Representing dynamic textures based on polarized gradient features

Thanh Tuan Nguyen, Thanh Phuong Nguyen, Frédéric Bouchara

Representing dynamic textures based on bipolar Gaussian-gradient features

Thanh Tuan Nguyen[a,b], Thanh Phuong Nguyen[a,*], Frédéric Bouchara[a]

[a] *Université de Toulon, Aix Marseille Université, CNRS, LIS, Marseille, France*
[b] *HCMC University of Technology and Education, Faculty of IT, Ho Chi Minh City, Vietnam*

Abstract

Efficiently representing dynamic textures (DTs) is one of significant challenges for video understanding in real implementations of computer vision applications. It is partly caused by the negative influence of the well-known issues : noise, changes of environments, illumination, and scaling. To diminish those problems, a new approach for an efficient DT description is introduced in this work by addressing the following prominent concepts. Firstly, high-order 2D/3D Gaussian-gradient filtering kernels are used for filtering a given video to obtain its Gaussian-gradient-filtered images/volumes. Secondly, taking advantage of the bipolar properties of these images/volumes allows that a competent model of decomposition is proposed to decompose them into corresponding collections of robust bipolar-filtered outcomes, which are complementary for DT representation. Finally, a simple variant of Local Binary Patterns (LBPs) is applied to extract local bipolar Gaussian-gradient features from the complemented collections for constructing discriminative bipolar-based descriptors. Experimental results in DT recognition on benchmark datasets have remarkably validated the interest of our proposal.

Keywords: Dynamic texture, Gaussian filter, Bipolar feature extraction, LBP, CLBP, Video representation.

1. Introduction

Dynamic textures (DTs) are prevalent repetition of textural appearances in temporal strings [1]. Efficiently recognizing them can be one of crucial contributions for real applications in computer vision, such as human interaction [2, 3], tracking motion objects [4, 5], object and event detection [6, 7], crowded people [8], background subtraction [9, 10], etc. Due to the disorientation of DTs in their motions and the negative influence of well-known issues (e.g., environmental changes, illumination, noise, etc.) on DT description, understanding turbulent DTs in effect is really a significant challenge. To this end, many works have been attempted to utilize different techniques in order to address spatio-temporal properties for DT representation. Roughly, it can be categorized them into the following groups of approaches.

Model-based approaches: Motivated by the concept of Linear Dynamical System (LDS) [1, 11], many efforts have taken it into account video analysis in order to model disorientational motions of DTs. Chan *et al.* [12] introduced an adaptation of LDS's observation constituent with a kernel-PCA (Principal Component Analysis) in order to be able to understand dynamic properties in more complex sequences, e.g., those with motions of DTs recorded by moving camera, etc. In another work, Chan *et al.* [13] adapted LDS for concentrating characteristics of movable objects in videos by using a DT mixture (DTM) model to cluster their similarities for DT description. Also motivated by LDS's concept, other efforts have been proposed to

*Corresponding author.
Email addresses: tuannt@hcmute.edu.vn (Thanh Tuan Nguyen), tpnguyen@univ-tln.fr (Thanh Phuong Nguyen), bouchara@univ-tln.fr (Frédéric Bouchara)

be compliant with modeling DT features: bag-of-systems (BoS) [14, 15], bag-of-words (BoW) [16, 17], BoS Tree [18], etc. Besides, Hidden Markov Model (HMM) have been taken into account modeling DT motions: spatial-HMM [19], multivariate-HMM [20]. In regard to the effectiveness in representing DTs, the model-based methods have usually obtained moderate results in DT recognition since they have principally focused on the spatial features of DTs while the dynamic ones are also influential information for DT representation. In case of addressing both of which, it can be more complicated to bring the modelings into implementations in practice [14].

Optical-flow-based approaches: Towards capturing DT features in natural ways, optical-flow-based approaches have taken advantage of magnitudes and directions of normal flow for DT description: shaping and tracing the motion paths of DTs in a video [21], based on the normal vector field and criteria of sequences [22, 23], the velocity and acceleration [24], local image distortions and their relation to optical flow [25]. Recently, Nguyen *et al.* [26, 27] exploited local spatio-temporal features of motion points subject to their trajectories extracted from a given sequence. In terms of effectiveness in DT representation, the optical-flow-based approaches have obtained moderate performance because most of them supposed the brightness constancy and local smoothness in their encodings as mentioned in [28], while the textural appearances, one of crucial evidences for DT understanding, have been less involved in.

Geometry-based approaches: Based on fractal techniques, geometry-based approaches have attempted to improve DT representation by reducing the negative elements of environmental changes in their video analyses. Dynamic Fractal Spectrum (DFS) [29] and Multi-Fractal Spectrum (MFS) [30] exploited the stochastic self-similar properties and fractal patterns to encode DTs. Since the information of spatial domain, one of important keys for representing DTs, has not been exploited in MFS, Ji *et al.* [31] fixed this issue in Wavelet-based MFS (WMFS) model, where MFS is integrated along with wavelet coefficients for describing DTs in more effect. Recently, Spatio-Temporal Lacunarity Spectrum (STLS) is introduced by Quan *et al.* [32] in order to take lacunarity analysis in slices of a video into account DT description to benefit by local lacunarity-based features. Baktashmotlagh *et al.* [33] presented Stationary Subspace Analysis (SSA) in consideration of video's stationary aspects to reduce dimension for DT description. Experiments in DT classification have validated that most of geometry-based methods often have good discrimination on simple datasets, e.g., UCLA [11], while being difficult to recognize DTs on the more challenging ones, e.g., DynTex [34] and DynTex++ [35]. It may be partly due to the lack of temporal features taken into account their fractal analyses.

Learning-based approaches: For learning DTs, most of learning-based approaches are situated into two trends as follows. The first one is based on deep learning frameworks, e.g., Convolutional Neural Networks (CNNs), in order to learn DT features in various directions: DT-CNN [36] and PCANet-TOP [37] learns DT features on three orthogonal planes of a given video, Transferred ConvNet Features (TCoF) [38] learns deep structures in still images, while D3 [39] uses concepts of "key frames" and "key segments" for learning static and dynamic properties of sequences. The second trend is based on dictionary learning methods to represent DT features: based on an atom-learned dictionary [40], based on a equiangular kernel [41]. With respect to efficiency of the learning-based approaches in DT recognition, while the dictionary-based methods have been at moderate levels in "understanding" DTs with complex motions, the deep models have significant performances. However, to learn enormous parameters, most of them needed complex learning algorithms in deep architectures of neural networks, e.g., up to ∼61M parameters for AlexNet and ∼6.8M for GoogleNet learned in the DT-CNN's deep model [36], ∼80M for C3D [42],

∼88M in MSOE-two-Stream [43], etc. In this work, our proposal is able to obtain competitive performance functioned by an efficient simple framework of shallow analysis.

Local-feature-based approaches: Based on Local Binary Pattern (LBP) [44] and its derivations for capturing local textural features in image description, their benefits have brought into DT representation. For a given DT video, Zhao *et al.* [3] addressed LBP on its three consecutive frames and its three orthogonal planes to structure Volume-LBP (VLBP) and LBP-TOP patterns respectively. Motivated by these fundamental techniques, many efforts have made in order to deal with the conventional drawbacks of LBP-based variants for further enhancement of the discrimination power for DT representation: sensitivity to noise [45, 46, 47, 48], rotation-invariant problems [49], near-uniform regions [50, 51, 52, 53, 54, 55], etc.

Filter-based approaches: Most of them have mainly based on the manual and learned filters to reduce the negative impacts of noise on DT encodings. Arashloo *et al.* [56] introduced Multi-scale Binarized Statistical Image Features extracted by applying filters learned by transformation of independent component analysis (ICA) to Three Orthogonal Planes of a given video (MBSIF-TOP). In another work, 3D filters, learned by PCA, ICA, sparse filtering, and k-means clustering, are exploited in [57] to extracted 3D filtered volumes of a given video for structuring local spatio-temporal features by Completed Local Binary Pattern (CLBP) operator [58]. Recently, Nguyen *et al.* [55, 53] addressed filtering models based on moment images/volumes in order to point out filtered outcomes of variance and mean features for further discriminative improvement. In the meanwhile, Gaussian-based filterings were utilized in [46, 47, 48, 59] to mitigate noise problems before LBP-based variants were taken into account the DT feature extraction. In terms of effectiveness in DT recognition, the filter-based methods have often achieved good performances on simple DT motions (e.g., those in UCLA [11] dataset) rather than those in complicated datasets, e.g., DynTex [34] and DynTex++ [35].

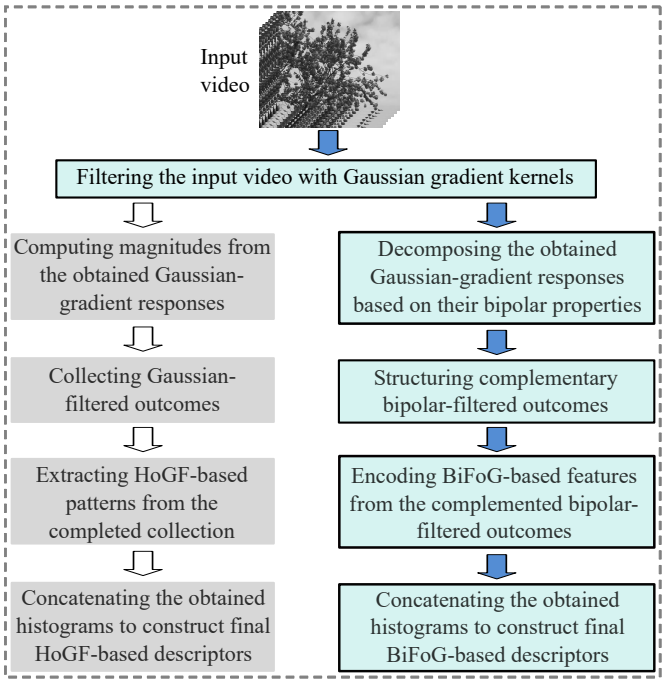


Figure 1: (Best viewed in color) A comparison of this proposal highlighted in blue background in comparison with our previous work [59], highlighted in dark background for DT representation.

As mentioned above, a pre-processing of the filters applied to video analyses for noise reduction has recently allowed to point out robust filtered outcomes for local-based DT encodings. It could be stated that addressing Gaussian-based filterings derived from the original Gaussian kernel for DT representation has made the obtained descriptors be at moderate levels of performances (referred to FoSIG [46], V-BIG [47], RUBIG [48]). This may be due to lack of complementary filtered outcomes taken into account for their DT encodings. Newly, instead of using the conventional Gaussian-filtered outcomes, our prior work [59] introduced Gaussian-gradient-based features and their magnitude information in order to construct HoGF-based descriptors, which have very good performances in comparison with state of the art. Different from those above, taking advantage of the bipolar properties of Gaussian-gradient-filtered outcomes is proposed in this work to structure discriminative BiFoG-based descriptors, which nearly have the same ability as that of

the HoGF-based ones [59] but in about two thirds smaller dimensions. Therefore, our proposed framework can be expected as one of appreciated solutions for real implementations in mobile devices and embedded systems. Figure 1 shows a comprehensive viewpoint of this proposal compared to our former work [59].

In general, our framework takes three main stages for DT representation as follows. Firstly, the high-order partial derivatives of a 2D/3D Gaussian kernel are taken into account the noise reduction in order to obtain robust Gaussian-gradient-filtered images/volumes. Secondly, a decomposing model is introduced to partition those into semi-bipolar-filtered outcomes, $\Theta^{2D/3D}$, subject to their bipolar properties. Finally, a simple LBP-based variant is utilized to extract local Bipolar Features of Gaussian-gradients (BiFoG) from the complementary components in $\Theta^{2D/3D}$. Discriminative descriptors BiFoG$^{2D/3D}$ are then pointed out correspondingly. Experimental results for DT recognition on benchmark datasets have clearly validated the interest of our proposal. In short, it can be listed our prominent contributions as

- Taking advantage of the bipolar properties of Gaussian-gradient filterings allows to point out more robust filtered outcomes for DT representation.
- An investigation of bipolar properties in high-orders of Gaussian-gradient filterings has been made so that the significant effectiveness of high-order bipolar-based features is comprehensively evaluated in comparison with those without decomposition taken into account.
- An efficient framework is presented to analyze and decompose the Gaussian-gradient images/volumes into separable bipolar-filtered features. Shallowly, discriminative BiFoG-based descriptors are constructed by locating a simple LBP-based operator on the complementary bipolar-filtered outcomes.
- In a small dimension, our BiFoG-based descriptors can achieve very good performance compared to all non-deep learning models, while ours results are also commensurate with those of deep learning methods in most of circumstances.

2. Related works

2.1. A brief of LBP and its completed model

For representing a 2D gray-scale textural image $\mathcal{I}$, Ojala *et al.* [44] introduced a simple encoding method to compute LBP patterns in consideration of gray-level differences between a center pixel $\mathbf{q}_c \in \mathcal{I}$ and its local neighbors $\{\mathbf{p}_i\}_{i=1}^{P}$ as

$$\mathrm{LBP}_{P,R}(\mathbf{q}_c) = \sum_{i=1}^{P} g\big(\mathcal{I}(\mathbf{p}_i) - \mathcal{I}(\mathbf{q}_c)\big) \times 2^{i-1} \qquad (1)$$

where $\mathcal{I}(.)$ returns the gray-level of a pixel; P is a number of concerning neighbors sampled by a local-circle region of center $\mathbf{q}_c$ and radius R; and the thresholding function $g(.)$ is defined as: $g(x) = 1$ if $x \geq 0$, and $g(x) = 0$ otherwise.

Accordingly, a histogram of 2^P bins is computed for depicting the whole textures of image $\mathcal{I}$. This leads to one of remarkable barriers for real applications in computer vision because of the curse of dimension. Hence, in order to deal with the problem, two following mappings [44] are ordinarily addressed for reductions of dimension in practice: $u2$ mapping for structuring uniform patterns (LBP^{u2}) with $P(P-1)+3$ bins, and $riu2$ mapping for rotation invariant uniform patterns (LBP^{riu2}) with only $P+2$ bins. In addition, other mappings have been also introduced for further consideration: $TAP^{\mathcal{A}}$ [60] for topological patterns, Local Binary Count [61] for an alternative of the $riu2$ ones.

For further enhancement of the discrimination power, Guo *et al.* [58] introduced a completed model of LBP (named CLBP) in consideration of forcefully capturing more LBP-based characteristics via three complementary components: CLBP$_S$ for the basic local features (i.e., the typical LBP patterns), CLBP$_M$ for the magnitude information, and CLBP$_C$ for the intensity difference of a center pixel versus the mean of all in a given image. Experiments

have shown that the description, which is formed by a 3D-joint integration of these components (i.e., $\text{CLBP}_{S/M/C}$), often obtains better performance than others. It could be referred to [58] for CLBP's components in more detail of formulas, samples of computing CLBP patterns, various combinations of those, and concerned others.

2.2. DT description based on LBP-based variants

To lay the foundation of taking LBP-based variants into account DT representation, Zhao *et al.* [3] took advantage of LBP's simple and efficient computation to introduce two kinds of DT features: volume of LBP-based patterns (VLBP) and LBP based on orthogonal planes of a given video (LBP-TOP). Therein, VLBP is proposed to describe a voxel based on its $3P$ neighbors that are located on the three consecutive frames. This leads to a crucial barrier for real applications in computer vision due to a very large dimension for video description caused by VLBP patterns with up to 2^{3P+2} bins. To mitigate the burden, LBP-TOP is introduced to encode a voxel based on its P neighbors that are placed on each of three orthogonal planes in a given video. As a result, it takes 3×2^P bins for DT representation.

Motivated by the simpleness and effectiveness of VLBP and LBP-TOP in encoding computation, many efforts have been made in order for improvement of performance by handling the conventional shortcomings of LBP-based variants as well as noise problems in DT representation: CVLBC [62] - an integration of CLBC [61] and VLBP; CVLBP [50] - a combination of CLBP [58] and VLBP; CLSP-TOP [52], CSAP-TOP [53], HLBP [51], MMDP [55], RUBIG [48], DDTP [27], etc. - dealing with issues of near uniform regions and sensitivity to noise in DT encodings.

2.3. Gaussian-based filterings

Filter-bank techniques have been exploited to denoise in texture analysis [63] for early years of 90s. Lately, Nguyen *et al.* [64, 55] introduced robust filters based on moment images/volumes to reduce noise for a LBP-based encoding in textural representation. With respect to DT description, several recent efforts [65, 56, 66, 57, 46, 47, 48, 55] have attempted to address different filter-bank approaches in order to treat the negative impact of noise issues. Therein, those [46, 47, 48], which take Gaussian-based filters into account video analysis, have noticeable effectiveness in noise reduction for local DT encoding. In general, a Gaussian filtering kernel is defined as

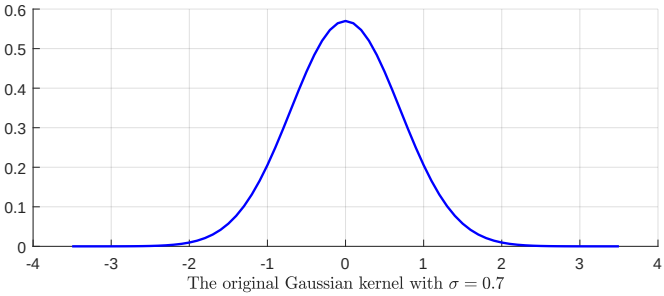


Figure 2: Profile of a 1D Gaussian kernel with a standard deviation $\sigma = 0.7$.

$$\mathrm{G}_{\sigma}^{n}(\gamma_n) = \frac{1}{(\sigma\sqrt{2\pi})^n}\exp\Big(-\frac{\lambda_1^2+\lambda_2^2+...+\lambda_n^2}{2\sigma^2}\Big) \qquad (2)$$

where $\gamma_n = \{\lambda_i\}_{i=1}^{n}$ is a set of n-dimensional spatial axes involved with the convolving operation; σ means a predefined standard deviation. The filtered results are in accordance with Gaussian distribution (see Figure 2 for the distribution of a 1D Gaussian kernel with a standard deviation $\sigma = 0.7$). Appropriately, the k^{th}-order partial derivative of $\mathrm{G}_{\sigma}^{n}(\gamma_n)$ with respect to a spatial domain λ_i is formulated as

$$\mathrm{G}_{\sigma,\partial\lambda_i^k}^{n}(\gamma_n) = \frac{\partial^k \mathrm{G}_{\sigma}^{n}(\gamma_n)}{\partial\lambda_i^k} \qquad (3)$$

in which "∂" denotes a gradient function. Figure 4 shows the distributions of different partial derivatives of a Gaussian kernel with a standard deviation $\sigma = 0.7$.

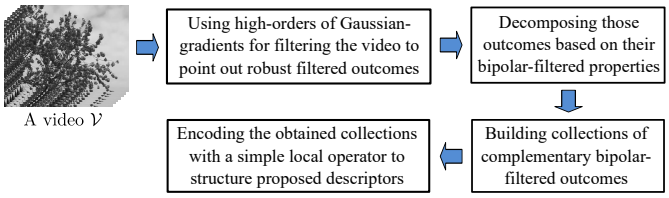


Figure 3: A general flowchart for encoding a given video $\mathcal{V}$ based on bipolar-filtered features.

3. Proposed method

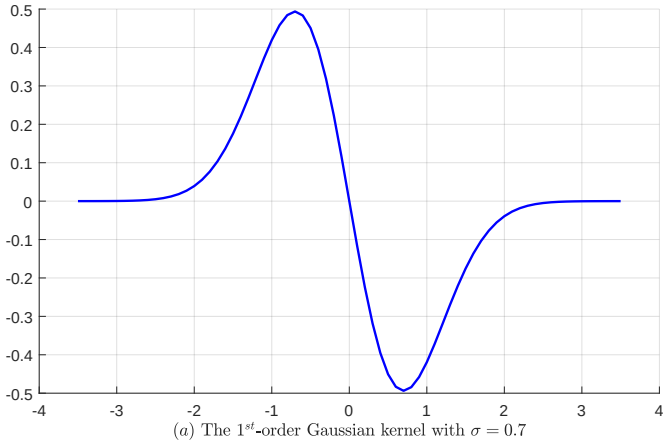
3.1. An overview of our proposal

As mentioned above, Nguyen *et al.* [46, 47, 48] have recently exploited the basic Gaussian-based filterings for noise reduction, but the obtained Gaussian-filtered outcomes have not been robust enough for DT representation yet. This has leaded to the moderate levels of performance in DT recognition due to lack of complementary filtered outcomes involved in the local encoding. Newly, a noticeable filtering based on high-order Gaussian gradients [59] is introduced and stated its efficiency in the denosing process through extracting the Gaussian-gradient-filtered features and gradient-based magnitudes for DT representation. Different from those, in this work, we present an efficient framework to take advantage of bipolar properties of the high-order Gaussian gradients to point out more robust filtered outcomes for the local DT encoding. It has nearly the same high performance as done in [59], but in a smaller dimension of output descriptors which are a considerable solution for mobile applications in practice. Generally, our proposed framework can be illustrated as in Figure 3. Accordingly, the high-order 2D/3D Gaussian-gradient filterings are addressed for noise reduction. A decomposing model is proposed to separate the obtained Gaussian-filtering responses to build collections of complementary bipolar-filtered outcomes (see Section 3.2). A local encoding framework is then presented to take the bipolar-filtered outcomes into account DT representation (see Section 3.3). As a result, it could be constructed robust BiFoG-based descriptors with very good performances on DT recognition in comparison with recent approaches. Hereunder, we express above processes in detail.

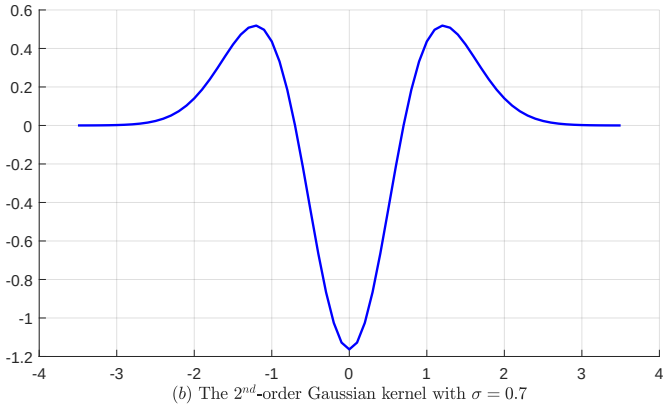
3.2. Bipolar features of Gaussian-gradient filterings

It can be deduced from Eq. (3) that there are several Gaussian-gradient kernels subject to the number of directions in γ_n taken into account a video filtering for denoising. This allows to point out more filtered outcomes for DT representation than the original Gaussian filtering due to Eq. (2). Furthermore, it can be seen from Figure 4 that the high-order Gaussian-gradient filterings could produce bipolar features allowing to decompose them into two separable filtered parts: positive and negative features which are together complementary for DT representation. In the meanwhile, only positive ones are responsed by the original Gaussian-based filtering (see Figure 2). All of those can enhance the discrimination power thanks to more complementary information extracted from the obtained bipolar-filtered outcomes in comparison with other previous efforts [46, 47, 48], where only the non-Gaussian-gradient kernels were addressed for the filterings instead of their partial derivatives as done in this work. It should be noted that the separable bipolar-filtered properties of Gaussian-gradients are addressed in this work, instead of taking the whole gradient-filtered features and their gradient-magnitudes as done in [59] (see Figure 1 for a comprehensive comparison). Hereafter, we detail the complementary collections of bipolar-filtered outcomes constructed for DT representation.

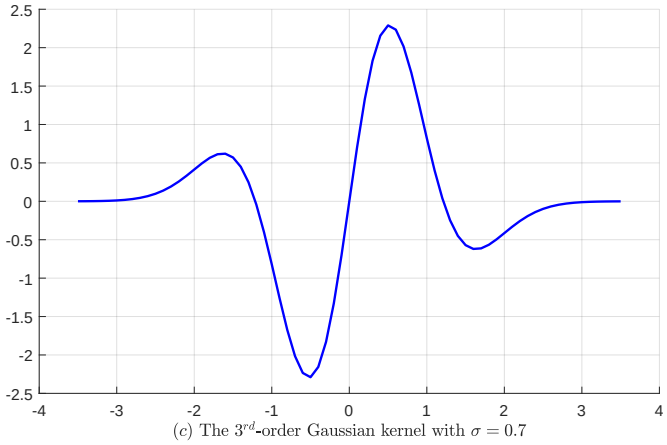
Let us consider a 2D (resp. 3D) Gaussian kernel in high-orders of its partial derivatives addressed for the filtering with a pre-defined standard deviation σ. According to the 2D filtering, two k^{th}-order Gaussian-gradient kernels $\mathrm{G}^{2D}_{\sigma,x^k}$ and $\mathrm{G}^{2D}_{\sigma,y^k}$ are taken into account as a pre-processing analysis of an image $\mathcal{I}$ to produce the following high-order



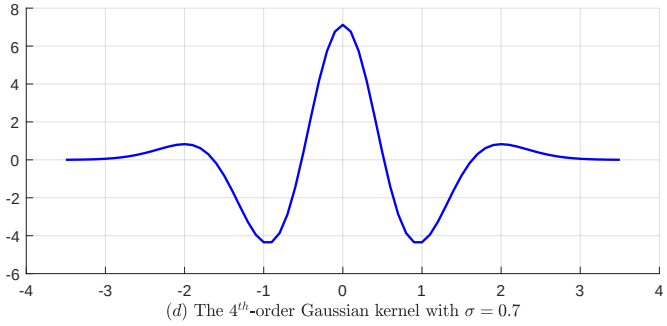
(a) The 1^{st}-order Gaussian kernel with $\sigma = 0.7$



(b) The 2^{nd}-order Gaussian kernel with $\sigma = 0.7$



(c) The 3^{rd}-order Gaussian kernel with $\sigma = 0.7$



(d) The 4^{th}-order Gaussian kernel with $\sigma = 0.7$

Figure 4: Profile of different gradients of a 1D Gaussian kernel with deviation $\sigma = 0.7$. Therein, (a): the profile for the 1^{st}-order, (b): for the 2^{nd}-order, (c): for the 3^{rd}-order, (d): for the 4^{th}- order.

gradient-filtered images as

$$\begin{cases} \mathcal{I}_{\sigma,\partial x^k} = \mathrm{G}^{2D}_{\sigma,\partial x^k}(x,y) * \mathcal{I} \\ \mathcal{I}_{\sigma,\partial y^k} = \mathrm{G}^{2D}_{\sigma,\partial y^k}(x,y) * \mathcal{I} \end{cases} \tag{4}$$

Figure 5 at line (b) shows some filtered images using the 1^{st}-order of this 2D filtering with $\sigma = 0.7$ to filter plane-images separated from a given video $\mathcal{V}$. After that, these filtered images are separably decomposed into $\{\mathcal{I}^{pos}_{\sigma,\partial x^k}, \mathcal{I}^{neg}_{\sigma,\partial x^k}\}$ for $\mathcal{I}_{\sigma,\partial x^k}$ and $\{\mathcal{I}^{pos}_{\sigma,\partial y^k}, \mathcal{I}^{neg}_{\sigma,\partial y^k}\}$ for $\mathcal{I}_{\sigma,\partial y^k}$, subject to the positive-negative properties of their pixels $\mathbf{q}$ as

$$\begin{cases} \mathcal{I}^{pos}_{\sigma,\partial x^k}(\mathbf{q}) = \mathcal{I}_{\sigma,\partial x^k}(\mathbf{q}) \text{ so that } \mathcal{I}_{\sigma,\partial x^k}(\mathbf{q}) \geq 0 \\ \mathcal{I}^{neg}_{\sigma,\partial x^k}(\mathbf{q}) = |\mathcal{I}_{\sigma,\partial x^k}(\mathbf{q})| \text{ so that } \mathcal{I}_{\sigma,\partial x^k}(\mathbf{q}) < 0 \\ \mathcal{I}^{pos}_{\sigma,\partial y^k}(\mathbf{q}) = \mathcal{I}_{\sigma,\partial y^k}(\mathbf{q}) \text{ so that } \mathcal{I}_{\sigma,\partial y^k}(\mathbf{q}) \geq 0 \\ \mathcal{I}^{neg}_{\sigma,\partial y^k}(\mathbf{q}) = |\mathcal{I}_{\sigma,\partial y^k}(\mathbf{q})| \text{ so that } \mathcal{I}_{\sigma,\partial y^k}(\mathbf{q}) < 0 \end{cases} \tag{5}$$

As a result, a complementary collection of bipolar-filtered outcomes for image $\mathcal{I}$ is structured as

$$\Theta^{2D}_{\sigma,k}(\mathcal{I}) = \{\mathcal{I}^{pos}_{\sigma,\partial x^k}, \mathcal{I}^{neg}_{\sigma,\partial x^k}, \mathcal{I}^{pos}_{\sigma,\partial y^k}, \mathcal{I}^{neg}_{\sigma,\partial y^k}\} \tag{6}$$

Figure 5 at line (c) shows an instance of bipolar filtered images decomposed subject to the 1^{st}-order 2D Gaussian-gradient-filtered images.

In respect of the 3D filtering, a given video $\mathcal{V}$ is filtered by three k^{th}-order Gaussian-gradient kernels $\mathrm{G}^{3D}_{\sigma,\partial x^k}$, $\mathrm{G}^{3D}_{\sigma,\partial y^k}$, $\mathrm{G}^{3D}_{\sigma,\partial z^k}$ in order to obtain corresponding gradient-filtered volumes as

$$\begin{cases} \mathcal{V}_{\sigma,\partial x^k} = \mathrm{G}^{3D}_{\sigma,\partial x^k}(x,y,z) * \mathcal{V} \\ \mathcal{V}_{\sigma,\partial y^k} = \mathrm{G}^{3D}_{\sigma,\partial y^k}(x,y,z) * \mathcal{V} \\ \mathcal{V}_{\sigma,\partial z^k} = \mathrm{G}^{3D}_{\sigma,\partial z^k}(x,y,z) * \mathcal{V} \end{cases} \tag{7}$$

Figure 6 at line (a) shows several filtered volumes using the 1^{st}-order of this 3D filtering with $\sigma = 0.7$ to filter a given video $\mathcal{V}$. Similar to the decomposing model of the 2D Gaussian-gradients, these volumes are then separated subject to the positive-negative properties of their voxels

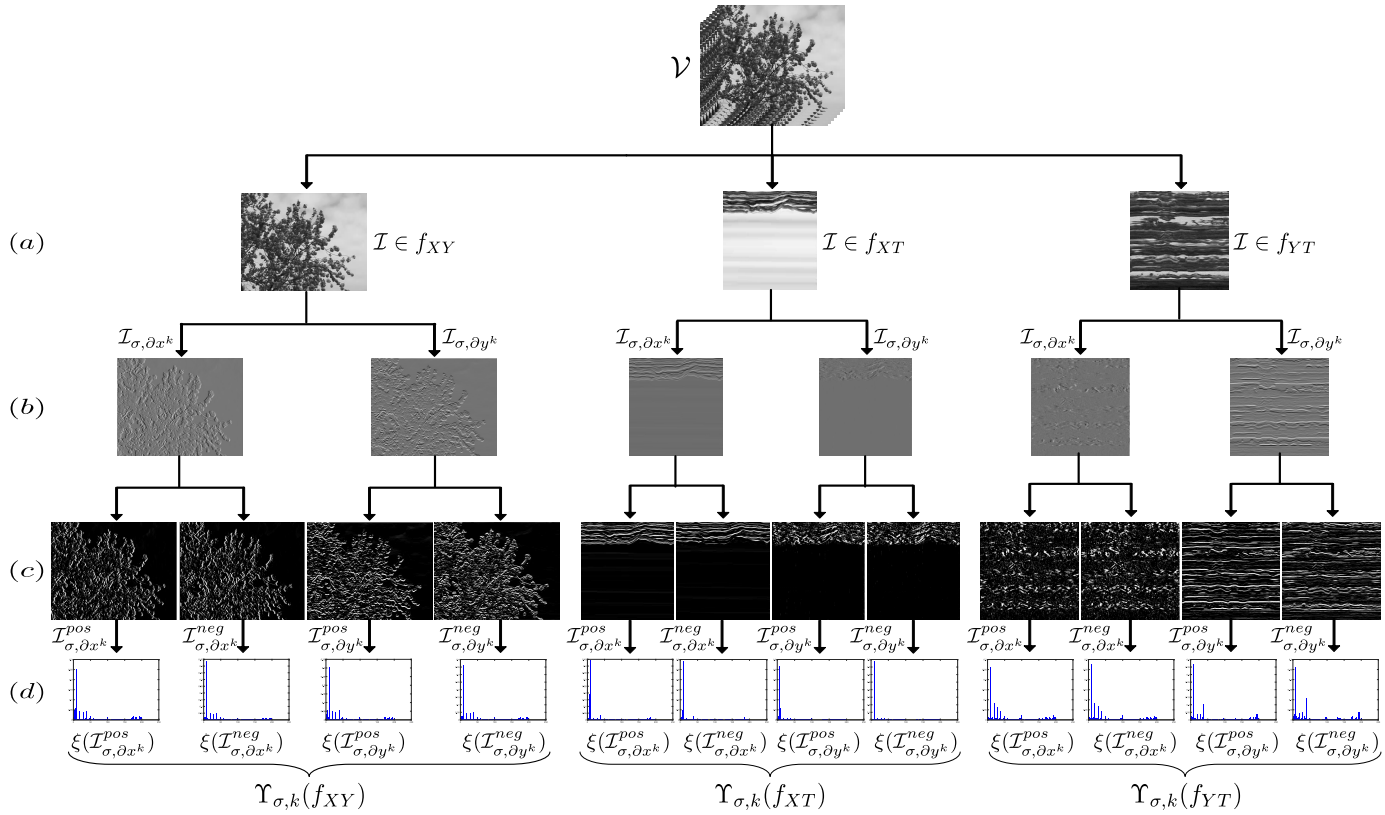


Figure 5: A framework for structuring a video $\mathcal{V}$ based on the 1^{st}-order 2D Gaussian-gradient filtering with $\sigma = 0.7$, and $\xi(.) = \text{CLBP}^{riu2}_{8,1}(.)$ to encode bipolar-filtered images of $\Theta^{2D}_{0.7,1^{st}}(\mathcal{V})$. Therein, line (a): partition $\mathcal{V}$ into collections of plane-images, line (b): the filterings with 2D Gaussian-gradient kernels, line (c): decomposing operations, line (d): encoding bipolar-filtered features for DT representation.

$\mathbf{p}$ as

$$\begin{cases} \mathcal{V}^{pos}_{\sigma,\partial x^k}(\mathbf{p}) = \mathcal{V}_{\sigma,\partial x^k}(\mathbf{p}) \text{ so that } \mathcal{V}_{\sigma,\partial x^k}(\mathbf{p}) \geq 0 \\ \mathcal{V}^{neg}_{\sigma,\partial x^k}(\mathbf{p}) = |\mathcal{V}_{\sigma,\partial x^k}(\mathbf{p})| \text{ so that } \mathcal{V}_{\sigma,\partial x^k}(\mathbf{p}) < 0 \\ \mathcal{V}^{pos}_{\sigma,\partial y^k}(\mathbf{p}) = \mathcal{V}_{\sigma,\partial y^k}(\mathbf{p}) \text{ so that } \mathcal{V}_{\sigma,\partial y^k}(\mathbf{p}) \geq 0 \\ \mathcal{V}^{neg}_{\sigma,\partial y^k}(\mathbf{p}) = |\mathcal{V}_{\sigma,\partial y^k}(\mathbf{p})| \text{ so that } \mathcal{V}_{\sigma,\partial y^k}(\mathbf{p}) < 0 \\ \mathcal{V}^{pos}_{\sigma,\partial z^k}(\mathbf{p}) = \mathcal{V}_{\sigma,\partial z^k}(\mathbf{p}) \text{ so that } \mathcal{V}_{\sigma,\partial z^k}(\mathbf{p}) \geq 0 \\ \mathcal{V}^{neg}_{\sigma,\partial z^k}(\mathbf{p}) = |\mathcal{V}_{\sigma,\partial z^k}(\mathbf{p})| \text{ so that } \mathcal{V}_{\sigma,\partial z^k}(\mathbf{p}) < 0 \end{cases} \tag{8}$$

As a result, a complementary collection of bipolar-filtered outcomes for video $\mathcal{V}$ is formed as

$$\Theta^{3D}_{\sigma,k}(\mathcal{V}) = \{\mathcal{V}^{pos}_{\sigma,\partial x^k}, \mathcal{V}^{neg}_{\sigma,\partial x^k}, \mathcal{V}^{pos}_{\sigma,\partial y^k}, \mathcal{V}^{neg}_{\sigma,\partial y^k}, \mathcal{V}^{pos}_{\sigma,\partial z^k}, \mathcal{V}^{neg}_{\sigma,\partial z^k}\} \tag{9}$$

Figure 6 at line (b) illustrates an instance of a 3D decomposition of gradient-filtered volumes using the 1^{st}-order 3D Gaussian-gradient filterings with $\sigma = 0.7$.

Hence, it could be pointed out several following benefits of the complementary bipolar-filtered components in $\Theta^{2D/3D}_{\sigma,k}$ to improve the discrimination power for DT representation compared to other Gaussian-based ones.

- It is clarified that addressing the high-order Gaussian gradients allows to produce more robust filtered components, while it is not for the conventional Gaussian kernels as done in the former works [46, 47, 48].
- Instead of exploiting the whole Gaussian-gradient-filtered features [59], decomposing them into separate bipolar-filtered ones based on their positive-negative properties grants more informative discrimination for DT representation.
- It should be noted that a separation was also introduced in [67] to split Different of Gaussians (DoG) features subject to a pre-defined meaningless threshold in order to avoid an issue of close-to-zero textural pixels caused by the responses of the DoG filtering.

Referred to Figure 4, it can be conducted that the negative impact of this problem on the DT encoding is inconsiderable due to the amplification of the high-order Gaussian-gradient filterings (also proved by experiments in Section 4.4).

- We can identify an important finding from Figure 4 : Gaussian-gradient kernels of even orders have symmetric property while that of odd orders are assymetric (semi-symmetric). This suggests that Gaussian-gradient kernels of odd and even orders are complementary and then a such combination of odd and even orders allows to enhance the discrimination power of the proposed framework.

3.3. DT representation based on $\Theta_{\sigma,k}^{2D/3D}$ features

As presented in Section 3.2, the 2D/3D decomposing models have pointed out sets of bipolar-filtered outcomes $\Theta_{\sigma,k}^{2D/3D}$, which are complementary together in potentially boosting the discrimination power. For DT representation, we proposed hereafter two efficient encoding models to structure robust descriptors corresponding to which of $\Theta_{\sigma,k}^{2D/3D}$ in high-orders is taken into account.

Proposed $\text{BiFoG}_{\sigma,\mathcal{F}}^{2D}$ ***descriptor:*** To be in accordance with the k-order 2D Gaussian-gradient filtering and its 2D decomposition, an input video $\mathcal{V}$ is split into collections of plane-images $\{f_{XY}, f_{XT}, f_{YT}\}$ subject to its three orthogonal planes $\{XY, XT, YT\}$ (see Figure 5 at line (a)). For an image $\mathcal{I} \in f_{XY}$, its bipolar-filtered outcomes $\Theta_{\sigma,k}^{2D}(\mathcal{I})$ are encoded as

$$\Upsilon_{\sigma,k}(f_{XY}) = \frac{1}{|f_{XY}|}\sum_{\mathcal{I}\in f_{XY}}\Big[\xi(\mathcal{I}_{\sigma,\partial x^k}^{pos}), \xi(\mathcal{I}_{\sigma,\partial x^k}^{neg}), \xi(\mathcal{I}_{\sigma,\partial y^k}^{pos}), \xi(\mathcal{I}_{\sigma,\partial y^k}^{neg})\Big] \tag{10}$$

where $|f_{XY}| = \textbf{card}(f_{XY})$ is the cardinality of plane-image collection f_{XY}; $\xi(.)$ stands for a simple LBP-based operator addressed for capturing local features from the bipolar-filtered images. Figure 5 at line (d) shows an instance of a local encoding with $\xi(.) = \text{CLBP}_{8,1}^{riu2}(.)$. This computation is applied to the rest of the plane-image collections f_{XT} and f_{YT} to obtain corresponding histograms $\Upsilon_{\sigma,k}(f_{XT})$ and $\Upsilon_{\sigma,k}(f_{YT})$ respectively. Consequently, the spatio-temporal Bipolar Features of high-order 2D Gaussian-gradients ($\text{BiFoG}_{\sigma,\mathcal{F}}^{2D}$) are formed for DT description as

$$\text{BiFoG}_{\sigma,\mathcal{F}}^{2D}(\mathcal{V}) = \biguplus_{k\in\mathcal{F}}\Big[\Upsilon_{\sigma,k}(f_{XY}), \Upsilon_{\sigma,k}(f_{XT}), \Upsilon_{\sigma,k}(f_{YT})\Big] \tag{11}$$

in which $\biguplus$ stands for a concatenating function of the achieved histograms; $\mathcal{F}$ denotes a set of Gaussian gradients taken into account a filtering, e.g., $\mathcal{F} = \{1^{st}, 2^{nd}\}$ means that the first and second partial derivatives of a 2D Gaussian kernel have being concerned with the DT encoding.

Proposed $\text{BiFoG}_{\sigma,\mathcal{F}}^{3D}$ ***descriptor:*** As presented in Section 3.2, for a given video $\mathcal{V}$, it could be pointed out 6 complementary bipolar-filtered volumes when a k-order 3D Gaussian-gradient kernel is taken into account the filtering. Those volumes are taken into account local analysis to construct a robust descriptor as follows. For a bipolar-filtered volume $\mathcal{V}_{\sigma,\partial x^k}^{pos} \in \Theta_{\sigma,k}^{3D}(\mathcal{V})$, subject to its three orthogonal planes $\{XY, XT, YT\}$, $\mathcal{V}_{\sigma,\partial x^k}^{pos}$ is firstly split into collections of plane-images $\{f'_{XY}, f'_{XT}, f'_{YT}\}$. In respect of a collection f'_{XY}, its histogram is computed and normalized as

$$\Gamma_{\mathcal{V}_{\sigma,\partial x^k}^{pos}}(f'_{XY}) = \frac{1}{|f'_{XY}|}\sum_{\mathcal{I}\in f'_{XY}}\xi(\mathcal{I}) \tag{12}$$

where $|f'_{XY}| = \textbf{card}(f'_{XY})$ is the cardinality of plane-image collection f'_{XY}; $\xi(.)$ stands for a simple LBP-based operator addressed for capturing local features from the bipolar-filtered volume. Similarly, we also have computations for the plane-images in f'_{XT} and f'_{YT} to obtain probability distributions $\Gamma_{\mathcal{V}_{\sigma,\partial x^k}^{pos}}(f'_{XT})$ and $\Gamma_{\mathcal{V}_{\sigma,\partial x^k}^{pos}}(f'_{YT})$ respectively. Accordingly, local spatio-temporal features for representing $\mathcal{V}_{\sigma,\partial x^k}^{pos}$ could be structured by concatenating all of those in a natural way as

$$\Psi(\mathcal{V}_{\sigma,\partial x^k}^{pos}) = \big[\Gamma_{\mathcal{V}_{\sigma,\partial x^k}^{pos}}(f'_{XY}), \Gamma_{\mathcal{V}_{\sigma,\partial x^k}^{pos}}(f'_{XT}), \Gamma_{\mathcal{V}_{\sigma,\partial x^k}^{pos}}(f'_{YT})\big] \tag{13}$$

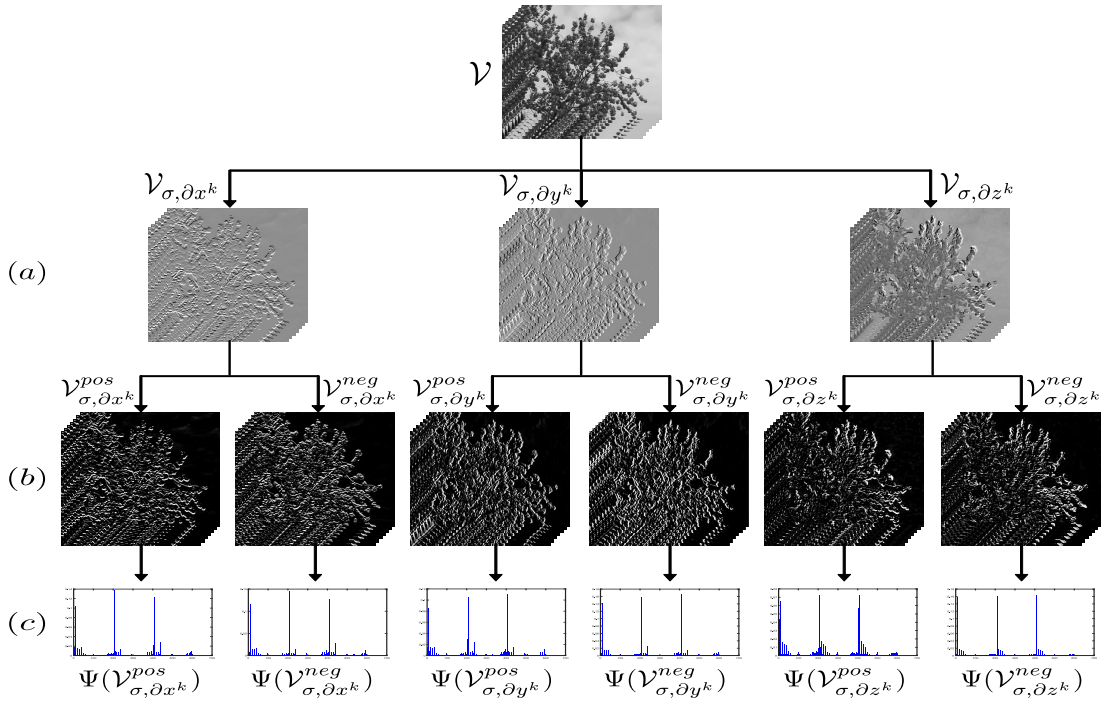


Figure 6: A framework for structuring a video $\mathcal{V}$ based on the 1^{st}-order 3D Gaussian-gradient filtering with $\sigma = 0.7$, and $\xi(.) = \text{CLBP}^{riu2}_{8,1}(.)$ to encode bipolar-filtered volumes of $\Theta^{3D}_{0.7,1^{st}}(\mathcal{V})$. Therein, line (a): the filterings with 3D Gaussian-gradient kernels, line (b): decomposing operations, line (c): encoding bipolar-filtered features for DT representation.

The computation of $\mathcal{V}^{pos}_{\sigma,\partial x^k}$ is then applied to the rest of the bipolar-filtered volumes in $\Theta^{3D}_{\sigma,k}(\mathcal{V})$ to figure out corresponding representations, i.e., $\Psi(\mathcal{V}^{neg}_{\sigma,\partial x^k})$, $\Psi(\mathcal{V}^{pos}_{\sigma,\partial y^k})$, $\Psi(\mathcal{V}^{neg}_{\sigma,\partial y^k})$, $\Psi(\mathcal{V}^{pos}_{\sigma,\partial z^k})$, and $\Psi(\mathcal{V}^{neg}_{\sigma,\partial z^k})$. As a result, the Bipolar Features of high-order 3D Gaussian-gradients ($\text{BiFoG}^{3D}_{\sigma,\mathcal{F}}$) are formed for DT description as

$$\text{BiFoG}^{3D}_{\sigma,\mathcal{F}}(\mathcal{V}) = \biguplus_{k\in\mathcal{F}} \Big[\Psi(\mathcal{V}^{pos}_{\sigma,\partial x^k}), \Psi(\mathcal{V}^{neg}_{\sigma,\partial x^k}), \Psi(\mathcal{V}^{pos}_{\sigma,\partial y^k}), \Psi(\mathcal{V}^{neg}_{\sigma,\partial y^k}), \Psi(\mathcal{V}^{pos}_{\sigma,\partial z^k}), \Psi(\mathcal{V}^{neg}_{\sigma,\partial z^k})\Big] \tag{14}$$

in which $\biguplus$ stands for a concatenating function of the obtained histograms; $\mathcal{F}$ denotes a set of Gaussian gradients taken into account a filtering, e.g., $\mathcal{F} = \{1^{st}, 2^{nd}\}$ means that the first and second partial derivatives of a 3D Gaussian kernel have being concerned with the DT encoding. Figure 6 at line (c) shows an instance of the entire encoding process for video $\mathcal{V}$ with specific parameters of $\mathcal{F} = \{1^{st}\}$, $\sigma = 0.7$, and $\xi(.) = \text{CLBP}^{riu2}_{8,1}(.)$.

Furthermore, in order to thoroughly assess the prominent performance of BiFoG-based features, we also present two more local non-$\text{BiFoG}^{2D/3D}$ descriptors based on the non-decomposed features, i.e., $\{\mathcal{I}_{\sigma,\partial x^k}, \mathcal{I}_{\sigma,\partial y^k}\}$ (see Eq. (4)) and $\{\mathcal{V}_{\sigma,\partial x^k}, \mathcal{V}_{\sigma,\partial y^k}, \mathcal{V}_{\sigma,\partial z^k}\}$ (see Eq. (7)) without decomposition involved in the DT encoding. Following to the construction of $\text{BiFoG}^{2D/3D}$, the non-BiFoG-based descriptors are computed as

$$\text{non-BiFoG}^{2D}_{\sigma,\mathcal{F}}(\mathcal{V}) = \biguplus_{k\in\mathcal{F}} \big[\Lambda_{\sigma,k}(f_{XY}), \Lambda_{\sigma,k}(f_{XT}), \Lambda_{\sigma,k}(f_{YT})\big] \tag{15}$$

where $\Lambda(.)$ is formulated as the same Eq. (10) but for encoding the non-decomposed images $\{\mathcal{I}_{\sigma,\partial x^k}, \mathcal{I}_{\sigma,\partial y^k}\}$. Similarly, $\text{non-BiFoG}^{3D}_{\sigma,\mathcal{F}}(\mathcal{V})$ is formed by using the non-decomposed volumes $\{\mathcal{V}_{\sigma,\partial x^k}, \mathcal{V}_{\sigma,\partial y^k}, \mathcal{V}_{\sigma,\partial z^k}\}$ as

$$\text{non-BiFoG}^{3D}_{\sigma,\mathcal{F}}(\mathcal{V}) = \biguplus_{k\in\mathcal{F}} \big[\Psi(\mathcal{V}_{\sigma,\partial x^k}), \Psi(\mathcal{V}_{\sigma,\partial y^k}), \Psi(\mathcal{V}_{\sigma,\partial z^k})\big] \tag{16}$$

It should be noted that the non-$\text{BiFoG}^{2D/3D}$ descriptors are identical to $\text{HoGF}^{2D/3D}$ [59], excluding the magnitude features of Gaussian gradients.

Consequently, based on the structure of our proposed

BiFoG$^{2D/3D}$ above, it could be pointed out several following beneficial properties to enhance the discrimination power compared to other Gaussian-based descriptions.

- Thanks to taking advantage of the separately bipolar-filtered components $\Theta_{\sigma,k}^{2D/3D}$, BiFoG$^{2D/3D}$ could be represented by more complementary textural features to improve the performance, while it is partly not for the conventional Gaussian-based descriptors: FoSIG [46], V-BIG [47], RUBIG [48]. This advantage is also consistent with the non-BiFoG$^{2D/3D}$ ones for a Gaussian-gradient filtering involved in the encoding.
- Both symmetric and asymmetric features correspondingly extracted from the odd and even orders are also decomposed and combined, which allows to exploit more forceful filtered patterns for DT representation.
- Different from descriptors HoGF$^{2D/3D}$ [59] where non-decomposed Gaussian gradients and their magnitudes were exploited, our BiFoG$^{2D/3D}$ ones are just based on the separately bipolar-filtered features of Gaussian gradients. The proposed descriptors have nearly the same performance of HoGF$^{2D/3D}$ but in about two thirds smaller dimension (see Section 4.5 for more thorough evaluations).
- Structuring BiFoG$^{2D/3D}$ is presented as an adaptive encoding model. It means that it is able to apply different LBP-based operators to the local encoding phase in consideration of further enhancement.

4. Experiments and evaluations

4.1. Datasets and protocols

In this section, benchmark DT datasets along with protocols, which are addressed for evaluating performance of our proposed BiFoG$^{2D/3D}$ descriptors, are explained in detail. After that, Table 1 is drawn out to present the short of their properties for a quick reference.

UCLA dataset: Saisan *et al.* [11] introduced UCLA, a simple dataset with 200 DT sequences. Each of sequence

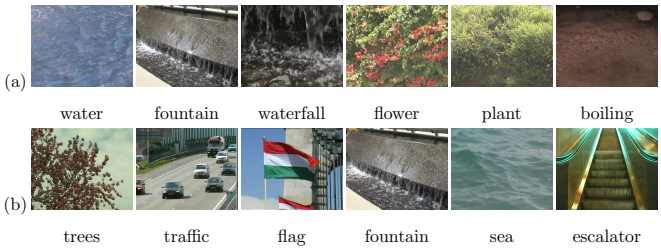


Figure 7: Several DT samples of UCLA (a) and DynTex (b).

in UCLA was recorded in dimension of $110 \times 160 \times 75$ resolution to describe turbulent motions of dynamic scenes such as fountain, boiling water, fire, plant, flower, waterfall, etc. (see Figure 7 at line (a) for some instances). In terms of DT recognition, UCLA is frequently composed in challenging schemes as follows:

- *50-class:* 50 categories are composed by taking 4 sequences for each from 200 DT videos of UCLA. Leave-one-out (LOO) and four cross-fold validation (4-fold) are two main protocols for recognizing DTs on this scheme [56, 51, 47, 46].
- *9-class and 8-class:* *9-class* scheme [14, 29] is composed by 200 DT videos of UCLA and arranged into 9 groups as "fountains(20)", "flowers(12)", "boiling water(8)", "sea(12)", "smoke(4)", "water(12)", "plants(108)", "waterfall(16)", and "fire(8)". Therein, the numbers in parentheses express the cardinalities of the corresponding groups. Due to the dominant cardinality of "plants" with 108 samples, the group is discarded to form a more challenging scheme, named *8-class* [14, 29]. 50%/50% protocol is often located for evaluating performance in DT recognition on two these scenarios. It means that a half of DT samples in each group is randomly picked out for the training and the rest for testing [35, 51, 46]. The mean of results in 20 trials is reported as a final rate for each scheme.

DynTex dataset: Péteri *et al.* [34] introduced DynTex, a more challenging dataset than UCLA, with more than 650 videos which their turbulent motions of DTs were recorded in various environmental conditions (see Figure

7 at line (b) for some of instances). Following the experimental settings in [37, 68, 47], a version of DynTex's videos in dimension of $352 \times 288 \times 250$ resolution is usually used for evaluating performances on DT classification. Accordingly, the LOO protocol is addressed for the evaluations on the following schemes:

- *DynTex35:* It is composed by taking 35 DynTex's sequences into account a clipping operation as follows. each of which is divided into 8 non-overlapping sub-videos using random partition points on X, Y, and T axes, but not half of them. For instance, a partition could be $x = 170$, $y = 130$, and $t = 100$ as addressed in [3, 56, 51, 53, 46]. Besides, two more sub-videos are obtained by only addressing the partition subject to the T axis (i.e., $t = 100$). In short, there are 10 sub-videos for a splitting process located as a category. Consequently, splitting 35 DynTex's sequences points out a challenging scheme with 35 categories.
- *Alpha:* 60 DT videos are picked out from DynTex to arranged into three categories, named "grass", "sea", and "trees". Each of which consists of 20 DT videos.
- *Beta:* It has 10 categories arranged by 162 DynTex's videos: "calm water(20)", "sea(20)", "smoke(16)", "rotation(10)", "vegetation(20)", "trees(20)", flags(20)", "fountains(20)", "escalator(7)", and "traffic(9)". Therein, the numbers in parentheses express the cardinalities of the corresponding categories.
- *Gamma:* It also has 10 categories arranged by 264 DynTex's videos: "grass(23)", "traffic(9)", "flowers(29)", "sea(38)", "naked trees(25)", "calm water(30)", "flags(31)", "foliage(35)", "escalator(7)", and "fountains(37)". Therein, the numbers in parentheses express the cardinalities of the corresponding categories.

DynTex++ dataset: Ghanem *et al.* [35] took 345 sequences from the DynTex collection in order to split and filter them so that the obtained results only consisted of major textural motions of DTs. Those were then arranged into 36 categories, i.e., 3600 sub-videos in total. As the experimental protocol in [35, 56, 69], 50%/50% protocol is addressed for evaluating performance in DT recognition. It means that a half of DT samples in each category is randomly picked out for the training and the rest for testing. The mean of results in 20 trials is reported as a final rate.

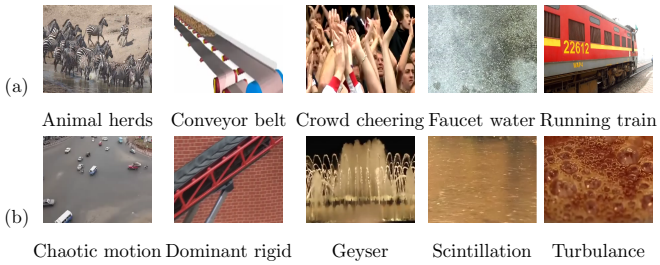


Figure 8: Some samples of DTDB, (a): Appearance, (b): Dynamics.

DTDB dataset: Hadji *et al.* [43] recently introduced Dynamic Texture DataBase (DTDB), a large scale collection of DT videos for principally evaluating performances of proposals in learning DT features based on deep-neural networks. Its over 10000 DT sequences with a total of ∼3.5M frames was collected from different sources: websites, handled cameras, etc. For DT recognition, two challenging scenarios of DTDB, *Dynamics* and *Appearance*, were arranged as follows.

- *Appearance* scheme consists of 45 categories, where its DT videos were selected from DTDB so that they mostly focus on features of spatial appearance, i.e., independent of dynamics (see Figure 8(a) for some instances).
- *Dynamics* scheme consists of 18 categories. Contrary to *Appearance*, its DT videos, selected from DTDB, just include features of dynamics, i.e., independent of spatial appearance (see Figure 8(b) for some instances).

Following protocol in [43], for each category, 70% of its samples is randomly picked out training and the rest (30%) for testing. The final result is then reported by the average rate of 10 repetitions.

Table 1: A brief of main properties of DT datasets.

Dataset	Sub-dataset	#Videos	Resolution	#Classes	Protocol
UCLA	50-class	200	$110 \times 160 \times 75$	50	LOO and 4-fold
	9-class	200	$110 \times 160 \times 75$	9	50%/50%
	8-class	92	$110 \times 160 \times 75$	8	50%/50%
DynTex	DynTex35	350	different dimensions	10	LOO
	Alpha	60	$352 \times 288 \times 250$	3	LOO
	Beta	162	$352 \times 288 \times 250$	10	LOO
	Gamma	264	$352 \times 288 \times 250$	10	LOO
DynTex++		3600	$50 \times 50 \times 50$	36	50%/50%
DTDB	Dynamics	> 10000	different dimensions	18	70%/30%
	Appearance	> 9000	different dimensions	45	70%/30%

4.2. Parameters for experimental implementation

For computing bipolar-filtered outcomes $\Theta_{\sigma,k}^{2D/3D}$*:* To construct the collections of the complementary bipolar-filtered outcomes $\Theta_{\sigma,k}^{2D/3D}$, we investigate partial derivatives ($\mathrm{G}_{\sigma,\partial\lambda_i^k}^{2D/3D}$) of a Gaussian filtering kernel in four levels of orders, i.e., $\mathcal{F} \subseteq \{1^{st}, 2^{nd}, 3^{rd}, 4^{th}\}$. Direction axes $x, y, z \in [-3\sigma, 3\sigma]$ are addressed for convolving operations of the filterings, where standard deviation σ is empirically conducted as $\sigma \in \{0.5, 0.7, 1, 1.3, 1.5, 2\}$.

For structuring $\mathrm{BiFoG}_{\sigma,\mathcal{F}}^{2D/3D}$ *descriptors:* In order to extract local spatio-temporal features from the bipolar-filtered outcomes $\Theta_{\sigma,k}^{2D/3D}$ for our proposed BiFoG-based descriptors, we simply apply CLBP[1] [58], one of the most popular local operators, to the local encoding with the 3D-joint setting of $riu2$ mapping and a supporting region $(P, R) = (8, 1)$. It means $\xi = \mathrm{CLBP}_{8,1}^{\mathrm{riu2}}$ corresponding to $\mathcal{H}_\xi = 2(P+2)^2$ bins for representing a pattern, where P denotes a number of neighbors concerned with the computation. Accordingly, it generally takes $3 \times |\Theta_{\sigma,k}^{2D/3D}| \times |\mathcal{F}| \times \mathcal{H}_\xi$ bins for $\mathrm{BiFoG}_{\sigma,\mathcal{F}}^{2D/3D}$ descriptors, where $|\Theta_{\sigma,k}^{2D/3D}| = \mathbf{card}(\Theta_{\sigma,k}^{2D/3D})$ denotes the number of bipolar-filtered images/volumes taken into account the DT representation; $|\mathcal{F}| = \mathbf{card}(\mathcal{F})$ denotes the number of k-orders in $\mathcal{F}$ involved with a multi-order analysis. For instance, in single-order analysis, i.e., $|\mathcal{F}| = 1$, the dimensions are $3 \times |\Theta_{\sigma,k}^{2D}| \times |\mathcal{F}| \times \mathcal{H}_\xi = 2400$ bins for $\mathrm{BiFoG}_{\sigma,\mathcal{F}}^{2D}$ and $3 \times |\Theta_{\sigma,k}^{3D}| \times |\mathcal{F}| \times \mathcal{H}_\xi = 3600$ bins for $\mathrm{BiFoG}_{\sigma,\mathcal{F}}^{3D}$. Table 2 demonstrates a comprehensive comparison between dimension of the proposed $\mathrm{BiFoG}^{2D/3D}$ descriptors and that of other LBP-based ones.

For structuring $\text{non-BiFoG}_{\sigma,\mathcal{F}}^{2D/3D}$ *descriptors:* In order to make an objective evaluation in comparison with our $\mathrm{BiFoG}_{\sigma,\mathcal{F}}^{2D/3D}$ descriptors, the identical settings should be situated for the DT encodings, i.e., the simple LBP-based variant $\xi = \mathrm{CLBP}_{8,1}^{\mathrm{riu2}}$. Accordingly, for a single-order analysis, it takes $6 \times |\mathcal{F}| \times \mathcal{H}_\xi = 1200$ bins for $\text{non-BiFoG}_{\sigma,\mathcal{F}}^{2D}$ and $9 \times |\mathcal{F}| \times \mathcal{H}_\xi = 1800$ bins for $\text{non-BiFoG}_{\sigma,\mathcal{F}}^{3D}$.

For DT classification: we utilize the linear multi-class SVM classifier implemented by LIBLINEAR [73] in order to measure performances of our proposed BiFoG-based descriptors. To be simple in the operation, the default parameters of the classifier are regarded in this work.

4.3. Complexity of our proposed $\mathrm{BiFoG}_{\sigma,\mathcal{F}}^{2D/3D}$ *descriptors*

In general, it could be stated that the complexity of structuring $\mathrm{BiFoG}_{\sigma,\mathcal{F}}^{2D/3D}$ is the same level as that of $\mathrm{HoGF}^{2D/3D}$ [59], $\text{non-BiFoG}_{\sigma,\mathcal{F}}^{2D/3D}$, and other local-feature-based approaches. Indeed, let us consider $\mathcal{Q}_{\mathrm{LBP}} = \mathcal{O}(P \times \mathcal{H} \times \mathcal{W})$ as the cost of structuring a textual $\mathcal{H} \times \mathcal{W}$ image based on the basic LBP [44] operator with P local concerning neighbors. Zhao *et al.* [3] introduced LBP-TOP for DT representation, in which its LBP-based features are encoded on the three orthogonal planes $\{XY, XT, YT\}$ of a given video $\mathcal{V}$, i.e., $\mathcal{Q}_{\text{LBP-TOP}} \approx 3 \times \mathcal{T} \times \mathcal{Q}_{\mathrm{LBP}}$, where $\mathcal{T}$ denotes the number of $\mathcal{V}$'s frames. Since three LBP-based components of CLBP [58] is computed independently (refer to [58] for detail of their formulas), it can be deduced that CLBP's cost for structuring a textural image is estimated as $\mathcal{Q}_{\mathrm{CLBP}} \approx 3 \times \mathcal{Q}_{\mathrm{LBP}}$. As presented in Sections 3.2 and 3.3, it can be seen that the complexity of our proposed $\mathrm{BiFoG}_{\sigma,\mathcal{F}}^{2D/3D}$ descriptors

[1] CLBP [58] operator is utilized in this work for a purpose of unity in implementing and evaluating the efficiency of the bipolar-filtered features for DT description. Definitely, it could address other robust local-based operators for further enhancement in practice, e.g., LDP-based [70, 55], CLBC [61], LRP [48], LVP-based [71, 27], MRELBP [72], etc.

Table 2: A comparison of LBP-based descriptors' dimension.

Method	#bins	$P = 8$
LBP-TOP^{u2} [3]	$3(P(P-1)+3)$	177
VLBP [3]	2^{3P+2}	-
CVLBP [50]	$3 \times 2^{3P+2}$	-
HLBP [51]	6×2^{P}	1536
CLSP-TOP^{riu2} [52]	$6(P+2)^2$	600
WLBPC [74]	6×2^{P}	1536
MEWLSP [69]	6×2^{P}	1536
CVLBC [62]	$2(3P+3)^2$	1458
CSAP-TOP^{riu2}[53]	$12(P+2)^2$	1200
FDT^{u2} [26]	$216P((P-1)+3)$	12744
FD-MAP$^{u2}_{L=2}$ [26]	$216P((P-1)+3))+16$	12760
HILOP [54]	$3P(P(P-1)+3)$	1416
FoSIG [46]	$12(P+2)^2$	1200
V-BIG [47]	$12(P+2)^2$	1200
RUBIG [48]	$36(P+2)^2$	3600
HoGF^{2D} [59]	$36(P+2)^2$	3600
HoGF^{3D} [59]	$48(P+2)^2$	4800
non-BiFoG$^{2D}_{\sigma,1^{st}}$	$12(P+2)^2$	1200
non-BiFoG$^{3D}_{\sigma,1^{st}}$	$18(P+2)^2$	1800
Our BiFoG$^{\mathbf{2D}}_{\boldsymbol{\sigma},\mathbf{1^{st}}}$	$24(P+2)^2$	2400
Our BiFoG$^{\mathbf{3D}}_{\boldsymbol{\sigma},\mathbf{1^{st}}}$	$36(P+2)^2$	3600

Note: P denotes the concerned neighbors. "-" means "not available". Dimension of all above descriptors is referred to their basic parameters used for encoding a given video.

relies on three main computing parts: Gaussian-gradient filtering, decomposing, and local encoding. Hereunder, we express those in detail.

Complexity of BiFoG^{2D}*:* Due to the computational independence of the Gaussian-gradient filtering and decomposing processes, it could be deduced from Eq. (10) that the computational cost to encode $\{f_{XY}\}$'s plane-images is evaluated as

$$\mathcal{Q}_{\Upsilon_{XY}} \approx 4 \times |f_{XY}| \times \mathcal{Q}_{\xi} + \mathcal{Q}_{\mathrm{G}^{2D}} + \mathcal{Q}_{\mathrm{S}^{2D}} \quad (17)$$

where $\mathcal{Q}_{\xi}$ is the cost of local encoding function $\xi(.)$; $\mathcal{Q}_{\mathrm{G}^{2D}}$ is the cost of a 2D Gaussian-gradient filtering (see Equation 4); and $\mathcal{Q}_{\mathrm{S}^{2D}}$ is the cost of a 2D splitting model (see Eq. (5)). According to Eq. (11), the complexity of BiFoG^{2D} could be estimated as

$$\mathcal{Q}_{\mathrm{BiFoG}^{2D}} = |\mathcal{F}| \times (\mathcal{Q}_{\Upsilon_{XY}} + \mathcal{Q}_{\Upsilon_{XT}} + \mathcal{Q}_{\Upsilon_{YT}}) \quad (18)$$

Due to the linear and separable properties of the Gaussian-gradient filtering and the splitting processes, as well as the much smaller value of $|\mathcal{F}|$ (e.g., $|\mathcal{F}| = 2$ for two orders of Gaussian gradients), $\mathcal{Q}_{\mathrm{G}^{2D}}$, $\mathcal{Q}_{\mathrm{S}^{2D}}$, and $|\mathcal{F}|$ can be disregarded. In addition, $max(|f_{XY}|, |f_{XT}|, |f_{YT}|) \approx \mathcal{T}$; $\xi(.) = \mathrm{CLBP}^{\mathrm{riu2}}_{8,1}(.)$ as located in Section 4.2. Consequently,

$$\mathcal{Q}_{\mathrm{BiFoG}^{2D}} \approx \mathcal{Q}_{\mathrm{CLBP}} \times \mathcal{T} \approx \mathcal{O}(P \times \mathcal{H} \times \mathcal{W} \times \mathcal{T}) \quad (19)$$

Complexity of BiFoG^{3D}*:* It can be seen from Eq. (12) that the cost for encoding a collection $\{f'_{XY}\}$ of a bipolar-filtered volume in Θ^{3D} is $\mathcal{Q}_{\Gamma_{XY}} = |f'_{XY}| \times \mathcal{Q}_{\xi}$. Hence, according to Eq. (13),

$$\mathcal{Q}_{\Psi} \approx max(\mathcal{Q}_{\Gamma_{XY}}, \mathcal{Q}_{\Gamma_{XT}}, \mathcal{Q}_{\Gamma_{YT}}) + \mathcal{Q}_{\mathrm{G}^{3D}} + \mathcal{Q}_{\mathrm{S}^{3D}} \quad (20)$$

where $\mathcal{Q}_{\mathrm{G}^{3D}}$ is the cost of a 3D Gaussian-gradient filtering (see Eq. (7)); and $\mathcal{Q}_{\mathrm{S}^{3D}}$ is the cost of a 3D splitting model (see Eq. (8)). Based on Eq. (14), the complexity of BiFoG^{3D} could be estimated as

$$\mathcal{Q}_{\mathrm{BiFoG}^{3D}} = 6 \times |\mathcal{F}| \times \mathcal{Q}_{\Psi} \approx \mathcal{O}(P \times \mathcal{H} \times \mathcal{W} \times \mathcal{T}) \quad (21)$$

This is since $\mathcal{Q}_{\mathrm{G}^{3D}}$, $\mathcal{Q}_{\mathrm{S}^{3D}}$, and $|\mathcal{F}|$ can be ignored while $max(|f'_{XY}|, |f'_{XT}|, |f'_{YT}|) \approx \mathcal{T}$ and $\xi(.) = \mathrm{CLBP}^{\mathrm{riu2}}_{8,1}(.)$ as located in Section 4.2.

As above analyzed, both our proposed descriptors has mostly the same computational cost, generally stipulated for $\mathcal{Q}_{\mathrm{BiFoG}}$ in further evaluations. Similarly, the complexity of non-BiFoG$^{2D/3D}$ is estimated as $\mathcal{Q}_{\text{non-BiFoG}} \approx \mathcal{O}(P \times \mathcal{H} \times \mathcal{W} \times \mathcal{T})$ due to Eqs. (15) and (16). Therefore, it can be asserted that our $\mathcal{Q}_{\mathrm{BiFoG}}$ is equivalent to that of local Gaussian-filtering-based descriptors:

Table 3: Comparison of processing time of encoding a video with $50 \times 50 \times 50$ dimension in DynTex++ dataset.

Descriptor	$\{\sigma\}/\{(\sigma,\sigma')\}$	Derivative	$\{(P,R)\}$	Mapping	Runtime
VLBP [3]	-	-	$\{(4,1)\}$	-	≈ 0.22s
LBP-TOP [3]	-	-	$\{(8,1)\}$	u2	≈ 0.15s
CLSP-TOP [52]	-	-	$\{(8,1)\}$	riu2	≈ 0.27s
CSAP-TOP [53]	-	-	$\{(8,1)\}$	riu2	≈ 0.50s
FoSIG [46]	$\{(0.5,6)\}$	-	$\{(8,1)\}$	riu2	≈ 0.37s
V-BIG [47]	$\{(0.5,6)\}$	-	$\{(8,1)\}$	riu2	≈ 0.35s
HoGF^{2D} [59]	$\{\sigma=1\}$	1^{st}-order	$\{(8,1)\}$	riu2	≈ 0.54s
HoGF^{3D} [59]	$\{\sigma=1\}$	1^{st}-order	$\{(8,1)\}$	riu2	≈ 0.70s
non-BiFoG^{2D}	$\{\sigma=1\}$	1^{st}-order	$\{(8,1)\}$	riu2	≈ 0.37s
non-BiFoG^{3D}	$\{\sigma=1\}$	1^{st}-order	$\{(8,1)\}$	riu2	≈ 0.55s
Our BiFoG^{2D}	$\{\sigma=1\}$	1^{st}-order	$\{(8,1)\}$	riu2	≈ 0.69s
Our BiFoG^{3D}	$\{\sigma=1\}$	1^{st}-order	$\{(8,1)\}$	riu2	≈ 0.91s

Note: "-" means "not available". Most of runtime results are referred to the implementation of Nguyen *et al.* [59]. It should be noted that all these implementations use raw MATLAB codes in single-threading, which are run on a 64-bit Linux desktop of CPU Core i7 3.4GHz 16G RAM.

non-BiFoG$^{2D/3D}$, HoGF$^{2D/3D}$ [59], V-BIG [47], FoSIG [46], RUBIG [48], as well as that of other LBP-based ones: CVLBC [62], CSAP-TOP [53], CVLBP [50], VLBP [3], etc. (refer to these works for computations in detail). In the meantime, our abilities on DT recognition are nearly the same order as those of HoGF$^{2D/3D}$ but in smaller dimension, while being significantly better than those of the others. (see Sections 4.4, 4.5, and 4.6 for comprehensive evaluations). In regard to encoding time, Table 3 shows ours in comparison with that of other approaches.

4.4. Contribution of separately bipolar-filtered features

As mentioned in Section 3.2, the decomposing model have decomposed the Gaussian-gradient filtered components into the crucial collections $\Theta^{2D/3D}_{\sigma,k}$ of separably bipolar-filtered outcomes. It could be verified that the accompaniment of their positive-negative filtered features allows to forcefully boost the discrimination power in DT representation. Indeed, Table 4 shows significant contributions of each kind of these features. Especially, the performance of DT recognition on the challenging schemes *Gamma* and DynTex++ is boosted up to about 3.5% on average when integrating all complementary elements of

Table 4: Recognition rate (%) of each 1^{st}-order filtered component in $\Theta^{3D}_{1.0,1^{st}}$ and its contribution for performance of BiFoG$^{3D}_{1.0,1^{st}}$.

	UCLA		DynTex			Dyn++
Component	50-LOO	50-4fold	Alpha	Beta	Gamma	
$\mathcal{V}^{pos}_{1.0,\partial x^1}$	99.00	99.00	95.00	92.59	91.67	91.91
$\mathcal{V}^{neg}_{1.0,\partial x^1}$	**100**	**100**	95.00	91.36	92.42	91.63
$\mathcal{V}^{pos}_{1.0,\partial y^1}$	**100**	**100**	95.00	90.74	88.64	91.53
$\mathcal{V}^{neg}_{1.0,\partial y^1}$	**100**	**100**	96.67	91.98	91.29	90.52
$\mathcal{V}^{pos}_{1.0,\partial z^1}$	94.00	95.00	98.33	95.06	89.77	93.02
$\mathcal{V}^{neg}_{1.0,\partial z^1}$	95.50	97.50	**100**	94.44	90.91	93.08
BiFoG$^{3D}_{1.0,1^{st}}$	**100**	**100**	**100**	**95.68**	**95.83**	**97.38**

Note: *50-LOO* and *50-4fold* mean results on *50-class breakdown* using leave-one-out and four cross-fold validation. Dyn++ is shortened for DynTex++.

$\Theta^{3D}_{1.0,1^{st}}$. In addition, the higher orders are taken into account the Gaussian-gradient filterings, the more filtered components are pointed out to be fed into the decomposing model for producing bipolar-filtered outcomes, i.e., the more positive-negative properties are allocated to enrich discriminative information of appearance and motion clues for DT description.

4.5. Performing assessments of BiFoG-based descriptors

We comprehensively discuss the noteworthy effectiveness of the high-order BiFoG-based features compared to the non-BiFoG-based ones as well as other Gaussian-gradient-magnitude-based features, i.e., HoGF$^{2D/3D}$ [59]. In general, experiments for DT recognition on benchmark datasets have validated that the BiFoG's spatio-temporal patterns are in better discrimination than non-BiFoG's. It has proved the benefits of bipolar-filtered features in dealing with the well-known issues of DT representation. As deliberated in Sections 3.3 and 4.4, it could be stated the following significant points relied on the experimental results:

- The higher level of standard deviation σ is taken into account the filterings, the less effectiveness in denosing has been responded. This also leads to a negative influence on the bipolar-filtered features partly due to the weaker appearance features caused by the amplifications of σ. Indeed, we have empirically pointed

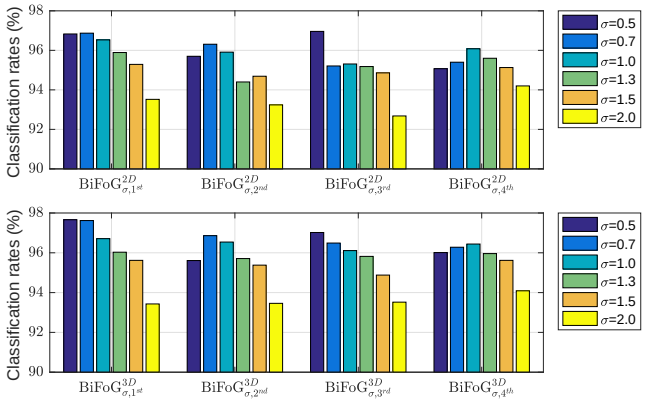


Figure 9: (Best viewed in color) A sharp reduction of BiFoG$^{2D/3D}_{\sigma,\mathcal{F}}$'s performances when an increase of σ is from 0.5 to 2.

Table 5: Comparison of encoding parameters of our BiFoG$^{2D/3D}$ and HoGF$^{2D/3D}$ [59], which are recommended for real implementations and comparison with existing methods.

Filtering	Descriptor	$\{\sigma_i\}$	$\{(P,R)\}$	Single-order	Multi-order	#Bins
$G^{2D}_{\sigma,\partial\lambda^k_i}$	HoGF^{2D}[59]	{1.0}	{(8, 1), (8, 2)}	-	$\{2^{nd}, 3^{rd}\}$	7200
	BiFoG^{2D}	{1.0}	{(8, 1)}	-	$\{1^{st}, 2^{nd}\}$	4800
	BiFoG^{2D}	{1.0}	{(8, 1)}	-	$\{1^{st}, 4^{th}\}$	4800
$G^{3D}_{\sigma,\partial\lambda^k_i}$	HoGF^{3D}[59]	{1.0}	{(8, 1), (8, 2)}	-	$\{3^{rd}, 4^{th}\}$	9600
	BiFoG^{3D}	{1.0}	{(8, 1)}	1^{st}-order	-	3600
	BiFoG^{3D}	{1.0}	{(8, 1)}	-	$\{1^{st}, 2^{nd}\}$	7200
	BiFoG^{3D}	{1.0}	{(8, 1)}	-	$\{1^{st}, 4^{th}\}$	7200

Note: "-" means "not available".

out that rates of DT classification on DynTex++ have decreased by from 1% to 3% corresponding to an increasing of σ from 0.5 to 2 (see Figure 9). Hence, in the remains of evaluations, we just report the results of BiFoG$^{2D/3D}_{\sigma,\mathcal{F}}$ implemented by $\sigma \in \{0.5, 0.7, 1\}$ as presented in Tables 6 and 7.

- It could be clarified that some single-scales of high-orders obtain good results with just a small dimension of 3600 bins, e.g., BiFoG$^{3D}_{0.7,1^{st}}$ and BiFoG$^{3D}_{1.0,2^{nd}}$ (see Table 6), but the betters come from the combinations of 2-scale orders (see Table 7). Therein, those of odd-even orders achieve the higher and more stable rates thanks to taking advantage of both symmetric and asymmetric patterns extracted from the odd and even bipolar-filtered outcomes respectively.
- The experimental results have validated that addressing the bipolar-filtered features obtains the better performances than the original Gaussian-gradient ones, i.e., the non-BiFoG-based characteristics. Figure 10 shows the higher rates of single-scale BiFoG$^{2D/3D}_{\sigma,\mathcal{F}}$ descriptors in DT recognition compared to those of non-BiFoG$^{2D/3D}_{\sigma,\mathcal{F}}$ in pairs of the filtering orders. This substantiates the interest of our proposal of the decomposing model involved in the DT encoding. In addition, it should be noted that despite being little inferior to the single-scale non-BiFoG-based descriptors in some circumstances, BiFoG$^{2D/3D}$ in 2-scale orders have significant rates compared to the 2-scale orders of non-BiFoG$^{2D/3D}$ (see Figure 11 for an instance of DT recognition on *Gamma*, the challenging DynTex's scheme). It has proved that integrating of high-orders is crucial in booting the discrimination of bipolar-filtered features.
- In regard to comparison with performances of HoGF$^{2D/3D}$ [59], ours have nearly the same levels (see Tables 8 and 9) but in two thirds smaller dimension (see Table 5). Particularly, in DT recognition on *Gamma* and DynTex++, our BiFoG$^{3D}_{1.0,\{1^{st},4^{th}\}}$ respectively obtains 97.73% and 97.94%, better than HoGF^{3D} with rates of 97.53% and 97.63% (see Table 9). Hence, the BiFoG-based features could be considered as one of potential solutions to deploy for functions in mobile devices and embedded systems. It should be recalled that the non-BiFoG-based features (see Section 3.3) could be regarded as the corresponding HoGF-based ones without the magnitude properties which are computed from different filtered components in the same level of Gaussian gradients. It means that the high performance of HoGF$^{2D/3D}$ [59] is partly thanks to the crucial contribution of these magnitudes. It may further enhance the discrimination power for real applications when combining the magnitude features with the BiFoG-based ones.

Briefly, based on above comprehensive assessments, we

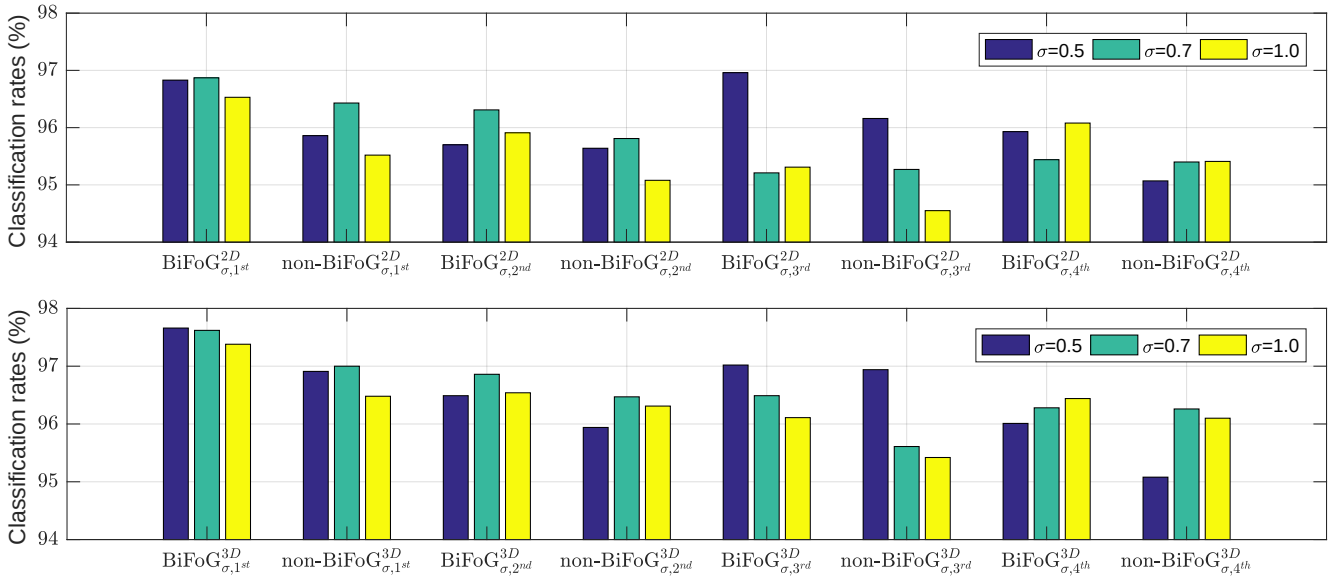


Figure 10: (Best viewed in color) The better performances of BiFoG$^{2D/3D}_{\sigma,\mathcal{F}}$ compared to non-BiFoG$^{2D/3D}_{\sigma,\mathcal{F}}$'s in classifying DTs on DynTex++.

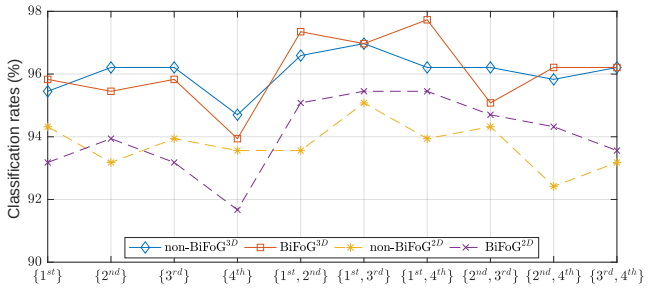


Figure 11: Performances on *Gamma* of BiFoG$^{2D/3D}_{\sigma,\mathcal{F}}$ in single and 2-scale orders of Gaussian-gradient filterings with $\sigma = 1$ compared to those of non-BiFoG$^{2D/3D}_{\sigma,\mathcal{F}}$.

recommend that the settings in Table 5 should be addressed for implementing applications in practice as well as for comparing to existing methods. Accordingly, the single-scale BiFoG$^{3D}_{1.0,1^{st}}$ descriptor can be considered to meet demands of small dimension, while those based on the 2-scale orders are for strict requirements of high precision on challenging datasets. Hereinafter, the performances of our BiFoG$^{2D/3D}_{\sigma,\mathcal{F}}$ descriptors are thoroughly discussed in comparison with state of the art. Therein, if their settings are not mentioned explicitly, the default parameters in Table 5 are referred to.

4.6. Comprehensive comparison to state of the art

In general, it can be observed from Tables 8 and 9 that performances of our proposed BiFoG-based descriptors in the smaller dimension are nearly the same as those of the HoGF-based [59] ones. These results are significantly better than all non-deep-learning methods. In the meantime, ours are also better than deep-learning-based approaches on UCLA while being close to those on DynTex, DynTex++, and DTDB. This is definitely thanks to the leverage contribution of the separably bipolar-based features. Hereinafter, we discuss in detail evaluations of those on each benchmark dataset.

4.6.1. Recognition on UCLA

It can be observed from Tables 6 and 7 that our BiFoG-based descriptors obtain substantial rates on UCLA's schemes compared to state of the art, including the deep-learning methods, i.e., DT-CNN [36] and PCANet-TOP [37]. Therein, those based on the decomposition of 3D Gaussian-gradient-filtered outcomes (refer to Eq. (8)) achieve better performance in more stability. In terms of settings for comprehensive comparison, our BiFoG$^{2D/3D}$ descriptors achieve the best rates of 100% on both *50-*

Table 6: Classification rates (%) on DT benchmark datasets of BiFoG$^{2D/3D}$ descriptors.

Dataset		UCLA								DynTex								DynTex++	
		50-LOO		50-4fold		9-class		8-class		DynTex35		Alpha		Beta		Gamma			
Order	$\{\sigma_i\}$	$\Theta^{2D}_{\sigma,k}$	$\Theta^{3D}_{\sigma,k}$	$\Theta^{2D}_{\sigma,k}$	$\Theta^{3D}_{\sigma,k}$	$\Theta^{2D}_{\sigma,k}$	$\Theta^{3D}_{\sigma,k}$	$\Theta^{2D}_{\sigma,k}$	$\Theta^{3D}_{\sigma,k}$	$\Theta^{2D}_{\sigma,k}$	$\Theta^{3D}_{\sigma,k}$	$\Theta^{2D}_{\sigma,k}$	$\Theta^{3D}_{\sigma,k}$	$\Theta^{2D}_{\sigma,k}$	$\Theta^{3D}_{\sigma,k}$	$\Theta^{2D}_{\sigma,k}$	$\Theta^{3D}_{\sigma,k}$	$\Theta^{2D}_{\sigma,k}$	$\Theta^{3D}_{\sigma,k}$
1^{st}	{0.5}	**100**	**100**	**100**	**100**	97.80	98.65	98.80	98.15	98.00	97.71	95.00	95.00	95.68	95.06	92.42	94.70	96.83	**97.66**
	{0.7}	**100**	**100**	**100**	**100**	98.65	99.60	98.26	97.17	**99.14**	**99.14**	98.33	98.33	94.44	95.06	**93.94**	95.08	96.87	97.62
	{1.0}	**100**	**100**	**100**	**100**	98.85	99.10	98.70	98.80	**99.14**	98.86	**100**	**100**	**96.30**	95.68	93.18	**95.83**	96.53	97.38
2^{nd}	{0.5}	**100**	**100**	**100**	**100**	98.45	98.85	98.26	98.37	98.29	98.86	98.33	98.33	91.98	91.98	91.67	92.05	95.70	96.49
	{0.7}	**100**	**100**	**100**	**100**	99.55	98.95	97.07	98.48	96.29	98.00	98.33	98.33	93.21	95.68	93.56	95.08	96.31	96.86
	{1.0}	**100**	**100**	**100**	**100**	**99.85**	99.50	99.24	98.26	**99.14**	98.86	98.33	98.33	93.83	95.68	**93.94**	95.45	95.91	96.54
3^{rd}	{0.5}	**100**	**100**	**100**	**100**	98.45	97.85	97.39	99.35	98.29	98.86	96.67	96.67	93.21	**96.30**	**93.94**	93.18	**96.96**	97.02
	{0.7}	99.50	**100**	99.50	**100**	98.95	98.00	97.39	97.07	96.86	98.00	98.33	98.33	91.98	93.83	92.80	93.56	95.21	96.49
	{1.0}	**100**	**100**	**100**	**100**	99.50	99.55	97.93	99.13	98.00	98.57	96.67	96.67	92.59	95.06	93.18	**95.83**	95.31	96.11
4^{th}	{0.5}	**100**	**100**	**100**	**100**	97.70	98.70	97.17	97.07	98.00	98.29	95.00	96.67	93.21	92.59	89.39	91.67	95.93	96.01
	{0.7}	97.50	**100**	98.00	**100**	98.25	99.45	97.50	99.35	98.00	98.86	98.33	96.67	91.36	93.21	92.42	91.67	95.44	96.28
	{1.0}	**100**	**100**	**100**	**100**	99.15	**99.75**	**99.35**	**99.57**	98.57	**99.14**	98.33	98.33	93.21	91.98	91.67	93.94	96.08	96.44

Note: Columns $\Theta^{2D/3D}_{\sigma,k}$ indicate that BiFoG$^{2D/3D}$ are encoded using the bipolar-filtered outcomes $\Theta^{2D/3D}_{\sigma,k}$ correspondingly. 50-LOO and 50-4fold denote results on 50-class breakdown using leave-one-out and four cross-fold validation.

Table 7: Classification rates (%) on DT benchmark datasets of 2-scale BiFoG$^{2D/3D}$ descriptors.

Dataset		UCLA								DynTex								DynTex++	
		50-LOO		50-4fold		9-class		8-class		DynTex35		Alpha		Beta		Gamma			
Multi-order	$\{\sigma_i\}$	$\Theta^{2D}_{\sigma,k}$	$\Theta^{3D}_{\sigma,k}$	$\Theta^{2D}_{\sigma,k}$	$\Theta^{3D}_{\sigma,k}$	$\Theta^{2D}_{\sigma,k}$	$\Theta^{3D}_{\sigma,k}$	$\Theta^{2D}_{\sigma,k}$	$\Theta^{3D}_{\sigma,k}$	$\Theta^{2D}_{\sigma,k}$	$\Theta^{3D}_{\sigma,k}$	$\Theta^{2D}_{\sigma,k}$	$\Theta^{3D}_{\sigma,k}$	$\Theta^{2D}_{\sigma,k}$	$\Theta^{3D}_{\sigma,k}$	$\Theta^{2D}_{\sigma,k}$	$\Theta^{3D}_{\sigma,k}$	$\Theta^{2D}_{\sigma,k}$	$\Theta^{3D}_{\sigma,k}$
$\{1^{st}, 2^{nd}\}$	{0.5}	**100**	**100**	**100**	**100**	98.95	98.90	98.26	97.39	98.57	98.86	96.67	96.67	94.44	96.30	94.32	95.45	97.38	97.62
	{0.7}	**100**	**100**	**100**	**100**	99.55	99.20	98.59	97.39	98.57	98.86	98.33	98.33	94.44	95.06	**95.45**	95.08	**97.67**	97.49
	{1.0}	**100**	**100**	**100**	**100**	99.30	99.30	**99.13**	99.13	**99.71**	99.43	98.33	98.33	95.06	**96.91**	95.08	97.35	97.56	97.68
$\{1^{st}, 3^{rd}\}$	{0.5}	**100**	**100**	**100**	**100**	97.40	98.90	98.91	97.28	98.00	98.86	96.67	96.67	94.44	95.68	93.94	95.08	97.43	97.57
	{0.7}	**100**	**100**	**100**	**100**	98.85	98.75	**99.13**	97.93	97.43	98.57	98.33	98.83	92.59	95.06	94.32	95.08	97.23	97.56
	{1.0}	**100**	**100**	**100**	**100**	98.20	99.10	98.91	98.37	99.14	99.43	**100**	**100**	93.21	95.68	**95.45**	96.97	96.83	97.37
$\{1^{st}, 4^{th}\}$	{0.5}	**100**	**100**	**100**	**100**	97.30	98.00	97.39	97.93	98.57	98.86	95.00	96.67	94.44	95.06	92.80	93.56	97.05	97.71
	{0.7}	99.50	**100**	99.50	**100**	99.30	98.80	97.83	98.26	99.14	98.86	98.33	96.67	93.21	96.30	93.94	95.45	97.38	97.67
	{1.0}	**100**	**100**	**100**	**100**	99.15	99.55	98.80	**99.35**	99.14	99.14	98.33	98.33	**95.68**	95.68	**95.45**	**97.73**	97.29	**97.94**
$\{2^{nd}, 3^{rd}\}$	{0.5}	**100**	**100**	**100**	**100**	98.95	99.20	98.26	97.28	98.29	**99.71**	96.67	96.67	**95.68**	**96.91**	93.94	93.94	97.31	97.46
	{0.7}	**100**	**100**	**100**	**100**	99.40	99.10	97.39	97.39	97.43	98.29	98.33	98.33	92.59	95.06	93.18	95.08	96.22	97.22
	{1.0}	**100**	**100**	**100**	**100**	99.35	98.50	98.37	98.70	99.43	99.43	98.33	98.33	93.21	95.06	94.70	95.08	96.37	97.17
$\{2^{nd}, 4^{th}\}$	{0.5}	**100**	**100**	**100**	**100**	98.05	98.70	98.91	97.61	98.29	98.57	93.33	96.67	91.97	91.98	91.67	92.05	96.44	97.05
	{0.7}	99.50	**100**	99.50	**100**	99.35	98.15	**99.13**	97.50	98.00	98.29	98.33	98.33	92.59	95.68	92.05	92.80	96.67	97.09
	{1.0}	**100**	**100**	**100**	**100**	98.85	99.15	98.26	98.48	98.86	99.14	98.33	98.33	94.44	96.30	94.32	96.21	96.76	97.03
$\{3^{rd}, 4^{th}\}$	{0.5}	**100**	**100**	**100**	**100**	98.45	98.80	97.72	98.04	98.29	**99.71**	96.67	96.67	94.44	95.68	93.56	93.56	97.09	97.48
	{0.7}	99.00	**100**	99.00	**100**	99.75	98.70	98.26	98.48	98.00	98.00	98.33	98.33	92.59	93.83	92.80	92.80	96.07	97.04
	{1.0}	**100**	**100**	**100**	**100**	**99.80**	**99.60**	98.80	99.24	98.00	99.43	96.67	98.33	91.98	95.68	93.56	96.21	96.44	97.49

Note: Columns $\Theta^{2D/3D}_{\sigma,k}$ indicate that BiFoG$^{2D/3D}$ are encoded using the bipolar-filtered outcomes $\Theta^{2D/3D}_{\sigma,k}$ correspondingly. 50-LOO and 50-4fold denote results on 50-class breakdown using leave-one-out and four cross-fold validation.

LOO and *50-4fold* schemes of UCLA, while also performing well on *9-class* and *8-class* scenarios. More specifically, BiFoG$^{3D}_{1.0,\{1^{st},4^{th}\}}$ obtains 99.55% on *9-class* and 99.35% *8-class*, not reach to the top due to the confused recognition as pointed out in Figures 12 and 13 respectively. According to that, BiFoG$^{3D}_{1.0,\{1^{st},4^{th}\}}$ has mainly confused in mutually recognizing DTs in videos of "Fountains", "Flowers", and "Fire" on both *9-class* and *8-class*. In addition, it is worth noting that several local-feature-based methods such as CVLBC (99.2%, 99.02%) [62], FD-MAP (99.35%, 99.57%) [26], and DNGP (99.6%, 99.4%) [28] also have the same levels in comparison with ours but they are not on

Table 8: Comparison of recognition rates (%) on UCLA.

Group	Encoding method	50-LOO	50-4fold	9-class	8-class
A	FDT [26]	98.50	99.00	97.70	99.35
	FD-MAP [26]	99.50	99.00	99.35	**99.57**
	DDTP [27]	99.00	99.50	98.75	98.04
B	AR-LDS [11]	89.90^{N}	-	-	-
	Chaotic vector [16]	-	-	85.10^{N}	85.00^{N}
C	3D-OTF [30]	-	87.10	97.23	99.50
	DFS [75]	-	**100**	97.50	99.20
	STLS [32]	-	99.50	97.40	99.50
D	MBSIF-TOP [56]	99.50^{N}	-	-	-
	DNGP [28]	-	-	**99.60**	99.40
	B3DF_SMC [57]	99.50^{N}	99.50^{N}	98.85^{N}	98.15^{N}
E	VLBP [3]	-	89.50^{N}	96.30^{N}	91.96^{N}
	LBP-TOP [3]	-	94.50^{N}	96.00^{N}	93.67^{N}
	CVLBP [50]	-	93.00^{N}	96.90^{N}	95.65^{N}
	HLBP [51]	95.00^{N}	95.00^{N}	98.35^{N}	97.50^{N}
	CLSP-TOP [52]	99.00^{N}	99.00^{N}	98.60^{N}	97.72^{N}
	MEWLSP [69]	96.50^{N}	96.50^{N}	98.55^{N}	98.04^{N}
	WLBPC [74]	-	96.50^{N}	97.17^{N}	97.61^{N}
	CVLBC [62]	98.50^{N}	99.00^{N}	99.20^{N}	99.02^{N}
	CSAP-TOP [53]	99.50	99.50	96.80	95.98
	FoSIG [46]	99.50	**100**	98.95	98.59
	V-BIG [47]	99.50	99.50	97.95	97.50
	HILOP [54]	99.50	99.50	97.80	96.30
	$\text{MMDP}_{D\_M/C}$ [55]	**100**	**100**	98.70	98.70
	$\text{MEMDP}_{D\_M/C}$ [55]	**100**	**100**	98.90	98.70
	RUBIG [48]	**100**	**100**	99.20	99.13
	HoGF^{2D} [59]	**100**	**100**	99.20	98.91
	HoGF^{3D} [59]	**100**	**100**	99.25	**99.57**
	Our $\textbf{BiFoG}^{2D}_{1.0,\{1^{st},2^{nd}\}}$	**100**	**100**	99.30	99.13
	Our $\textbf{BiFoG}^{2D}_{1.0,\{1^{st},4^{th}\}}$	**100**	**100**	99.15	98.80
	Our $\textbf{BiFoG}^{3D}_{1.0,1^{st}}$	**100**	**100**	99.10	98.80
	Our $\textbf{BiFoG}^{3D}_{1.0,\{1^{st},2^{nd}\}}$	**100**	**100**	99.30	99.13
	Our $\textbf{BiFoG}^{3D}_{1.0,\{1^{st},4^{th}\}}$	**100**	**100**	99.55	99.35
F	DL-PEGASOS [35]	-	97.50	95.60	-
	PI-LBP+super hist [45]	-	**100**^{N}	98.20^{N}	-
	Orthogonal Tensor DL [40]	-	99.80	98.20	99.50
	PCANet-TOP [37]	99.50*	-	-	-
	DT-CNN-AlexNet [36]	-	99.50*	98.05*	98.48*
	DT-CNN-GoogleNet [36]	-	99.50*	98.35*	99.02*

Note: "-" means "not available". Superscript "*" indicates results using deep learning algorithms. "N" is rate with 1-NN classifier. 50-LOO and 50-4fold are results on 50-class breakdown using leave-one-out and four cross-fold validation respectively. Group A is *optical-flow-based methods*, B: *model-based*, C: *geometry-based*, D: *filter-based*, E: *local-feature-based*, F: *learning-based*.

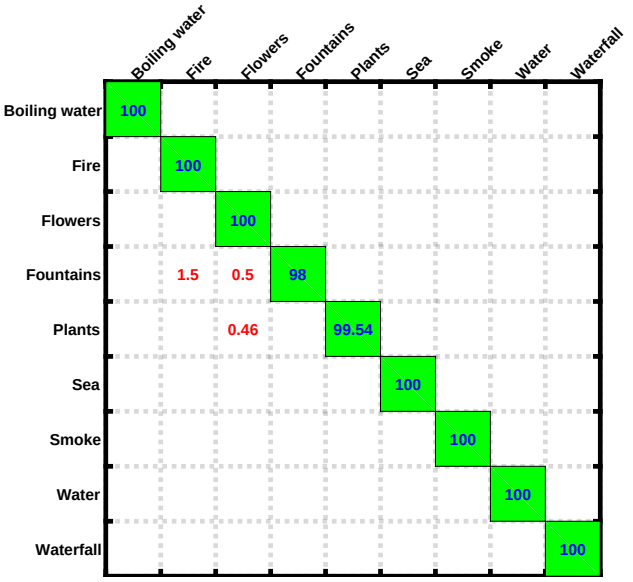


Figure 12: Confusion matrix (%) of $\text{BiFoG}^{3D}_{1.0,\{1^{st},4^{th}\}}$ on *9-class*.

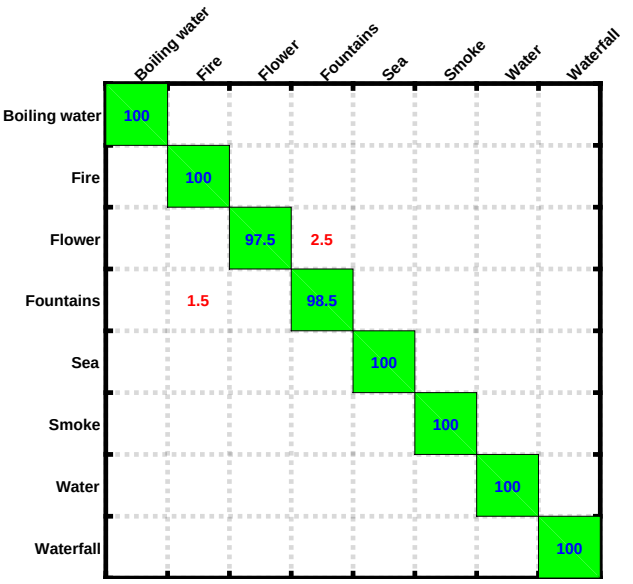


Figure 13: Confusion matrix (%) of $\text{BiFoG}^{3D}_{1.0,\{1^{st},4^{th}\}}$ on *8-class*.

50-LOO and *50-4fold* of UCLA (see Table 8), as well as not on DynTex and DynTex++ (see Table 9). Therein, CVLBC and DNGP have been deficient in the crucial validations on the challenging schemes of DynTex, i.e., *Alpha*, *Beta*, and *Gamma*. In the meanwhile, the non-Gaussian-gradient-based methods such as V-BIG [47], FoSIG [46], and RUBIG [48] only perform well in understanding simple motions on UCLA but not on DynTex and DynTex++ with more complex turbulence of DTs (see Table 9).

Table 9: Comparison of rates (%) on DynTex and DynTex++.

Group	Encoding method	Dyn35	Alpha	Beta	Gamma	Dyn++
A	FDT [26]	98.86	98.33	93.21	91.67	95.31
	FD-MAP [26]	98.86	98.33	92.59	91.67	95.69
	DDTP [27]	99.71	96.67	93.83	91.29	95.09
C	3D-OTF [30]	96.70	83.61	73.22	72.53	89.17
	DFS [75]	97.16	85.24	76.93	74.82	91.70
	2D+T [68]	-	85.00	67.00	63.00	-
	STLS [32]	98.20	89.40	80.80	79.80	94.50
D	MBSIF-TOP [56]	98.61^{N}	90.00^{N}	90.70^{N}	91.30^{N}	97.12^{N}
	DNGP [28]	-	-	-	-	93.80
	B3DF_SMC [57]	99.71^{N}	95.00^{N}	90.12^{N}	90.91^{N}	95.58^{N}
E	VLBP [3]	81.14^{N}	-	-	-	94.98^{N}
	LBP-TOP [3]	92.45^{N}	98.33	88.89	84.85^{N}	94.05^{N}
	DDLBP with MJMI [76]	-	-	-	-	95.80
	CVLBP [50]	85.14^{N}	-	-	-	-
	HLBP [51]	98.57^{N}	-	-	-	96.28^{N}
	CLSP-TOP [52]	98.29^{N}	95.00^{N}	91.98^{N}	91.29^{N}	95.50^{N}
	MEWLSP [69]	99.71^{N}	-	-	-	98.48^{N}
	WLBPC [74]	-	-	-	-	95.01^{N}
	CVLBC [62]	98.86^{N}	-	-	-	91.31^{N}
	CSAP-TOP [53]	**100**	96.67	92.59	90.53	-
	FoSIG [46]	99.14	96.67	92.59	92.42	95.99
	V-BIG [47]	99.43	**100**	95.06	94.32	96.65
	HILOP [54]	99.71	96.67	91.36	92.05	96.21
	$\text{MMDP}_{D\_M/C}$ [55]	99.43	98.33	96.91	92.05	95.86
	$\text{MEMDP}_{D\_M/C}$ [55]	99.71	96.67	96.91	93.94	96.03
	RUBIG [48]	98.86	**100**	95.68	93.56	97.08
	HoGF^{2D} [59]	99.71	**100**	97.53	96.59	97.19
	HoGF^{3D} [59]	99.43	98.33	98.15	97.53	97.63
	Our $\textbf{BiFoG}^{\textbf{2D}}_{\textbf{1.0},\{1^{st},2^{nd}\}}$	99.71	98.33	95.06	95.08	97.56
	Our $\textbf{BiFoG}^{\textbf{2D}}_{\textbf{1.0},\{1^{st},4^{th}\}}$	99.14	98.33	95.68	95.45	97.29
	Our $\textbf{BiFoG}^{\textbf{3D}}_{\textbf{1.0},1^{st}}$	98.86	**100**	95.68	95.83	97.38
	Our $\textbf{BiFoG}^{\textbf{3D}}_{\textbf{1.0},\{1^{st},2^{nd}\}}$	99.43	98.33	96.91	97.35	97.68
	Our $\textbf{BiFoG}^{\textbf{3D}}_{\textbf{1.0},\{1^{st},4^{th}\}}$	99.14	98.33	95.68	97.73	97.94
F	DL-PEGASOS [35]	-	-	-	-	63.70
	PCA-cLBP/PI/PD-LBP [45]	-	-	-	-	92.40
	Orthogonal Tensor DL [40]	-	87.80	76.70	74.80	94.70
	Equiangular Kernel DL [41]	-	88.80	77.40	75.60	93.40
	SOE-Net [77]	-	96.70	95.70	92.20	94.40
	st-TCoF [38]	-	**100***	**100***	98.11*	-
	PCANet-TOP [37]	-	96.67*	90.74*	89.39*	-
	D3 [39]	-	**100***	**100***	98.11*	-
	DT-CNN-AlexNet [36]	-	**100***	99.38*	**99.62***	98.18*
	DT-CNN-GoogleNet [36]	-	**100***	**100***	**99.62***	**98.58***

Note: "-" is "not available". Superscript "*" are results using deep learning algorithms. "N" is rate with 1-NN classifier. Dyn35 and Dyn++ stand for DynTex35 and DynTex++ sub-datasets. Group A denotes *optical-flow-based methods*, C: *geometry-based*, D: *filter-based*, E: *local-feature-based*, F: *learning-based*.

4.6.2. Recognition on DynTex

It can be seen from Table 9 that our proposal is one of the best compared to all non-deep-learning methods. Specifically, the proposed $\text{BiFoG}^{2D/3D}$ descriptors have nearly the same performance as $\text{HoGF}^{2D/3D}$'s [59] but in the smaller dimension. Also, it should be emphasized that our $\text{BiFoG}^{3D}_{1.0,\{1^{st},4^{th}\}}$ obtains 97.73% on *Gamma*, a little better than HoGF^{3D}'s (97.53%) (see Table 9). Furthermore, ours is also from over 1% to 4% higher enhancement on the challenging schemes (i.e., *Beta* and *Gamma*) than those of MDP-based [55] and RUBIG [48], which are very recently the potent approaches based on local features for DT representation. Due to the very similarity of DT motions in two categories as highlighted in red in Figure 14, $\text{BiFoG}^{2D}_{1.0,\{1^{st},2^{nd}\}}$ obtains 99.71% on *DynTex35*, a little lower than CSAP-TOP [53] but in much smaller dimension (4800 bins) compared to CSAP-TOP's (13200 bins). Moreover, CSAP-TOP is not better than ours on the rest scenarios of DynTex (i.e., *Alpha*, *Beta*, and *Gamma*), as well as on UCLA (see Table 8). In terms of comparison with deep-learning methods, our highest rates of 100%, 96.91%, and 97.73% on *Alpha*, *Beta*, and *Gamma* respectively are very close to those of the deep-learning techniques: DT-CNN [36], st-TCoF [38], and D3 [39] (see Table 9). It should be pointed out that those have usually used complex algorithms to learn tremendous parameters, while we just address shallow analyses for DT representation. For further consideration of improvement, we detail the confusions of the BiFoG-based descriptors in DT recognition on challenging schemes *Beta* and *Gamma*. Accordingly, $\text{BiFoG}^{3D}_{1.0,\{1^{st},2^{nd}\}}$ has mainly confused "Rotation" with the others (see Figure 15), while $\text{BiFoG}^{3D}_{1.0,\{1^{st},4^{th}\}}$ is more miscellaneous, confused "Escalator" with "Flags" and "Grass", "Calm water" with "Sea" and "Fountains" (see Figure 16).

4.6.3. Recognition on DynTex++

Our BiFoG-based descriptors have also very good performance on this scheme, with over 97% for BiFoG^{3D} in 2-scale of high-orders (see Table 7). Particularly, $\text{BiFoG}^{3D}_{1.0,\{1^{st},4^{th}\}}$ achieves rate of 97.94% because of the challenge of DynTex++'s categories impressed in Figure

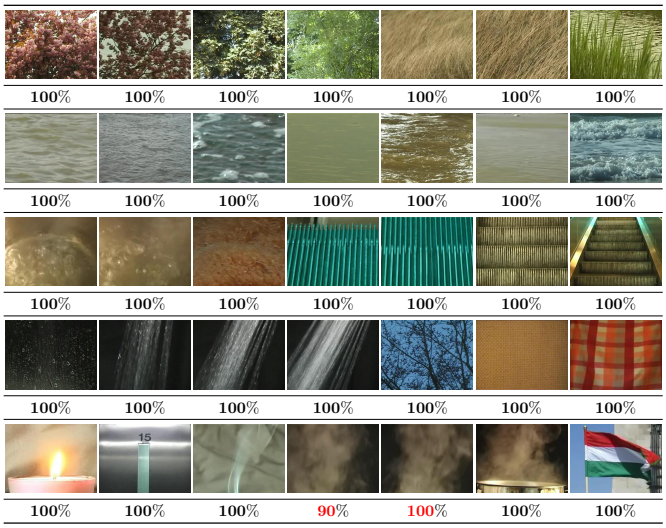


Figure 14: (Best viewed in color) Rates of $\text{BiFoG}^{3D}_{1.0,\{1^{st},2^{nd}\}}$ on specific categories of *DynTex35*.

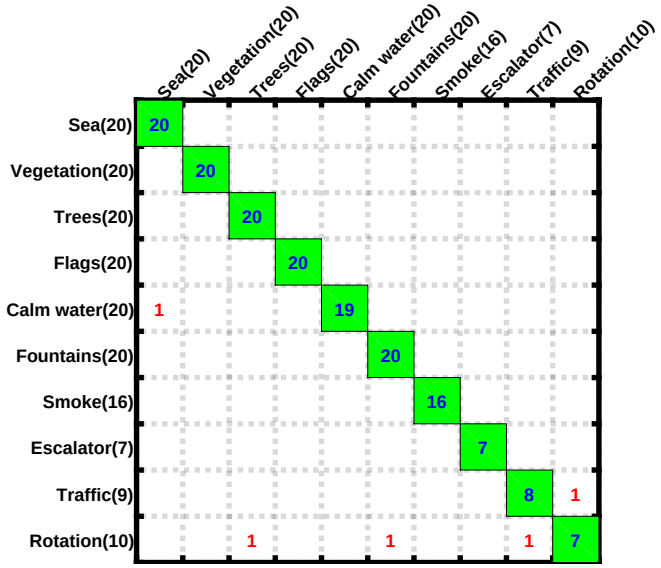


Figure 15: Confusion matrix of $\text{BiFoG}^{3D}_{1.0,\{1^{st},2^{nd}\}}$ on *Beta*.

17. This result is mostly the best compared to that of all existing methods, except MEWLSP's [69] (98.48%) and DT-CNN's [36] (98.18% for AlexNet and 98.58% for GoogleNet) (see Table 9). It should be noted that the execution of MEWLSP is lower than ours on UCLA (see Table 8), as well as it has not been verified on the challenging schemes of DynTex, i.e., *Alpha*, *Beta*, *Gamma*. In the meanwhile, DT-CNN is about 0.2∼0.6% better than ours but it learned an enormous number of parameters using deep-learning frameworks on each particular dataset.

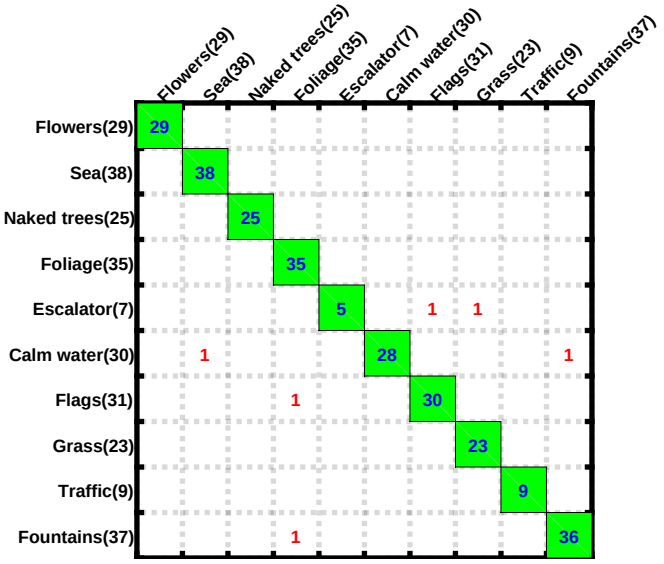


Figure 16: Confusion matrix of $\text{BiFoG}^{3D}_{1.0,\{1^{st},4^{th}\}}$ on *Gamma*.

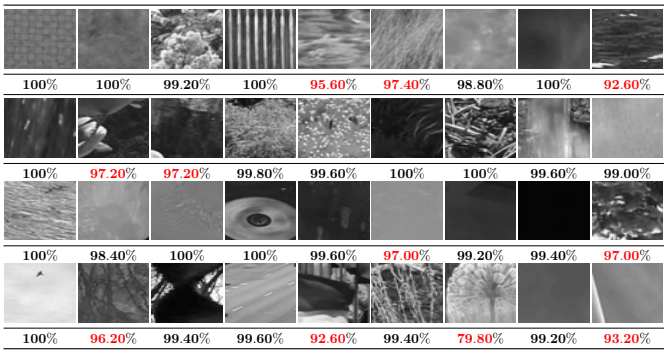


Figure 17: (Best viewed in color) Specific recognition results of $\text{BiFoG}^{3D}_{1.0,\{1^{st},4^{th}\}}$ on DynTex++'s categories. The challenging ones for the proposed descriptor are in red rates.

4.6.4. Recognition on DTDB

Due to the large scale of DTDB [43] dataset, we just implement the best settings of 2-scale high-orders, as discussed on Section 4.5, in order to evaluate the ability of our BiFoG-based descriptors, i.e., the partial derivatives of high-orders $\{\{1^{st},2^{nd}\},\{1^{st},4^{th}\}\}$ and the standard deviation $\sigma = 1$. It should be noted that the HoGF-based descriptors [59] have not been verified on this large scale dataset. For thoroughly evaluating the effectiveness of the bipolar-filtered features compared to the Gaussian-gradient-filtered ones, we also implement the HoGF-based descriptors [59] using their best settings: 2-scale analysis of local neighborhoods $\{(P,R)\} = \{(8,1),(8,2)\}$, the

standard deviation $\sigma = 1$, the 2-scale orders $\{2^{nd}, 3^{rd}\}$ for HoGF^{2D} and $\{3^{rd}, 4^{th}\}$ for HoGF^{3D} (refer to [59] for more detail). Furthermore, the typical LBP-based methods (i.e., LBP-TOP [3] and CLBP [58]) are also addressed for a purpose of comparison. Table 10 shows results of those implementations for DT recognition on two challenging DTDB's subsets: *Dynamics* and *Appearance*.

It can be seen from Table 10 that our BiFoG-based descriptors have very good performance on both DTDB's schemes. Indeed, rates of BiFoG^{2D} are about 11% and 9% better than LBP-TOP's [3] and CLBP's [58] respectively, while BiFoG^{3D}'s are over 2% higher than those of BiFoG^{2D}. It means that addressing CLBP on the high-order bipolar-filtered outcomes $\Theta^{2D/3D}$ has pointed out local spatio-temporal features with much more discrimination power than doing it on the raw DT sequences. Furthermore, it should be emphasized that with two thirds smaller dimension, our BiFoG$^{2D/3D}$ descriptors have noticeably better performances of DT recognition on both the challenging DTDB's schemes, e.g., BiFoG$^{2D}_{1.0,\{1^{st},4^{th}\}}$ with 4800 bins obtains rates of (69.74%, 69.64%) on schemes (*Dynamics*, *Appearance*) respectively compared to (69.38%, 69.56%) of HoGF^{2D} [59] with 7200 bins, while BiFoG$^{3D}_{1.0,\{1^{st},4^{th}\}}$ with 7200 bins obtains (71.73%,71.60%), also about 0.6% higher than (71.08%, 71.03%) of HoGF^{3D} [59] with 9600 bins. Those above have consolidated the interest of our proposal.

In respect of comparison to the learning-based methods, our BiFoG-based descriptors are comparable to those methods. Particularly, on *Appearance* scheme, BiFoG^{3D} with rates of over 71% are about 7% better than that of Flow Stream [78] (just 64.80%), while being very close to that of MSOE Stream [65] (72.20%). On *Dynamics* scheme, ours is nearly the same level as that of Flow Stream [78]. Also, it should be emphasized that SOE-Net [77] mostly obtains the highest rates on DTDB but it is not on DynTex and DynTex++. Actually, it can be verified from Table 9 that the performance of SOE-Net is the same ours on *Beta* but about 4∼5% inferior to our BiFoG^{3D}'s on *Gamma* and DynTex++.

Table 10: Comparison of rates (%) on two challenging schemes of the large scale DTDB dataset.

Group	Encoding method	$\{(P,R)\}$	Dynamics	Appearance
E	LBP-TOP^{u2} [3]	$\{(8,1)\}$	48.52	47.32
	CLBP$^{riu2}_{S/M/C}$ [58]	$\{(8,1)\}$	60.45	60.73
	HoGF^{2D} [59]	$\{(8,1),(8,2)\}$	69.38	69.56
	HoGF^{3D} [59]	$\{(8,1),(8,2)\}$	71.08	71.03
	Our BiFoG$^{2D}_{1.0,\{1^{st},2^{nd}\}}$	$\{(8,1)\}$	69.66	69.08
	Our BiFoG$^{2D}_{1.0,\{1^{st},4^{th}\}}$	$\{(8,1)\}$	69.74	69.64
	Our BiFoG$^{3D}_{1.0,\{1^{st},2^{nd}\}}$	$\{(8,1)\}$	71.57	71.33
	Our BiFoG$^{3D}_{1.0,\{1^{st},4^{th}\}}$	$\{(8,1)\}$	71.73	71.60
F	MSOE Stream [65]	-	80.10	72.20
	SOE-Net [77]	-	**86.80**	79.00
	C3D [42]	-	74.90*	75.50*
	RGB Stream [78]	-	76.40*	76.10*
	Flow Stream [78]	-	72.60*	64.80*
	MSOE-two-Stream [43]	-	84.00*	**80.00***

Note: "-" means "not available". Superscript "*" expresses results using deep learning algorithms. Group E denotes *local-feature-based* methods, while F: *learning-based*. The results of above learning-based methods are referred to [43], while those of LBP-TOP [3], CLBP [58], and HoGF$^{2D/3D}$ [59] are reported by our implementations. "$_{S/M/C}$" denotes a 3D-jointed histogram of CLBP's components.

4.7. Global discussions

As thoroughly evaluated in Sections 4.3, 4.4, 4.5, and 4.6, it could be asserted that addressing the bipolar-filtered features of Gaussian-gradient filterings for DT representation is an considerable solution for implementation in practice. Beside those evaluations, it can be made more following statements in order to consolidate the effectiveness of the BiFoG-based descriptors in further experiments:

- It can be observed from Tables 6 and 7 that the bipolar-filtered features extracted from the 3D Gaussian-gradient filterings are more discriminative to boost the performance than those decomposed from the 2D ones thanks to a joint consideration of shape and motion cues in the first ones. Therefore, the proposed 3D decomposition should be recommended for real applications.

Table 11: Rates (%) of the BiFoG-based descriptors in multi-analyses of high-orders $\mathcal{F}$ and standard deviations σ.

	BiFoG-based descriptor	#Bins	Beta	Gamma	DynTex++
(a)	$\text{BiFoG}^{2D}_{1.0,\{1^{st},2^{nd},3^{nd}\}}$	7200	93.21	94.70	96.99
	$\text{BiFoG}^{2D}_{1.0,\{1^{st},2^{nd},3^{nd},4^{th}\}}$	9600	94.44	94.32	97.36
	$\text{BiFoG}^{3D}_{1.0,\{1^{st},2^{nd},3^{nd}\}}$	10800	95.68	96.59	97.73
	$\text{BiFoG}^{3D}_{1.0,\{1^{st},2^{nd},3^{nd},4^{th}\}}$	14400	96.91	**96.97**	97.48
(b)	$\text{BiFoG}^{2D}_{\{0.5,1.0\},\{1^{st}\}}$	4800	95.06	93.56	97.32
	$\text{BiFoG}^{2D}_{\{0.5,0.7,1.0\},\{1^{st}\}}$	7200	94.44	93.94	97.21
	$\text{BiFoG}^{3D}_{\{0.5,1.0\},\{1^{st}\}}$	7200	95.68	**96.97**	97.82
	$\text{BiFoG}^{3D}_{\{0.5,0.7,1.0\},\{1^{st}\}}$	10800	96.30	96.21	97.64
(c)	$\text{BiFoG}^{2D}_{\{0.5,1.0\},\{1^{st},2^{nd}\}}$	9600	95.68	95.08	97.83
	$\text{BiFoG}^{2D}_{\{0.5,0.7,1.0\},\{1^{st},2^{nd}\}}$	14400	95.68	95.08	97.80
	$\text{BiFoG}^{3D}_{\{0.5,1.0\},\{1^{st},2^{nd}\}}$	14400	**97.53**	96.21	98.11
	$\text{BiFoG}^{3D}_{\{0.5,0.7,1.0\},\{1^{st},2^{nd}\}}$	21600	96.30	95.83	**98.16**

- The experimental results have also indicated that the BiFoG-based descriptors have performed well in comparison with the non-BiFoG ones. In further context, the bipolar-filtered features can be combined with the informative magnitudes of the concerning Gaussian-gradient-filtered outcomes in order to improve the discrimination power.
- Addressing multi-scales of more than two high-orders $\mathcal{F}$ as well as of more than one standard deviation σ seems not to enhance the performance while the dimension increases dramatically. Indeed, Table 11 (a) shows that the rates of $\text{BiFoG}^{2D/3D}$ are not improved when multi-scales of three and four scales of high-orders are taken into account. It is the same for multi-scales of standard deviations (see Table 11 (b)). In the meanwhile, combinations of multi-scales of both high-orders and standard deviations obtain little higher rates on *Beta* (97.53%) and DynTex++ (98.16%) but in much larger dimension (see Table 11 (c)). Therefore, those should not be recommended for real applications.

In current community of computer vision, methods based on deep-learning networks are one of major streams. In spite of achieving good performances in learning DT features for recognition issues (see Tables 8, 9, and 10), they have usually taken a large cost to learn millions of parameters by implementing complicated learning algorithms in deeply multi-layer frameworks, e.g., ∼61M for AlexNet and ∼6.8M for GoogleNet for DT-CNN [36], ∼80M for C3D [42], and ∼88M for MSOE-two-Stream [43]. Certainly, it is one of decisive obstructions so that they can be applied to real implementations for embedded sensor systems as well as mobile devices, those which are in strict requirements of tightly computing resources for their executions.

Our proposal in this work could partly deal with that barrier by using a shallow architecture of video analysis in low computational complexity. Indeed, it just exploits a simple operator to structure local bipolar-based patterns from the filtered outcomes extracted by the Gaussian-gradient filterings. In small dimension, our proposed BiFoG-based descriptors are one of the best among the local-feature-based approaches while the BiFoG's performances are also close to those of the deep-learning ones. Tables 8, 9, and 10 show the significant rates of our 2-order $\text{BiFoG}^{2D/3D}_{1.0,\{1^{st},2^{nd}\}}$ and $\text{BiFoG}^{2D/3D}_{1.0,\{1^{st},4^{th}\}}$ with only 4800 bins for the 2D ones and 7200 bins for the 3D. Substantially, those can be considered as some of appreciated solutions for mobile applications. In addition, instead of addressing CLBP [58], other local potent operators can be investigated for a purpose of further improvements such as CLBC [61], MRELBP [72], LVP-based [71, 27], LRP [48], LDP-based [70, 55], etc.

5. Conclusions

In this paper, the bipolar properties of 2D/3D Gaussian-gradient filterings have been introduced for DT representation. Accordingly, the decomposing model has been proposed to split the Gaussian-gradient-filtered images/volumes (i.e., $\mathcal{I}^{pos}_{\sigma,\partial\lambda^k_i}/\mathcal{V}_{\sigma,\partial\lambda^k_i}$) into bipolar-filtered outcomes $\Theta^{2D/3D}_{\sigma,k}$, which have been proved the robustness to

noise. An efficient simple framework has been then presented to take advantage of the bipolar-filtered features, extracted from these complementary outcomes, in order to construct discriminative descriptors $\text{BiFoG}_{\sigma,\mathcal{F}}^{2D/3D}$. The experiments in DT recognition have verified that the performances of $\text{BiFoG}_{\sigma,\mathcal{F}}^{2D/3D}$ are very good in comparison with those of state of the art. For perspectives, it could be considered to decompose filtered components $\mathcal{I}_{\sigma,\partial\lambda_i^k}/\mathcal{V}_{\sigma,\partial\lambda_i^k}$ into more sub-outcomes in consideration of the influence of the close-to-zero pixels/voxels [67]. Encoding these obtained filtered elements may capture more robust bipolar-filtered features for DT representation. However, the increase of their final dimension should be treated for real implementations. In addition, instead of using CLBP [58], it can apply other LBP-based variants to the encoding phase in order for further improvement of performance.

[1] G. Doretto, A. Chiuso, Y. N. Wu, S. Soatto, Dynamic textures, IJCV 51 (2) (2003) 91–109.

[2] X. S. Nguyen, T. P. Nguyen, F. Charpillet, N.-S. Vu, Local derivative pattern for action recognition in depth images, Multimedia Tools Appl 77 (7) (2018) 8531–8549.

[3] G. Zhao, M. Pietikäinen, Dynamic texture recognition using local binary patterns with an application to facial expressions, IEEE Trans. PAMI 29 (6) (2007) 915–928.

[4] T. P. Nguyen, A. Manzanera, M. Garrigues, N. Vu, Spatial motion patterns: Action models from semi-dense trajectories, IJPRAI 28 (7).

[5] O. J. Makhura, J. C. Woods, Learn-select-track: An approach to multi-object tracking, Sig. Proc.: Image Comm. 74 (2019) 153–161.

[6] X. Wu, X. Lu, H. Leung, Video smoke separation and detection via sparse representation, Neurocomputing 360 (2019) 61–74.

[7] C. Zhang, F. Zhou, B. Xue, W. Xue, Stabilization of atmospheric turbulence-distorted video containing moving objects using the monogenic signal, Sig. Proc.: Image Comm. 63 (2018) 19–29.

[8] V. Hoang, K. Jo, Joint components based pedestrian detection in crowded scenes using extended feature descriptors, Neurocomputing 188 (2016) 139–150.

[9] H. Sajid, S. S. Cheung, N. Jacobs, Motion and appearance based background subtraction for freely moving cameras, Sig. Proc.: Image Comm. 75 (2019) 11–21.

[10] Z. Xu, B. Min, R. C. C. Cheung, A robust background initialization algorithm with superpixel motion detection, Sig. Proc.: Image Comm. 71 (2019) 1–12.

[11] P. Saisan, G. Doretto, Y. N. Wu, S. Soatto, Dynamic texture recognition, in: CVPR, 2001, pp. 58–63.

[12] A. B. Chan, N. Vasconcelos, Classifying video with kernel dynamic textures, in: CVPR, 2007, pp. 1–6.

[13] A. Mumtaz, E. Coviello, G. R. G. Lanckriet, A. B. Chan, Clustering dynamic textures with the hierarchical EM algorithm for modeling video, IEEE Trans. PAMI 35 (7) (2013) 1606–1621.

[14] A. Ravichandran, R. Chaudhry, R. Vidal, View-invariant dynamic texture recognition using a bag of dynamical systems, in: CVPR, 2009, pp. 1651–1657.

[15] L. Wang, H. Liu, F. Sun, Dynamic texture video classification using extreme learning machine, Neurocomputing 174 (2016) 278–285.

[16] Y. Wang, S. Hu, Chaotic features for dynamic textures recognition, Soft Computing 20 (5) (2016) 1977–1989.

[17] Y. Wang, S. Hu, Exploiting high level feature for dynamic textures recognition, Neurocomputing 154 (2015) 217–224.

[18] A. Mumtaz, E. Coviello, G. R. G. Lanckriet, A. B. Chan, A scalable and accurate descriptor for dynamic textures using bag of system trees, IEEE Trans. PAMI 37 (4) (2015) 697–712.

[19] Y. Qiao, L. Weng, Hidden markov model based dynamic texture classification, IEEE Signal Process. Lett. 22 (4) (2015) 509–512.

[20] Y. Qiao, Z. Xing, Dynamic texture classification using multivariate hidden markov model, IEICE Transactions 101-A (1) (2018) 302–305.

[21] C. Peh, L. F. Cheong, Synergizing spatial and temporal texture, IEEE Trans. IP 11 (10) (2002) 1179–1191.

[22] R. Péteri, D. Chetverikov, Qualitative characterization of dynamic textures for video retrieval, in: K. W. Wojciechowski, B. Smolka, H. Palus, R. Kozera, W. Skarbek, L. Noakes (Eds.), ICCVG, Vol. 32 of Computational Imaging and Vision, 2004, pp. 33–38.

[23] R. Péteri, D. Chetverikov, Dynamic texture recognition using normal flow and texture regularity, in: J. S. Marques, N. P. de la Blanca, P. Pina (Eds.), IbPRIA, Vol. 3523 of LNCS, 2005, pp. 223–230.

[24] Z. Lu, W. Xie, J. Pei, J. Huang, Dynamic texture recognition by spatio-temporal multiresolution histograms, in: WACV/MOTION, 2005, pp. 241–246.

[25] S. Fazekas, D. Chetverikov, Analysis and performance evaluation of optical flow features for dynamic texture recognition, Sig. Proc.: Image Comm. 22 (7-8) 680–691.

[26] T. T. Nguyen, T. P. Nguyen, F. Bouchara, X. S. Nguyen, Directional beams of dense trajectories for dynamic texture recognition, in: J. Blanc-Talon, D. Helbert, W. Philips, D. Popescu, P. Scheunders (Eds.), ACIVS, 2018, pp. 74–86.

[27] T. T. Nguyen, T. P. Nguyen, F. Bouchara, Directional dense-

trajectory-based patterns for dynamic texture recognition, IET Computer Vision 14 (4) (2020) 162–176.

[28] A. R. Rivera, O. Chae, Spatiotemporal directional number transitional graph for dynamic texture recognition, IEEE Trans. PAMI 37 (10) (2015) 2146–2152.

[29] Y. Xu, Y. Quan, H. Ling, H. Ji, Dynamic texture classification using dynamic fractal analysis, in: ICCV, 2011, pp. 1219–1226.

[30] Y. Xu, S. B. Huang, H. Ji, C. Fermüller, Scale-space texture description on sift-like textons, CVIU 116 (9) (2012) 999–1013.

[31] H. Ji, X. Yang, H. Ling, Y. Xu, Wavelet domain multifractal analysis for static and dynamic texture classification, IEEE Trans. IP 22 (1) (2013) 286–299.

[32] Y. Quan, Y. Sun, Y. Xu, Spatiotemporal lacunarity spectrum for dynamic texture classification, CVIU 165 (2017) 85–96.

[33] M. Baktashmotlagh, M. T. Harandi, B. C. Lovell, M. Salzmann, Discriminative non-linear stationary subspace analysis for video classification, IEEE Trans. Pattern Anal. Mach. Intell. 36 (12) (2014) 2353–2366.

[34] R. Péteri, S. Fazekas, M. J. Huiskes, Dyntex: A comprehensive database of dynamic textures, Pattern Recognition Letters 31 (12) (2010) 1627–1632.

[35] B. Ghanem, N. Ahuja, Maximum margin distance learning for dynamic texture recognition, in: K. Daniilidis, P. Maragos, N. Paragios (Eds.), ECCV, Vol. 6312 of LNCS, 2010, pp. 223–236.

[36] V. Andrearczyk, P. F. Whelan, Convolutional neural network on three orthogonal planes for dynamic texture classification, Pattern Recognition 76 (2018) 36 – 49.

[37] S. R. Arashloo, M. C. Amirani, A. Noroozi, Dynamic texture representation using a deep multi-scale convolutional network, JVCIR 43 (2017) 89 – 97.

[38] X. Qi, C.-G. Li, G. Zhao, X. Hong, M. Pietikainen, Dynamic texture and scene classification by transferring deep image features, Neurocomputing 171 (2016) 1230 – 1241.

[39] S. Hong, J. Ryu, W. Im, H. S. Yang, D3: recognizing dynamic scenes with deep dual descriptor based on key frames and key segments, Neurocomputing 273 (2018) 611–621.

[40] Y. Quan, Y. Huang, H. Ji, Dynamic texture recognition via orthogonal tensor dictionary learning, in: ICCV, 2015, pp. 73–81.

[41] Y. Quan, C. Bao, H. Ji, Equiangular kernel dictionary learning with applications to dynamic texture analysis, in: CVPR, 2016, pp. 308–316.

[42] D. Tran, L. D. Bourdev, R. Fergus, L. Torresani, M. Paluri, Learning spatiotemporal features with 3d convolutional networks, in: ICCV, 2015, pp. 4489–4497.

[43] I. Hadji, R. P. Wildes, A new large scale dynamic texture dataset with application to convnet understanding, in: ECCV, 2018, pp. 334–351.

[44] T. Ojala, M. Pietikäinen, T. Mäenpää, Multiresolution gray-scale and rotation invariant texture classification with local binary patterns, IEEE Trans. PAMI 24 (7) (2002) 971–987.

[45] J. Ren, X. Jiang, J. Yuan, Dynamic texture recognition using enhanced LBP features, in: ICASSP, 2013, pp. 2400–2404.

[46] T. T. Nguyen, T. P. Nguyen, F. Bouchara, Smooth-invariant gaussian features for dynamic texture recognition, in: ICIP, 2019, pp. 4400–4404.

[47] T. T. Nguyen, T. P. Nguyen, F. Bouchara, N. Vu, Volumes of blurred-invariant gaussians for dynamic texture classification, in: M. Vento, G. Percannella (Eds.), CAIP, 2019, pp. 155–167.

[48] T. T. Nguyen, T. P. Nguyen, F. Bouchara, Rubik gaussian-based patterns for dynamic texture classification, Pattern Recognition Letters 135 (2020) 180–187.

[49] G. Zhao, T. Ahonen, J. Matas, M. Pietikäinen, Rotation-invariant image and video description with local binary pattern features, IEEE Trans. IP 21 (4) (2012) 1465–1477.

[50] D. Tiwari, V. Tyagi, Dynamic texture recognition based on completed volume local binary pattern, MSSP 27 (2) (2016) 563–575.

[51] D. Tiwari, V. Tyagi, A novel scheme based on local binary pattern for dynamic texture recognition, CVIU 150 (2016) 58–65.

[52] T. T. Nguyen, T. P. Nguyen, F. Bouchara, Completed local structure patterns on three orthogonal planes for dynamic texture recognition, in: IPTA, 2017, pp. 1–6.

[53] T. T. Nguyen, T. P. Nguyen, F. Bouchara, Completed statistical adaptive patterns on three orthogonal planes for recognition of dynamic textures and scenes, J. Electronic Imaging 27 (05) (2018) 053044.

[54] T. T. Nguyen, T. P. Nguyen, F. Bouchara, Dynamic texture representation based on hierarchical local patterns, in: ACIVS, 2020, pp. 277–289.

[55] T. T. Nguyen, T. P. Nguyen, F. Bouchara, X. S. Nguyen, Momental directional patterns for dynamic texture recognition, CVIU 194 (2020) 102882.

[56] S. R. Arashloo, J. Kittler, Dynamic texture recognition using multiscale binarized statistical image features, IEEE Trans. Multimedia 16 (8) (2014) 2099–2109.

[57] X. Zhao, Y. Lin, L. Liu, J. Heikkilä, W. Zheng, Dynamic texture classification using unsupervised 3d filter learning and local binary encoding, IEEE Trans. Multimedia 21 (7) (2019) 1694–1708.

[58] Z. Guo, L. Zhang, D. Zhang, A completed modeling of local binary pattern operator for texture classification, IEEE Trans. IP 19 (6) (2010) 1657–1663.

[59] T. T. Nguyen, T. P. Nguyen, F. Bouchara, Prominent local representation for dynamic textures based on high-order gaussian-

gradients, IEEE Trans. Multimedia in press (2020) 1–1.

[60] T. P. Nguyen, A. Manzanera, W. G. Kropatsch, X. S. N'Guyen, Topological attribute patterns for texture recognition, Pattern Recognition Letters 80 (2016) 91–97.

[61] Y. Zhao, D.-S. Huang, W. Jia, Completed Local Binary Count for Rotation Invariant Texture Classification, IEEE Trans. IP 21 (10) (2012) 4492–4497.

[62] X. Zhao, Y. Lin, J. Heikkilä, Dynamic texture recognition using volume local binary count patterns with an application to 2d face spoofing detection, IEEE Trans. Multimedia 20 (3) (2018) 552–566.

[63] A. K. Jain, F. Farrokhnia, Unsupervised texture segmentation using gabor filters, Pattern Recognition 24 (12) (1991) 1167–1186.

[64] T. P. Nguyen, N. Vu, A. Manzanera, Statistical binary patterns for rotational invariant texture classification, Neurocomputing 173 (2016) 1565–1577.

[65] K. G. Derpanis, R. P. Wildes, Spacetime texture representation and recognition based on a spatiotemporal orientation analysis, IEEE Trans. PAMI 34 (6) (2012) 1193–1205.

[66] Y. Jansson, T. Lindeberg, Dynamic texture recognition using time-causal and time-recursive spatio-temporal receptive fields, Journal of Mathematical Imaging and Vision 60 (9) (2018) 1369–1398.

[67] N. Vu, T. P. Nguyen, C. Garcia, Improving texture categorization with biologically-inspired filtering, Image Vision Comput. 32 (6-7) (2014) 424–436.

[68] S. Dubois, R. Péteri, M. Ménard, Characterization and recognition of dynamic textures based on the 2d+t curvelet transform, Signal, Image and Video Processing 9 (4) (2015) 819–830.

[69] D. Tiwari, V. Tyagi, Dynamic texture recognition using multiresolution edge-weighted local structure pattern, Computers & Electrical Engineering 62 (2017) 485–498.

[70] B. Zhang, Y. Gao, S. Zhao, J. Liu, Local derivative pattern versus local binary pattern: Face recognition with high-order local pattern descriptor, IEEE Trans. IP 19 (2) (2010) 533–544.

[71] K. Fan, T. Hung, A novel local pattern descriptor - local vector pattern in high-order derivative space for face recognition, IEEE Trans. IP 23 (7) (2014) 2877–2891.

[72] L. Liu, S. Lao, P. W. Fieguth, Y. Guo, X. Wang, M. Pietikäinen, Median robust extended local binary pattern for texture classification, IEEE Trans. IP 25 (3) (2016) 1368–1381.

[73] R. Fan, K. Chang, C. Hsieh, X. Wang, C. Lin, LIBLINEAR: A library for large linear classification, JMLR 9 (2008) 1871–1874.

[74] D. Tiwari, V. Tyagi, Improved weber's law based local binary pattern for dynamic texture recognition, Multimedia Tools Appl. 76 (5) (2017) 6623–6640.

[75] Y. Xu, Y. Quan, Z. Zhang, H. Ling, H. Ji, Classifying dynamic textures via spatiotemporal fractal analysis, Pattern Recognition 48 (10) (2015) 3239–3248.

[76] J. Ren, X. Jiang, J. Yuan, G. Wang, Optimizing LBP structure for visual recognition using binary quadratic programming, SPL 21 (11) (2014) 1346–1350.

[77] I. Hadji, R. P. Wildes, A spatiotemporal oriented energy network for dynamic texture recognition, in: ICCV, 2017, pp. 3085–3093.

[78] K. Simonyan, A. Zisserman, Two-stream convolutional networks for action recognition in videos, in: NIPS, 2014, pp. 568–576.