Signal, Image and Video Processing (2019) 13:575–582
https://doi.org/10.1007/s11760-018-1384-8

ORIGINAL PAPER

Sparse binarised statistical dynamic features for spatio-temporal texture analysis

Shervin Rahimzadeh Arashloo[1]

Received: 14 May 2018 / Revised: 15 September 2018 / Accepted: 24 October 2018 / Published online: 31 October 2018
© Springer-Verlag London Ltd., part of Springer Nature 2018

Abstract
The paper presents a new spatio-temporal learning-based descriptor called binarised statistical dynamic features (BSDF) for representation and classification of dynamic texture. The BSDF descriptor operates by applying three-dimensional spatio-temporal filters on local voxels of an image sequence where the filters are learned via an independent component analysis, maximising independence over spatial and temporal domains concurrently. The BSDF representation is formed by binarising filter responses which are then converted into codewords and summarised using histograms. A robust representation of the BSDF descriptor is finally obtained via a sparse representation approach yielding very discriminative features for classification. The effects of different hyper-parameters on performance including the number of filters, the number of scales, temporal depth, number of samples drawn are also investigated. The proposed approach is evaluated on the most commonly used dynamic texture databases and shown to perform very well compared to the existing methods.

Keywords Dynamic texture · Spatio-temporal filtering · Independent component analysis · Sparse representation

1 Introduction

Dynamic or spatio-temporal texture refers to spatial patterns exhibiting motion over time. Recognition of dynamic texture (DT) has been an active research area finding applications in a wide variety of different tasks including video indexing and retrieval [25], visual speech recognition [51], dynamic scene classification [38], activity recognition [18], traffic monitoring [15], environmental monitoring [2], tracking [9]. Despite decades of research in this direction, the problem still remains challenging partly due to variations in image sequence content caused by environmental changes in addition to the inherent variations in shape and appearance of a dynamic texture as a function of time.

The existing approaches for recognition of dynamic texture may be coarsely categorised into generative and non-generative approaches. The generative approaches are typically based on some hypothesised model whose parameters are used for recognition purposes, while the non-generative methods avoid the challenges of modelling and inferring system parameters and typically use aggregate statistics in local neighbourhoods of DT sequences for recognition. A drawback of the generative model-based approaches is their inability to generalise to DT sequences which are generated by some irregular physical process with complexities beyond those which can be accommodated by the hypothesised models. On the other hand, the success of non-generative local statistical methods is partly determined by the discriminatory properties of the features capturing local statistics which very often cannot be determined a priori. Nevertheless, as the non-generative methods focus on the underlying classification problem rather than modelling the generative process, they tend to yield better performance in practical applications. A successful group of non-generative statistical methods among others is the family of LBP-based approaches [5,34,36,52] which consider an image sequence as three orthogonal planes. While computationally attractive, characterising an image sequence as three orthogonal planes partly compromises the discriminatory capability of the representation. Moreover, the use of hand-crafted feature extraction procedures such as those in [36,52] may lead to a suboptimal performance.

In this work, a new descriptor called binarised statistical dynamic features (BSDF) for characterisation of dynamic texture is presented which addresses the aforementioned

✉ Shervin Rahimzadeh Arashloo
S.Rahimzadeh@cs.bilkent.edu.tr

[1] Department of Computer Engineering, Faculty of Engineering, Bilkent University, Ankara, Turkey

shortcomings of some of the previous methods as follows. First, instead of characterising an image sequence via three orthogonal planes, the spatial and temporal information content of an image sequence is jointly encoded via three-dimensional spatio-temporal filters. Next, instead of using hand-crafted feature extraction procedures, the 3D filters used in the proposed BSDF representation are learned via an ICA analysis on local space–time data. The use of ICA is motivated by the observations confirming that applying ICA on video sequences of natural scenes produces results with qualitatively similar spatio-temporal properties as those of simple cells in primary visual cortex [26]. The use of ICA is also motivated by its success in learning 2D filters for characterisation of static/dynamic texture [5,31]. As a consequence of using ICA for filter learning, the proposed approach jointly maximises statistical independence over both space and time which contrasts with some alternative methods, maximising independence either over time or over space only [5]. In order to form the final BSDF representation, the filter responses are binarised and converted into codewords which are then summarised using histograms. For improved robustness of the BSDF representation to unwanted degradations and inconsistencies in imaging conditions, a sparse representation of the BSDF features for characterisation of dynamic textures is proposed. An analysis of the effects of various hyper-parameters associated with the design of BSDF representation including the numbers of filters (and consequently different lengths for codewords), a multi-scale extension of the representation as well as the effects of temporal depths of filters on system performance is also carried out. Moreover, a random sampling approach is examined and shown to be instrumental in reducing the computational complexity of the proposed approach to a large extent without much compromising system performance. The main contributions of this work can be summarised as: (1) a new descriptor (BSDF) for characterisation of a DT sequence based on spatio-temporal filtering. In contrast to some earlier approaches, the new descriptor uses ICA in the filter design procedure to jointly maximise independence over both space and time concurrently which leads to more informative features for recognition purposes; (2) an analysis of the effect of the number of BSDF filters on performance; (3) a multi-scale analysis of the proposed BSDF representation; (4) an analysis of the effect of filter depths in the time dimension on performance; (5) a random sampling approach for reduced computational complexity of the proposed representation; (6) a sparse representation of the multi-scale BSDF features for robust representation and classification of an image sequence and (7) extensive evaluation of the proposed approach on different databases along with a comparison to the state-of-the-art methods.

The rest of the paper is organised as follows: In Sect. 2, we review the relevant literature. In Sect. 3, the proposed spatio-temporal BSDF approach is presented. In Sect. 4, the results of an extensive experimental evaluation of the proposed approach, analysing different aspects of the proposed representation along with a comparison to the state-of-the-art methods on different databases is presented. Finally, conclusions are drawn in Sect. 5.

2 Related work

As noted earlier, the existing methods for dynamic texture recognition may be classified into generative and non-generative methods. As examples of the former group, one may consider [10,20,22–24,32,33,39–42,46,48]. In contrast to the generative methods, there exist non-generative methods using statistical properties of local descriptors. As instances, one may consider the LBP-based methods [5,36,52]. Other approach [34] presented a variant of the LBP-TOP approach for action recognition. In [49], a collaborative representation dynamic texture classification method is presented where the dynamic texture sequence is divided into sub-sequences along the temporal axis for each of which an LBP histogram is extracted. Other work in [47] presents a semantic decomposition of spatio-temporal information. In [29], a texture descriptor for both static and dynamic textures is built using the wavelet-based spatial-frequency analysis. Other work in [21] proposes the use of the 2D+T curvelet transform for characterisation of dynamic textures in image sequences. In [30], the one-class SVM is proposed for DT recognition purposes resulting in relatively robust features. Other work in [11] proposes a local ternary pattern for background modelling for saliency detection in video sequences. In [7], the authors address the face presentation attack detection from a dynamic texture point of view and use aggregated local weighted gradient orientation. In [43], a dynamic micro-texture descriptor capturing the spatial structure and motion of a local neighbourhood is proposed. The work in [6] proposed a method for explicitly separating the parts shared among all videos from those specific to individual videos. Other work in [45] proposed to represent videos using unsupervised learning of motion features. The authors in [13] proposed a DT segmentation method based on appearance and motion. In addition to the methods mentioned above, there exists a further category of methods using deep networks [8], examples of which include [3,4,12,14,38].

3 Methodology

3.1 Linear filtering

Linear filtering in the spatial domain is an effective and widely employed approach in characterising static image content. When dealing with an image sequence, the oper-

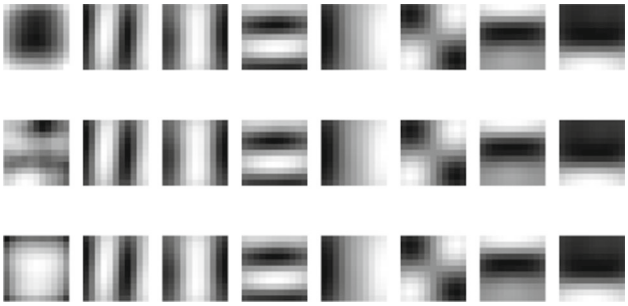

Fig. 1 Sample spatio-temporal filters obtained for 25 fps sampling rate. Each column represents spatial slices of a spatio-temporal filter of depth 3 in time and sizes of 11×11 in the spatial domain

ation would be linear spatio-temporal filtering, which can be considered as the direct extension of the linear spatial filtering. In order to analyse the dynamic statistics of data beyond covariances, independent component analysis (ICA) has been successfully employed in earlier studies [28]. This is in fact the approach followed in the BSIF representation [31] for static images, and the proposed BSDF approach can be considered as a spatio-temporal extension of the method in [31].

3.2 Binarised statistical dynamic features (BSDF)

The proposed BSDF approach uses spatio-temporal volumes as data in the ICA model. For a more detailed analysis, let $\mathbf{f}$ denote the voxel values of a local volume arranged into a vector. The order in which the voxels are arranged in $\mathbf{f}$ is arbitrary but otherwise fixed. Using ICA, $\mathbf{f}$ can be represented as

$$\mathbf{f} = \mathcal{G}\mathbf{h} \tag{1}$$

where the elements of the vector $\mathbf{h}$ are random variables which are unknown and statistically as independent of each other as possible. $\mathcal{G}$ is a feature matrix with constant elements. Using a training set of local volumes, the feature matrix $\mathcal{G}$ can be approximated without explicitly knowing the latent vector $\mathbf{h}$ [28]. Equivalently, one may infer matrix $\mathbf{G}$ which represents $\mathbf{h}$ as the output of a number of linear spatio-temporal filters as

$$\mathbf{h} = \mathbf{G}\mathbf{f} \tag{2}$$

where each row of $\mathbf{G}$ now represents a spatio-temporal filter. In practice, a whitening transformation is commonly applied to the data before an ICA analysis. Following such an approach and multiplying both sides of Eq. 1 by matrix $\mathbf{M}$ (which performs whitening) yields

$$\mathbf{M}\mathbf{f} = \mathbf{y} = \mathbf{M}\mathcal{G}\mathbf{h} = \mathcal{T}\mathbf{h} \tag{3}$$

where $\mathbf{y}$ is whitened data while matrix $\mathcal{T}$ is obtained by pre-multiplying matrix $\mathcal{G}$ by the pre-processing transformation matrix, $\mathbf{M}$. In practice, one would like to obtain $\mathbf{h}$ as a function of $\mathbf{y}$. In this case, the relation $\mathbf{y} = \mathcal{T}\mathbf{h}$ needs to be inverted for which the number of independent components is required to match the number of components of $\mathbf{y}$. If this condition is satisfied, the system $\mathbf{y} = \mathcal{T}\mathbf{h}$ would be invertible in a unique way, producing the vector $\mathbf{h}$ as a linear function of $\mathbf{y}$ as

$$\mathbf{h} = \mathbf{T}\mathbf{y} \tag{4}$$

where $\mathbf{T} = \mathcal{T}^{-1}$. As a result, the independent components (h_i's) of vector $\mathbf{h}$ are explicitly derived from voxel values as

$$\mathbf{h} = \mathbf{T}\mathbf{M}\mathbf{f} \tag{5}$$

Comparing Eqs. 2 and 5, it can be easily verified that the filter matrix $\mathbf{G}$ in Eq. 2 can be obtained as

$$\mathbf{G} = \mathbf{T}\mathbf{M} \tag{6}$$

Once a spatio-temporal feature detector has been learned from the data, it can be visualised as an image sequence. Sample filters learned using the aforementioned procedure are depicted in Fig. 1. Once BSDF filters are inferred, given a local volume $\mathbf{f}$, one applies N filters to the elements of $\mathbf{f}$ and obtains N responses, stacked into the vector $\mathbf{h}$. The filter responses are binarised next by thresholding at zero to produce the binarised features b_i's as

$$b_i = \begin{cases} 1 & h_i > 0, \\ 0 & \text{otherwise.} \end{cases} \tag{7}$$

The binarised features b_i's are then converted to codewords and summarised using histograms. If N BSDF spatio-temporal filters are inferred, an N-bit binary code would be obtained. The BSDF histogram representation ($\mathbf{H}$) is then derived as below.

$$\begin{aligned} &\mathbf{H} = [H^0, H^1, \ldots, H^{L-1}] \\ &H^i = \sum_{v} \mathbb{1}\{\text{BSDF}(v) = i\} \\ &L = 2^N \end{aligned} \tag{8}$$

where v is a voxel of the DT sequence, $\mathbb{1}\{.\}$ is the indicator function, equal to one when its argument is true and zero otherwise. L is the number of histogram bins, and N represents number of BSDF filters. When the dynamic textures to be compared are of different sizes, $\mathbf{H}$ is normalised to yield a coherent description:

$$\tilde{\mathbf{H}} = \frac{\mathbf{H}}{\sum_{i=0}^{L-1} H^i} \tag{9}$$

3.3 Multi-scale extension

The BSDF representation can be easily extended into a multi-scale framework by varying the sizes of spatio-temporal filters. For this purpose, in this work, BSDF filters are inferred at six different spatial scales of $3 \times 3, \ldots, 13 \times 13$ and in different temporal scales. By concatenating all the histograms estimated at different scales into a single vector, the multi-resolution BSDF descriptor for Z scales is obtained as

$$\mathbf{H_{ms}} = [\tilde{H}_1, \tilde{H}_2, \ldots, \tilde{H}_Z] \tag{10}$$

where $\tilde{H}_i$ denotes the normalised histogram obtained for the scale of i.

3.4 Sparse representation

In the context of linear modelling where an unknown sample is represented as a linear combination of available atoms, the sparse representation (SR) method can be considered as one of the most representative methodologies [50]. In the context of the current work, a BSDF descriptor derived from a test sequence is first approximated as a sparse linear combination of all training samples:

$$\hat{\boldsymbol{\alpha}} = \arg\min_{\boldsymbol{\alpha}} \|\boldsymbol{\alpha}\|_p \quad \text{s.t. } \|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\alpha}\|_2^2 \leq \epsilon \tag{11}$$

where $\boldsymbol{X}$ is the set of training samples, $\boldsymbol{y}$ represents the probe sample, $\boldsymbol{\alpha}$ is the sparse coefficients vector, and ϵ is a small threshold. For the $l1$-minimisation problem ($p = 1$ in Eq. 11), efficient methods exist when the solution is known to be very sparse. The homotopy algorithms [19,35] are used in this work due to the fact that they tend to be faster than some alternative solvers. For classification, the reconstruction residual of a test sample using the sparse coefficients of each class is used as a dissimilarity criterion for hypothesis selection:

$$\min_i r_i(y) = \|y - X\delta_i(\hat{\boldsymbol{\alpha}})\|_2^2 \tag{12}$$

In Eq. 12, $r_i(y)$ estimates the residual for probe y reconstructed as a linear combination of samples of class i and $\delta_i(.)$ is a characteristic function that selects the coefficients associated with the ith class.

4 Experimental evaluation

In this section, an experimental evaluation of the proposed BSDF representation is provided. The implementation is performed in the Matlab R2017a environment. For the ICA analysis, the FastICA algorithm [27] is used, while for the homotopy-based sparse representation approach the software provided in [1] is utilised.

4.1 The Dyntex database

The DynTex database [37] is one of the most widely used dynamic texture data sets, which is comprised of 656 videos. Image sequences in this database are divided into three subsets of alpha, beta and gamma corresponding to 60 ,162 and 264 image sequences, respectively.

4.2 The UCLA database

The UCLA data set is comprised of 50 scenes, each represented by four image sequences. A second version of this database where each sequence is clipped to a 48×48 window containing the key statistical and dynamical features is used in this work [44]. We have evaluated the proposed approach in the 50-class breakdown scenario and similar to [5,16,44] and followed a leave-one-out classification procedure where the true decision for a test sequence is defined as having one of the three remaining samples of the same class as its nearest neighbour.

4.3 Multi-scale BSDF

In this experiment, the effect of a multi-scale extension of the BSDF approach is investigated where six scales of $3 \times 3, \ldots, 13 \times 13$ for the spatial dimensions of the filters are considered. The temporal widths of filter are fixed at $T = 3$, and $N = 8$ filters are used. The multi-scale representation is constructed in two alternative ways. First, one may start with smaller filters and incrementally add larger filters. Alternatively, a reverse approach may be pursued where one starts with larger filters and gradually adds smaller filters to form the multi-scale representation. The results of this experiment on the Dyntex and UCLA data sets are reported in Table 1. In the table, scale 1 corresponds to the smallest filters (i.e. filters of size $3 \times 3 \times 3$) and scale 6 represents the largest ones. The [a b] notation indicates that all the scales in the range between a and b, including the scales of a and b, are used to form the representation. From Table 1, one observes that regardless of the approach taken to form the multi-scale representation (direct/reverse) the multi-scale approach improves performance. However, pursuing a reverse approach, a lower number of filters are required to obtain a similar performance as that of a direct approach. It may be concluded that it is beneficial to start with larger filters and incrementally include smaller filters to form a multi-scale BSDF representation.

Table 1 The effect of using a multi-scale representation ($T = 3$ and $N = 8$)

Scales used	Alpha (%)	Beta (%)	Gamma (%)	UCLA (%)
1	95.00	87.65	90.53	97.50
[1 2]	98.33	91.36	91.29	99.00
[1 3]	96.67	93.21	89.77	99.50
[1 4]	96.67	92.59	89.39	99.50
[1 5]	98.33	91.98	90.53	99.50
[1 6]	100.00	93.21	90.91	99.50
6	98.33	92.59	89.02	98.00
[5 6]	100.00	90.74	89.39	99.00
[4 6]	100.00	90.74	90.53	99.50
[3 6]	100.00	91.98	91.67	99.00
[2 6]	100.00	92.59	92.05	99.50
[1 6]	100.00	93.21	90.91	99.50

Table 2 Performance of the 16-bit BSDF ($N = 16$ and $T = 3$)

Scales used	Alpha (%)	Beta (%)	Gamma (%)	UCLA (%)
1	88.33	85.80	84.47	97.00
[1 2]	88.33	88.89	85.98	97.50
[1 3]	91.67	86.42	86.36	99.00
[1 4]	93.33	88.27	88.26	99.00
[1 5]	95.00	90.12	89.77	99.00
[1 6]	95.00	90.12	90.15	99.50
6	100.00	94.44	91.29	99.50
[5 6]	100.00	91.98	92.80	99.50
[4 6]	100.00	93.21	93.56	100.00
[3 6]	100.00	92.59	92.42	100.00
[2 6]	98.33	93.21	92.05	100.00
[1 6]	95.00	90.12	90.15	100.00

4.4 The effect of codeword lengths

By using only $N = 8$ leading eigenvectors for filter design, one may lose a lot of variance. In this respect, using more filters might be beneficial in forming the representation. In this section, this aspect of the BSDF representation is investigated where sixteen filters ($N = 16$) are inferred. resulting in a 2^{16}-bin histogram representation. The results of this analysis on the Dyntex and UCLA data sets are reported in Table 2. From Table 2, it is observed that on all data sets, using a 16-bit representation results in improved performance. A further observation from the tables is that as one uses a longer codeword, the number of scales to achieve the peak performance reduces. It can be concluded that a 16-bit variant of the proposed BSDF is superior in terms of both performance and the number of scales required to form an effective representation.

Table 3 Performance of the 16-bit BSDF representation when $T = X = Y$

Scales used	Alpha (%)	Beta (%)	Gamma (%)	UCLA (%)
1	88.33	85.19	86.36	88.00
[1 2]	91.67	87.65	86.36	97.50
[1 3]	91.67	86.42	89.77	99.50
[1 4]	93.33	88.27	88.64	99.50
[1 5]	93.33	87.04	89.77	100.00
[1 6]	93.33	87.04	89.39	100.00
6	98.33	89.51	89.77	98.00
[6 5]	98.33	91.36	92.42	99.50
[6 4]	98.33	91.36	92.80	99.50
[6 3]	96.67	91.36	93.18	99.50
[6 2]	95.00	91.36	91.29	99.50
[6 1]	93.33	87.04	89.39	100.00

Table 4 Performance of the 16-bit BSDF representation when $T = 13$

Scales used	Alpha (%)	Beta (%)	Gamma (%)	UCLA (%)
1	93.33	87.65	90.53	88.00
[1 2]	93.33	88.27	91.29	97.50
[1 3]	93.33	89.51	91.67	99.50
[1 4]	93.33	90.12	92.42	99.50
[1 5]	93.33	89.51	92.42	100.00
[1 6]	93.33	90.74	92.05	100.00
6	98.33	90.12	92.42	98.00
[6 5]	98.33	91.98	92.05	99.50
[6 4]	98.33	91.36	92.80	99.50
[6 2]	96.67	90.12	92.42	99.50
[6 1]	93.33	90.74	92.05	100.00

4.5 The effect of temporal depth

In this section, in order to examine the effect of using different temporal depths, two experiments are conducted. First, the temporal width of filters is fixed at 13 irrespective of their spatial dimensions. Next, the temporal width of a filter is varied in accordance with its spatial size, resulting in filters of sizes $d \times d \times d$. The results of this experiment are reported in Tables 3 and 4.

As the results reported in the corresponding tables for the case when $T = 3$ are typically better than or equal to those obtained with larger filters, it can be concluded that a temporal width of three ($T = 3$) is the optimal choice for constructing the BSDF representation.

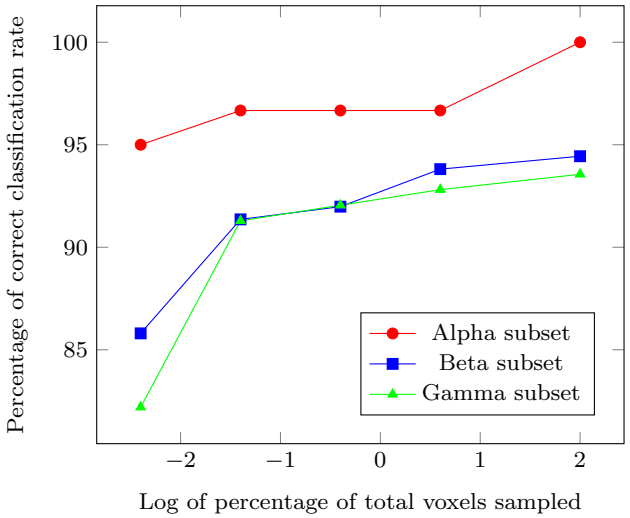

Fig. 2 The effect of number of random samples on recognition performance on the three subsets of the Dyntex database

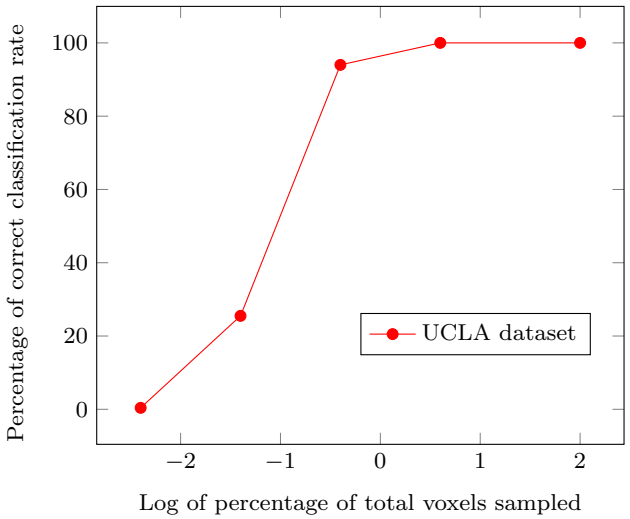

Fig. 3 The effect of number of random samples on recognition performance on the UCLA database

4.6 Random sampling

In this experiment, samples are drawn uniformly from an image sequence and the final BSDF representation is constructed using only the samples drawn to moderate the computational cost of the proposed approach. The results obtained are depicted in Figs. 2 and 3 for the Dyntex and UCLA data sets using the best performing parameters of the proposed representation, i.e. $N = 16$ and $T = 3$. From Figure 2, it is observed that the recognition performance of the BSDF representation on the Dyntex data set is quite stable over a wide range of sampling rates. For the UCLA data set, from Figure 3, it is again evident that the system performance is stable with respect to the number of samples drawn.

Table 5 Comparison of the performance of the proposed method to some other methods on the UCLA data set in the 50-class breakdown scenario

Method	UCLA (50-class scenario) (%)
The proposed approach	**100.0**
BSIF-TOP [5]	99.5
Martin Distance [44]	89.5
L2 Bhattacharyya [16]	81.0
PCANet-TOP [3]	99.5

Best performance indicated in bold

Table 6 Comparison of the performance of the proposed method to some other methods on the alpha, beta and gamma subsets of the Dyntex data set

Method	Alpha (%)	Beta (%)	Gamma (%)
The proposed approach	**100**	**93.21**	**93.56**
PCANet-TOP [3]	96.67	90.74	89.39
BSIF-TOP [5]	90	90.74	91.3
The method of [21]	88.3	69.8	68.3
LBP-TOP [52]	96.67	87.65	87.12

Best performances are indicated in bold

4.7 Comparison to the state-of-the-art methods

In this experiment, the proposed BSDF representation is constructed using a multi-scale 16-bit approach with a temporal width of $T = 3$. The multi-scale representation is constructed using filters of dimensions $13 \times 13 \times 3$, $11 \times 11 \times 3$ and $9 \times 9 \times 3$. A comparison of the proposed BSDF approach to other methods on the UCLA data set is provided in Table 5. As can be seen from the table, the proposed method obtains perfect recognition rate, ranking first among other methods. It is interesting to note that the proposed approach performs better than some other multi-layer convolutional networks such as [3]. The results of a comparison on the Dyntex data set are provided in Table 6. Table 7 reports the results of a statistical analysis for the significance of the results using the Friedman Test [17]. From the tables, one can observe that the proposed approach achieves the best performance among other competitors.

The PCANet-TOP method of [3] is a an example of multi-layer networks for DT recognition which is outperformed by the BSD approach presented in this work. In addition to the methods in Table 6, the deep convolutional network-based method proposed in [38] has been evaluated on the Dyntex database, obtaining 100%, 100% and 98.11% recognition performances on the alpha, beta and gamma subsets of the Dyntex database, respectively. The advantages of the proposed approach in this work over very deep approaches such as that of [38] are being faster in the application phase due to the lower number of parameters, being unsupervised,

Table 7 Average rankings of the algorithms (Friedman). Friedman statistic (distributed according to chi-square with 4 degrees of freedom: 10.066666666666663

Algorithm	Ranking
The proposed approach	1.0
PCANet-TOP [3]	2.67
BSIF-TOP [5]	2.83
The method of [21]	5.0
LBP-TOP [52]	3.5

P value computed by Friedman Test: 0.039319539964143835

requiring less training samples and computationally less intensive training procedure. Nevertheless, the price paid for the aforementioned advantages is a slightly lower recognition performance compared to very deep networks.

5 Conclusion and future work

The paper presented a new spatio-temporal descriptor (BSDF) for representation and classification of dynamic texture. The proposed descriptor operates by applying linear spatio-temporal filters on local voxels of a video sequence. Constructing the BSDF representation entails binarising filter responses and forming codes which are then summarised using histograms. Several aspects of the design procedure were analysed, and for reduced computational complexity, a random sampling approach was examined. For classification, a robust representation of the BSDF feature was obtained via a sparse representation approach. The proposed approach was evaluated on the most commonly used dynamic texture databases and was shown to perform very well compared to the existing methods.

As a future research direction, one may consider extending the proposed approach to benefit from colour information. In addition, one may consider the possibility of using a nonlinear ICA-based filter learning paradigm for improved performance.

References

1. http://www.yongxu.org/lunwen.html
2. Ali, W., Georgsson, F., Hellstrom, T.: Visual tree detection for autonomous navigation in forest environment. In: Intelligent Vehicles Symposium, 2008 IEEE, pp. 560–565 (2008). https://doi.org/10.1109/IVS.2008.4621315
3. Arashloo, S.R., Amirani, M.C., Noroozi, A.: Dynamic texture representation using a deep multi-scale convolutional network. J. Vis. Commun. Image Represent. **43**, 89–97 (2017). https://doi.org/10.1016/j.jvcir.2016.12.015
4. Arashloo, S.R., Kittler, J.: Hierarchical image matching for pose-invariant face recognition. In: Cavallaro, A., Prince, S., Alexander D. (eds.) BMVC. British Machine Vision Association, London, UK (2009)
5. Arashloo, S.R., Kittler, J.: Dynamic texture recognition using multiscale binarized statistical image features. IEEE Trans. Multimed. **16**(8), 2099–2109 (2014). https://doi.org/10.1109/TMM.2014.2362855
6. Baktashmotlagh, M., Harandi, M., Lovell, B.C., Salzmann, M.: Discriminative non-linear stationary subspace analysis for video classification. IEEE Trans. Pattern Anal. Mach. Intell. **36**(12), 2353–2366 (2014). https://doi.org/10.1109/TPAMI.2014.2339851
7. Beham, M.P., Roomi, S.M.M.: Anti-spoofing enabled face recognition based on aggregated local weighted gradient orientation. Signal Image Video Process. **12**(3), 531–538 (2018). https://doi.org/10.1007/s11760-017-1189-1
8. Bengio, Y., Courville, A., Vincent, P.: Representation learning: a review and new perspectives. IEEE Trans. Pattern Anal. Mach. Intell. **35**(8), 1798–1828 (2013). https://doi.org/10.1109/TPAMI.2013.50
9. Cannons, K.J., Gryn, J.M., Wildes, R.P.: Visual Tracking Using a Pixelwise Spatiotemporal Oriented Energy Representation, pp. 511–524. Springer, Berlin (2010)
10. Chan, A.B., Vasconcelos, N.: Probabilistic kernels for the classification of auto-regressive visual processes. In: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), vol. 1, pp. 846–851 (2005). https://doi.org/10.1109/CVPR.2005.279
11. Chan, K.L.: Saliency detection in video sequences using perceivable change encoded local pattern. Signal Image Video Process. **12**(5), 975–982 (2018). https://doi.org/10.1007/s11760-018-1242-8
12. Chan, T.H., Jia, K., Gao, S., Lu, J., Zeng, Z., Ma, Y.: Pcanet: a simple deep learning baseline for image classification? IEEE Trans. Image Process. **24**(12), 5017–5032 (2015). https://doi.org/10.1109/TIP.2015.2475625
13. Chen, J., Zhao, G., Salo, M., Rahtu, E., Pietikainen, M.: Automatic dynamic texture segmentation using local descriptors and optical flow. IEEE Trans. Image Process. **22**(1), 326–339 (2013). https://doi.org/10.1109/TIP.2012.2210234
14. Culibrk, D., Sebe, N.: Temporal dropout of changes approach to convolutional learning of spatio-temporal features. In: K.A. Hua, Y. Rui, R. Steinmetz, A. Hanjalic, A. Natsev, W. Zhu (eds.) ACM Multimedia, pp. 1201–1204. ACM (2014)
15. Derpanis, K.G., Wildes, R.P.: Classification of traffic video based on a spatiotemporal orientation analysis. In: 2011 IEEE Workshop on Applications of Computer Vision (WACV), pp. 606–613 (2011). https://doi.org/10.1109/WACV.2011.5711560
16. Derpanis, K.G.P., Wildes, R.: Spacetime texture representation and recognition based on a spatiotemporal orientation analysis. IEEE Trans. Pattern Anal. Mach. Intell. **34**(6), 1193–1205 (2012). https://doi.org/10.1109/TPAMI.2011.221
17. Derrac, J., Garca, S., Molina, D., Herrera, F.: A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms. Swarm Evol. Comput. **1**(1), 3–18 (2011). https://doi.org/10.1016/j.swevo.2011.02.002
18. Dollar, P., Rabaud, V., Cottrell, G., Belongie, S.: Behavior recognition via sparse spatio-temporal features. In: 2005 IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance, pp. 65–72 (2005). https://doi.org/10.1109/VSPETS.2005.1570899
19. Donoho, D.L., Tsaig, Y.: Fast solution of l1-norm minimization problems when the solution may be sparse. IEEE Trans. Inf. Theory **54**(11), 4789–4812 (2008). https://doi.org/10.1109/TIT.2008.929958
20. Doretto, G., Chiuso, A., Wu, Y.N., Soatto, S.: Dynamic textures. Int. J. Comput. Vis.: IJCV **51**(2), 91–109 (2003)

21. Dubois, S., Pteri, R., Mnard, M.: Characterization and recognition of dynamic textures based on the 2d+t curvelet transform. Signal Image Video Process. **9**(4), 819–830 (2015). https://doi.org/10.1007/s11760-013-0532-4
22. Fitzgibbon, A.W.: Stochastic rigidity: image registration for nowhere-static scenes. In: Proceedings. Eighth IEEE International Conference on Computer Vision, 2001. ICCV 2001, vol. 1, pp. 662–669 (2001). https://doi.org/10.1109/ICCV.2001.937584
23. Ghanem, B., Ahuja, N.: Phase based modelling of dynamic textures. In: 2007 IEEE 11th International Conference on Computer Vision, pp. 1–8 (2007). https://doi.org/10.1109/ICCV.2007.4409094
24. Ghanem, B., Ahuja, N.: Extracting a fluid dynamic texture and the background from video. In: IEEE Conference on Computer Vision and Pattern Recognition, 2008. CVPR 2008, pp. 1–8 (2008). https://doi.org/10.1109/CVPR.2008.4587547
25. Haas, M., Rijsdam, J., Thomee, B., Lew, M.S.: Relevance feedback: perceptual learning and retrieval in bio-computing, photos, and video. In: Proceedings of the 6th ACM SIGMM International Workshop on Multimedia Information Retrieval, MIR '04, pp. 151–156. ACM, New York, NY, USA (2004). https://doi.org/10.1145/1026711.1026737
26. van Hateren, J.H., Ruderman, D.L.: Independent component analysis of natural image sequences yields spatio-temporal filters similar to simple cells in primary visual cortex. Proc. R. Soc. Biol. Sci. **265**(1412), 2315–2320 (1998). https://doi.org/10.1098/rspb.1998.0577
27. Hyvarinen, A.: Fast and robust fixed-point algorithms for independent component analysis. IEEE Trans. Neural Netw. **10**(3), 626–634 (1999). https://doi.org/10.1109/72.761722
28. Hyvrinen, A., Hurri, J., Hoyer, P.O.: Natural Image Statistics: A Probabilistic Approach to Early Computational Vision, 1st edn. Springer, Berlin (2009)
29. Ji, H., Yang, X., Ling, H., Xu, Y.: Wavelet domain multifractal analysis for static and dynamic texture classification. IEEE Trans. Image Process. **22**(1), 286–299 (2013). https://doi.org/10.1109/TIP.2012.2214040
30. Junejo, I.N., Bhutta, A.A., Foroosh, H.: Single-class svm for dynamic scene modeling. Signal Image Video Process. **7**(1), 45–52 (2013). https://doi.org/10.1007/s11760-011-0230-z
31. Kannala, J., Rahtu, E.: Bsif: Binarized statistical image features. In: 2012 21st International Conference on Pattern Recognition (ICPR), pp. 1363–1366 (2012)
32. Kung, T.J., Richards, W.: Inferring "water" from images. In: Richards, W. (ed.) Natural Computation, Chap. 16, pp. 224–233. M.I.T. Press, Cambridge, MA (1988)
33. Mumtaz, A., Coviello, E., Lanckriet, G.R.G., Chan, A.B.: Clustering dynamic textures with the hierarchical em algorithm for modeling video. IEEE Trans. Pattern Anal. Mach. Intell. **35**(7), 1606–1621 (2013). https://doi.org/10.1109/TPAMI.2012.236
34. Nanni, L., Brahnam, S., Lumini, A.: Local ternary patterns from three orthogonal planes for human action classification. Expert Syst. Appl. **38**(5), 5125–5128 (2011). https://doi.org/10.1016/j.eswa.2010.09.137
35. Osborne, M., Presnell, B., Turlach, B.: A new approach to variable selection in least squares problems. IMA J. Numer. Anal. **20**(3), 389 (2000)
36. Päivärinta, J., Rahtu, E., Heikkilä, J.: Volume Local Phase Quantization for Blur-Insensitive Dynamic Texture Classification, pp. 360–369. Springer, Berlin (2011)
37. Péteri, R., Fazekas, S., Huiskes, M.J.: DynTex : a comprehensive database of dynamic textures. Pattern Recogn. Lett. https://doi.org/10.1016/j.patrec.2010.05.009
38. Qi, X., Li, C.G., Zhao, G., Hong, X., Pietikainen, M.: Dynamic texture and scene classification by transferring deep image features. Neurocomputing **171**, 1230–1241 (2016). https://doi.org/10.1016/j.neucom.2015.07.071
39. Qiao, Y., Weng, L.: Hidden markov model based dynamic texture classification. IEEE Signal Process. Lett. **22**(4), 509–512 (2015). https://doi.org/10.1109/LSP.2014.2362613
40. Quan, Y., Huang, Y., Ji, H.: Dynamic texture recognition via orthogonal tensor dictionary learning. In: 2015 IEEE International Conference on Computer Vision (ICCV), pp. 73–81 (2015). https://doi.org/10.1109/ICCV.2015.17
41. Ravichandran, A., Chaudhry, R., Vidal, R.: View-invariant dynamic texture recognition using a bag of dynamical systems. In: IEEE Conference on Computer Vision and Pattern Recognition, 2009. CVPR 2009, pp. 1651–1657 (2009). https://doi.org/10.1109/CVPR.2009.5206847
42. Ravichandran, A., Chaudhry, R., Vidal, R.: Categorizing dynamic textures using a bag of dynamical systems. IEEE Trans. Pattern Anal. Mach. Intell. **35**(2), 342–353 (2013). https://doi.org/10.1109/TPAMI.2012.83
43. Rivera, A.R., Chae, O.: Spatiotemporal directional number transitional graph for dynamic texture recognition. IEEE Trans. Pattern Anal. Mach. Intell. **37**(10), 2146–2152 (2015). https://doi.org/10.1109/TPAMI.2015.2392774
44. Saisan, P., Doretto, G., Wu, Y.N., Soatto, S.: Dynamic texture recognition. In: Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2001. CVPR 2001, vol. 2, pp. II-58–II-63 (2001). https://doi.org/10.1109/CVPR.2001.990925
45. Thriault, C., Thome, N., Cord, M.: Dynamic scene classification: learning motion descriptors with slow features analysis. In: 2013 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2603–2610 (2013). https://doi.org/10.1109/CVPR.2013.336
46. Wang, Y., Chun Zhu, S.: Modeling textured motion: particle, wave and sketch. In: IEEE International Conference on Computer Vision, ICCV'03, pp. 213–220 (2003)
47. Wildes, R.P., Bergen, J.R.: Qualitative Spatiotemporal Analysis Using an Oriented Energy Representation, pp. 768–784. Springer, Berlin (2000)
48. Woolfe, F., Fitzgibbon, A.W.: Shift-invariant dynamic texture recognition. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV (2), Lecture Notes in Computer Science, vol. 3952, pp. 549–562. Springer, Berlin (2006)
49. Xie, J., Fang, Y.: Dynamic texture recognition with video set based collaborative representation. Image Vis. Comput. **55**(Part 2), 86–92 (2016)
50. Zhang, Z., Xu, Y., Yang, J., Li, X., Zhang, D.: A survey of sparse representation: algorithms and applications. IEEE Access **3**, 490–530 (2015). https://doi.org/10.1109/ACCESS.2015.2430359
51. Zhao, G., Barnard, M., Pietikainen, M.: Lipreading with local spatiotemporal descriptors. IEEE Trans. Multimed. **11**(7), 1254–1265 (2009). https://doi.org/10.1109/TMM.2009.2030637
52. Zhao, G., Pietikainen, M.: Dynamic texture recognition using local binary patterns with an application to facial expressions. IEEE Trans. Pattern Anal. Mach. Intell. **29**(6), 915–928 (2007). https://doi.org/10.1109/TPAMI.2007.1110