posC}, b_{posN}, b_{curN}, b_{preN}, b_{preC}) \in B$, that means a_{preN} , a_{curN} , a_{posN} are rotated same time to get the combined minimal value b . Obviously, $b_{preC} = a_{preC}$, $b_{posC} = a_{posC}$, $b_{preN} \geq a_{preN}$, $b_{curN} \geq a_{curN}$, and $b_{posN} \geq a_{posN}$. If $b_{preN} > a_{preN}$, $b_{curN} > a_{curN}$, or $b_{posN} > a_{posN}$, the combined code b would be bigger than a, which is inconsistent to that b is minimal. So $b_{preN} = a_{preN}$, $b_{curN} = a_{curN}$, and $b_{posN} = a_{posN}$, then $b = a$. Namely, $a \in B$.

2) But for the codes $b' = (b^{'}_{posC}, b^{'}_{posN}, b^{'}_{curN}, b^{'}_{preN}, b^{'}_{preC}) \in B$, in which at least one of $b^{'}_{curN}$ and $b^{'}_{posN}$ is not the minimal patterns in their own rotation variation ($b^{'}_{posN}$ must be the minimal to make the b minimal in whole combination), after the ri#1 operation, obtained $a' = (a^{'}_{posC}, a^{'}_{posN}, a^{'}_{curN}, a^{'}_{preN}, a^{'}_{preC}) \in A$ is definitely not equal to b' , so b' does not belong to the set A ($b' \notin A$). For example: $b' = (1, 0101, 1101, 0011, 1) \in B$ after the r1#1 operation will be $a' = (1, 0101, 0111, 0011, 1) \in A$, so $b' \neq a'$, then $b' \notin A$.

In summary, $A \subset B$ can be concluded.

3. Combination of Uniform and Non-uniform Patterns

As an extension to the original VLBP operator, we will use non-uniform patterns jointly with the uniform ones. The uniformity measure of a pattern is defined as the number of bitwise transitions from 0 to 1 or vice versa when the bit pattern is considered circular [16]. A pattern is called uniform if its uniformity measure is at most 2. Otherwise, it is non-uniform.

Ojala et al. noticed in their experiments with texture images that uniform patterns account for a bit less than 90% of all patterns when using the (8,1) neighborhood and for around 70% in the (16,2) neighborhood [16]. Therefore, to make the feature vector compact only the uniform patterns are used while the non-uniform ones are grouped and labeled with a single label.

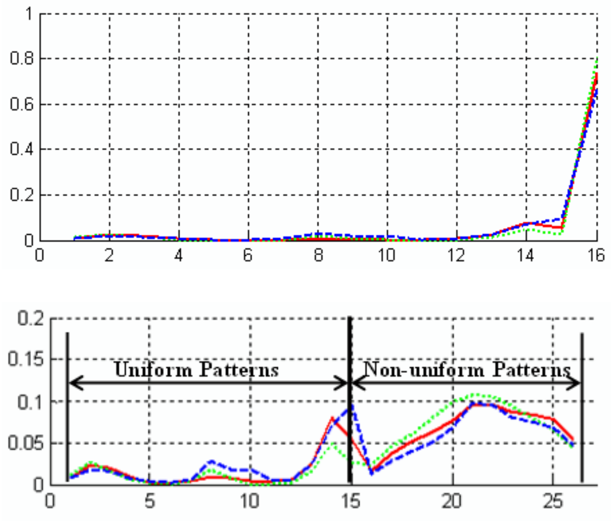

Fig. 1 Histograms of three DTs using only uniform patterns (top); Histograms of three DTs using uniform patterns and non-uniform patterns (bottom)

But in the VLBP, the neighboring points are sampled from three frames with time interval *L* and the number of neighboring points is *3P+2*, so the uniform patterns are not majority any more, as Fig. 1 (top) shows, the last label reserved for all non-uniform patterns will dominate. So to reduce the feature vector length and keep effective representation, the original label used to express all non-uniform patterns is split into several patterns by computing their number of "1" bits. The combination of rotation invariant uniform and non-uniform VLBP code is denoted as $VLBP^{riunu2}_{L,P,R}$:

$$VLBP^{riunu2}_{L,P,R} = \left\{ \begin{array}{ll} \sum_{q=0}^{3P+1} v_q & if \ U(VLBP^{ri}_{L,P,R}) \leq 2) \\ \sum_{q=0}^{3P+1} v_q + (3P+1) & otherwise \end{array} \right\},$$

where, $VLBP^{ri}_{L,P,R}$ is the rotation invariant pattern and $U(VLBP^{ri}_{L,P,R}) = \left| v^{'}_{3P+1} - v^{'}_{0} \right| + \sum_{q=1}^{3P+1} \left| v^{'}_{q} - v^{'}_{q-1} \right|$. $V' = (v^{'}_{0}, ..., v^{'}_{q}, ..., v^{'}_{3P+1})$ expresses the code after rotation invariant transform. Superscript $riu2$ reflects the use of rotation invariant uniform patterns that have U value of at most 2 while $riunu2$ is the combination of uniform and non-uniform patterns. So the total number of $VLBP^{riunu2}_{L,P,R}$'s is: $(3P+3) + (3P-1) = 6P+2$. The first 3P+3 are uniform patterns and the last 3P-1 are non-uniform ones.

The uniform and non-uniform pattern histograms of the same three DTs to Fig. 1 (top) are demonstrated in Fig. 1 (bottom). It can be seen that only using the uniform patterns, these three DTs are almost indiscriminating. However, the dispersed non-uniform patterns can help the discrimination.

4. Experiments

To evaluate the performance of our algorithm, DynTex, a large and varied database of dynamic textures, was selected for the experiments. Fig.2 shows example DTs from this dataset.

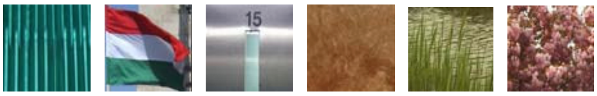

Fig. 2. DynTex database.

In classification, the dissimilarity between a sample and a model VLBP distribution is measured using the log-likelihood statistic [15]. Other dissimilarity measures like histogram intersection or Chi square distance could also be used.

4.1. Experiments on original dataset without rotation

After obtaining the VLBP features on the basis of different parameters of L , P and R , a leave-one-group-out classification test was carried out based on the nearest class. In the experiments, the same test setups as in [15] were used. Each sequence was divided into 8 non-overlapping subsets, but not half in X , Y and T . The segmentation position in volume was selected randomly. These eight samples do not overlap each other, and they have different spatial and temporal information. Sequences with the original size but only cut in time direction were also included in the experiments. So we can get 10 samples of each class and every sample is different in image size and sequence length to each other.

Table 1 presents the overall classification rates. Results from combination of uniform and non-uniform patterns (last row) are better than those from only the uniform patterns (first row) with a slight increase of the feature vector length.

Table 1. Results (%) in DynTex dataset. (*riu2* is rotation invariant uniform, while *riunu2* is rotation invariant combining uniform and non-uniform patterns. The numbers inside the parentheses demonstrate the lengths of corresponding feature vectors.)

	$VLBP_{1,2,1}$	$VLBP_{2,2,1}$	$VLBP_{1,4,1}$	$VLBP_{2,4,1}$
riu2[4]	83.43 (10)	83.43 (10)	88.57 (16)	85.14(16)
riunu2	85.71 (14)	86.86 (14)	89.71 (26)	88.86(26)

4.2. Experiments on rotated sequences

To evaluate the true rotation invariance of the methods, each sequence was rotated by 15 degree intervals as shown in Fig. 3, obtaining 24 sequences in total. Every sequence was cut in length into two sequences; so totally we have 35 classes each with 48 samples. In our experiments, two sequences with 0 degree (no rotation) were used as training samples, and the remaining ones were test sequences. Hence, in this suite, there are 70 (35x2) training models and 1610 (35x46) testing samples.

Table 2 demonstrates the results of different patterns for all rotation tests. It can be seen that a combined use of uniform and non-uniform patterns in rotation invariant $VLBP_{1,4,1}$ obtained 87.14% for all rotations (last row in Table 2) with just 26 bins. All this kind of descriptors achieved better results than those using only uniform patterns, which proves the effectiveness of including non-uniform patterns.

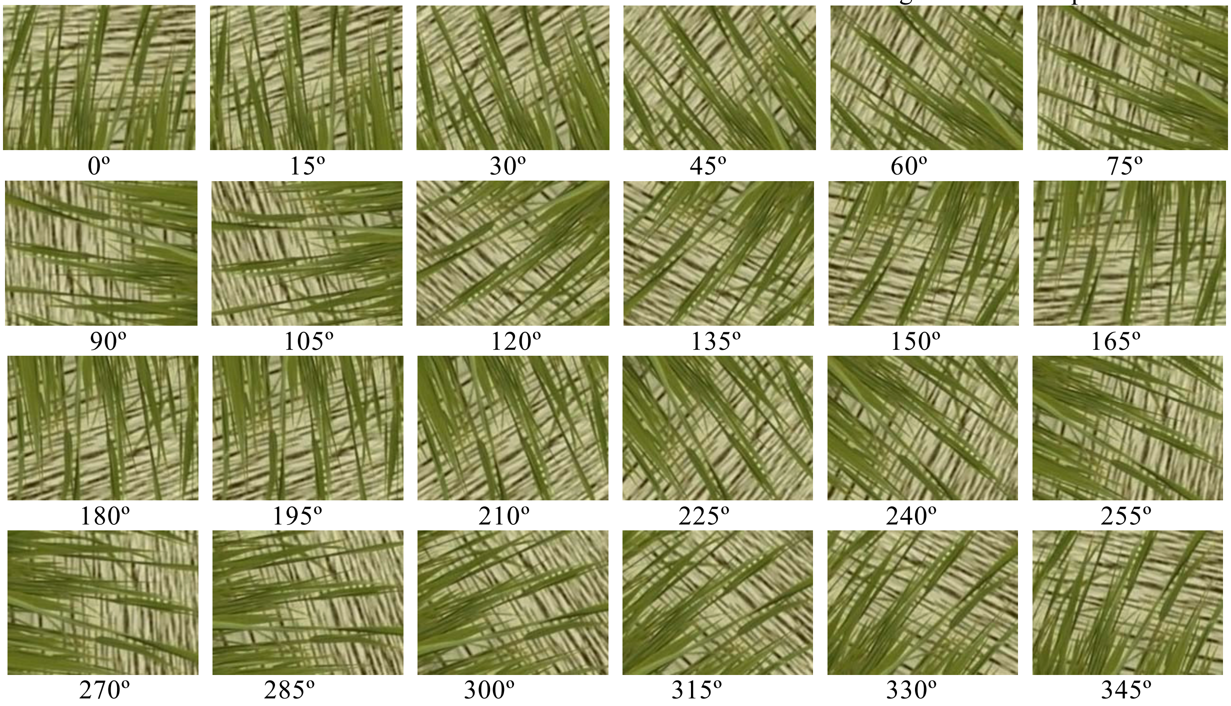

Fig. 3. Images after rotating by 15° intervals

Table 3 lists the results for testing just in every 90 degrees and 45 degrees rotation. *ri(ordinary)* in Table 3 represents to perform the a circular bit-wise right shift on the $3P+2$-bit VLBP code and select the minimal one as rotation invariant patterns, which is same to that done in static textures [16]. But this approach considers the VLBP code as a whole and assumes the texture rotates round one point, which is not suitable for the dynamic texture rotation.

It can be seen that ordinary rotation invariance (first, fourth and seventh rows) cannot deal with the rotation at all, not even when testing with rotations of 90 and 45 degrees. Rotation invariant version #2 outperforms #1. Combination of uniform and non-uniform rotation invariant patterns performs better than only uniform patterns (last three rows vs. middle three rows).

Table 2. Results using different patterns. (The numbers inside the parentheses demonstrate the lengths of corresponding feature vectors)

	$VLBP_{1,2,1}$	$VLBP_{2,2,1}$	$VLBP_{1,4,1}$	$VLBP_{2,4,1}$
Basic VLBP	56.71 (256)	59.81 (256)	48.76 (16384)	48.26 (16384)
riu2 #2	59.07(10)	60.62(10)	82.8(16)	79.75(16)
riunu2 #2	**60.37(14)**	**62.24(14)**	**87.14(26)**	**81.74(26)**

Table 3. Comparison for every 90° and 45° rotation.

$VLBP_{1,4,1}$	90°	45°
ri (ordinary)	58.1%	52.24%
ri #1	97.14%	75.92%
ri #2	98.57%	78.78%
riu2 (ordinary)	69.52%	61.63%
riu2 #1	97.14%	80.82%
riu2 #2	97.14%	82.45%
riunu2 (ordinary)	74.29%	68.57%
riunu2 #1	97.14%	82.45%
riunu2 #2	97.14%	85.71%

Our results are very good compared to the state-of-the-art. In [8], a classification rate of 98.1% was reported for 26 classes of the DynTex database without rotations. Furthermore, their test and training samples were only different in the length of the sequence, but the spatial variation was not considered. This means that their experimental setup was much simpler. When we experimented using all 35 classes with same experimental setup, a 100% classification rate was obtained.

5. Discussion

Improvements of the rotation invariance of the recently proposed VLBP operator were considered. The relation between two versions of rotation invariant VLBP patterns was proven. Methods for obtaining rotation invariance were experimentally evaluated and an approach for combining uniform and non-uniform patterns was proposed. Because the neighboring points in the original VLBP are from different frames, the patterns were shown to be not uniform any more. The original method of using one label to express all non-uniform patterns was substituted by dispersing the non-uniform patterns and labeling them separately as we did for uniform patterns. In the experiments on the dynamic textures recognition with and without rotation variation, the operators combining uniform and non-uniform patterns outperformed those using only uniform patterns.

References

[1] G. Doretto, A.Chiuso, S. Soatto, Y.N. Wu: "Dynamic textures," *International Journal of Computer Vision*, vol. 51, no.2, pp. 91-109, 2003.

[2] R. Péteri, D.Chetverikov: "Dynamic texture recognition using normal flow and texture regularity," IbPRIA 2005, pp. 223-230, 2005.

[3] D. Chetverikov, R. Péteri: "A brief survey of dynamic texture description and recognition," *In Proc. of 4th Int. Conf. on Computer Recognition Systems*, pp. 17-26, 2005.

[4] G. Zhao and M. Pietikäinen: "Dynamic texture recognition using local binary patterns with an application to facial expressions," *PAMI*, in press.

[5] R. Fablet, P. Bouthemy: "Motion recognition using nonparametric image motion models estimated from temporal and multiscale co-occurrence statistics," *PAMI 25*, pp. 1619-1624, 2003.

[6] Z. Lu, W. Xie, J. Pei, J. Huang: "Dynamic texture recognition by spatiotemporal multiresolution histogram," *In Proc. IEEE Workshop on Motion and Video Computing*, vol. 2, pp. 241-246, 2005.

[7] R. Péteri, D.Chetverikov: "Qualitative characterization of dynamic textures for video retrieval," *In Proc. International Conference on Computer Vision and Graphics*, 2004.

[8] S. Fazekas, D. Chetverikov: "Normal versus complete flow in dynamic texture recognition: a comparative study," *Texture 2005*.

[9] K. Otsuka, T. Horikoshi, S. Suzuki, M. Fujii: "Feature extraction of temporal texture based on spatiotemporal motion trajectory," *In ICPR*, pp. 1047-1051, 1998.

[10] J. Zhong, S. Scarlaro: "Temporal texture recognition model using 3D features," *Technical report*, MIT Media Lab Perceptual Computing, 2002.

[11] P.P. Wildes and J.R. Bergen. "Qualitative spatiotemporal analysis using an oriented energy representation." *ECCV*, pp. 768-784, 2000.

[12] J.R. Smith, C.-Y. Lin, M. Naphade: "Video texture indexing using spatiotemporal wavelets," *ICIP*, Vol. 2, pp. 437-440, 2002.

[13] P. Saisan, G. Doretto, Y.N. Wu, S. Soatto: "Dynamic texture recognition," *CVPR*, pp. 58-63, 2001.

[14] K. Fujita, S.K. Nayar: "Recognition of dynamic textures using impulse responses of state variables," *Texture 2003*, pp. 31-36, 2003.

[15] G. Zhao and M. Pietikäinen: "Dynamic Texture Recognition Using Volume Local Binary Patterns," *Proc. ECCV 2006 Workshop on Dynamical Vision*, 12 p., 2006.

[16] T. Ojala, M. Pietikäinen, T. Mäenpää: "Multiresolution gray scale and rotation invariant texture analysis with local binary patterns," *PAMI*: 24(7), pp. 971-987, 2002.