Spatiotemporal Derivative Pattern: A Dynamic Texture Descriptor for Video Matching

Farshid Hajati[1], Mohammad Tavakolian[1], Soheila Gheisari[1,2], and Ajmal Saeed Mian[3]

[1]Electrical Engineering Department, Tafresh University, Tafresh, Iran
{hajati,m_tavakolian}@tafreshu.ac.ir
[2]Electrical Eng. Dep., Central Tehran Branch, Islamic Azad University, Tehran, Iran
s.gheisari@iauctb.ac.ir
[3]Computer Science and Software Engineering, The University of Western Australia, WA 6009, Australia
ajmal.mian@uwa.edu.au

Abstract. We present Spatiotemporal Derivative Pattern (SDP), a descriptor for dynamic textures. Using local continuous circular and spiral neighborhoods within video segments, SDP encodes the derivatives of the directional spatiotemporal patterns into a binary code. The main strength of SDP is that it uses fewer frames per segment to extract more distinctive features for efficient representation and accurate classification of the dynamic textures. The proposed SDP is tested on the Honda/UCSD and the YouTube face databases for video based face recognition and on the Dynamic Texture database for dynamic texture classification. Comparisons with existing state-of-the-art methods show that the proposed SDP achieves the overall best performance on all three databases. To the best of our knowledge, our algorithm achieves the highest results reported to date on the challenging YouTube face database.

1 Introduction

Automatic visual motion analysis has attracted the interest of many researchers [1, 2]. In nature, visual motions are classified into three categories [3]: motions, activities, and dynamic textures. Motions are one-time occurring phenomena, such as a door closing, that are not repetitive in either spatial or temporal domains. Activities are events, such as running, that are temporally periodic while spatially restricted. Dynamic textures present a statistical regularity having indeterminate spatial and temporal extent [4]. In other words, image sequences of moving scenes that exhibit certain stationary time properties are defined as dynamic texture [5]. Dynamic textures such as smoke, fire, sea-waves, blowing flags, and waterfalls are periodic in the temporal domain and repetitive in the spatial domain. In some cases, dynamic textures such as smoke can be partially transparent and the object's spatiotemporal appearance may change over time. In other cases, the shape and appearance of an object may be fixed but the camera may exhibit motions such as zooming, rotation, or panning. Figure 1

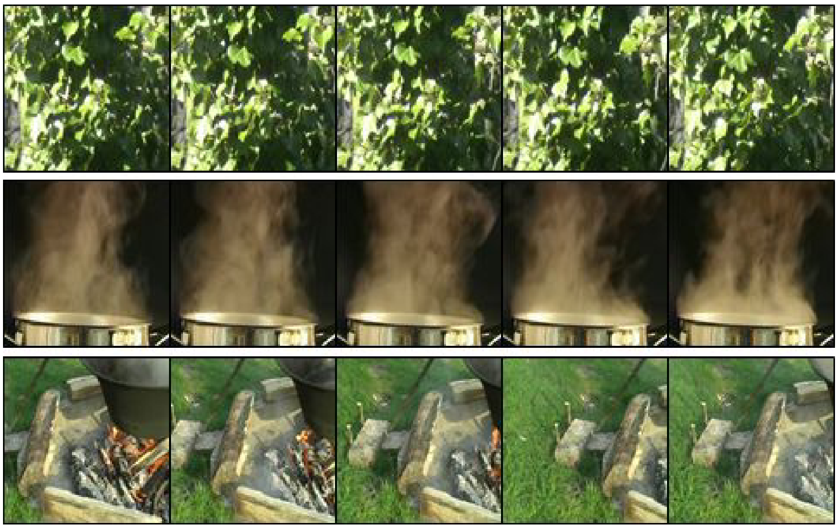

Fig. 1. Examples of the dynamic texture from the DynTex database [6]. **Top:** shaking leaves as dynamic texture with fixed shape. **Middle:** rising steam as a translucent dynamic texture. **Bottom:** the camera's panning.

illustrates three examples of dynamic textures; shaking leaves, rising steam, and moving camera.

Representation of dynamic textures has attracted the attention of computer vision community because of its applications in surveillance systems, video retrieval, space-time texture synthesis, and image registration. Dynamic textures are able to abstract a wide range of complicated appearances and motions into a spatiotemporal model. Research has shown that the redundancy contained in dynamic textures can improve an algorithm's performance [7]. Considering the above mentioned intrinsic properties of dynamic textures, more effective representations can be obtained from image sequences. However, due to its unknown and stochastic spatial and temporal properties, dynamic texture representation is more challenging compared to static textures.

2 Related Work

Earlier dynamic texture approaches were based on still images. Their goal was to select representative frames from a given sequence and applying traditional still-image based algorithms on the selected frames. Principle Component Analysis (PCA) [8], Linear Discriminant Analysis (LDA) [9], Local Binary Pattern (LBP) [10], Locality Preserving Projections (LPP) [11], and Local Directional Number pattern (LDN) [12] are examples of the still-image based methods that have been used in dynamic texture recognition. The major drawback of these methods is that they are not able to capture the temporal periodic properties of the image sequences. Moreover, when there are only a few frames per video in the training set, the performance of the still-image based methods decreases dramatically and popular still image methods such as LBP cannot extract discriminative features for matching.

Existing methods in dynamic texture representation can be divided into two categories: those which completely ignore the temporal relationships between the frames and those which assume that adjacent frames are temporally contiguous. The first category mainly consists of image set classification approaches [13, 14] that only consider the spatial cues and treat the images as points on a high dimensional manifold. In image set classification, each class is represented by multiple images and the algorithm assigns a label to the query image set by measuring the minimum distance to the gallery sets. Hu et al. [15] proposed the Sparse Approximate Nearest Point (SANP) for image set classification. SANPs are the nearest points of two image sets that can be sparsely approximated by a subset of the corresponding image set. Wang et al. [16] proposed Covariance Discriminative Learning (CDL) to represent image sets as a covariance matrix. They treat the image set classification problem as the problem of point classification on a Riemannian manifold spanned by symmetric positive-definite matrices. Wang et al. [17] represented an image set by a nonlinear manifold and proposed Manifold to Manifold Distance (MMD) to measure the similarity between manifolds. A manifold learning technique which represented a manifold by a set of locally linear subspaces was proposed to compute the MMD. They used MMD to integrate the distances between each pair of subspaces. Coviello et al. [18] introduced Bag-of-Systems (BoS) representation for motion description in the dynamic texture. In their framework, the dynamic texture codewords represents the typical motion patterns in spatiotemporal patches extracted from the video. They proposed the BoS Tree which constructs a bottom-up hierarchy of codewords that enables mapping of videos to the BoS codebook.

The methods in the second category attempt to represent the dynamic textures by capturing both spatial and temporal cues of the image sequence. We can further divide these approaches into holistic and local methods. Liu et al. [19] used an adaptive Hidden Markov Model (HMM) [20] to learn the statistics and the temporal dynamics of the training video. Global temporal features of the query video were represented over time and the recognition task is performed by the likelihood score computed using the HMMs' comparison. Rahman et al. [1] proposed a motion-based temporal texture characterization technique using first-order global motion co-occurrence features. Doretto et al. [21] extended the Active Appearance Model (AAM) [22] to represent dynamic shapes, motions, and appearances globally by conditionally linear models using the joint variations of shape and appearance of portions in the image sequence. Derpanis et al. [23] investigated the impact of multi-scale orientation measurements on scene classification. These measurements in visual space, $x-y$, and spacetime, $x-y-t$, were recovered by a bank of spatiotemporal oriented energy filters.

Compared to global representations, local spatiotemporal representations can capture temporal relationships more effectively [2]. Xu et al. [24] assumed dynamic textures as the product of nonlinear stochastic dynamic systems. Based on the assumption, they proposed a dynamic texture descriptor using an extension of dynamic fractal analysis called Dynamic Fractal Spectrum (DFS). DFS captures the stochastic self-similarities in the local structure of the dynamic textures for classification. Zhao et al. [2] proposed two variants of the LBP features namely Volume Local Binary Pattern (VLBP) and Local Binary Pattern from Three Orthogonal Planes (LBP-TOP). VLBP is an extension of the LBP to dynamic textures by combining the appearance and the motion. VLBP is extracted by considering a local volumetric neighborhood around each pixel. Comparing the gray-level of the center pixel and the neighbors, VLBP generates a binary representative code. Local Binary Pattern from Three Orthogonal Planes (LBP-TOP) models the textures by concatenating the extracted LBPs on the three orthogonal planes ($x-y$ plane, $x-t$ plane, and $y-t$ plane) and considering the co-occurrence statistics on the three planes. The main drawback of VLBP and LBP-TOP is that they only represent the first-order derivative pattern of an image sequence and cannot extract detailed information of the image sequence. Moreover, they sample the image for coding on a discrete rectangular grid which can have aliasing effects. However, high-order derivatives, such as second order derivatives, capture more detailed spatial and temporal discriminant information contained in the image sequence. Similarly, extracting derivative patterns based on continuous circular neighborhoods are likely to generate more robust features.

We present a novel dynamic texture descriptor called Spatiotemporal Derivative Pattern (SDP) that overcomes the above two limitations. SDP describes dynamic textures by characterizing image sequences by a feature vector computed from the high-order spatiotemporal derivative patterns within local neighborhoods. Neighborhoods are defined using continuous circular and spiral regions to avoid the aliasing effects. Using the high-order spatiotemporal derivatives, SDP captures more detailed information about the given image sequence. Comprehensive experiments are conducted on the Honda/UCSD [25], the YouTube [26], and the DynTex [6] datasets. Comparisons with existing state-of-the-art techniques show the effectiveness of the proposed method for dynamic texture analysis. To the best of our knowledge, our proposed technique achieves the highest recognition rate reported on the challenging YouTube database [26]. Our experiments demonstrate that the SDP needs fewer numbers of frames for dynamic texture recognition compared to existing methods.

3 Spatiotemporal Derivative Pattern (SDP)

Video-based dynamic texture recognition is a sequential process where every incoming frame adds to the information provided by the previous frames [27]. However, the limitation of system memory is an important issue in dynamic texture recognition. Therefore, representing the dynamic texture with the least number of frames is desirable but challenging at the same time. This can be done by considering only the M number of contiguous frames that optimally

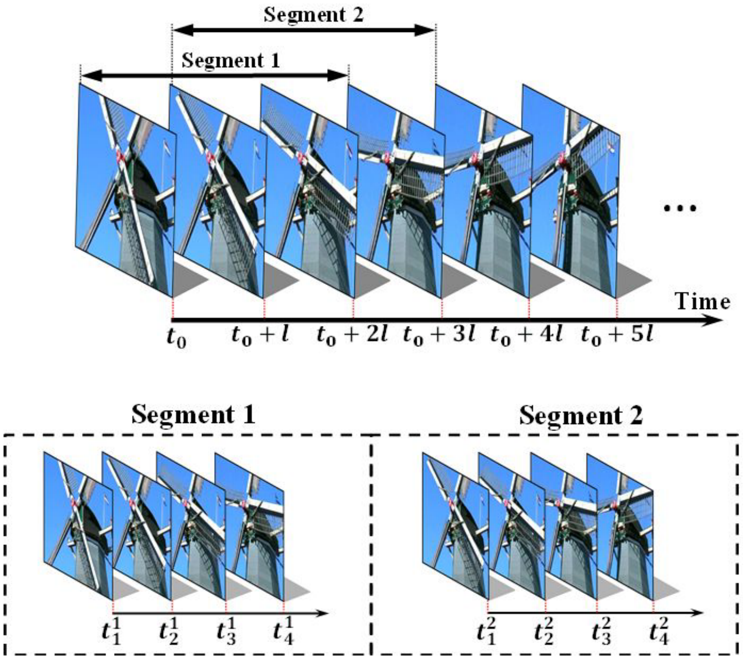

Fig. 2. Definition of segments in an image sequence from the DynTex database [6].

represent the scene. SDP characterizes the image segments by encoding them into binary patterns based on the spatiotemporal directional variations within segments. A segment is a subset of the image sequence with a determined number of frames. Figure 2 illustrates the definition of two 4-frame segments in a given image sequence.

We partition the given image sequence into M-frame overlapping segments (see Figure 2). Let $f(x, y, t)$ be a texture point in the given image sequence, where x and y are the spatial coordinates of the texture point, and t denotes the time of the frame in which the texture point is located. The k-th frame of the s-th segment, $f_{s,k}(x, y, t_k^s)$, is defined as

$$f_{s,k}(x, y, t_k^s) = f(x, y, t_0 + (s + k - 2)l) \qquad (1)$$
$$s = 1, 2, \cdots, (L - M + 1) \ \ \& \ \ k = 1, \cdots, M$$

where L is the length of the image sequence and l is the time interval between two successive frames. M and s denote the total number of frames in the segment and the segment's index number, respectively. t_k^s denotes the time of the k-th frame in the s-th segment.

We define the first frame of each segment as the reference frame of the segment. The SDP algorithm considers the texture points of the s-th segment's

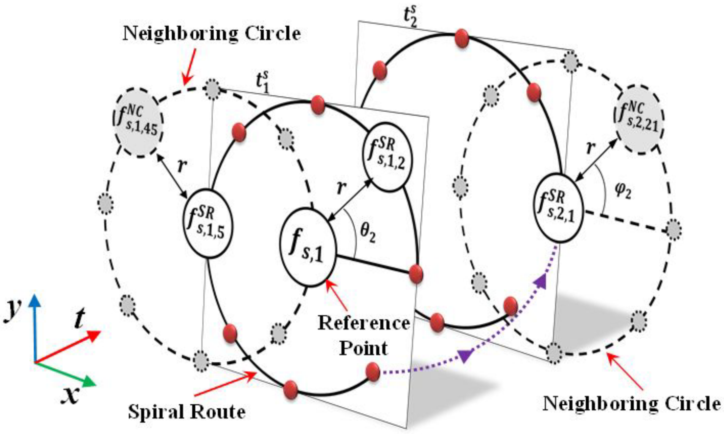

Fig. 3. Defining the spiral route and the neighboring circle in a two-frame segment. A neighboring (planar) circle is defined for every point in the spiral route.

reference frame, $f_{s,1}(x, y, t_1^s)$, as the reference point of a spatiotemporal 'Spiral Route' with radius r. The spiral route starts from the reference point and passes through all the segment's frames, making a spiral path. A spiral route with radius r has $8r$ points in each frame; the total number of spiral route's points in a M-frame segment is $8r \times M$. Figure 3 illustrates the spiral route's points (solid points) and the reference point in a two-frame segment. The spiral route specifies a local spatiotemporal neighborhood for the reference point. To compute the local spatiotemporal derivatives within segments, we consider another circular neighborhood for each point of the spiral route with a similar radius r. We call this circular neighborhood as the 'Neighboring Circle'. We consider $8r$ neighboring points on a neighboring circle with radius r. Examples of the neighboring circle are also illustrated in Figure 3. Since the effectiveness of the circular neighborhood has been proven in 2D image description [28], we use circular schemes in both the space and the time domains to represent dynamic textures.

We move the neighboring circle on the spiral route and compute the derivatives using the points along its circumference. Using the k-th frame of s-th segment, $f_{s,k}(x, y, t_k^s)$, the i-th texture point in the k-th frame of the spiral route, $f^{SR}_{s,k,i}(x_i, y_i, t_k^s)$, is defined as

$$f^{SR}_{s,k,i}(x_i, y_i, t_k^s) = f_{s,k}(x + r\cos(\theta_i), y + r\sin(\theta_i), t_k^s) \tag{2}$$
$$i = 1, \cdots, 8r \;\;\&\;\; k = 1, \cdots, M$$

where t_k^s, is the time of the k-th frame in the s-th segment. r is the radius of the spiral route and $\theta_i = 2\pi(i-1)/8r$ is the angle of the i-th spiral route's point with respect to the x axis. x_i and y_i are the coordinates of the i-th texture point of the spiral route.

As already mentioned, each texture point of the spiral route is surrounded by $8r$ neighbors on the neighboring circle. Denoting the k-th frame of the s-th segment as $f_{s,k}(x, y, t_k^s)$, the j-th neighboring texture points of the i-th spiral route's point in the s-th segment, $f_{s,k,ji}^{NC}(x_j, y_j, t_k^s)$, is defined as

$$f_{s,k,ji}^{NC}(x_j, y_j, t_k^s) = f_{s,k}(x_i + r\cos(\varphi_{ji}), y_i + r\sin(\varphi_{ji}), t_k^s) \tag{3}$$
$$i = 1, \cdots, 8r \;\;\&\;\; j = 1, \cdots, 8r \;\;\&\;\; k = 1, \cdots, M$$

where $\varphi_{ji} = 2\pi(j-1)/8r$ is the angle of the j-th neighboring texture point of the i-th spiral route's point with respect to the x axis.

SDP considers the spatiotemporal directional texture transitions within segments for description. Here, we have two directions: θ and φ. Using the texture points of the spiral route and the neighboring circles, the first-order spatiotemporal directional derivatives along θ direction, $\partial f_{s,1,a}(x, y, t_1^s)/\partial\theta$, and along φ direction, $\partial f_{s,k,bc}(x, y, t_k^s)/\partial\varphi$, are defined as

$$\partial f_{s,1,a}(x, y, t_1^s)/\partial\theta = f_{s,1}(x, y, t_1^s) - f_{s,1,a}^{SR}(x_a, y_a, t_1^s) \tag{4}$$
$$a = 1, \cdots, 4r$$
$$\partial f_{s,k,bc}(x, y, t_k^s)/\partial\varphi = f_{s,k,bc}^{NC}(x_b, y_b, t_k^s) - f_{s,k,c}^{SR}(x_c, y_c, t_k^s) \tag{5}$$
$$c = 1, \cdots, 8r \;\;\&\;\; b = 1, \cdots, 4r$$

where $f_{s,1}(x, y, t_1^s)$ denotes the texture point of the s-th segment's reference point and $f_{s,1,a}^{SR}(x_a, y_a, t_1^s)$ is the a-th spiral route's texture point in the reference frame. In order to compute the directional derivatives, we only consider the first $4r$ points ($a = 1, \cdots, 4r$) on the spiral route in the reference frame and the first $4r$ points ($b = 1, \cdots, 4r$) on each neighboring circle. The remaining neighboring points are considered in the derivative calculation by changing the spatial coordinates of the reference point.

We compute the second-order spatiotemporal directional derivatives from the first-order derivatives. The second-order spatiotemporal directional derivatives along θ direction, $\partial^2 f_{s,1,a}(x, y, t_1^s)/\partial\theta^2$, and along φ direction, $\partial^2 f_{s,k,bc}(x, y, t_k^s)/\partial\varphi^2$, are computed as

$$\partial^2 f_{s,1,a}(x, y, t_1^s)/\partial\theta^2 = \partial f_{s,1}(x, y, t_1^s)/\partial\theta \;\; - \partial f_{s,1,a}^{SR}(x_a, y_a, t_1^s)/\partial\theta \tag{6}$$
$$a = 1, \cdots, 4r$$
$$\partial^2 f_{s,k,bc}(x, y, t_k^s)/\partial\varphi^2 = \partial f_{s,k,bc}^{NC}(x_b, y_b, t_k^s)/\partial\varphi \;\; - \partial f_{s,k,c}^{SR}(x_c, y_c, t_k^s)/\partial\varphi \tag{7}$$
$$c = 1, \cdots, 8r \;\;\&\;\; b = 1, \cdots, 4r$$

where $\partial f_{s,1}(x, y, t_1^s)/\partial\theta$, $\partial f_{s,1,a}^{SR}(x_a, y_a, t_1^s)/\partial\theta$, and $\partial f_{s,k,bc}^{NC}(x_b, y_b, t_k^s)/\partial\varphi$ are the first-order directional derivative along θ direction in the reference point, the first-order directional derivative of the a-th spiral route's texture point along θ direction in the reference frame, and the first-order directional derivative of the b-th neighbor of the c-th spiral route's texture points along φ direction, respectively, computed as

$$\partial f_{s,1}(x, y, t_1^s)/\partial\theta = f_{s,1}(x, y, t_1^s) - f_{s,1}(x + r\cos(\theta), y + r\sin(\theta), t_1^s) \tag{8}$$

$$\partial f^{SR}_{s,k,a}(x_a, y_a, t^s_k)/\partial\theta = f^{SR}_{s,k,a}(x_a, y_a, t^s_k) - f_{s,k}(x_a + r\cos(\theta), y_a + r\sin(\theta), t^s_k) \tag{9}$$

$$\partial f^{NC}_{s,k,bc}(x_b, y_b, t^s_k)/\partial\varphi = f^{NC}_{s,k,bc}(x_b, y_b, t^s_k) - f_{s,k}(x_b + r\cos(\varphi), y_b + r\sin(\varphi), t^s_k) \tag{10}$$

Generally, the n^{th}-order spatiotemporal directional derivatives are computed from the $(n-1)^{th}$-order spatiotemporal directional derivatives along θ and φ direction using the following recursive equations:

$$\partial^n f_{s,1,a}(x, y, t^s_1)/\partial\theta^n = \partial^{n-1} f_{s,1}(x, y, t^s_1)/\partial\theta^{n-1} - \partial^{n-1} f^{SR}_{s,1,a}(x_a, y_a, t^s_1)/\partial\theta^{n-1} \quad a = 1, \cdots, 4r \tag{11}$$

$$\partial^n f_{s,k,bc}(x, y, t^s_k)/\partial\varphi^n = \partial^{n-1} f^{NC}_{s,k,bc}(x_b, y_b, t^s_k)/\partial\varphi^{n-1} - \partial^{n-1} f^{SR}_{s,k,c}(x_c, y_c, t^s_k)/\partial\varphi^{n-1} \quad c = 1, \cdots, 8r \;\; \& \;\; b = 1, \cdots, 4r \tag{12}$$

where $\partial^{n-1} f_{s,1}(x, y, t^s_1)/\partial\theta^{n-1}$ and $\partial^{n-1} f^{SR}_{s,1,a}(x_a, y_a, t^s_1)/\partial\theta^{n-1}$ are the $(n-1)^{th}$-order directional derivative along θ direction in the reference point and the $(n-1)^{th}$-order directional derivative of the a-th spiral route's texture point along θ direction in the reference frame, respectively. $\partial^{n-1} f^{NC}_{s,k,bc}(x_b, y_b, t^s_k)/\partial\varphi^{n-1}$ denotes the $(n-1)^{th}$-order spatiotemporal directional derivative of the b-th neighbor of the c-th spiral route's texture points along φ direction.

We encode the computed derivatives into binary bits using the unit step function, thereby forming a binary bit pattern analogous to LBP features. Essentially, we are only encoding the direction of derivative which is more robust to changes compared to the derivative value itself. By concatenating the coded spatiotemporal directional derivatives, we define the n^{th}-order Derivative Pattern (DP) along $\alpha_b = 2\pi(b-1)/8r$ direction for the s-th segment of the given image sequence, $DP^{(n)}_{s,\alpha_b}(f(x, y, t))$, as

$$DP^{(n)}_{s,\alpha_b}(f(x, y, t)) = \{u(\frac{\partial^n f_{s,1,b}(x, y, t^s_1)}{\partial\alpha^n_b} \times \frac{\partial^n f_{s,k,bc}(x, y, t^s_k)}{\partial\alpha^n_b}) \;|c = 1, \cdots, 8r\,;\, k = 1, \cdots, M\} \quad b = 1, \cdots, 4r \tag{13}$$

where $u(\cdot)$ is the unit step function which encodes the transitions' direction.

Using the n^{th}-order derivative pattern along α_b direction, we compute the n^{th}-order Spatiotemporal Derivative Pattern (SDP) within the s-th segment along α_b direction as

$$SDP^{(n)}_{s,\alpha_b}(f(x, y, t)) = \frac{1}{2^{8r\times M}} \sum_{q=1}^{8r\times M} 2^{(8r\times M)-q} \times DP^{(n)}_{s,\alpha_b,q}(f(x, y, t)) \quad b = 1, \cdots, 4r \tag{14}$$

where $DP^{(n)}_{s,\alpha_b,q}(f(x, y, t))$ is the q-th component of the n^{th}-order derivative pattern along α_b direction computed using Equation (13).

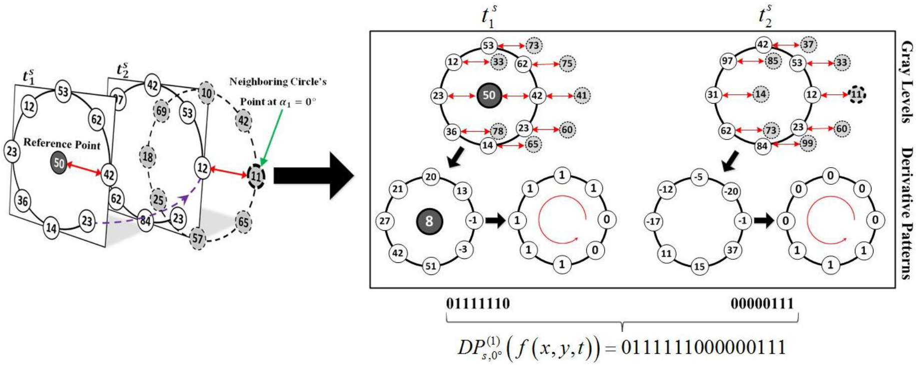
t_1^s
t_2^s
Reference Point
Neighboring Circle's Point at $\alpha_1 = 0°$
Gray Levels
Derivative Patterns
01111110
00000111
$DP^{(1)}_{s,0°}(f(x,y,t)) = 0111111000000111$

Fig. 4. Example of obtaining the first-order SDP along 0° direction in a two-frame segment. **Left:** the texture points on the spiral route and a sample neighboring circle. **Right:** derivative pattern calculation in 2D plane.

An example of computing the first-order SDP along $\alpha_1 = 0°$ direction in a two-frame segment is illustrated in Figure 4. For better illustration, the neighboring circles are omitted in the right and just the points on each neighboring circle along the 0° direction are shown. The shaded circles denote the gray level of the points on the neighboring circle along the 0° direction. The first-order spatiotemporal directional derivatives are calculated using Equations (4) and (5). Then, a 16-bit binary code $DP^{(1)}_{s,0°}(f(x,y,t)) = 0111111000000111$ is generated by concatenating the two 8-bit derivative patterns from the two frames in the segment. Using Equation (14), the first-order SDP along 0° direction, $SDP^{(1)}_{s,0°}(f(x,y,t))$, is determined as 0.4923.

According to the described algorithm, SDP generates a feature vector for each segment of the image sequence along α_b $(b = 1, \cdots, 4r)$ directions. Figure 5 illustrates the extracted SDPs along 0° direction for a 10-frame segment from the Honda/UCSD face database [25]. As can be seen, more details are extracted from the image sequence as the order of the derivatives increases. By increasing the radius r, the number of texture points on the spiral route and the neighboring circle increases and the derivatives are taken in a larger local area. Hence, the accuracy of the descriptor decreases and less detailed information is extracted from the given image sequence.

SDP extracts high-order local features for each segment's texture points. We model the distribution of the SDP by the spatial histogram [29] because of its robustness against variations [28]. For this purpose, the SDP along α_b $(b = 1, \cdots, 4r)$ directions is partitioned into P non-overlapping equal-sized square regions represented by $R_1, \cdots, R_P$ and the spatial histograms of the regions are concatenated using Equation (15).

$$HSDP_{s,\alpha_b} = \{H_{SDP_{s,\alpha_b}}(R_p) | p = 1, \cdots, P\} \qquad (15)$$
$$b = 1, \cdots, 4r$$

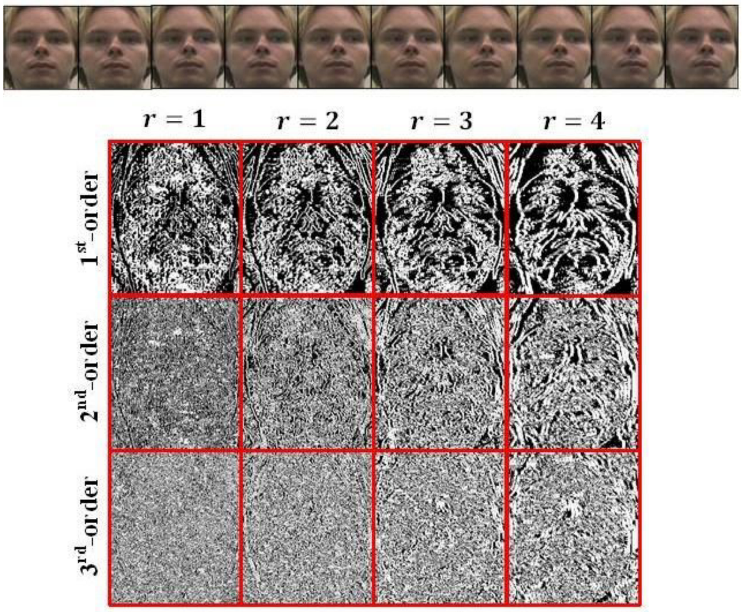

Fig. 5. Visualization of SDPs along 0° direction for a 10-frame segment from Honda/UCSD face database. **Top:** the given segment. **Bottom:** SDPs with different orders and radii.

where $HSDP_{s,\alpha_b}$ denotes the spatial histogram of the SDP of the s-th segment along α_b direction, and $H_{SDP_{s,\alpha_b}}(R_p)$ is the spatial histogram of the p-th region in the SDP of the s-th segment along α_b direction. After calculating the spatial histograms along all directions, we concatenate the computed histograms to extract a histogram vector for the whole segment as

$$HSDP_s(f(x,y,t)) = \{HSDP_{s,\alpha_b} | b = 1, \cdots, 4r\} \tag{16}$$

where $HSDP_s(f(x,y,t))$ denotes the histogram vector of the s-th segment in the given image sequence.

We partition a given image sequence into M-frame query segments using Equation (1). For each query segment, the matching is performed by considering the minimum Euclidean distance between the histogram of the segment computed using Equation (16) and the histogram of M-frame model segments in the gallery. The model in the gallery with the minimum distance is considered as the correct match.

4 Experimental Results

We evaluate the performance of the proposed method using the Honda/UCSD [25], the YouTube [26], and the DynTex [6] databases. The Honda/UCSD and the YouTube databases are designed for video-based face recognition while the DynTex dataset is used for dynamic texture recognition task.

4.1 Parameters Determination

The spiral route and the neighboring circles' radius, r, and the SDP's order are two free parameters which are determined experimentally using Honda/UCSD database. Once their optimal values are determined, we use the same values for all experiments.

The Honda/UCSD video database is widely used for evaluating face tracking and recognition algorithms. It contains 59 video sequences of 20 different subjects. The video sequences are recorded in an indoor environment for at least 15 seconds at 15 frames per second. The resolution of each video sequence is 640×480 in AVI format. All the video sequences contain both in-plane and in-depth head rotations. In this paper, we use the standard training/test configuration provided in [25]; 20 sequences (one per subject) are used as the gallery and the remaining 39 sequences are used as the probes. All gallery and probe sequences are partitioned into M-frame overlapping segments using Equation (1). To have a better comparison with the benchmarks, all the faces in the video segments are detected, cropped, and resized to 40×40 frames using the algorithm proposed in [15].

The average rank-1 recognition rate of the proposed algorithm is computed using SDPs with different radius of the spiral route and the neighboring circles, r, versus the SDP's order (see Figure 6). In this experiment, we computed the average rank-1 recognition rate of SDP for different segment's length (i.e. 5, 10, 50, and 100 frames and full length of image sequence). As can be seen, the recognition rate is significantly improved when the order of local pattern is increased from the first-order SDP to the second-order SDP for all radii. Then, the performance drops when the SDP's order increases. On the other hand, as the radius of the spiral route and the neighboring circles increase the accuracy of the SDP drops. This means that SDP captures more detailed information in small local regions. The results prove the effectiveness of the second-order SDP in extracting more distinctive features from the given video segment. Moreover, the best value of the radius of the spiral route and the neighboring circles' radius is $r = 1$. Therefore, we used the second-order SDP with radius $r = 1$ in all the remaining experiments.

4.2 Results on the Honda/UCSD Database

The rank-1 recognition rate of SDP using query segments with different length on the Honda/UCSD face databases is compared to the state-of-the-art approaches in Table 1. As can be seen, SDP achieves 21.7%, 10.9%, 4.0%, and 3.1% improvement over Volume Local Binary Pattern (VLBP) [2], VLBP + AdaBoost [30], Extended Volume Local Binary Pattern (EVLBP + AdaBoost) [30], and Manifold-Manifold Distance (MMD) [17], respectively. Notice that Sparse Approximated Nearest Point (SANP) [15] and Kernel Approximated Nearest Point (KSANP) [15] achieved 100% recognition rate but using the full length image sequences (varying between 275 and 1168 for each sequence). On the other hand, SDP achieved 100% recognition rate using only 70 frames per each query segment.

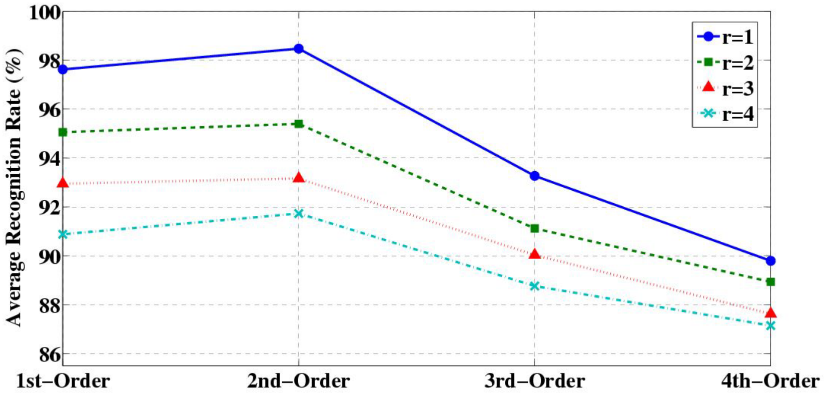

Fig. 6. Average rank-1 recognition rate versus SDP's order and the radius.

Table 1. Comparison of rank-1 recognition rates (%) on the Honda/UCSD dataset [25].

Method	Number of frames per segment	Recognition rate (%)
*VLBP [2]	N/A	78.30
*VLBP+AdaBoost [30]	N/A	89.10
*EVLBP+AdaBoost [30]	N/A	96.00
MMD [17]	300 – 400	96.90
SANP [15]	Full Length (275 – 1168)	100
KSANP [15]	Full Length (275 – 1168)	100
SDP	**10**	**95.32**
SDP	**50**	**97.51**
SDP	**70**	**100**

*The results are from [30].

4.3 YouTube Database

The YouTube database [26] contains 3,425 videos of 1,595 individuals. All the videos are from the YouTube website containing the subjects of the Labeled Faces in the Wild LFW database [31]. The videos have a frame rate of 24 frames/second, and an average of 2.15 videos are available for each subject. The shortest video duration is 48 frames, while the longest one is 6,070 frames, and

Table 2. Comparison of average recognition rates ± standard deviations (%) and Equal Error Rates (%) on the YouTube database [26].

Method	**Number of frames per segment**	**Average recognition rate ± deviation (%)**	**EER (%)**
*CSLBP [32]	N/A	63.10 ± 1.1	37.40
*FPLBP [33]	N/A	65.60 ± 1.8	35.60
*LBP [10]	N/A	65.70 ± 1.7	35.20
SDP	**10**	**90.00 ± 1.5**	**12.24**
SDP	**20**	**90.24 ± 1.5**	**12.17**
SDP	**Full length (48-6070)**	**91.42 ± 1.2**	**10.31**

*The results are from [26].

the average length of a video sequence is 181.3 frames. The faces are detected, cropped, and resized using the procedure in [26].

In order to make a direct comparison with the results reported in [26], the same experimental strategy used in [26] is adopted in our experiments. We conduct ten-fold experiments by splitting the data randomly similar to [26]. The average rank-1 recognition rate, the standard deviation, and the Equal Error Rate (EER) of the SDP and the benchmark approaches on YouTube database are summarized in Table 2. The results demonstrate that the proposed method consistently achieves the highest recognition rate and the smallest EER compared to the benchmarks. SDP improves the average recognition accuracy by 28.32%, 25.82%, and 25.72% over Center-Symmetric Local Binary Pattern (CSLBP) [32], Four-Path Local Binary Pattern (FPLBP) [33], and Local Binary Pattern (LBP) [10]. It also improves the EER by 24.89% compared to the smallest EER of the benchmarks. To the best of our knowledge, these are the highest recognition rates achieved on the challenging YouTube database.

4.4 DynTex Database

DynTex (Dynamic Texture) [6] is a standard database for dynamic texture research containing high-quality dynamic texture videos. It contains over 650 dynamic texture shots in different conditions. Dynamic texture sequences were recorded in PAL format at 25 frames per second with a frame resolution of 720 × 576. DynTex's standard video length is 250 frames.

Table 3 compares the rank-1 recognition rate of the SDP for different query segment's lengths with the dynamic texture benchmark approaches. SDP achieves 0.9% error rate which is equivalent to 21.05% reduction in error rate compared to the nearest competitor BoS Tree [18] (1.14%).

Table 3. Comparison of rank-1 recognition rates (%) on the dynamic texture database [6].

Method	**Number of frames per segment**	**Recognition rate (%)**
VLBP [2]	N/A	95.71
LBP-TOP [2]	N/A	97.14
DFS [24]	N/A	97.63
BoS Tree [18]	N/A	98.86
SDP	**10**	**97.94**
SDP	**100**	**98.23**
SDP	**Full length (250)**	**99.10**

5 Conclusion

In this paper, we proposed a novel dynamic texture descriptor namely Spatiotemporal Derivative Pattern (SDP). SDP captures the spatiotemporal variations of a video segment using the spatiotemporal directional high-order derivatives. The SDP algorithm encodes the video segments into directional patterns based on the spatiotemporal directional derivatives computed within segments. The spatiotemporal directional derivatives are computed using continuous circular and spiral paths to avoid the aliasing effects. A binary code is obtained by comparing the gray-level transitions of the segment's points. The most important characteristic of the SDP is using fewer training sample frames compared to the benchmark methods.

The proposed SDP was tested on three standard datasets: the Honda/UCSD and the YouTube databases for video-based face recognition and the DynTex database for dynamic texture classification. In all experiments, the algorithm was compared with state-of-the-art benchmarks. It is a very encouraging finding that the SDP performs consistently superior to all benchmarks under the video-based face recognition and the dynamic texture classification tasks. Especially, our results demonstrate that the proposed method consistently achieves the best performances for the challenging YouTube database. This research reveals that the Spatiotemporal Derivative Pattern provides a new solution for the dynamic texture description.

References

1. Rahman, A., Murshed, M.: A temporal texture characterization technique using block-based approximated motion measure. IEEE Trans. on Circuits and Systems

for Video Technology **17** (2007) 1370–1382

2. Zhao, G., Pietikainen, M.: Dynamic texture recognition using local binary patterns with an application to facial expressions. IEEE Trans. on PAMI **29** (2007) 915–928
3. Polana, R., Nelson, R.: Temporal texture and activity recognition. In: Motion-Based Recognition. Volume 9 of Computational Imaging and Vision. (1997) 87–124
4. Chetverikov, D., Peteri, R.: A brief survey of dynamic texture description and recognition. In: Proc. Intl Conf. Computer Recognition Systems. (2005) 17–26
5. Doretto, G., Chiuso, A., Wu, Y.N., Soatto, S.: Dynamic textures. IJCV **51** (2003) 91–109
6. Peteri, R., Fazekas, S., Huiskes, M.J.: Dyntex: A comprehensive database of dynamic textures. Pattern Recognition Letters **31** (2010) 1627–1632
7. O'Toole, A.J., Roark, D.A., Abdi, H.: Recognizing moving faces: A psychological and neural synthesis. Trends in Cognitive Sciences **6** (2002) 261–266
8. Turk, M., Pentland, A.: Eigenfaces for recognition. Journal of Cognitive Neuroscience **3** (1991) 71–86
9. Etemad, K., Chellappa, R.: Discriminant analysis for recognition of human face images. Journal of Optical Society of America A **14** (1997) 1724–1733
10. Ojala, T., Pietikainen, M., Maenpaa, T.: Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. IEEE Trans. on PAMI **24** (2002) 971–987
11. He, X., Yan, S., Hu, Y., Niyogi, P., Zhang, H.J.: Face recognition using laplacianfaces. IEEE Trans. on PAMI **27** (2005) 328–340
12. Rivera, A.R., Castillo, R., Chae, O.: Local directional number pattern for face analysis: Face and expression recognition. IEEE Trans. Image Processing **22** (2013) 1740–1752
13. Cevikalp, H., Triggs, B.: Face recognition based on image sets. In: IEEE CVPR. (2010) 2567–2573
14. Kim, T.K., Kittler, J., Cipolla, R.: Discriminative learning and recognition of image set classes using canonical correlations. IEEE Trans. on PAMI **29** (2007) 1005–1018
15. Hu, Y., Mian, A.S., Owens, R.: Face recognition using sparse approximated nearest points between image sets. IEEE Trans. on PAMI **34** (2012) 1992–2004
16. Wang, R., Guo, H., Davis, L.S., Dai, Q.: Covariance discriminative learning: A natural and efficient approach to image set classification. In: IEEE CVPR. (2012) 2496–2503
17. Wang, R., Shan, S., Chen, X., Gao, W.: Manifold-manifold distance with application to face recognition based on image set. In: IEEE CVPR. (2008) 1–8
18. Coviello, E., Mumtaz, A., Chan, A., Lanckriet, G.: Growing a bag of systems tree for fast and accurate classification. In: IEEE CVPR. (2012) 1979–1986
19. Liu, X., Chen, T.: Video-based face recognition using adaptive hidden markov models. In: Proc. of IEEE CVPR. (2003) 340–345
20. Rabiner, L.R.: A tutorial on hidden markov models and selected applications in speech recognition. Proceedings of the IEEE **77** (1989) 257–286
21. Doretto, G., Soatto, S.: Dynamic shape and appearance models. IEEE Trans. on PAMI **28** (2006) 2006–2019

22. Cootes, T.F., Edwards, G.J., Taylor, C.J.: Active appearance models. IEEE Trans. on PAMI **23** (2001) 681–685
23. Derpanis, K., Lecce, M., Daniilidis, K., Wildes, R.: Dynamic scene understanding: The role of orientation features in space and time in scene classification. In: IEEE CVPR. (2012) 1306–1313
24. Xu, Y., Quan, Y., Ling, H., Ji, H.: Dynamic texture classification using dynamic fractal analysis. In: IEEE ICCV. (2011) 1219–1226
25. Lee, K.C., Ho, J., Yang, M.H., Kriegman, D.: Video-based face recognition using probabilistic appearance manifolds. In: Proc. of IEEE CVPR. (2003) 313–320
26. Wolf, L., Hassner, T., Maoz, I.: Face recognition in unconstrained videos with matched background similarity. In: IEEE CVPR. (2011) 529–534
27. Mian, A.S.: Online learning from local features for video-based face recognition. Pattern Recognition **44** (2011) 1068–1075
28. Ahonen, T., Hadid, A., Pietikainen, M.: Face description with local binary patterns: Application to face recognition. IEEE Trans. on PAMI **28** (2006) 2037–2041
29. Zhang, H., Gao, W., Chen, X., Zhao, D.: Object detection using spatial histogram features. Image and Vision Computing **24** (2006) 327–341
30. Hadid, A., Pietikainen, M.: Combining appearance and motion for face and gender recognition from videos. Pattern Recognition **42** (2009) 2818–2827
31. Huang, G.B., Ramesh, M., Berg, T., Miller, E.L.: Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical report, University of Massachusetts, Amherst (2007)
32. Heikkil, M., Pietikainen, M., Schmid, C.: Description of interest regions with center-symmetric local binary patterns. In: Computer Vision, Graphics and Image Processing. (2006) 58–69
33. Wolf, L., Hassner, T., Taigman, Y.: Descriptor based methods in the wild. In: Real-Life Images Workshop in ECCV. (2008)