To Appear at CAIP'2019

Volumes of Blurred-Invariant Gaussians for Dynamic Texture Classification

Thanh Tuan Nguyen[1,2], Thanh Phuong Nguyen[1], Frédéric Bouchara[1], and Ngoc-Son Vu[3]

[1] Universit de Toulon, Aix Marseille Universit, CNRS, LIS, Marseille, France
[2] HCMC University of Technology and Education, Faculty of IT, HCM City, Vietnam
[3] ETIS UMR 8051, Université de Paris Seine, UCP, ENSEA, CNRS, Cergy, France

Abstract. An effective model, which jointly captures shape and motion cues, for dynamic texture (DT) description is introduced by taking into account advantages of volumes of blurred-invariant features in three main following stages. First, a 3-dimensional Gaussian kernel is used to form smoothed sequences that allow to deal with well-known limitations of local encoding such as near uniform regions and sensitivity to noise. Second, a receptive volume of the Difference of Gaussians (DoG) is figured out to mitigate the negative impacts of environmental and illumination changes which are major challenges in DT understanding. Finally, a local encoding operator is addressed to construct a discriminative descriptor of enhancing patterns extracted from the filtered volumes. Evaluations on benchmark datasets (i.e., UCLA, DynTex, and DynTex++) for issue of DT classification have positively validated our crucial contributions.

Keywords: Dynamic Texture · DoG · Gaussian Filter · LBP · CLBP.

1 Introduction

Dynamic textures (DTs) are textural characteristics repeated in temporal ranges, such as waves, trees, fire, clouds, fountain, blowing flag, etc. Analyzing to clarify them is an important task for different applications in computer vision. Various methods have been introduced with diverse procedures for describing DTs. In general, those can be roughly categorized into the following groups. *Optical-flow-based approaches* [20, 18] efficiently represent the turbulent motion properties of DT videos in natural modes for issues of classifying DTs. In the meanwhile, filtering techniques are taken into account in *filtering-based approaches* to reduce the negative impacts of noise and illumination on encoding DT sequences. It should be noted that this technique is also effective for analyzing 2D texture images [14, 35]. The group of *model-based approaches* [30, 36] has mainly inherited the computational types of Linear Dynamical System [30] (LDS) and its variants to model motions of DTs in spatio-temporal aspects, while *geometry-based approaches* [37, 38, 25] geometrically capture dynamic features for DT representation based on fractal analyses, such as dynamic fractal spectrum (DFS) [38], Multi-fractal spectrum (MFS) [37], and wavelet-based MFS [11]. Recently, *learning-based approaches* have been considerable due to their promising effect in DT recognition, which can be divided into two principle trends: deep learning methods (e.g., Convolutional Neural Networks (CNNs)) [22, 1, 2] utilize deep algorithms for learning features, and the other is dictionary-learning-based techniques [23, 24] which are based on kernel sparse coding to produce learned dictionaries for DT description. Finally, *local-feature-based approaches* [15, 31–34, 16] are involved with Local Binary Pattern (LBP) operator which is fortunately applied for encoding still images thanks to its computational efficiency. For video representation, they mostly rely on two main LBP-based variants to enhance the capacity of discriminative power as follows: Volume LBP (VLBP) [39] formed on contiguous frames, and LBP-TOP [39] computed on three orthogonal planes.

Although achieving the positive performance on classifying DTs, some limitations have been enduring, such as in the filtering-based approaches: issues of noise and illumination [3]; in the local-feature-based methods: near uniform patterns, sensitivity to noise [32, 15], and large dimensional problems [39, 28, 31]. In this paper, we propose an effectively computational framework to diminish these restrictions in the following steps. Firstly, a 3D Gaussian kernel is taken into account for analyzing videos as a pre-processing to point out blurred sequences with less sensitive to noise and near uniform regions. A receptive volume of DoG is then computed from these sequences to deal with the influences of environmental and illumination changes. Finally, a robust descriptor is structured by exploiting a local encoding operator (e.g., LBP, CLBP, etc) to jointly capture shape and motion cues of blurred-invariant features from three orthogonal planes of the filtered volumes. Experiments on various benchmarks have shown that our proposal promisingly performs compared to the state-of-the-art methods.

2 Proposed Method

Exploiting local features for video representation with an effective computation, the local-feature-based approaches have acquired promising results on DT recognition. In spite of that, their performance is still in restriction due to the problems of illumination, near uniform regions, and sensitivity to noise. In this section, we firstly recall LBP and its variants as well as Gaussian-based filtering kernels. We then introduce an efficient framework for DT representation based on above materials to address typical limitations of local encoding operators.

2.1 A Brief Review of LBP and Its Completed Model

A typical LBP code is defined as a chain of bits for describing local relationships between a center pixel and others in neighborhoods of a still image [19]. Accordingly, let $\mathcal{I}$ signify a 2D gray-scale image. A binary string for each pixel $\mathbf{q} \in \mathcal{I}$ is formed by estimating the difference of gray-scale values of $\mathbf{q}$ and local neighbors $\{\mathbf{p}_i\}$ sampled its surrounding regions as

$$\mathrm{LBP}_{P,R}(\mathbf{q}) = \left\{ sign\big(\mathcal{I}(\mathbf{p}_i) - \mathcal{I}(\mathbf{q})\big) \right\}\Big|_{i=0}^{P-1} \tag{1}$$

where P denotes a number of neighbors interpolated on a circle of radius R, $\mathcal{I}(.)$ comes out the gray-level of a pixel, and function $sign(.)$ is defined as

$$sign(x) = \begin{cases} 1, x \geq 0 \\ 0, \text{otherwise.} \end{cases} \tag{2}$$

In calculation of probability distributions for image texture representation, the LBP codes form a histogram with a large dimension of 2^P distinct values. In practice, two popular mappings are utilized to treat this shortcoming: $u2$ with $P(P-1)+3$ bins for uniform features and $riu2$ with $P+2$ bins for rotation invariant uniform features. In addition, other considerable mappings are also expected to improve the structuring operation, such as LBC [41] - an alternative of uniform patterns, $TAP^{\mathcal{A}}$ mapping [13] for addressing topological information.

Guo et al. [9] proposed a completed model of LBP (CLBP) in which three complemented components are incorporated in different ways for enhancing the performance: CLBP_S that is identical to the typical LBP, CLBP_M for capturing magnitude information, and CLBP_C for obtaining the difference between the gray-level of a center pixel and that of the mean on the whole image. Experiments in [9] also validated that the joint of three components (i.e., $\text{CLBP}_{S/M/C}$), which is used in our proposal, outperforms other configurations.

2.2 Blurred-Invariant Gaussian Volumes

A Gaussian filtering is a process of convolving a Gaussian kernel on a spatial domain. It should be in accordance with the regulation of a Gaussian distribution. In general, the n-dimensional Gaussian kernel is defined as follows.

$$\text{G}^n_\sigma(x_1, x_2, ..., x_n) = \frac{1}{(\sigma\sqrt{2\pi})^n}\exp\big(-\frac{x_1^2 + x_2^2 + ... + x_n^2}{2\sigma^2}\big) \tag{3}$$

in which σ means a pre-defined standard derivation, n denotes a number of spatial axes $\{x_i\}_{i=1}^n$ taken into account the convolutions.

For analysis of DT videos, the 3D Gaussian kernel should be applied as

$$\text{G}^{3D}_\sigma(x, y, t) = \frac{1}{(\sigma\sqrt{2\pi})^3}\exp\big(-\frac{x^2 + y^2 + t^2}{2\sigma^2}\big) \tag{4}$$

where x, y denote the spatial coordinates, t indicates the temporal coordinate. As a result of that, the difference of 3D Gaussian filters is computed as

$$\text{DoG}^{3D}_{\sigma_1,\sigma_2}(x, y, t) = \text{G}^{3D}_{\sigma_1}(x, y, t) - \text{G}^{3D}_{\sigma_2}(x, y, t) \tag{5}$$

Two above kernels $\text{G}^{3D}_{\sigma_1}$ and $\text{DoG}^{3D}_{\sigma_1,\sigma_2}$ are used to filter a DT video resulting in filtered volumes of blurred ($\mathcal{V}_\text{G}$) and invariant ($\mathcal{V}_\text{DoG}$) features as follows.

$$\begin{cases} \mathcal{V}_{\text{G}_{\sigma_1}} = \text{G}^{3D}_{\sigma_1}(x, y, t) * \mathcal{V} \\ \mathcal{V}_{\text{DoG}_{\sigma_1,\sigma_2}} = |\text{DoG}^{3D}_{\sigma_1,\sigma_2}(x, y, t)| * \mathcal{V} \end{cases} \tag{6}$$

where $\mathcal{V}$ means a volume of DTs, $\sigma_1 < \sigma_2$, "*" is the convolutional operator. This volume filtering is illustrated in the second image line of Figure 1.

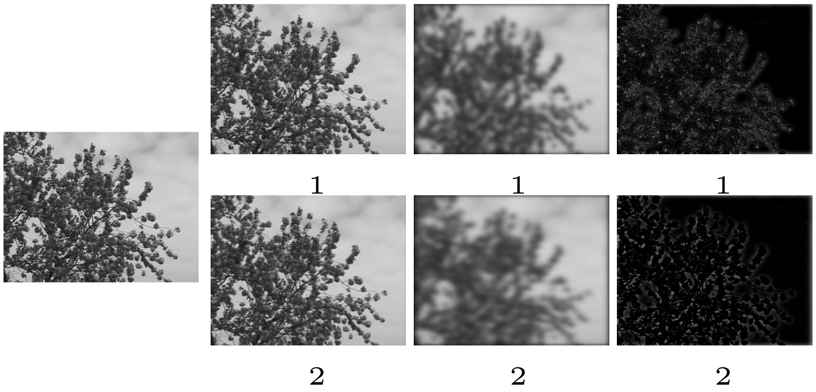

Fig. 1. A sample of Gaussian filterings. (a) is an input gray-scale frame in a DT video. (b$_1$), (c$_1$) are 2D smoothed images of (a) using $\sigma_1 = 0.5$, $\sigma_2 = 4$ respectively, and (d$_1$) denotes the 2D DoG of them [17]. In the meanwhile, (b$_2$), (c$_2$) are 3D blurred frames of (a) with the above standard derivations, and (d$_2$) is the 3D DoG of (b$_2$) and (c$_2$).

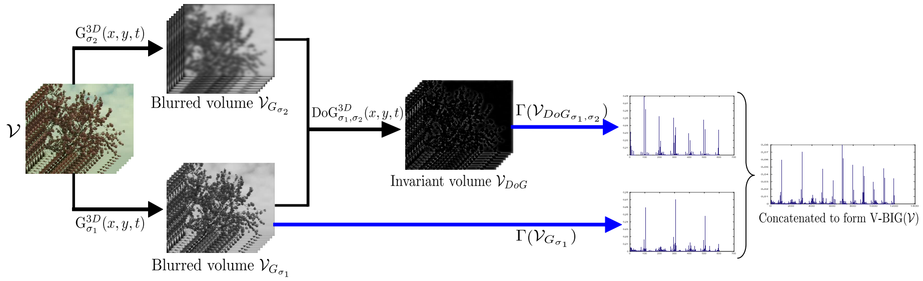
$G^{3D}_{\sigma_2}(x,y,t)$
Blurred volume $\mathcal{V}_{G_{\sigma_2}}$
$\mathcal{V}$
$DoG^{3D}_{\sigma_1,\sigma_2}(x,y,t)$
Invariant volume $\mathcal{V}_{DoG}$
$\Gamma(\mathcal{V}_{DoG_{\sigma_1,\sigma_2}})$
Concatenated to form V-BIG($\mathcal{V}$)
$G^{3D}_{\sigma_1}(x,y,t)$
$\Gamma(\mathcal{V}_{G_{\sigma_1}})$
Blurred volume $\mathcal{V}_{G_{\sigma_1}}$

Fig. 2. (Best viewed in color) Our proposed framework for structuring volumes of blurred-invariant features. Therein, the black arrow denotes a pre-processing of Gaussian filters while the blue implies a process of encoding the filtered volumes.

2.3 Proposed DT Descriptor

In this section, we propose a simple framework to efficiently capture appearance information and motion cues for DT representation. For a given video $\mathcal{V}$, the proposed framework consists of three major steps to encode DT characteristics (see Fig. 2 for graphical illustration). Firstly, the 3D Gaussian-based filters $\mathrm{G}^{3D}_{\sigma_1}$ and $\mathrm{DoG}^{3D}_{\sigma_1,\sigma_2}$ are taken into account for analyzing $\mathcal{V}$ as a pre-processing stage to figure out its corresponding filtered volumes, i.e., $\mathcal{V}_{\mathrm{G}_{\sigma_1}}$ and $\mathcal{V}_{\mathrm{DoG}_{\sigma_1,\sigma_2}}$, against the issues of noise and illumination. Secondly, each of these volumes is broken into the separative frames of image textures according to its three orthogonal planes $\{XY, XT, YT\}$. A local encoding operator Ψ is then utilized for these planes to extract spatial information and motion properties of DTs as follows.

$$\Gamma(\mathcal{V}_{\mathrm{G}_{\sigma_1}/\mathrm{DoG}_{\sigma_1,\sigma_2}}) = \left[\Psi(f_i \in XY), \Psi(f_j \in XT), \Psi(f_k \in YT)\right] \tag{7}$$

where frames f_i, f_j, and f_k belong to the corresponding planes of a filtered volume, $\Psi(.)$ denotes a function of local encoding structures (e.g., LBP, CLBP, etc.) for capturing smoothing characteristics of spatio-temporal cues from $\mathcal{V}_{\mathrm{G}_{\sigma_1}}$ and invariant features of those from $\mathcal{V}_{\mathrm{DoG}_{\sigma_1,\sigma_2}}$. An instance for encoding $\mathcal{V}_{\mathrm{G}_{\sigma_1}}$ is graphically illustrated in Fig. 3. Finally, the achieved histograms are normalized and concatenated to produce a robust descriptor based on the volumes of blurred-invariant Gaussians V-BIG$(\mathcal{V}_{\sigma_1,\sigma_2})$ to enhance the performance.

$$\text{V-BIG}(\mathcal{V}_{\sigma_1,\sigma_2}) = \left[\Gamma(\mathcal{V}_{\mathrm{G}_{\sigma_1}}), \Gamma(\mathcal{V}_{\mathrm{DoG}_{\sigma_1,\sigma_2}})\right] \tag{8}$$

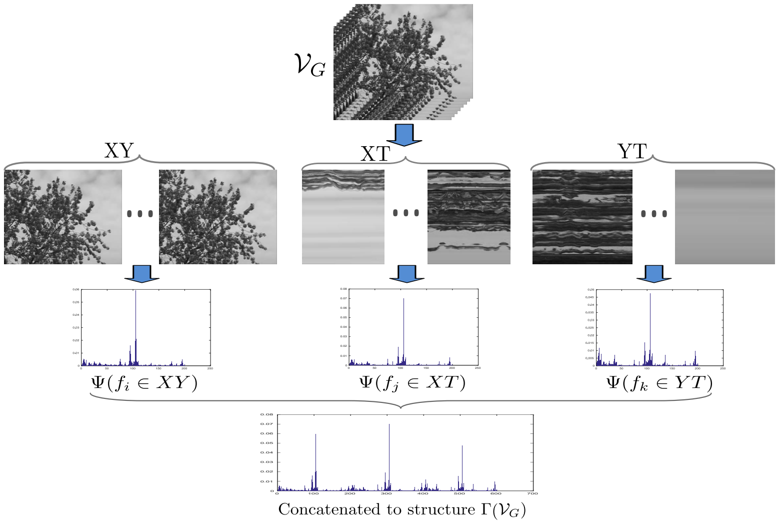

Fig. 3. Encoding model for a filtered volume $\mathcal{V}_G$.

2.4 Beneficial Properties of the Proposed Descriptor

Similar to FoSIG [17], our descriptor takes advantage of the following beneficial properties to enhance the discriminant power. In addition, V-BIG are involved with an important feature of informative voxel discrimination that leads to its outstanding performance compared to FoSIG in most of circumstances. Figure 1 shows a specific comparison of using 2D and 3D Gaussian filtering kernels, in which the outputs of the 3D filter seem more "stable" than that of 2D.

- *Robustness to changes of illumination and environment:* Filtered volumes $\mathcal{V}_{\mathrm{G}}$ and $\mathcal{V}_{\mathrm{DoG}}$ are robust against changes of illumination thanks to 3D Gaussian filtering kernels. Furthermore, the receptive DoG volume against scale changes, computed by two Gaussians of different scales, allows to capture features with more robustness to the major remaining problems of DT description: illumination, scale, and changes of environment.

- *Robustness to noise:* Instead of encoding on a raw video $\mathcal{V}$, its filtered volumes, i.e., $\mathcal{V}_{\mathrm{G}}$ and $\mathcal{V}_{\mathrm{DoG}}$, are addressed to exploit local features with more intensities to noise. It should be noted that the 2D Gaussian filtering kernel has been taken into account for analyzing neighborhoods at various regional scales of a pixel for textural image description [12], and capturing spatio-temporal information based on filtered images of the planes in a video [17]. Different from those, the 3D Gaussian filtering kernels is used to enrich discriminative information of voxels in consideration of the whole sequence. On the other hand, our proposal also integrates DoG filters to make descriptor more robust against environmental and illumination changes.
- *Forceful discriminative factors:* Well-known as an approximation of Laplacian of Gaussian (LoG), volume of $\mathcal{V}_{\mathrm{DoG}}$ provides useful receptive properties for encoding DT features. In the meanwhile, volume of $\mathcal{V}_{\mathrm{G}}$ produces robust smoothing characteristics for DT representation. As a result, these complemented components have significantly contributed to improve the performance of classification (see Table 2 for their contributions in detail).
- *Informative voxel discrimination:* Thanks to using 3D Gaussian filtering kernels, each voxel is enriched by informative discrimination that allows to *jointly capture shape and motion cues* of a DT video. It is different from FoSIG [17], in which just spatio-temporal characteristics of a voxel are calculated on 2D Gaussian filtered images of the planes in a sequence. Experiments in Section 3 validate the interest of this approach compared to [17].
- *Low computational cost:* Using a raw MATLAB code on a Linux laptop with configurations of CPU Intel Core i7 1.9 Ghz and 4G RAM, our encoding algorithm just takes less 0.84s to handle a video of 48 × 48 × 75 dimension. It is about 0.08s faster than that of FoSIG [17] (0.92s).

3 Experiments

To evaluate the performance of our proposition, we address descriptor V-BIG for task of DT recognition on different benchmark DT datasets, i.e., UCLA [30], DynTex [21], and DynTex++ [8]. For classifying, we utilize a linear multi-class SVM algorithm with the default parameters which is implemented in the LIBLINEAR[1] library [7]. The obtained rates are then evaluated in comparison with the state-of-the-art results.

3.1 Experimental Settings

To structure filtered volumes, we investigate $\sigma_1 = 0.5$, $\sigma_2 = \{1, 2, 3, 4, 5, 6\}$, and $x, y, t \in [-3\sigma, 3\sigma]$. For calculating the proposed descriptor, CLBP operator is exploited to capture local features of these volumes, i.e., $\Psi = \mathrm{CLBP}^{\mathrm{riu2}}_{\{(\mathrm{P},\mathrm{R})\}}$ with joint parameters of $riu2$ mapping, $\{(P, R)\} = \{(8, 1)\}$ for single-scale, and $\{(8, 1), (8, 2)\}$ for multi-scale to acquire more local relationships in further regions. As a result of those, the obtained descriptors have dimensions of 1200 and

[1] https://www.csie.ntu.edu.tw/∼cjlin/liblinear

2400 bins respectively. For comparison with the state of the art, the setting of $(\sigma_1, \sigma_2) = (0.5, 6)$ for the multi-scale encoding is appointed thanks to its outperformance on most of DT datasets. Empirically, it should be addressed $\sigma_2 \in [1, 6]$ due to a reduction of spatial information in case of $\sigma_2 > 6$.

3.2 Datasets and Protocols

The properties of benchmark datasets as well as experimental protocols, which are used for verifying our proposal, are exposed in this section. The summary in brief of those is shown in Table 1 for a convenient search.

UCLA Dataset: Saisan et al. [30] composed 50 categories of 200 DT sequences in 110 × 160 × 75 dimension with four videos for each of groups. In experiments of DT recognition, a split version of 48 × 48 × 75 is often used and divided into the challenging subsets as follows.

- *50-class:* DT classification using the original 50 categories with two protocols: *leave-one-out* (LOO) [3, 33] and *4-fold cross validation* [15, 32].
- *9-class* and *8-class:* 50 categories are readjusted in a different way to construct a sub-dataset of 9 groups named as "boiling water" (8), "plants" (108), "sea" (12), "fire" (8), "flowers" (12), "fountains" (20), "smoke" (4), "water" (12), and "waterfall" (16), in which the numbers in parentheses mean their quantities. As the dominance of "plants" class, it is discarded to form an 8-class scheme with more challenges [38]. Similarly, the protocol is set as in [8, 15], that a half of samples in each group is randomly taken out for training and the remain for testing. The average rates of 20 runtimes on these schemes are reported as the final results.

DynTex Dataset: Péteri et al. [21] recorded more than 650 high-quality DT videos in differences of environmental conditions. Identical to [2, 3, 6], LOO is used to evaluate DT classification rates for all of the following sub-datsets.

- *DynTex35:* It is constructed as a challenging sub-dataset from clipping 35 videos as follows: randomly splitting each video at different cutting points but not in the half of X, Y, and T axes to acquire 8 non-overlapping sub-videos; further splitting along its T axis to obtain 2 more. As a result of that, *DynTex35* is ranged into 10 categories [3, 32, 39].
- *Alpha:* It consists of three categories of 20 sequences.
- *Beta:* It includes 162 videos grouped into 10 classes with different quantities.
- *Gamma:* It contains 10 classes of 264 DT videos with varied cardinalities.

DynTex++ Dataset: 345 raw videos of DynTex are pre-processed so that just the main turbulent motions are taken out and fixed in dimension of 50 × 50 × 50 [8]. They are then grouped into 36 categories with 100 sub-videos for each, i.e., 3600 DTs in total. Similar to [3, 8], a half of items of each group is randomly chosen for training and the remain for testing. The mean of 10 runtimes is reported as the final recognition rate.

Table 1. A summary of main properties of DT datasets.

Dataset	Sub-dataset	#Videos	Resolution	#Classes	Protocol
UCLA	50-class	200	48 × 48 × 75	50	LOO and 4fold
	9-class	200	48 × 48 × 75	9	50%/50%
	8-class	92	48 × 48 × 75	8	50%/50%
DynTex	DynTex35	350	different dimensions	10	LOO
	Alpha	60	352 × 288 × 250	3	LOO
	Beta	162	352 × 288 × 250	10	LOO
	Gamma	264	352 × 288 × 250	10	LOO
DynTex++		3600	50 × 50 × 50	36	50%/50%

Note: LOO and 4fold are leave-one-out and four cross-fold validation respectively. 50%/50% denotes a protocol of taking randomly 50% samples for training and the remain (50%) for testing.

Table 2. Comparison rates (%) on DynTex++ between FoSIG [17] on filtered images and V-BIG on filtered volumes using settings of $riu2$ mapping and $\{(P, R)\} = \{(8, 1)\}$.

Descriptor	FoSIG$^{riu2}_{8,1}$ of [17]			Our V-BIG$^{riu2}_{8,1}$		
(σ_1, σ_2)	$G^{2D}_{\sigma_1}$	$DoG^{2D}_{\sigma_1,\sigma_2}$	$G^{2D}_{\sigma_1} + DoG^{2D}_{\sigma_1,\sigma_2}$	$G^{3D}_{\sigma_1}$	$DoG^{3D}_{\sigma_1,\sigma_2}$	$G^{3D}_{\sigma_1} + DoG^{3D}_{\sigma_1,\sigma_2}$
$(0.5, 3)$	95.73	93.19	96.38	96.01	94.61	96.45
$(0.5, 4)$	95.73	93.33	96.39	96.01	94.55	96.33
$(0.5, 5)$	95.73	93.52	96.12	96.01	94.26	96.14
$(0.5, 6)$	95.73	93.78	95.99	96.01	94.43	96.59

3.3 Experimental Results

Evaluations of our proposed descriptor V-BIG on the benchmark DT datasets are presented in Table 3, in which the highest rates are in bold. In the meanwhile, Table 2 shows the important contributions of each kind of filtered features in performing DT recognition. It can be verified from these tables that exploiting the filtered volumes of smooth-invariant patterns in video representation figures out a robust descriptor with outstanding operation. The experimental results are compared to those of the existing methods in Table 4. In general, our proposal is more efficient than the others, except deep-learning-based approaches utilizing a giant computational cost for DT description. It should be noted that V-BIG also outperforms significantly FoSIG [17] with the same single-scale settings of CLBP$^{\text{riu2}}_{\{(8,1)\}}$ and $(\sigma_1, \sigma_2) = (0.5, 6)$ (see Tables 3, 4). Hereafter, the proficiency of V-BIG on the specific datasets are assessed in detail.

UCLA Dataset: In this scenario, V-BIG outperforms on schemes of *50-LOO* and *50-4fold* with the settings for comparison (see Table 3). With rate of 99.5% for both of them, the proposed method deals with the same as performances of all existing approaches, including deep-learning-based techniques, i.e., DT-CNN [1] and PCANet-TOP [2] (see Table 4). In aspects of DT classification on *9-class* and *8-class*, our proposal achieves comparative results compared to the local-feature-based methods. More specifically, it can be seen in Table 4 that V-BIG's performance obtains recognition rates of 97.95%, 97.5% respectively, about 1.3% lower than those of the recent local-feature-based approaches, such as FoSIG's [17] (98.95%, 98.59%), CVLBC's [40] (99.20%, 99.02%), and MEWLSP's [34]

Table 3. Classification rates (%) on DT benchmark datasets.

Dataset	UCLA				DynTex				Dyn++
$\{(P,R)\},(\sigma_1,\sigma_2)$	50-LOO	50-4fold	9-class	8-class	Dyn35	Alpha	Beta	Gamma	
$\{(8,1)\},(0.5,1)$	98.50	99.00	96.90	96.74	98.57	**100**	93.83	94.32	96.53
$\{(8,1)\},(0.5,2)$	**99.50**	**99.50**	97.70	96.96	97.43	**100**	93.83	93.56	96.51
$\{(8,1)\},(0.5,3)$	99.00	**99.50**	97.75	96.74	98.57	**100**	93.21	92.42	96.45
$\{(8,1)\},(0.5,4)$	98.50	99.00	**98.00**	96.41	98.57	**100**	93.21	92.80	96.33
$\{(8,1)\},(0.5,5)$	98.50	99.00	97.65	97.72	98.57	**100**	93.21	92.80	96.14
$\{(8,1)\},(0.5,6)$	99.00	**99.50**	97.80	**98.04**	98.86	**100**	93.83	92.80	96.59
$\{(8,1),(8,2)\},(0.5,1)$	99.00	99.00	97.55	96.30	**99.43**	**100**	94.44	93.94	96.59
$\{(8,1),(8,2)\},(0.5,2)$	99.00	99.00	97.15	96.96	98.57	**100**	94.44	**94.70**	96.52
$\{(8,1),(8,2)\},(0.5,3)$	98.50	99.00	97.00	96.63	99.14	**100**	94.44	**94.70**	96.57
$\{(8,1),(8,2)\},(0.5,4)$	99.00	99.00	97.65	96.30	98.57	**100**	**95.06**	94.32	96.42
$\{(8,1),(8,2)\},(0.5,5)$	99.00	99.00	97.45	96.20	99.14	**100**	**95.06**	94.32	96.61
$\{(8,1),(8,2)\},(0.5,6)$	**99.50**	**99.50**	97.95	97.50	**99.43**	**100**	**95.06**	94.32	**96.65**

Note: 50-LOO and 50-4fold denote results on 50-class breakdown using leave-one-out and four cross-fold validation. Dyn35 and Dyn++ are shortened for DynTex35 and DynTex++ sub-datasets respectively.

(98.55%, 98.04%). It should be noted that their abilities are either not verified on the other challenging datasets (MEWLSP, CVLBC) or not better than ours on DynTex and DynTex++ (FoSIG).

DynTex Dataset: It can be observed from Table 4, our method obtains rate of 99.43% on *DynTex35*, the best result compared to all approaches, except MEWLSP [34] (99.71%) and CSAP-TOP [16] (100%). However, MEWLSP has not been verified on the challenging DT datasets (i.e., *Alpha*, *Beta*, *Gamma*) as well as not perform better than ours in schemes of *50-LOO* and *50-4fold*. In the meanwhile, CSAP-TOP is only little higher than ours on this scheme, but not on the others (see Table 4). In terms of DT classification on the other variants of DynTex datasets, V-BIG achieves the best performance on *Alpha* with rate of 100% among the state of the art, over 3% better than FoSIG's [17] (96.67%) and the same as that of the deep-learning-based methods, i.e., DT-CNN [1], st-TCoF [22], and D3 [10]. It is also verified that our method outperforms prominently on *Beta* and *Gamma* sub-datasets, obtaining the best results compared to all non-deep-learning methods. Specifically, with rates of 95.06% and 94.32% on *Beta* and *Gamma* respectively, V-BIG is about 2% higher than FDT's [18] with 93.21% and FoSIG's [17] with 92.42% (see Table 4).

DynTex++ Dataset: On this scheme, our method gains the highest rate of 96.65% using the settings chosen for comparison (see Table 3). This performance is the best compared to the state-of-the-art results, excluding MEWLSP [34] with 98.48%, MBSIF-TOP [3] 97.12%, and DT-CNN [1] 98.18%, 98.58% for AlexNet and GoogleNet frameworks respectively (see Table 4). However, MEWLSP and MBSIF-TOP have either not been evaluated on the variants of DynTex (i.e., *Alpha*, *Beta*, *Gamma*) or not better than ours in recognition on most of datasets. Meanwhile, DT-CNN takes huge computation to learning DT features with complex algorithms. In comparison with FoSIG's [17], ours is also higher, 96.59% versus 95.99%, with the same settings of $\{(P,R)\} = \{(8,1)\}$ and $(\sigma_1,\sigma_2) = (0.5,6)$ (see Tables 2 and 3).

Table 4. Comparison of recognition rates (%) on benchmark DT datasets

Group	Dataset	UCLA				DynTex				Dyn++
	Encoding method	50-LOO	50-4fold	9-class	8-class	Dyn35	Alpha	Beta	Gamma	
A	FDT [18]	98.50	99.00	97.70	99.35	98.86	98.33	93.21	91.67	95.31
	FD-MAP [18]	99.50	99.00	99.35	**99.57**	98.86	98.33	92.59	91.67	95.69
B	AR-LDS [30]	89.90^{N}	-	-	-	-	-	-	-	-
	KDT-MD [4]	-	97.50	-	-	-	-	-	-	-
	NLDR [26]	-	-	-	80.00	-	-	-	-	-
	Chaotic vector [36]	-	-	85.10^{N}	85.00^{N}	-	-	-	-	-
C	3D-OTF [37]	-	87.10	97.23	99.50	96.70	83.61	73.22	72.53	89.17
	WMFS [11]	-	-	97.11	96.96	-	-	-	-	-
	NLSSA [5]	-	-	-	-	-	-	-	-	92.40
	KSSA [5]	-	-	-	-	-	-	-	-	92.20
	DKSSA [5]	-	-	-	-	-	-	-	-	91.10
	DFS [38]	-	**100**	97.50	99.20	97.16	85.24	76.93	74.82	91.70
	2D+T [6]	-	-	-	-	-	85.00	67.00	63.00	-
	STLS [25]	-	99.50	97.40	99.50	98.20	89.40	80.80	79.80	94.50
D	MBSIF-TOP [3]	99.50^{N}	-	-	-	98.61^{N}	90.00^{N}	90.70^{N}	91.30^{N}	97.12^{N}
	DNGP [29]	-	-	**99.60**	99.40	-	-	-	-	93.80
E	VLBP [39]	-	89.50^{N}	96.30^{N}	91.96^{N}	81.14^{N}	-	-	-	94.98^{N}
	LBP-TOP [39]	-	94.50^{N}	96.00^{N}	93.67^{N}	92.45^{N}	98.33	88.89	84.85^{N}	94.05^{N}
	DDLBP with MJMI [28]	-	-	-	-	-	-	-	-	95.80
	CVLBP [31]	-	93.00^{N}	96.90^{N}	95.65^{N}	85.14^{N}	-	-	-	-
	HLBP [32]	95.00^{N}	95.00^{N}	98.35^{N}	97.50^{N}	98.57^{N}	-	-	-	96.28^{N}
	CLSP-TOP [15]	99.00^{N}	99.00^{N}	98.60^{N}	97.72^{N}	98.29^{N}	95.00^{N}	91.98^{N}	91.29^{N}	95.50^{N}
	MEWLSP [34]	96.50^{N}	96.50^{N}	98.55^{N}	98.04^{N}	99.71^{N}	-	-	-	98.48^{N}
	WLBPC [33]	-	96.50^{N}	97.17^{N}	97.61^{N}	-	-	-	-	95.01^{N}
	CVLBC [40]	98.50^{N}	99.00^{N}	99.20^{N}	99.02^{N}	98.86^{N}	-	-	-	91.31^{N}
	CSAP-TOP [16]	**99.50**	99.50	96.80	95.98	**100**	96.67	92.59	90.53	-
	FoSIG [17]	**99.50**	**100**	98.95	98.59	99.14	96.67	92.59	92.42	95.99
	Our V-BIG	**99.50**	99.50	97.95	97.50	99.43	**100**	95.06	94.32	96.65
F	DL-PEGASOS [8]	-	97.50	95.60	-	-	-	-	-	63.70
	PI-LBP+super hist [27]	-	**100^{N}**	98.20^{N}	-	-	-	-	-	-
	PD-LBP+super hist [27]	-	**100^{N}**	98.10^{N}	-	-	-	-	-	-
	PCA-cLBP [27]	-	-	-	-	-	-	-	-	92.40
	Orthogonal Tensor DL [24]	-	99.80	98.20	99.50	-	87.80	76.70	74.80	94.70
	Equiangular Kernel DL [23]	-	-	-	-	-	88.80	77.40	75.60	93.40
	st-TCoF [22]	-	-	-	-	-	**100***	**100***	98.11*	-
	PCANet-TOP [2]	99.50*	-	-	-	-	96.67*	90.74*	89.39*	-
	D3 [10]	-	-	-	-	-	**100***	**100***	98.11*	-
	DT-CNN-AlexNet [1]	-	99.50*	98.05*	98.48*	-	**100***	99.38*	**99.62***	98.18*
	DT-CNN-GoogleNet [1]	-	99.50*	98.35*	99.02*	-	**100***	**100***	**99.62***	**98.58***

Note: "-" means "not available". Superscript "*" indicates results using deep learning algorithms. "N" indicates rates with 1-NN classifier. 50-LOO and 50-4fold denote results on 50-class breakdown using leave-one-out and four cross-fold validation respectively. Dyn35 and Dyn++ are abbreviated for DynTex35 and DynTex++ datasets respectively. Evaluations of VLBP and LBP-TOP operators are referred to the evaluations of implementations in [32, 22]. Group A denotes *optical-flow-based approaches*, B: *model-based*, C: *geometry-based*, D: *filter-based*, E: *local-feature-based*, F: *learning-based*.

4 Conclusions

In this work, an efficient framework for DT representation has been proposed by exploiting the benefits of smooth-invariant features which are extracted from 3D Gaussian filtered volumes in order to construct a robust descriptor against the problems of illumination and noise. Evaluations for DT classification on the different benchmark datasets have verified that our method outperforms significantly compared to the state of the art. Furthermore, the experiments have also validated that encoding DT features based on the 3D filtered volumes allows to enrich more information of shape and motion cues than capturing spatio-temporal patterns based on the 2D Gaussian filtered images in the planes of a DT video [17]. In the further contexts, the advantages of these properties can be taken into account to form a descriptor with more discrimination.

References

1. Andrearczyk, V., Whelan, P.F.: Convolutional neural network on three orthogonal planes for dynamic texture classification. Pattern Recognition **76**, 36 – 49 (2018)
2. Arashloo, S.R., Amirani, M.C., Noroozi, A.: Dynamic texture representation using a deep multi-scale convolutional network. JVCIR **43**, 89 – 97 (2017)
3. Arashloo, S.R., Kittler, J.: Dynamic texture recognition using multiscale binarized statistical image features. IEEE Trans. Multimedia **16**(8), 2099–2109 (2014)
4. B. Chan, A.B., Vasconcelos, N.: Classifying video with kernel dynamic textures. In: CVPR. pp. 1–6 (2007)
5. Baktashmotlagh, M., Harandi, M.T., , A., C. Lovell, B.C., Salzmann, M.: Discriminative non-linear stationary subspace analysis for video classification. IEEE Trans. PAMI **36**(12), 2353–2366 (2014)
6. Dubois, S., Péteri, R., Ménard, M.: Characterization and recognition of dynamic textures based on the 2d+t curvelet transform. Signal, Image and Video Processing **9**(4), 819–830 (2015)
7. Fan, R., Chang, K., Hsieh, C., Wang, X., Lin, C.: LIBLINEAR: A library for large linear classification. JMLR **9**, 1871–1874 (2008)
8. Ghanem, B., Ahuja, N.: Maximum margin distance learning for dynamic texture recognition. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV. LNCS, vol. 6312, pp. 223–236 (2010)
9. Guo, Z., Zhang, L., Zhang, D.: A completed modeling of local binary pattern operator for texture classification. IEEE Trans. IP **19**(6), 1657–1663 (2010)
10. Hong, S., Ryu, J., Im, W., Yang, H.S.: D3: recognizing dynamic scenes with deep dual descriptor based on key frames and key segments. Neurocomputing **273**, 611–621 (2018)
11. Ji, H., Yang, X., Ling, H., Xu, Y.: Wavelet domain multifractal analysis for static and dynamic texture classification. IEEE Trans. IP **22**(1), 286–299 (2013)
12. Mäenpää, T., Pietikäinen, M.: Multi-scale binary patterns for texture analysis. In: SCIA. pp. 885–892 (2003)
13. Nguyen, T.P., Manzanera, A., Kropatsch, W.G., N'Guyen, X.S.: Topological attribute patterns for texture recognition. Pattern Recog. Letters **80**, 91–97 (2016)
14. Nguyen, T.P., Vu, N., Manzanera, A.: Statistical binary patterns for rotational invariant texture classification. Neurocomputing **173**, 1565–1577 (2016)
15. Nguyen, T.T., Nguyen, T.P., Bouchara, F.: Completed local structure patterns on three orthogonal planes for dynamic texture recognition. In: IPTA. pp. 1–6 (2017)
16. Nguyen, T.T., Nguyen, T.P., Bouchara, F.: Completed statistical adaptive patterns on three orthogonal planes for recognition of dynamic textures and scenes. J. Electronic Imaging **27**(05), 053044 (2018)
17. Nguyen, T.T., Nguyen, T.P., Bouchara, F.: Smooth-invariant gaussian features for dynamic texture recognition. In: ICIP (2019)
18. Nguyen, T.T., Nguyen, T.P., Bouchara, F., Nguyen, X.S.: Directional beams of dense trajectories for dynamic texture recognition. In: Blanc-Talon, J., Helbert, D., Philips, W., Popescu, D., Scheunders, P. (eds.) ACIVS. pp. 74–86 (2018)
19. Ojala, T., Pietikäinen, M., Mäenpää, T.: Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. IEEE Trans. PAMI **24**(7), 971–987 (2002)
20. Peh, C., Cheong, L.F.: Synergizing spatial and temporal texture. IEEE Trans. IP **11**(10), 1179–1191 (2002)

21. Péteri, R., Fazekas, S., Huiskes, M.J.: Dyntex: A comprehensive database of dynamic textures. Pattern Recognition Letters **31**(12), 1627–1632 (2010)
22. Qi, X., Li, C.G., Zhao, G., Hong, X., Pietikainen, M.: Dynamic texture and scene classification by transferring deep image features. Neurocomputing **171**, 1230 – 1241 (2016)
23. Quan, Y., Bao, C., Ji, H.: Equiangular kernel dictionary learning with applications to dynamic texture analysis. In: CVPR. pp. 308–316 (2016)
24. Quan, Y., Huang, Y., Ji, H.: Dynamic texture recognition via orthogonal tensor dictionary learning. In: ICCV. pp. 73–81 (2015)
25. Quan, Y., Sun, Y., Xu, Y.: Spatiotemporal lacunarity spectrum for dynamic texture classification. CVIU **165**, 85–96 (2017)
26. Ravichandran, A., Chaudhry, R., Vidal, R.: View-invariant dynamic texture recognition using a bag of dynamical systems. In: CVPR. pp. 1651–1657 (2009)
27. Ren, J., Jiang, X., Yuan, J.: Dynamic texture recognition using enhanced LBP features. In: ICASSP. pp. 2400–2404 (2013)
28. Ren, J., Jiang, X., Yuan, J., Wang, G.: Optimizing LBP structure for visual recognition using binary quadratic programming. SPL **21**(11), 1346–1350 (2014)
29. Rivera, A.R., Chae, O.: Spatiotemporal directional number transitional graph for dynamic texture recognition. IEEE Trans. PAMI **37**(10), 2146–2152 (2015)
30. Saisan, P., Doretto, G., Wu, Y.N., Soatto, S.: Dynamic texture recognition. In: CVPR. pp. 58–63 (2001)
31. Tiwari, D., Tyagi, V.: Dynamic texture recognition based on completed volume local binary pattern. MSSP **27**(2), 563–575 (2016)
32. Tiwari, D., Tyagi, V.: A novel scheme based on local binary pattern for dynamic texture recognition. CVIU **150**, 58–65 (2016)
33. Tiwari, D., Tyagi, V.: Improved weber's law based local binary pattern for dynamic texture recognition. Multimedia Tools Appl. **76**(5), 6623–6640 (2017)
34. Tiwari, D., Tyagi, V.: Dynamic texture recognition using multiresolution edge-weighted local structure pattern. Computers & Electrical Engineering **62**, 485–498 (2017)
35. Vu, N., Nguyen, T.P., Garcia, C.: Improving texture categorization with biologically-inspired filtering. Image Vision Comput. **32**(6-7), 424–436 (2014)
36. Wang, Y., Hu, S.: Chaotic features for dynamic textures recognition. Soft Computing **20**(5), 1977–1989 (2016)
37. Xu, Y., Huang, S.B., Ji, H., Fermüller, C.: Scale-space texture description on sift-like textons. CVIU **116**(9), 999–1013 (2012)
38. Xu, Y., Quan, Y., Zhang, Z., Ling, H., Ji, H.: Classifying dynamic textures via spatiotemporal fractal analysis. Pattern Recognition **48**(10), 3239–3248 (2015)
39. Zhao, G., Pietikäinen, M.: Dynamic texture recognition using local binary patterns with an application to facial expressions. IEEE Trans. PAMI **29**(6), 915–928 (2007)
40. Zhao, X., Lin, Y., Heikkilä, J.: Dynamic texture recognition using volume local binary count patterns with an application to 2d face spoofing detection. IEEE Trans. Multimedia **20**(3), 552–566 (2018)
41. Zhao, Y., Huang, D.S., Jia, W.: Completed Local Binary Count for Rotation Invariant Texture Classification. IEEE Trans. IP **21**(10), 4492–4497 (2012)