Dynamic Texture Recognition Using Volume Local Binary Patterns

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

Abstract. Dynamic texture is an extension of texture to the temporal domain. Description and recognition of dynamic textures has attracted growing attention. In this paper, a new method for recognizing dynamic textures is proposed. The textures are modeled with volume local binary patterns (VLBP), which are an extension of the LBP operator widely used in still texture analysis, combining the motion and appearance together. A rotation invariant VLBP is also proposed. Our approach has many advantages compared with the earlier approaches, providing a better performance for two test databases. Due to its rotation invariance and robustness to gray-scale variations, the method is very promising for practical applications.

1 Introduction

Dynamic textures or temporal textures are textures with motion [1]. Dynamic textures (DT) encompass the class of video sequences that exhibit some stationary properties in time [2]. There are lots of dynamic textures in real world, including sea-waves, smoke, foliage, fire, shower and whirlwind. Description and recognition of DT is needed, for example, in video retrieval systems, which have attracted growing attention. Because of their unknown spatial and temporal extend, the recognition of DT is a challenging problem compared with the static case [3].

Polana and Nelson classify visual motion into activities, motion events and dynamic textures [4]. Recently, a brief survey of DT description and recognition of dynamic texture was given by Chetverikov and Péteri [5]. In their paper, the existing approaches to temporal texture recognition are classified into five classes: methods based on optic flow, methods computing geometric properties in the spatiotemporal domain, methods based on local spatiotemporal filtering, methods using global spatiotemporal transforms and, finally, model-based methods that use estimated model parameters as features. Methods based on optic flow [3,4,6-13] are currently the most popular ones [5], because optic flow estimation is a computationally efficient and natural way to characterize the local dynamics of a temporal texture. Péteri and Chetverikov [3] proposed a method that combines normal flow features with periodicity features, in an attempt to explicitly characterize motion magnitude, directionality and periodicity. Their features are rotation-invariant, and the results are promising. But

they did not consider the multi-scale properties of DT. Lu et al. proposed a new method using spatio-temporal multi-resolution histograms based on velocity and acceleration fields [10]. Velocity and acceleration fields of different spatio-temporal resolution image sequences are accurately estimated by the structure tensor method. Their method is also rotation-invariant and provides local directionality information. Fazekas and Chetverikov compared normal flow features and regularized complete flow features in DT classification [14]. They conclude that normal flow contains information on both dynamics and shape. Saisan et al. [15] applied a dynamic texture model [1] to the recognition of 50 different temporal textures. Despite this success, their method assumes stationary DTs well-segmented in space and time, and the accuracy drops drastically if they are not. Fujita and Nayar [16] modified the approach [15] by using impulse responses of state variables to identify model and texture. Their approach shows less sensitivity to non-stationarity. However, the problem of heavy computational load and the issues of scalability and invariance remain open. Fablet and Bouthemy introduced temporal co-occurrence [8,9] that measures the probability of co-occurrence in the same image location of two normal velocities (normal flow magnitudes) separated by certain temporal intervals. Recently, Smith et al. dealt with video texture indexing using spatiotemporal wavelets [17]. Spatiotemporal wavelets can decompose motion into local and global, according to the desired degree of detail. Otsuka et al. [18] assume that DTs can be represented by moving contours whose motion trajectories can be tracked. They consider trajectory surfaces within 3D spatio-temporal volume data and extract temporal and spatial features based on the tangent plane distribution. The latter is obtained using 3D Hough transform. Two groups of features, spatial and temporal, are then calculated. The spatial features include the directionality of contour arrangement and the scattering of contour placement. The temporal features characterize the uniformity of velocity components, the ash motion ratio and the occlusion ratio. The features were used to classify four DTs. Zhong and Sclaro [19] modified [18] and used 3D edges in spatiotemporal domain. Their DT features are computed for voxels taking into account the spatiotemporal gradient.

Two key problems of dynamic texture recognition are: 1) how to combine motion features with appearance features, and 2) how to define features with robustness to affine transformations and insensitivity to illumination variations. To address these issues, we propose a novel, theoretically and computationally simple approach in which dynamic textures are modeled with volume local binary patterns. The local binary pattern (LBP) histogram model developed for ordinary texture [20,21] is extended to a volume model. The sequence is thought as a 3d volume in $X-Y-T$ space. A new volume LBP is defined for the sequence. The texture features extracted in a small local neighborhood of the volume are not only insensitive with respect to translation and rotation, but also robust with respect to illumination changes.

2 Volume Local Binary Patterns

The main difference between a dynamic texture and ordinary texture is that the notion of self-similarity central to conventional image texture is extended to the spatiotempo-

ral domain [5]. Therefore, combining motion and appearance together to analyze DT is well justified. Varying lighting conditions greatly affect the gray scale properties of dynamic texture. At the same time, the textures may also be arbitrarily oriented, which suggests using rotation-invariant features. Therefore, it is important to define features, which are robust with respect to gray scale changes, rotations and translation. So we propose the volume local binary patterns (VLBP) to address these problems.

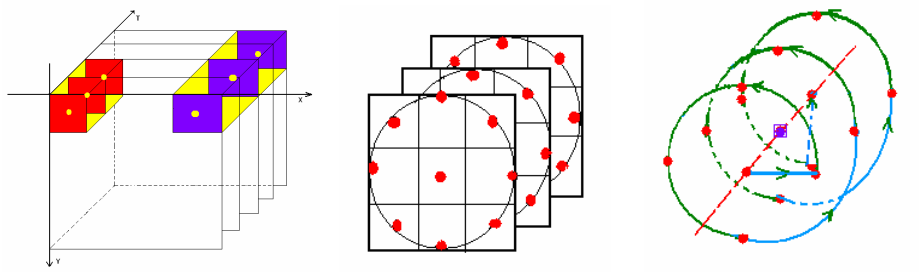
The basic LBP operator was first introduced as a complementary measure for local image contrast [20]. It is a gray-scale invariant texture primitive statistic, which has shown excellent performance in the classification of various kinds of textures [21]. For each pixel in an image, a binary code is produced by thresholding its neighborhood with the value of the center of pixel. A histogram is created to collect up the occurrences of different binary patterns. The definition of neighbors can be extended to include all circular neighborhoods with any number of pixels. In this way, one can collect larger-scale texture primitives.

2.1 Basic Volume LBP

To extend LBP to DT analysis, we define dynamic texture V in a local neighborhood of a monochrome dynamic texture sequence as the joint distribution of the gray levels of $3P+3(P \succ 1)$ image pixels.

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

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 values of the center pixels in the previous and posterior neighboring frames 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 $R(R \succ 0)$ in image t, which form a circularly symmetric neighbor set.



(a) (b) (c)

Fig. 1. (a) Volume in dynamic texture (Right volume with $L = 1$, while left volume with $L = 2$). (b) Circularly symmetric neighbor sets in volume ($R = 1$ and $P = 8$). (c) Neighboring points along the helix on the surface of cylinder ($P = 4$).

Suppose the coordinates of $g_{t_c,c}$ are (x_c, y_c, t_c), the coordinates of $g_{t_c,p}$ are given by $((x_c - R\sin(2\pi p/P), y_c + R\cos(2\pi p/P), t_c)$, and the coordinates of $g_{t_c \pm L,p}$ are given by $((x_c - R\sin(2\pi p/P), y_c + R\cos(2\pi p/P), t_c \pm L)$. The values of neighbors that do not fall exactly on pixels are estimated by closest point. Fig.1(a) shows the volume model with various time interval L in dynamic texture sequence, and Fig.1(b) illustrates circularly symmetric volume neighbor sets for (P, R). Fig.1(c) illustrates the sampling along the helix in the cylinder constructed with the neighboring frames and circularly symmetric neighbor set. The purple point is the center pixel $g_{t_c,c}$, and the red points are sampling points in the neighboring frames and circles. Blue lines illustrate the connection of neighboring frames and cyan lines the order of sampling.

To get gray-scale invariance, the distribution is thresholded similar to [21]. The gray value of the volume center pixel ($g_{t_c,c}$) is subtracted from the gray values of the circularly symmetric neighborhood $g_{t,p}(t = t_c - L, t_c, t_c + L; p = 0, ..., P-1)$, giving:

$$\begin{aligned} V = v(&g_{t_c-L,c} - g_{t_c,c}, g_{t_c-L,0} - g_{t_c,c}, ..., g_{t_c-L,P-1} - g_{t_c,c}, \\ &g_{t_c,c}, g_{t_c,0} - g_{t_c,c}, ..., g_{t_c,P-1} - g_{t_c,c}, \\ &g_{t_c+L,0} - g_{t_c,c}, ..., g_{t_c+L,P-1} - g_{t_c,c}, g_{t_c+L,c} - g_{t_c,c}). \end{aligned} \tag{2}$$

Then we can get:

$$\begin{aligned} V \approx v(&g_{t_c-L,c} - g_{t_c,c}, g_{t_c-L,0} - g_{t_c,c}, ..., g_{t_c-L,P-1} - g_{t_c,c}, \\ &g_{t_c,0} - g_{t_c,c}, ..., g_{t_c,P-1} - g_{t_c,c}, \\ &g_{t_c+L,0} - g_{t_c,c}, ..., g_{t_c+L,P-1} - g_{t_c,c}, g_{t_c+L,c} - g_{t_c,c}). \end{aligned} \tag{3}$$

This is a highly discriminative texture operator. It records the occurrences of various patterns in the neighborhood of each pixel in a ($2(P+1)+P = 3P+2$)-dimensional histogram.

We achieve invariance with respect to the scaling of the gray scale by considering just the signs of the differences instead of their exact values:

$$\begin{aligned} V \approx v(&s(g_{t_c-L,c} - g_{t_c,c}), s(g_{t_c-L,0} - g_{t_c,c}), ..., s(g_{t_c-L,P-1} - g_{t_c,c}), \\ &s(g_{t_c,0} - g_{t_c,c}), ..., s(g_{t_c,P-1} - g_{t_c,c}), \\ &s(g_{t_c+L,0} - g_{t_c,c}), ..., s(g_{t_c+L,P-1} - g_{t_c,c}), s(g_{t_c+L,c} - g_{t_c,c})). \end{aligned} \tag{4}$$

where $s(x) = \begin{Bmatrix} 1, & x \geq 0 \\ 0, & x \prec 0 \end{Bmatrix}$.

To simplify the expression of V, we use $V = v(v_0, ..., v_q, ..., v_{3P+1})$, and q corresponds to the index of values in V orderly. By assigning a binomial factor 2^q, for each sign $s(g_{t,p} - g_{t_c,c})$, we transform (4) into a unique $VLBP_{L,P,R}$ number that characterizes the spatial structure of the local volume dynamic texture:

$$VLBP_{L,P,R} = \sum_{q=0}^{3P+1} v_q 2^q . \tag{5}$$

Fig.2 gives the whole computing procedure for $VLBP_{1,4,1}$. Firstly, sampling neighboring points in the volume (Purple points), then thresholding its neighborhood with the value of the center pixel to get a binary value, and finally the VLBP code is produced by multiplying the thresholded values with weights given to the corresponding pixel and summing up the result.

Let us assume we are given a $X \times Y \times T$ dynamic texture ($x_c \in \{0,...,X-1\}$, $y_c \in \{0,...,Y-1\}$, $t_c \in \{0,...,T-1\}$). In calculating $VLBP_{L,P,R}$ distribution for this DT, the central part is only considered because a sufficiently large neighborhood can not be used on the borders in this 3D space. The basic VLBP code is calculated for each pixel in the cropped portion of the DT, and the distribution of the codes is used as a feature vector, denoted by D:

$$\begin{aligned} &D = v(VLBP_{L,P,R}(x,y,t)), \\ &x \in \{\lceil R \rceil,...,X-1-\lceil R \rceil\}, y \in \{\lceil R \rceil,...,Y-1-\lceil R \rceil\}, t \in \{\lceil L \rceil,...,T-1-\lceil L \rceil\}. \end{aligned} \tag{6}$$

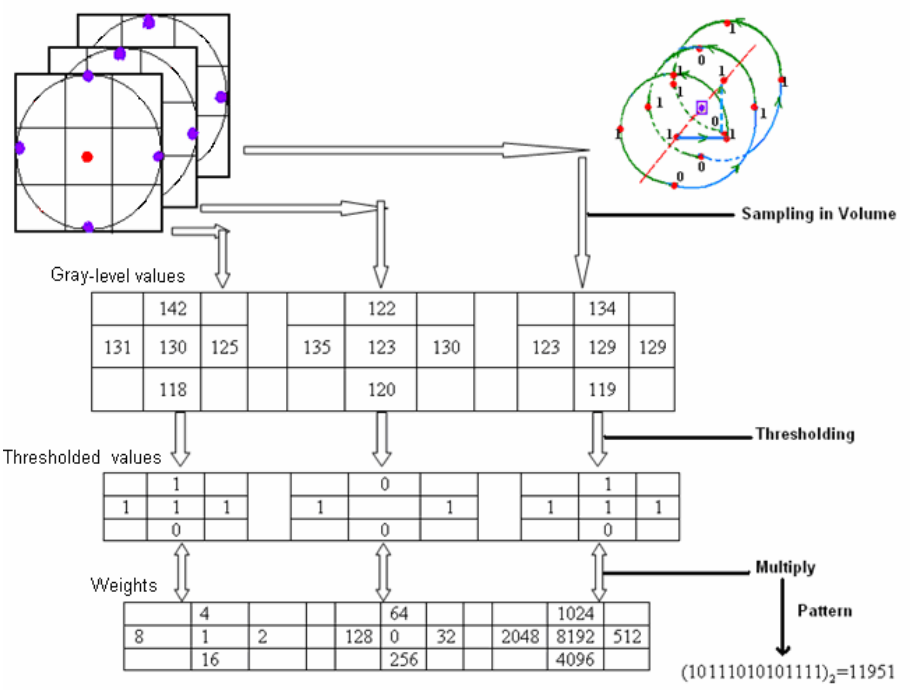


Fig.2. Procedure of $VLBP_{1,4,1}$.

So the time complexity is O(XYT). Because the dynamic texture is looked as sets of volumes and their features are extracted on the basis of those volume textons, the VLBP combines the motion and appearance together to describe dynamic textures.

2.2 Rotation Invariant VLBP

Dynamic textures may also be arbitrarily oriented, and DT also often rotates. The most important difference between rotation in a still texture image and DT is that the whole sequence of the DT rotates round one axis or multi-axes (if the camera rotates during capturing), while the still texture rotates round one point. Therefore, we cannot deal with VLBP as a whole to get rotation invariant code as in [21] which assumed rotation round the center pixel in the static case. We first divide the whole VLBP code from (4) into 5 parts:

$$\begin{aligned} V \approx v([s(g_{t_c-L,c}-g_{t_c,c})],[s(g_{t_c-L,0}-g_{t_c,c}),...,s(g_{t_c-L,P-1}-g_{t_c,c})],\\ [\mathrm{s}(g_{t_c,0}-g_{t_c,c}),...,s(g_{t_c,P-1}-g_{t_c,c})],\\ [s(g_{t_c+L,0}-g_{t_c,c}),...,s(g_{t_c+L,P-1}-g_{t_c,c})],\mathrm{s}(g_{t_c+L,c}-g_{t_c,c})), \end{aligned} .$$

Then we mark those as V_{preC} , V_{preN} , V_{curN} , V_{posN} , V_{posC} orderly, and V_{preN} , V_{curN} and V_{posN} represent the LBP code of neighboring points in previous, current and posterior frames, respectively, while V_{preC} and V_{posC} represent the binary values of the center pixels in previous and posterior frames.

$$LBP_{t,P,R}=\sum_{p=0}^{P-1}s(g_{t,p}-g_{t_c,c})2^p,\quad t=t_c-L,t_c,t_c+L. \tag{7}$$

Using formula (7), we can get $LBP_{t_c-L,P,R}$, $LBP_{t_c,P,R}$ and $LBP_{t_c+L,P,R}$.

To remove the effect of rotation, we firstly use:

$$LBP^{ri}_{t,P,R}=\min\{ROR(LBP_{t,P,R},i)\mid i=0,1,...,P-1\}. \tag{8}$$

where $ROR(x,i)$ performs a circular bit-wise right shift on the P -bit number x i times [21]. In terms of image pixels, formula (8) simply corresponds to rotating the neighbor set in one frame clockwise so many times that a maximum number of the most significant bits, starting from $g_{t,P-1}$, is 0. After getting the respective rotation variant LBP code $LBP^{ri}_{t_C-L,P,R}$, $LBP^{ri}_{t_C,P,R}$, $LBP^{ri}_{t_C+L,P,R}$, we can combine them together to get the rotation invariant VLBP, and we denote it as $VLBP^{ri}_{L,P,R}$.

$$\begin{aligned} VLBP^{ri}_{L,P,R}=&(VLBP_{L,P,R}\,and\,(2^{3P+2}-1))+ROR(LBP^{ri}_{t_C+L,P,R},2P+1)\\ &+ROR(LBP^{ri}_{t_C,P,R},P+1)+ROR(LBP^{ri}_{t_C-L,P,R},1)+(VLBP_{L,P,R}\ and\ 1) \end{aligned} . \tag{9}$$

For example, for the original VLBP code $(1,1010,1101,\ 1100,1)_2$, its codes after rotating anticlockwise 90, 180, 270 degrees are $(1,0101,1110,0110,1)_2$, $(1,1010,0111,0011,1)_2$ and $(1,0101,1011,1001)_2$ respectively. Their rotation invariant code should be $(1,0101,\ 0111,0011,1)_2$, and not $(00111010110111)_2$ obtained by using the VLBP as a whole.

Because neighboring points are sampled in volume, the number of bins is large. If the number of neighboring points in one frame is P, the number of the basic VLBP bins is 2^{3P+2}. Even for rotation invariant code, for example, the number of features for $VLBP^{ri}_{1,2,1}$ is 108. The occurrence frequencies of large number of individual patterns incorporated in $VLBP_{L,P,R}$ or $VLBP^{ri}_{L,P,R}$ vary greatly and may not provide very good discrimination, as concluded in [21]. So to reduce the feature vector length and get compact features, we borrow the idea of "uniform" patterns from [21] and compute the rotation invariant uniform VLBP code which is denoted as $VLBP^{riu2}_{L,P,R}$.

$$VLBP^{riu2}_{L,P,R} = \begin{cases} \sum_{q=0}^{3P+1} v_q & if\ U(VLBP^{ri}_{L,P,R}) \le 2) \\ 3P+3 & otherwise, \end{cases} \quad (\mathbf{10})$$

where, $U(VLBP^{ri}_{L,P,R}) = \left|v'_{3P+1} - v'_0\right| + \sum_{q=1}^{3P+1}\left|v'_q - v'_{q-1}\right|$. $V' = (v'_0, ..., v'_q, ..., v'_{3P+1})$ 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. So the total number of $VLBP^{riu2}_{L,P,R}$'s is: $3P+4$.

3 Experiments

To evaluate the performance of VLBP, two databases were selected for the experiments. The first one is the MIT dataset, which is the most frequently used collection of dynamic textures so far [1]. The second one is DynTex, which is a new large database.

VLBP histograms are used as texture models. The histograms are normalized with respect to volume size variations by setting the sum of their bins to unity. In classification, the dissimilarity between a sample and a model VLBP distribution is measured using the log-likelihood statistic:

$$L(S, M) = -\sum_{b=1}^{B} S_b \log M_b. \quad (\mathbf{11})$$

where, B is the number of bins and S_b and M_b correspond to the sample and model probabilities at bin b, respectively. Other dissimilarity measures like histogram intersection or Chi square distance could also be used.

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. If one DT includes m samples, we separate all DT samples into m groups, evaluate performance by letting each sample group be unknown and training on the rest $m-1$ samples groups. The mean VLBP features of all $m-1$ samples are computed as the feature for the class. The omitted sample is classified or verified

according to its difference with respect to the class. The k-nearest neighbor method (k=1) is used for classification.

3.1 Experiments on MIT Dataset

Fourteen classes from the MIT database were used for evaluation. To the convenience of comparison, each sequence was divided into 8 non-overlapping subsets or samples, half in X, Y and T in the same way as in [3], as Figs. 3 and 4 show. Fig.5 lists all the 14 classes and the different parts we used. Each column represents a class of dynamic texture and contains different samples of this class. First row is the top left, second row is the bottom left, third row is the top right, and last row is the bottom right of the original image.

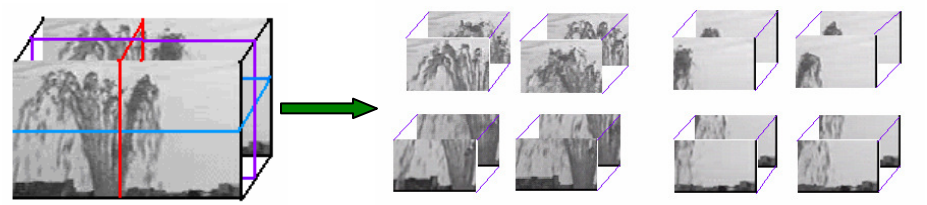


Fig.3. Each class with 8 samples in MIT DT database.

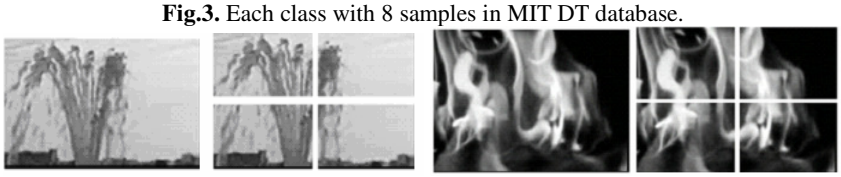


Fig.4. Two examples of segmentation in image.

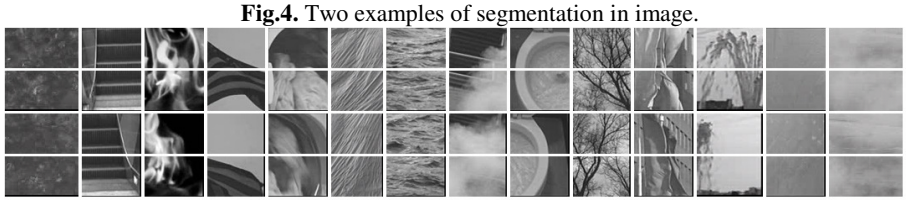


Fig.5. MIT DT dataset.

Classification results for the 14 MIT DT classes using rotation invariant uniform patterns are shown in Table 1. The best result of 100% was achieved, however, using the basic rotation invariant $VLBP_{1,4,1}$ (not shown in the table). Peteri and Chetverikov experimented on 10 classes achieving 93.8% [3]. Fazekas and Chetverikov obtained 95.4% on the same dataset [14]. Rahman and Murshed [22] used 9 classes of the MIT dataset and their classification rate was 98%, and they also gave results for 10 classes obtaining 86% [23]. Szummer and Picard [1] classified the MIT data based on the top 8 matches, and obtained an accuracy of 95%. Otsuka et al. [18] used only 4 classes from the MIT data achieving a classification rate of 98.3%. But except [3], which used the same segmentation as we but with only 10 classes, all other papers used simpler datasets which did not include the variation in space and time. Therefore, these results cannot be directly compared to ours, but we can say that our approach provided excellent results in a more difficult experimental setup.

Table 1. Results(%) for MIT dataset. ($riu2$ is rotation invariant uniform)

Features $riu2$	$VLBP_{1,2,1}$	$VLBP_{2,2,1}$	$VLBP_{1,4,1}$	$VLBP_{2,4,1}$	$VLBP_{1,8,1}$	$VLBP_{2,8,1}$
Results	88.39	91.07	91.96	95.54	98.21	98.21

3.2 Experiments on DynTex Dataset

The DynTex dataset (http://www.cwi.nl/projects/dyntex/) is a large and varied database of dynamic textures. The quality of the sequences is much better than that of the MIT data. Fig.6 shows example DTs from this dataset. The image size is 400×300.

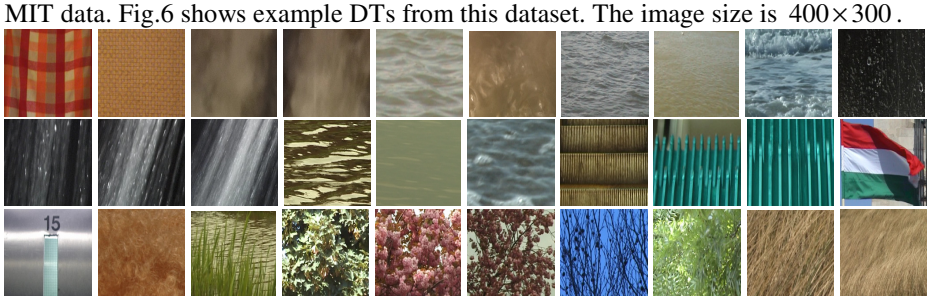


Fig.6. DynTex database.

In the experiments on DynTex database, 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. For example, we select the transverse plane with $x = 170$, lengthways plane with y= 130, and in time direction with t=100. These 8 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 are 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. Fig.7(a) demonstrates the segmentation, and Fig.7(b) shows some segmentation examples in space. These ten subsets are symbolized as A_S(short sequence with original image size), A_L(long sequence with original image size), TL_S(short sequence with top left of image), TL_L(long sequence with top left of image), BL_S(short sequence with bottom left of image), BL_L(long sequence with bottom left of image), TR_S(short sequence with top right of image), TR_L(long sequence with top right of image), BR_S(short sequence with bottom right of image), BR_L(long sequence with bottom right of image). We can see that this sampling method increases the challenge of recognition in a large database.

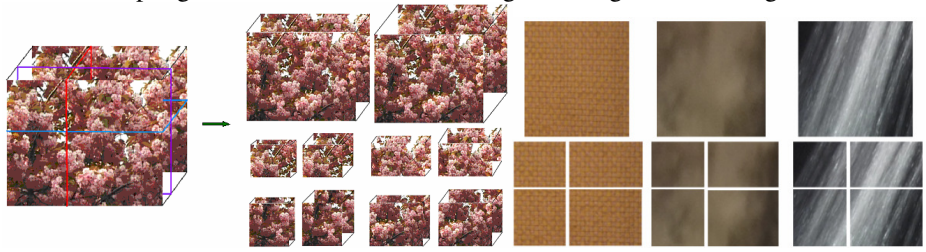


Fig.7. (a) Segmentation of DT sequence. (b) Examples of segmentation in space.

Table 2 presents the overall classification rates, while Table 3 provides more detailed results for each test dataset. When using the simple $VLBP^{riu2}_{1,2,1}$, we get good results of over 85%. By using all 256 bins of the basic $VLBP_{1,2,1}$ provides an excellent performance of 91.71% (not shown in the table).

Table 2. Results(%) in DynTex dataset.($riu2$ is rotation invariant uniform)

Features $riu2$	$VLBP_{1,2,1}$	$VLBP_{2,2,1}$	$VLBP_{1,4,1}$	$VLBP_{2,4,1}$	$VLBP_{1,8,1}$	$VLBP_{2,8,1}$
Results	85.43	81.43	87.71	84	90.57	88.29

Table 3 shows detailed results of all datasets in terms of rank order statistic [24], defined as the (cumulative) probability that the actual class of a test measurement is among its k top matches; k is called the rank. It should be mentioned that the CCR (Correct Classification Rate) is equivalent to the top 1. We can see that in the first four datasets: A_S, A_L, TL_S, and TL_L, a 100% accuracy is achieved. In the top five ranking, all the datasets are recognized correctly. This is very promising considering practical applications of DT recognition. In [14], a classification rate of 98.1% was reported for 26 classes. However, 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 samples having the original image size and only different in sequence length, a 100% classification rate was obtained with the $VLBP^{u2}_{1,8,1}$ feature.

Table 3. Results(%) of each test dataset using $VLBP^{riu2}_{1,8,1+2,8,3}$.

Test	Top 1	Top 2	Top 4	Top 5
A_S	100	100	100	100
A_L	100	100	100	100
TL_S	100	100	100	100
TL_L	100	100	100	100
BL_S	93.55	93.55	96.77	100
BL_L	87	96.77	100	100
TR_S	96.77	96.77	100	100
TR_L	90.32	96.77	100	100
BR_S	96.77	100	100	100
BR_L	96.77	100	100	100
Average	96.13	98.39	99.68	100

4 Discussion

A novel approach to dynamic texture recognition was proposed, in which volume LBP operators are used to combine the motion and appearance together. Experiments on two databases with a comparison to the state-of-the-art results showed that our method is efficient for DT recognition. Classification rates of 100% and 92% were obtained for the MIT and DynTex databases, respectively, using more difficult experimental setups than in the earlier studies. Our approach is robust in terms of grayscale and rotation variations, making it very promising for real application problems.

There are parameters L, P and R, that can be chosen to optimize the performance of the proposed algorithm. P determines the number of features. A large P produces a long histogram and thus calculating the distance matrix gets slower. Using a small P makes the feature vector shorter but also means losing more information. A small radius R and time interval L make the information encoded in the histogram more local in the spatial and temporal domain, respectively.

Results for two DT databases and comparison with the state-of-the-art show that our approach is very powerful. A topic for future research is to study in more detail how to reduce the long feature vectors of operators with many sampling points. The parameter P determines the number of features and a large value of P could produce a long histogram. Sampling in specific planes from volume will be considered to shorten the feature vector. Moreover, multiscale extensions of the method will be considered, as well as ways of computing VLBP features on the basis of blocks in order to focus on local region characteristics. Applications to other dynamic events, such as facial expression recognition, will also be investigated.

Acknowledgements

The authors would like to thank Dr. Renaud Péteri for providing the DynTex database used in experiments. This work was supported by the Academy of Finland.

References

1. Szummer, M., Picard, R.W.: Temporal texture modeling. In Proc. IEEE International Conference on Image Processing. Volume 3, (1996) 823-826
2. Doretto, G., Chiuso, A., Soatto, S., Wu, Y.N.: Dynamic textures. International Journal of Computer Vision 51(2) (2003) 91-109
3. Péteri, R., Chetverikov, D.: Dynamic texture recognition using normal flow and texture regularity. In Proc. Iberian Conference on Pattern Recognition and Image Analysis (IbPRIA 2005), Estoril, Portugal (2005) 223-230
4. Polana, R., Nelson, R.: Temporal texture and activity recognition. In Motion-based Recognition. Kluwer Academic (1997) 87-115
5. Chetverikov, D., Péteri, R.: A brief survey of dynamic texture description and recognition. In Proc. of 4th Int. Conf. on Computer Recognition Systems. Poland (2005) 17-26
6. Nelson, R.C., Polana, R.: Qualitative recognition of motion using temporal texture. CVGIP: Image Understanding 56 (1992) 78-89
7. Bouthemy, P., Fablet, R.: Motion characterization from temporal co-occurrences of local motion-based measures for video indexing. In Proc. Int. Conf. Pattern Recognition. Volume 1, Brisbane, Australia (1998) 905-908
8. Fablet, R., Bouthemy, P.: Motion recognition using spatio-temporal random walks in sequence of 2D motion-related measurements. In IEEE Int. Conf. on Image Processing, (ICIP 2001). Thessalonique, Greece (2001) 652-655
9. Fablet, R., Bouthemy, P.: Motion recognition using nonparametric image motion models estimated from temporal and multiscale co-occurrence statistics. IEEE Transactions on Pattern Analysis and Machine Intelligence 25 (2003) 1619-1624

10. Lu, Z., Xie, W., Pei, J., Huang, J.: Dynamic texture recognition by spatiotemporal multiresolution histogram. In Proc. IEEE Workshop on Motion and Video Computing (WACV/MOTION'05). Volume 2 (2005) 241-246
11. Peh, C.H., Cheong, L.-F.: Exploring video content in extended spatiotemporal textures. In Proc.1st European Workshop on Content-Based Multimedia Indexing. Toulouse, France (1999) 147-153
12. Peh, C.H., Cheong, L.-F.: Synergizing spatial and temporal texture. IEEE Transactions on Image Processing 11 (2002) 1179-1191
13. Péteri, R., Chetverikov, D.: Qualitative characterization of dynamic textures for video retrieval. In Proc. International Conference on Computer Vision and Graphics (ICCVG 2004). Warsaw, Poland (2004)
14. Fazekas, S., Chetverikov, D.: Normal versus complete flow in dynamic texture recognition: a comparative study. Texture 2005: 4th International Workshop on Texture Analysis and Synthesis, Beijing (2005). http://visual.ipan.sztaki.hu/publ/texture2005.pdf
15. Saisan, P., Doretto, G., Wu, Y.N., Soatto, S.: Dynamic texture recognition. In Proceedings of the Conference on Computer Vision and Pattern Recognition. Volume 2, Kauai, Hawaii (2001) 58-63
16. Fujita, K., Nayar, S.K.: Recognition of dynamic textures using impulse responses of state variables. In Proc. Third International Workshop on Texture Analysis and Synthesis (Texture 2003). Nice, France (2003) 31-36
17. Smith, J.R., Lin, C.-Y., Naphade, M.: Video texture indexing using spatiotemporal wavelets. In IEEE Int. Conf. on Image Processing (ICIP 2002). Volume 2 (2002) 437-440
18. Otsuka, K., Horikoshi, T., Suzuki, S., Fujii, M.: Feature extraction of temporal texture based on spatiotemporal motion trajectory. In ICPR. Volume 2 (1998) 1047-1051
19. Zhong, J., Scarlaro, S.: Temporal texture recognition model using 3D features. Technical report, MIT Media Lab Perceptual Computing (2002)
20. Ojala. T., Pietikäinen, M., Harwood, D.: A comparative study of texture measures with classification based on feature distributions. Pattern Recognition 29 (1996) 51-59
21. Ojala, T., Pietikäinen, M., Mäenpää, T.: Multiresolution gray scale and rotation invariant texture analysis with local binary patterns. IEEE Transactions on Pattern Analysis and Machine Intelligence 24(7) (2002) 971-987
22. Rahman, A., Murshed, M.: Real-time temporal texture characterisation using block-based motion co-ocurrence statistics. In Proc. IEEE International Conference on Image Processing (2004) 1593-1596
23. Rahman, A., Murshed, M.: A robust optical flow estimation algorithm for temporal textures. International Conference on Information Technology: Coding and Computing (ITCC-05). Las Vegas, USA (2005) 72–76
24. Cutting, J.T., Proffitt, D.R., Kozlowski, L.T.: A biomechanical invariant for gait perception. J.Exp. Psych.: Human Perception and Performance (1978) 357-372