

# An efficient and sparse approach for large scale human action recognition in videos

Cyrille Beaudry<sup>1</sup> · Renaud Péteri<sup>1</sup> · Laurent Mascarilla<sup>1</sup>

Received: 19 February 2015 / Revised: 29 January 2016 / Accepted: 17 February 2016 / Published online: 28 March 2016  
© Springer-Verlag Berlin Heidelberg 2016

**Abstract** This paper focuses on human action recognition in video sequences. A method based on optical flow estimation is presented, where critical points of this flow field are extracted. Multi-scale trajectories are generated from those points and are characterized in the frequency domain. Finally, a sequence is described by fusing this frequency information with motion orientation and shape information. This method has been tested on video datasets with recognition rates among the highest in the state of the art. Contrary to recent dense sampling strategies, the proposed method only requires critical points of motion flow field, thus permitting a lower computational cost and a better sequence description. A cross-dataset generalization is performed to illustrate the robustness of the method to recognition dataset biases. Results, comparisons and prospects on complex action recognition datasets are finally discussed.

**Keywords** Action recognition in videos · Critical points · Frequency characterization of motion trajectories · Cross-dataset generalization

## 1 Introduction

### 1.1 Context

Recognizing human actions in video sequences has recently gained increasing attention in computer vision. Its goal is to discriminate different actions of one or several subjects in a video sequence, using algorithmic methods trained on labeled video sequences. The research domain is driven by a growing number of applications in a large set of areas. Recent video-surveillance systems integrate automatic intrusions detection or potential acts of violence. Action recognition is also present in classification of video database by automatic annotation of human actions. For example, one can try to retrieve in a database of soccer matches different video sequences where a technical movement is performed by a specific player. Action recognition is also found in human-machine interaction applications, like programs which help to prepare a recipe by recognizing actions executed by the user. Motion crowd analysis, facial expression recognition and many other applications illustrate the need for analyzing and recognizing daily activities.

### 1.2 Recognition of human actions: an active research topic

Action recognition is encouraged by a growing demand of applications, bringing new challenges to tackle. This is the reason why this domain is still an active research topic with relevant problems in computer vision. In real world, video sequence acquisition is generally unconstrained. It is common rule to find occlusions, changing viewpoints, untimely fast illumination and camera motions. It is clear that visual data resulting from different video sequences of a same action would present huge variability.

---

✉ Renaud Péteri  
renaud.peteri@univ-lr.fr

Cyrille Beaudry  
cyrille.beaudry@univ-lr.fr

Laurent Mascarilla  
laurent.mascarilla@univ-lr.fr

<sup>1</sup> Univ. La Rochelle, 23 avenue Albert Einstein, BP 33060,  
17031 La Rochelle, France

The action recognition task also suffers from semantic issues. Indeed, actions like “open a door” and “open a bag” can be considered as two different action classes, but they illustrate the same concept of action. Moreover, spatial context where actions are performed can be the most discriminant feature for action recognition (“play piano” and “play guitar” actions can completely be discriminated by visual information about the musical instrument).

Popularization of mobile camera phone in recent years and democratization of video as a media support has increased dramatically the amount of video data. Thus, 30 % of internet traffic is generated by video data. YouTube, for example, receives 100 hours of video every single minute. Facing this challenge, recent databases include a huge amount of information (more than two million frames for UCF-101 [41]) and several action classes to discriminate. They are still challenging for some state-of-the-art methods, for instance when involving corporal movements, behavior actions, or actions of short duration with high visual context correlation (e.g. the action “smoking”).

Current researches are, therefore, dedicated to build robust and effective methods to deal with such actions. Improving computational complexity is also an issue to be able to process an expanding amount of data.

For all these reasons, action recognition task in video sequences has become one of the most active and challenging issues in computer vision, and numerous methods have been developed recently.

### 1.3 State of the art on action recognition

Different models have been studied for action recognition, and most methods are based on a global representation using a single feature vector.

The local approach used in document or image retrieval context has proven its efficiency compared to more global approaches. It consists in detecting features for selecting interest points or regions in video sequence being discriminative of an action. Descriptors are then computed around these interest points to characterize the video. The sequence is then represented by a collection of local feature vectors. In the final stage, a classification process trained on a labeled database allows to recognize actions present in the video.

The main differences between action recognition methods are in the feature extraction phase, their descriptors and the way they are used in the classification process.

The first approaches of feature extraction in video sequences are based on sparse representation methods from the image retrieval and classification paradigm. They result from temporal extension of 2D interest point detector. [22] was the first to extract spatio-temporal points (STIP) by proposing a temporal extension of the Harris–Laplace 2D detector [10]. It detects points where the local neighborhood has a

significant variation in space (corner) and also in time (fast displacement). STIP is still a usual method today, and the framework proposed by Laptev has been applied in several recent approaches. Harris 3D shows good results on constrained video datasets. However, its assumption of high variation in time of spatial corners describes only a certain type of temporal variation in video sequences. It also uses several parameters which have to be fit to maximize the detection performance. Moreover, experiments show that it is inefficient on more complex movements like behavior actions [6].

In [6], Dollar et al. provide a method for analyzing actions with the cuboid detector and descriptor. The cuboid detector is obtained by applying a 2D gaussian in the spatial domain and a pair of 1D Gabor wavelets in the temporal domain. The response obtained by this detector is significant for periodic movements and actions, like facial expressions. Authors introduce the cuboid descriptor containing gradient and optical flow information around the interest points. This process is fast to implement and improves results on certain datasets [51]. It is still used in recent methods as an efficient sparse approach to detect local perturbations in video sequences [34]. Furthermore, this method is efficient for movements with strong periodicity. It has been experimented on datasets where periodical movements do not represent realistic situations (KTH [36]). The authors also make the assumption of an acquisition with a fixed camera, which limits the performance of the method on unconstrained video datasets.

In [55], Willem et al. extend the SURF 2D detector in the temporal domain and detect saliency by using the determinant of the 3D Hessian matrix. Its computational efficiency results from the use of the so-called integral video.

Nevertheless, experiments show lack of performance of sparse representation methods on recent databases. [51] illustrate how dense sampling outperforms the sparse representation strategy, especially for realistic videos.

Recent authors focus their researches on dense sampling approaches chiefly with temporal motion models such as trajectories.

In [51], Wang et al. use a dense sampling approach at different scales to obtain interest points. The dense approach shows better results compared to state-of-the-art sparse approaches. Laptev et al. [23] propose an improvement of their previous framework by avoiding scale selection in the optimization of the STIP detection. The goal is to compute spatio-temporal points at different scales to maximize the number of features and to be more efficient on realistic human actions from movies. The authors also present a method for automatic action labeling and recognition based on movie scripts.

Wang et al. [49] use dense sampling and add a temporal extension by tracking points at regular time intervals. The use of trajectories enables to capture more temporal infor-

mation (2.6 % of gain compared to information contained in a cuboid of same length) and this approach shows better results on realistic videos. This method has been improved using human person detection and camera motion compensation by estimation of homography parameters between two consecutive frames [52]. Since then, several methods have retained this approach for the action recognition task in realistic video sequences. Raptis et al. [33] propose the tracklet descriptor, which encodes descriptor features along trajectories estimated by tracking salient points. Ullah et al. [47] cluster trajectories coming from body part movements, estimated on synthetic dataset, to retrieve actions on generic video sequences. Vrigkas et al. [48] extract motion curves with the optical flow field. Actions are represented by a Gaussian mixture model by clustering the motion curves of every video sequence. When using PCA on the motion curves to force them to be of equal length, this method reaches among the highest recognition rates on well-known datasets (KTH [36], UCF-11 [25]).

However, each exposed method based on dense features suffers from the same drawback: the dense sampling approach leads to huge computational time and massive amount of data. The sustainability of dense sampling strategies can be questioned by the increasing amount of data included in the recent databases and the development of real-time action recognition applications.

Some authors are addressing this problem by providing methods to reduce the number of features used to characterize a video sequence.

In [37], Raptis et al. show that using a fixed number of features, selected in a dense set, achieves results close to those obtained in recent publications. With a dense sampling approach, features are randomly extracted every 160 frames. A total of 10k features are kept. This allows to keep more points on finer grid scales and to control their number. Results obtained on HMDB51 [20] dataset give a gain of 1 % compared to the state-of-the-art dense strategies, but it is still far behind on other large datasets like UCF-50 [34].

Murthy *et al.* in [26] propose a method which selects few dense trajectories to reduce the amount of data. Authors match similar trajectories and merge them into a new sequence of points, called ordered trajectories. With half less trajectories and the same parameters, this method allows to obtain on the UCF-50 dataset [34] a slightly better recognition rate than the classical dense trajectories approach (gain of 0.5 %). However, the matching step also requires an extraction of dense trajectories which does not reduce the computation time nor avoids the dense trajectories storage.

Although methods of feature reduction provide improvements on recognition rates, the number of features generated is still high compared to some sparse approaches (10 or 20 times more than in average) and the computation time is expensive. Though dense cuboid and dense trajectories meth-

ods outperform sparse trajectories approaches (like the SIFT trajectories method), the relative gain is not outstanding. In fact, the temporal information of trajectories is not fully exploited in most state-of-the-art methods. Indeed, information extracted along trajectories or voxels is typically the same [26,33,49]. While sparse representation methods provide a better computation time and less complexity, they also cumulate substantial drawbacks, (i.e. a large number of parameters to tune, and are not efficient enough to analyze realistic video scenes). Similarly, dense strategies show efficiency on generic video datasets but become too expensive in storage and data processing, which is problematic for large datasets and real-time applications.

The most recent approaches are tackling human action recognition in large-scale dataset using deep convolutional network. Deep learning methods provide a significant improvement for several computer vision problems such as object recognition [19], facial recognition [7] or image classification [38]. These methods have recently been applied for human action recognition. Features from convolutional neuronal network (CNN) allow to treat large dataset and reach very good results on recent datasets of the literature. Recent architecture proposed in the literature reflect the advances made on this domain [11,27,39,53].

In the following work, we have made the choice to characterize actions in video based on movement informations. Optical flow is a common way to estimate the movement in video sequences. The estimated motion field permits to analyze with precision different spatial and temporal characteristics at different motion scales. Information brought by intrinsic motion allows to perform well on realistic and unconstrained videos while lowering complexity and the number of generated features.

#### 1.4 Main contributions of the paper

In this paper, we attempt to answer this question: “how to get a better exploitation of the movement in video sequences to enhance the discrimination task while keeping a low amount of data ?”

An approach based on the optical flow estimation is presented. It extracts robust interest features, such as critical points of the flow field, without any additional parameters. Trajectories are estimated from these critical points and are described in the frequency domain using Fourier transform coefficients. Frequency information of motion is not often used for action recognition. However, its rigorous use brings out different characteristics of movement, especially actions with multiple frequency intervals. The complementary of motion frequency with shape and motion orientation of movement in action analysis is also shown, all three components being weakly correlated. We reach among the best recognition rates of the literature while keeping a low compu-

tational time due to the analysis of only relevant points from the optical flow. An efficient way to add a camera motion compensation using the optical flow without extra computation process is presented.

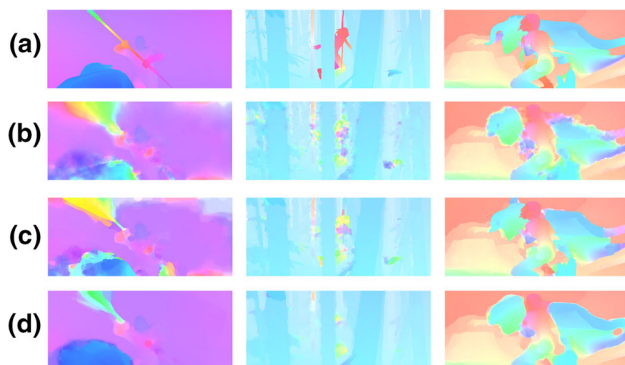
This paper is structured as follows: Sect. 2 describes the estimation of critical points from the optical flow. It also details multi-scale trajectories extraction and the camera motion compensation approach. Sect. 3 presents the descriptors used for these different features and details benefits of Fourier coefficients to characterize multi-scale trajectories. It also describes the bag of features approach used to combine all three components of movement as well as a boosting method. In Sect. 4, experimental results on different types of datasets are presented and a comparison with different state-of-the-art methods is performed. The performance brought by each features is also analyzed. The genericity of the method is assessed with a cross dataset generalization experiment.

## 2 Critical points and their trajectories

### 2.1 Critical points of a vector flow field

Optical flow estimation is used to characterize actions performed in video sequences. Several optical flow estimation methods exist [15,54]. We have focused on the optical flow estimation provided by Sun et al. [44] which is based on the Horn and Schunck approach. It has very good performance on different datasets such as MiddleBury [1] and Sintel [56]. Figure 1 illustrates results obtained with this method (row c), compared to other methods from the literature. The method accuracy with respect to the ground truth can be observed near motion borders.

For each frame of the sequence, the flow field is separated into two components, curl and divergence. Being  $\mathbf{F}_t = (u_t, v_t)$  an optical flow field, with  $u_t$  and  $v_t$  being the horizontal and vertical components, curl and divergence



**Fig. 1** Optical flow comparison between three examples from the Sintel dataset. **a** Ground truth; **b** Horn and Schunck; **c** Sun et al.; **d** DeepFlow



**Fig. 2** Critical points of typical flow fields : vortex, whirl, attractive and repulsive point

are defined as follows:

$$\text{Curl}(\mathbf{F}_t) = \nabla \wedge \mathbf{F}_t = \frac{\partial v_t}{\partial x} - \frac{\partial u_t}{\partial y}$$

$$\text{Div}(\mathbf{F}_t) = \nabla \cdot \mathbf{F}_t = \frac{\partial u_t}{\partial x} + \frac{\partial v_t}{\partial y}$$

Curl and divergence of the flow are two characteristics related to the evolution in time of the vector field:

- Curl gives information on how a fluid may rotate locally.
- Divergence represents to what extent a small volume around a point is a source or a sink for the vector field.

Extrema of these components are correlated with certain critical points of the flow (swirl points, attractive and repulsive points). These critical points correspond to local area with high deformation of the flow field which are potentially related to human movements (Fig. 2).

### 2.2 Extraction of multi-scale trajectories

#### 2.2.1 Trajectories of critical points

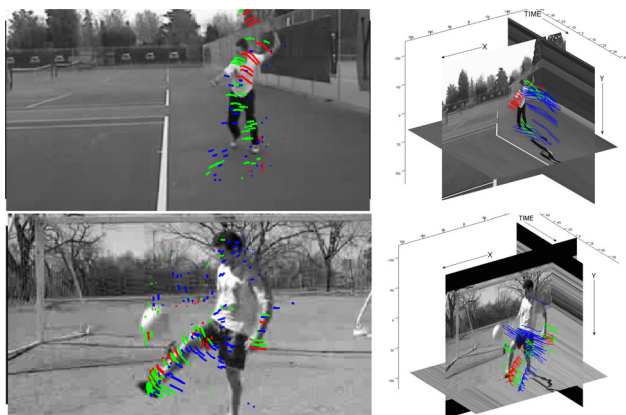
To go beyond the STIP concept, trajectories of optical flow critical points are computed using the dense trajectories methods of [49]. This method shows high performance compared to other methods. Given an optical flow field  $\mathbf{F}_t = (u_t, v_t)$ , position of a point  $P_t = (x_t, y_t)$  at frame  $t$  is estimated at  $t + 1$  as follows:

$P_{t+1} = (x_{t+1}, y_{t+1})$  such that  
 $P_{t+1} = (x_{t+1}, y_{t+1}) = (x_t, y_t) + \text{Med}_{F_t}(V_{(x_t, y_t)})$   
 with  $\text{Med}_{F_t}$ , a spatial median filter applied on  $\mathbf{F}_t$  at  $V_{(x_t, y_t)}$  a neighborhood centered on  $P_t$ .

Trajectories due to irrelevant movements in the video sequence are then removed and only trajectories of interest points are kept.

#### 2.2.2 Characterization of multi-scale trajectories

To analyze different frequencies of movement, a multi-scale approach of our method is proposed. The goal is to estimate critical points and their trajectories at different spatial and temporal scales. A spatio-temporal dyadic subdivision is performed on video sequences with a gaussian kernel, to suppress high frequencies in space and time. Optical flow is



**Fig. 3** Example of extracted multi-scale trajectories

estimated on each sub-sequence which corresponds to a scale of the pyramid. This way, critical points corresponding to different scales are extracted. Because of the dyadic subdivision in time, trajectories have the same length and characterize, at each scale, different frequencies of movement. Fast movements with high frequencies in the first scale, slower motions with lower frequencies as the scale increases.

With this approach, trajectories are computed in a larger interval of frequency of movement. A better analysis and a better characterization of movements contained in video is obtained. Figure 3 illustrates such multi-scale trajectories.

### 2.3 Camera motion compensation

Keeping low error estimation of the trajectory position in time is the main difficulty of the trajectory estimation step. In unconstrained video, this problem can be more complex, due to multiple camera motion that may impact trajectories estimation process.

The emergence of datasets which contain video sequences without acquisition constraints enhances the importance of camera motion compensation for action recognition. Among the existing strategies to address this problem, Wang et al. [52] assume that two consecutive frames are related by a homography. The estimation of the homography parameters between two consecutive frames is performed using SURF features for matching these frames, as they are robust to motion blur. This process gives a 2.6 % gain on the UCF-50 dataset with a recognition rate of 91.2 % (Table 2). In return, this approach adds a significant complexity using an ad hoc human detection process and a RANSAC method for estimating the homography.

Jain et al. in [13] suppose that movement can be separated in two parts, the dominant motion due to camera motion and the residual motion related to actions. The dominant motion is extracted by estimating the 2D affine motion model between

two consecutive frames. The compensation is obtained by subtracting the estimation of affine motion flow from the estimation of optical flow. This method shows good results on recent human action datasets. However, it implies the computation of two flow fields, the optical flow field and the affine flow, related to camera motion information.

The method presented here allows a compensation for the dominant motion but avoids the computation of an additional flow field.

#### 2.3.1 Global motion estimation by a pyramidal approach

To minimize the effect of camera motion while keeping a low computation time and avoiding ad hoc methods, we exploit the optical flow already estimated in Sect. 2.1. More precisely, we will use a pyramidal estimation of the optical flow to compensate the global motion of the camera. The displacement field at time  $t$  between two scales  $I^L$  and  $I^{L+1}$  of the pyramid is such that

$$\mathbf{F}_t^L = E_2 \left( \mathbf{F}_t^{L+1} \right) + f \left( \left[ I_t^L + E_2(\mathbf{F}_t^{L+1}) \right], I_{t+1}^L \right),$$

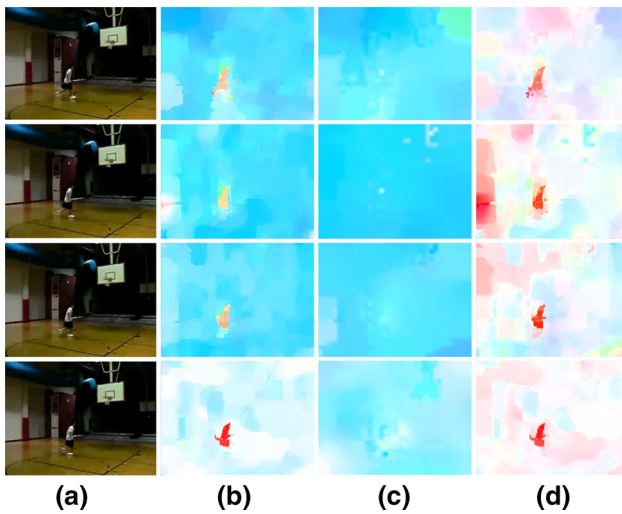
$0 \leq L < 4$ , being a level of the pyramid,  $E_2$  an upsampling operator of factor 2, and  $f$  the estimated optical flow between two consecutive frames.

At maximal scale  $L$ , the estimated vector flow field corresponds to the largest movements in the video sequence due to the camera movements. Small movements are not generally included in this flow. This “global motion” flow is used in the same way as the dominant flow of [13]. Finally we obtain

$$\mathbf{F}_{comp}^0 = \mathbf{F}_{original}^0 - \mathbf{F}_{original}^L$$

with  $\mathbf{F}_{original}^0$  the original optical flow estimation of the sequence,  $\mathbf{F}_{original}^L$  the original optical flow at the last level  $L$  of the pyramid, which represents global camera motion.  $\mathbf{F}_{comp}^0$  is the optical flow estimation of the sequence with camera movement compensation.

Figure 4 illustrates the result of this method on a video sequence from UCF-11. From the second to the fourth column, the color represents the motion orientation between two consecutive frames. The second column corresponds to the  $\mathbf{F}_{original}^0$  vector flow field which contains a large amount of pixels with the same angular displacement, related to a camera translation. The third column shows the computation of  $\mathbf{F}_{original}^L$ , which only keeps the dominant motion present in the sequence and does not take into account the player movements. The fourth column illustrates the  $\mathbf{F}_{comp}^0$  flow field which permits to retrieve the original motion orientation and intensity of the players by compensating camera movements. Table 2 shows the improvement of the trajectory descriptor.



**Fig. 4** First column four consecutive frames with a lateral camera movement on the first three frames; second column optical flow estimation  $F^0_{original}$ ; third column global flow estimation  $F^N_{original}$ ; fourth column camera movement compensation  $F^0_{comp}$

The estimation of the global motion of the camera is carried out directly during the estimation of the optical flow. This method thus allows motion compensation without any additional computational time.

### 3 Descriptors computed from critical points and their trajectories

#### 3.1 Trajectories descriptors based on Fourier transform coefficients

##### 3.1.1 Frequency analysis of trajectories

A robust action recognition method should extract descriptors with low intra-class variability by ensuring invariance to different kinds of transformations. Here, multi-scale trajectories obtained are described by Fourier transform coefficients.

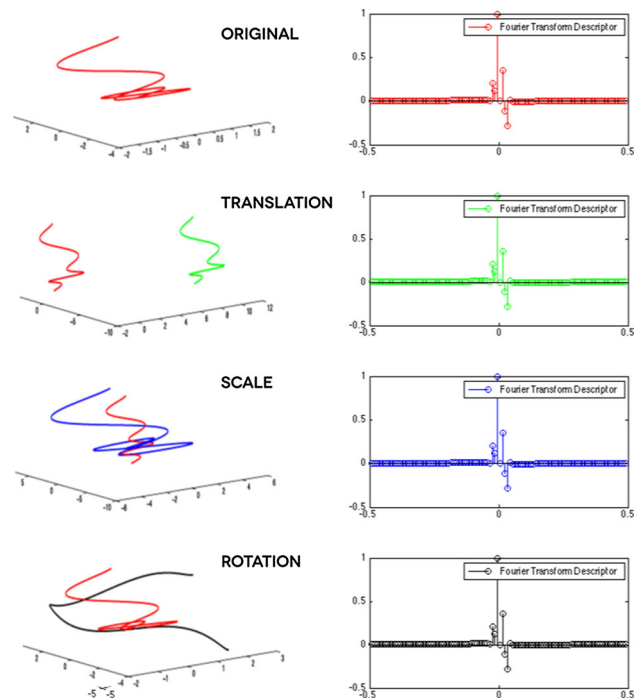
The choice of Fourier coefficients is motivated by invariances to certain geometric transformations (translation, rotation and scaling), which are natural in the frequency domain. Figure 5 illustrates these invariances.

##### 3.1.2 Invariance of the proposed descriptor

Given a trajectory  $T_N$  with  $N$  sequential points  $T_N = [P_1, P_2, \dots, P_t, \dots, P_N]$   
 $P_t$  being a point of the trajectory at position  $(x_t, y_t)$ .

The Fourier transform of trajectory  $T_N$  is  $TF[T_N] = [X_0, X_1, \dots, X_k, \dots, X_{N-1}]$  with:

$$X_k = \sum_{n=0}^{N-1} e^{-i2\pi kn} \cdot P_n, \quad k \in \llbracket 0, N-1 \rrbracket$$



**Fig. 5** Different geometric transformations applied on the original trajectory. The same values are obtained for the FCD descriptor

To obtain translation invariance, the mean point value on this trajectory  $T_N$  is subtracted to each point  $(x_n, y_n)$ .

$$\tilde{x}_n = x_n - \sum_{t=1}^N \frac{x_t}{N} \text{ et } \tilde{y}_n = y_n - \sum_{t=1}^N \frac{y_t}{N}$$

To obtain rotation invariance, trajectories  $T_N$  are considered as complex number vectors:

$$T_{iN} = [P_{i1}, P_{i2}, \dots, P_{it}, \dots, P_{iN}]$$

with  $P_{it} = \tilde{x}_t + i\tilde{y}_t$  being the complex representation of point  $P_t$ . For a trajectory  $T_{\theta iN}$  which represents a rotation by  $\theta$  of the initial trajectory  $T_{iN}$ , the modulus of the Fourier transform of  $T_{\theta iN}$  and  $T_{iN}$  are equal, giving rotation invariance.

Scale invariance is insured by normalizing the Fourier transform with the first non-zero frequency component:

$$\tilde{X}_k = \frac{X_k}{|X_0|}, \quad k \in \llbracket 0, N-1 \rrbracket$$

Finally, descriptors based on the Fourier coefficients (FCD) are

$$FCD_{[T_{iN}]} = [|\tilde{X}_0|, |\tilde{X}_1|, \dots, |\tilde{X}_k|, \dots, |\tilde{X}_{N-1}|], \quad k \in \llbracket 0; N-1 \rrbracket \text{ with}$$

$$X_k = \sum_{n=0}^{N-1} e^{-i2\pi kn} \cdot P_{in}, \quad k \in \llbracket 0, N-1 \rrbracket$$

As all trajectories have the same size  $N$ , the *FCD* descriptor is also of fixed size.

Trajectories are finally smoothed by removing Fourier coefficients corresponding to high frequencies, which are assimilated to noise or tracking drift. This process improves robustness with respect to small motion perturbations.

### 3.2 Critical points descriptor based on shape variation and orientation of movement

To characterize critical points, we use HOG and HOF descriptors [23]. HOG descriptor (histogram of 2D gradients) is based on the 2D gradient around critical points and characterizes the shape information of local movements present in the video sequence.

HOF descriptor (histogram of orientation of optical flow) encodes the local optical flow field orientation around critical points. This descriptor has proven its performance in the action recognition task.

Both characteristics, associated with frequency information brought by the FCD descriptor, are highly relevant information and have the benefit of sharing weak correlation.

HOG is based on the spatial gradient of the image, HOF corresponds to the optical flow estimation and FCD characterizes the different frequencies of movement along the sequence. To take advantage of their low correlation, these features are combined with a late fusion approach in the classification task, which is detailed thereafter.

## 4 Evaluation of the method

To evaluate the method, we use four datasets from the literature: two with video captured in constrained conditions (static camera, homogeneous background, . . .) and two composed of realistic movie-clip, from YouTube or movie films.

### 4.1 Database used

#### 4.1.1 The KTH dataset

The KTH dataset [36] consists of six human action classes: Walking, Jogging, Running, Boxing, Waving and Handclapping. Each action is performed several times by 25 subjects with four different scenarios. All sequences were shot with homogeneous backgrounds and a static camera.

#### 4.1.2 Weizmann dataset

The Weizmann dataset [9] is a collection of 90 video sequences captured with the same constraints and with no

camera motion. There are ten different actions, some being similar like Jack, Run, Skip, Side.

#### 4.1.3 UCF-11 dataset

The UCF-11 dataset [25] contains unconstrained realistic videos from YouTube. It is a challenging dataset due to large variations in camera motion, object appearance and pose, object scale, viewpoint, cluttered background, illumination conditions, etc. There are 11 action categories: Basketball shooting, Biking/cycling, Diving, Golf swinging, Horse back riding, Soccer juggling, Swinging, Tennis swinging, Jumping, Spiking and Walking with a dog.

#### 4.1.4 Olympic sport dataset

The Olympic sports dataset [28] contains videos of athletes practicing different sports. It contains 16 actions class performed in realistic condition of acquisition. This dataset is one of the most challenging sport dataset in the literature.

#### 4.1.5 UCF-50 dataset

The UCF-50 dataset [34] is an extension of UCF-11 with 50 action categories, consisting of realistic videos taken from YouTube.

#### 4.1.6 HMDB-51 dataset

The HMDB-51 dataset [20] is a large and recent dataset of videos from various sources (movies, archives, YouTube, . . .). It contains 6849 clips divided into 51 actions categories. This dataset is one of the most challenging for action recognition.

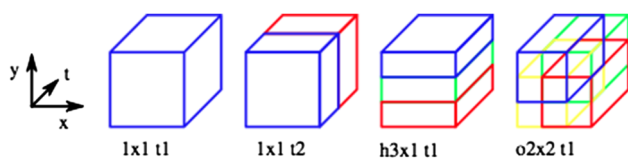
## 4.2 Experiments

### 4.2.1 Vector quantization

An important step after feature extraction is vector quantization. We have used two methods (bag of features and Fisher vector) to perform quantization and test them in the classification process.

#### *Bag of Features approach*

The first method used for feature quantization is the bag of features (BoF) approach [24]. This approach was used initially for the document retrieval task. It is now commonly used for image classification and for action recognition in videos. It assumes that a video can be described with a dictionary of “visual words”. This dictionary is built by clustering



**Fig. 6** Example of channels (from [23]). The  $1 \times 1 \times 1$  grid refers to the standard BoF representation.  $1 \times 1 \times 2$  corresponds to a temporal subdivision in two cells, while  $h3 \times 1 \times 1$  corresponds to a horizontal subdivision in three cells and  $o2 \times 2 \times 1$  a horizontal and vertical subdivisions overlapping in the center

the set of features computed on the database, generally by a  $k$ -means algorithm.

The obtained centroids constitute the “visual words” of the dictionary. A feature vector is then represented by its closest word (in the Euclidean distance sense). Finally, a video sequence is represented by an occurrence histogram of visual words of the dictionary.

In addition to the traditional approach, we have integrated the multi-channel method. Introduced by [23], it allows a more localized approach by computing a spatio-temporal bag of features. The video is subdivided following a particular grid structure. The bag of features approach is computed on each cell. Finally, the histogram of the video sequence, associated with the grid structure, is the concatenation of all histograms of its cells.

Each grid structure is called a channel. Figure 6 illustrates different channels with their cells highlighted with different colors.

The spatio-temporal bag of features approach uses different channels to combine more local information.

#### Fisher vectors approach

The other method used for feature quantization is the Fisher vectors approach (FV). It takes into account a wider set of information compared to the BoF approach. The Fisher vectors method encodes first- and second-order statistics between video features and a Gaussian mixture model (GMM). This approach is one of the state-of-the-art features encoding method for image classification and human action recognition.

In our experiments, we have used this quantization on the HMDB-51 dataset to be able to fairly compare with other state-of-the-art methods. We have used the same set of Fisher vectors parameters as Wang et al. [52]: the number of Gaussians in the GMM is set to  $K = 256$ , a subset of 200,000 features from the training set is used and L2-normalization is applied to the Fisher vectors as in [30].

The Fisher vectors are computed on each descriptor (FCD, HOG and HOF). Finally, a video is represented as the concatenation of the Fisher vectors obtained on each associated feature.

#### 4.2.2 SVM classification

A supervised SVM classification [5] is finally performed on the obtained quantized features. We use a multi-dimensional gaussian kernel to establish a distance between video sequences represented by several histograms from different channels [57]. This kernel is the RBF kernel, defined as follows:

$$K_{RBF}(x_i, x_j) = \exp\left(-\sum_{c \in C} \frac{1}{A_c} D(H_i^c, H_j^c)\right), \quad (1)$$

where  $H_i^c$  and  $H_j^c$  are, respectively, the histograms of videos  $x_i$  and  $x_j$  and correspond to a channel  $c$ .  $D(H_i^c, H_j^c)$  is the  $\chi^2$  distance and  $A_c$  is a normalizing coefficient. The classifier is trained on each descriptor. We then use the fusion of estimated probabilities obtained by the SVM classification from each descriptor by the multi-class Adaboost algorithm [12]. It allows an efficient exploitation of the complementarity between the characteristics. Recent researches have shown the efficiency of this late fusion in the action recognition task [29].

For action recognition datasets with high amounts of action classes and video sequences such as UCF-11 and UCF-50 a linear kernel is used in the SVM to reduce computation time [8]:

$$K_{Linear}(x_i, x_j) = (H_i^C)^T H_j^C, \quad (2)$$

where  $H_i^C$  and  $H_j^C$  are the concatenation of all histogram channels in the set  $C$ . The BoF approach ensures a sparse representation of the video sequence. Linear kernel is efficient for sparse data with high-dimensional features. The computation time is then lower than a non linear kernel. Another advantage is that a linear kernel allows to compute BoF with larger codebook size.

### 4.3 Results

Results of the method on the different datasets previously introduced are exposed in Table 1.

#### 4.3.1 Parameters of the method

The method uses very few parameters. They are

- $C_p$ , the number of critical points.
- $N$ , the size of trajectory.
- $C$ , the channel structure
- $s$ , the number of spatio-temporal scales for the multi-scale trajectories approach.



**Table 1** Summary of different recognition rates obtained on various datasets (in %)

	KTH	Weizman	UCF-11	UCF-50	Olympic sport	HMDB51
FCD	85.47	90.12	66.42	53.58	51.15	6.76
HOG	91.98	92.59	86.98	84.88	65.85	41.14
HOF	91.98	95.06	74.43	73.80	60.86	32.46
Combined	95.32	100	89.99	88.30	65.98	49.60

**Table 2** Summary of recognition rates with the multi-scale parameter  $s$  and the camera compensation on UCF-11 dataset (in %)

UCF 11	$s = 1$	$s = 3$	Camera compensation + $s = 1$	Camera compensation + $s = 3$
FCD	53.50	57.42	64.05	66.42
HOG	80.34	84.53	83.34	86.98
HOF	70.06	74.34	74.06	74.43
Combined	82.07	86.98	87.89	89.99

$N$  has been fixed to 16 frames for each database. The influence of the variation of parameter  $s$  on the database UCF-11 has been studied and detailed in Table 2.

#### 4.3.2 Discussion of the results

The recognition rates of our approach are presented in Table 1 for each dataset. The gain obtained with the late fusion Adaboost illustrates the complementarity of the chosen characteristics (3.88 % of mean gain on all datasets).

Experiments show that among these parameters,  $C$  and  $C_p$  are the main parameters impacting recognition rate. The number of spatio-temporal scales for trajectories helps to improve the recognition rate on realistic videos datasets, where the frequency information is much richer than on constrained videos. For the FCD descriptor, the increase from one to three spatio-temporal scales improves the results by 3.92 %. The HOG descriptor shows good results on generic video datasets (UCF-11, UCF-50). The spatial context is very relevant for some actions that are performed in a well-defined framework, especially for objet-interaction actions or sport actions.

The influence of the camera motion compensation stage is presented in Table 2. Motion compensation has been computed for two cases:  $s = 1$  and  $s = 3$ . On the UCF-11 dataset which contains video sequences with camera motion, the gain for the FCD is 10.55 % with  $s = 1$  and 8.83 % with  $s = 3$ . This result shows the importance of camera motion compensation in the trajectory estimation stage. The increased performance of the method when using compensation before computing HOG and HOF descriptor shows that the estimated optical flow of the video sequence is more reliable. Critical points related to movements are better located and the information encoded by HOF descriptor is less disturbed and more relevant. With the best setting, the global gain with camera motion compensation is 2.1 % for the UCF-

**Table 3** Mean features per frame ratio for UCF-50 dataset

Method	Wang et al.	Shi et al.	Our approach
Features/frame	230	65.3	70.6
%	91.2	83.3	88.3

11 dataset. We reach a recognition rate of 89.99 %, one of the best in the literature for this dataset (Table 4).

#### 4.3.3 Comparison with the state of the art

For the different datasets used, the approach proposed is compared with other methods of the literature in Tables 4 and 5. Table 3 shows the number of features per frame generated by our method on the UCF-50 dataset compared to method of [50] and [37]. It gives an indication of the number of features to generate to achieve a given recognition rate.

Shi et al. [37] propose a random selection of 10k features from a dense sampling strategy. Wang et al. [50] have one of the best recognition rates in the literature but generate a very high number of features. Moreover, it uses 8 spatial scales of trajectories and 30 channels of bag of features. 15 % of the execution time in this method is dedicated to data storage. Murthy et al. [26] compare the number of features per frame generated by the proposed method to the one of [50] after the step of ordered trajectories. When using one channel and trajectories of 15 frames, it uses 1.85 time less trajectories (11,657 versus 21,647 features). Referring to the features per frame rate of [50], it would give an average of 124.32 for [26] on the UCF-50 dataset with a recognition rate of 87.3%. For the UCF-50 dataset, our method uses 1200 critical points per scale and per channel, which gives a total of 10,800 points per video sequence and a features/frame of 70.6. The slight improvement obtained by the best methods compared to our approach (see Tables 4 and 5) has to be put into perspective with the significant increased complexity.

**Table 4** Summary of different recognition rates obtained on various datasets (in %)

KTH	Weizmann	UCF-11	UCF-50	Olympic sport
Dollar et al. 89.1 [6]	Gorelick et al. 97.8 [9]	J. Liu et al. 71.2 [14]	Murthy et al. 87.3 [32]	Laptev et al. 62.0 [28]
Laptev et al. 92.1 [23]	Blank et al. 99.6 [3]	Wang et al. 85.4 [49]	Wang et al. 91.2 [52]	Tang et al. 66.8 [45]
Wang et al. 94.2 [49]	Vrigkas et al. 100 [48]	Reddy et al. 87.1 [34]	Bilinski et al. 92.1 [2]	Niebles et al. 72.1 [28]
Vrigkas et al. 98.3 [48]	Wang et al. 100 [52]	Vrigkas et al. 95.1 [48]	Lan et al. 94.4 [21]	Li et al. 78.2 [53]
Our approach 95.3	Our approach 100	Our approach 89.9	Our approach 88.3	Our approach 65.9

**Table 5** Summary of different recognition rates obtained on HMDB-51 (in %)

HMDB-51	
Simonyan et al. [39]	59.4
Wang et al. [52]	57.2
Jain et al. [13]	52.1
Our approach	49.6
Wang et al. [49]	46.6
Srivastava et al. [42]	44.1
Jiang et al. [14]	40.7
Can et al. [4]	39.0
Klipper-Gross et al. [18]	29.2
Solmaz et al. [40]	29.2
Sadanand et al. [35]	26.9
Kuehne et al. [20]	23.0
Kuehne et al. [20]	20.0

For the HMDB51 dataset, the global recognition rate obtained is 49.6 %. As can be seen in Table 5, the proposed approach performs reasonably well compared to other methods<sup>1</sup>. Specifically, it outperforms well-known approaches based on local features such as [14,49], or on global features such as GIST [40] and action Bank [35]. As a matter of fact, it provides one of the best classification results among handcrafted features based methods in this dataset. Only very recent approaches based on convolutional neuronal network [39] surpass it by a vast margin. However, some observations could be done to explain the weakness of the proposed method relatively to such approaches and to provide a guideline to significantly increase classification rates. One explanation is that FCD descriptors do not perform well on this dataset mainly because of the great number of shot transitions in many of the videos of the HMDB-51 database. This introduces perturbations in the optical flow estimation process and makes trajectories estimates all along the sequences difficult. A temporal segmentation of videos by some cut detection algorithm would certainly be relevant as a preprocessing stage before applying our algorithm.

<sup>1</sup> See also: <http://serre-lab.clps.brown.edu/resource/hmdb-a-large-human-motion-database/>.

Second, it can be observed that, in this dataset, a high proportion of actions does require few temporal information to be recognized. This observation was also made on large video datasets such as Sports-1M Dataset [16]. Authors have found that motion information provided by a convolutional network leads to a gain of only 1.6 % compared to a single-frame model. They suggest that in large-scale dataset, methods using static information descriptors such as HOG can reach a good recognition rate without the need of temporal and motion descriptors. It can be observed that in the case of the proposed method, HOG remains the best descriptor in terms of recognition rate. As a consequence, classification rate could be improved if more static, single frame based descriptors were used in the process.

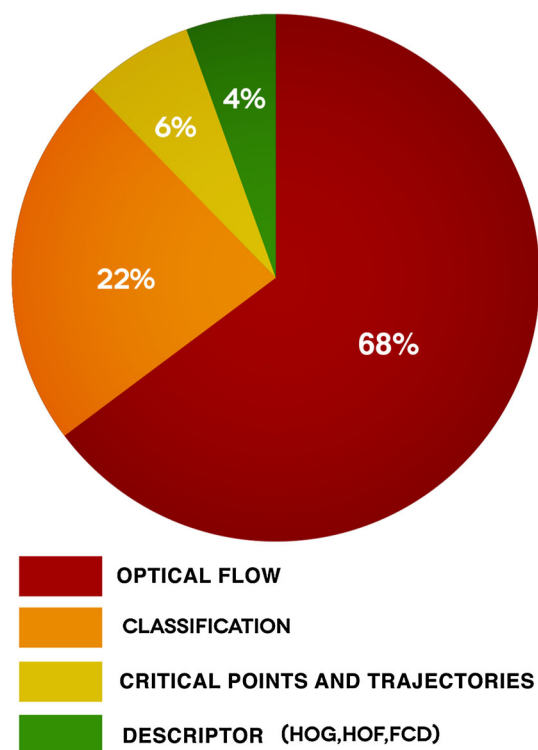
#### 4.3.4 Computation time

Results presented in our method have been computed using Matlab running on a workstation with 2 Quadcore CPU at 3.1 GHZ and 24 GB RAM. It takes 2.03 s/frame to compute the optical flow and 1.71 s/video to process the features. We have performed in Fig. 7 an analysis of the computation time for each step of the method.

We are using an optical flow estimation method based on the Horn and Schunk model proposed by Sun et al. [44]. Most of the computation time is spent for the optical flow computation, and improving this step by implementing it on GPU is part of our future work. The feature extraction and description steps, which constitute the main innovative parts of our work, are the fastest steps.

#### 4.3.5 Cross-dataset generalization

In this section, an original way to evaluate the genericity of our method is introduced. Our experiments are based on recent studies of Efros et al. [46]. The initial goal of this work is to highlight the visual bias contained in some state-of-the-art recognition datasets. This issue is very important in pattern recognition but largely neglected in the literature. Datasets are collected for representing an information as varied and rich as possible to mimic the real world. But in practice, they appear to contain representation biases due to the way that they are built. Authors point out different



**Fig. 7** Proportions of computation time for each step of the method

causes of these visual biases: selection bias (data sources), capture bias (constraints of acquisition, habits of capture), negative set bias (the representation of the rest of the real world).

Authors in [46] try to answer this question: how a classifier trained on one dataset generalizes when tested on other datasets, compared to its performances on the “native” test set?

In the context of action recognition, this methodology is used to evaluate the genericity of the presented approach. The purpose is to see how it characterizes and generalizes human actions while being robust to visual biases contained in each datasets. For this experiment we have picked up four popular databases; the KTH and Weizmann datasets (videos with acquisition constraints), UCF-11 and HMDB51 datasets (generic videos). We consider the *walk* and *wave* action classes, which are common to all chosen datasets (note that for UCF-11, *wave* action is represented by *golf* action class which is the closest action for representing a hand wave).

The experimental protocol is based on Efros et al. [46]. The method is trained with 200 positive and 200 negative examples for each dataset (oversampling has been performed on datasets which contain a too small amount of data). The test was performed with 100 positive and 100 negative examples from the other datasets. This proportion was chosen by

considering the one of Efros et al. which uses a smaller test set than the train set. We also take into account the fact that video dataset contains much less examples than image datasets. The goal of this work was to observe the difference in performance between train and test datasets.

Tables 6 and 8 expose the obtained results. Table 8 exposes the obtained results. Rows correspond to training on one dataset and testing on all the others. Columns correspond to the performance obtained when testing on one dataset and training on all the others. As observed in [46], the best results are obtained when training and testing on the same dataset (94.7 % in average for *walk* and 95.1 % for *wave*).

Weizmann and UCF-11 are the less efficient datasets in generalization (respectively 39.75 and 35.35 percent drop in average for the two actions). Strong acquisition constraints and the few examples in the Weizmann dataset can explain the difficulty to reach a good generalization rate with this database. Kuehne et al. [20] point out the fact that videos from YouTube contain low-level biases due to some amateur director habits. It can explain the lack of generalization of UCF-11 (42 % percent drop for *walk* action class) compared to HMDB51 which contains different video sources like YouTube, Google videos, movies or archives (15 % percent drop for the *walk* action class).

Lack of comparison with other approaches does not allow us to conclude totally on the robustness of the method with respect to dataset bias. Nevertheless, one can observe a fairly good generalization behavior when training on one dataset and testing on all the others (64.2 % in average). One can think that the *walk* and *wave* action classes have been globally well generalized with the presented approach.

Selected datasets represent different aspects of the *walk* and *wave* action classes. KTH and Weizmann contain videos performed by people acting and represent those action classes in a canonical way. In UCF and HMDB, action classes are not acted and are represented in different situations and contexts. It brings visual variabilities and noise (movement which do not correspond to the observed action). They provide a representation of a *walk* and *wave* action classes “in the wild”. Both, acted and generic dataset contains complementary information about an action. In can be observed in Tables 6 and 8 that KTH and HMDB, respectively a constrained (acted) and a generic video dataset, perform well in generalization.

We explore the representation generalization of human actions by enhancing the previous process using a weighted mixture of datasets in the training phase. We use the percent drop of each dataset as a weight to build a new dataset giving more importance to videos from datasets with good generalization. This new dataset contains 200 positive and 200 negative examples from each dataset drawn proportionally to their weight obtained by normalizing the percent drop.

**Table 6** Cross-dataset generalization for the “walk” action class when training on one dataset (rows) and testing on another (columns)

Action	Train/test	KTH (%)	Weizmann (%)	UCF-11 (%)	HMDB51 (%)	Self (%)	Mean others (%)	% drop
Walk	KTH	<b>97</b>	96	56	62	97	71.3	26.4
	Weizmann	66	<b>100</b>	51	55	100	57.3	42.7
	UCF-11	54	50	<b>95</b>	61.5	95	55.1	42
	HMDB51	79	79.5	62.5	<b>87</b>	87	73.6	15
	Mean others	66.3	75.1	56.5	59.5	94.7	64.3	32.1

Bold value emphasizes the results obtained by our method

**Table 7** Cross-dataset generalization for the “walk” action class when training on the “mixture dataset” and testing on another (columns)

Action	Train/test	Mixture dataset (%)	KTH (%)	Weizmann (%)	UCF-11 (%)	HMDB51 (%)	Self (%)	Mean others (%)	% drop
Walk	Mixture dataset	<b>90</b>	85	93	74.5	82	90	83.6	7.1

Bold value emphasizes the results obtained by our method

**Table 8** Cross-dataset generalization for the “wave” action class when training on one dataset (rows) and testing on another (columns)

Action	Train/test	KTH (%)	Weizmann (%)	UCF-11 (%)	HMDB51 (%)	Self (%)	Mean others (%)	% drop
Wave	KTH	<b>99.5</b>	73.5	60	50	99.5	61.1	38.5
	Weizmann	65	<b>100</b>	73.5	51	100	63.1	36.8
	UCF-11	58.5	85.5	<b>94.5</b>	58	94.5	67.3	28.7
	HMDB51	50.5	75	69.5	<b>86.5</b>	86.5	65	24
	Mean others	58	78	67.6	53	95.1	64.1	32

Bold value emphasizes the results obtained by our method

**Table 9** Cross-dataset generalization for the “wave” action class when training on the “mixture dataset” and testing on another (columns)

Action	Train/test	Mixture dataset (%)	KTH (%)	Weizmann (%)	UCF-11 (%)	HMDB51 (%)	Self (%)	Mean others (%)	% drop
Wave	Mixture dataset	<b>89.5</b>	81	93.5	88	71	89.5	83.3	6.8

Bold value emphasizes the results obtained by our method

Tables 7 and 9 show results obtained with this “mixture” dataset. The average rate when testing on all the other is fairly high compared to rates obtained in Table 8 (83.6 % for walk and 83.3 % for wave).

The percent drop is just 6.9 % in average, which is half of the percent drop of HMDB51 which is the best in generalization among the other datasets (15 % for walk and 24 % for wave). This new dataset, which is a mix of previous datasets drawn proportionally to their generalization rates, provides a robust representation of the walk and wave action classes.

This dataset bias is new and not addressed in the literature, and only few papers point out this issue and cross dataset generalization ([17, 43]).

Building mixed dataset from different datasets according to their generalization capacity is a preliminary work, but it brings some guidelines for a robust representation of human actions, especially in concrete applications where action recognition methods are used.

## 5 Conclusion

This paper presents a novel approach of human actions recognition in video sequences. Video sequences are characterized by critical points estimated from the optical flow field and trajectories of critical points at different spatial and temporal scales.

The characterization in the frequency domain of the movement trajectories combined with motion orientation and shape information enable to reach among the best rate of recognition of the literature. Only the movement of critical points is characterized, which represents a significant advantage in terms of complexity. Indeed, obtained recognition rates are close to dense strategy approaches but with the computation of fewer features. Critical points are well reflecting movements present in the tested video sequences, and the fusion process shows efficiency in the action recognition task.

Recognition rates on different datasets illustrate the performance of the proposed method for different cases: recognition of actions with constrained acquisition conditions (KTH) or in realistic videos (UCF-11); discrimination of different actions with strong visual similarities (Weizmann); discrimination of a large number of action classes (UCF-50).

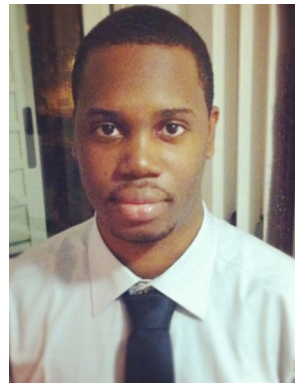
The introduction of cross-dataset generalization provides a good robustness for describing elementary actions. It illustrates the ability of the approach to robustly characterize an action despite dataset bias.

The obtained results open the way for future studies. A current prospect is to test our method for recognizing complex actions or activities by representing them as a sequence of elementary actions. Another application field can also be the analysis and the recognition of dynamic textures [31]. We believe that the use of critical points and frequency information may be particularly relevant for periodic motions of fluids.

## References

- Baker, S., Scharstein, D., Lewis, J.P., Roth, S., Black, M.J., Szeliski, R.: A database and evaluation methodology for optical flow. In: Proceedings of international conference on computer vision (2007)
- Bilinski, P., Bremond, F.: Video covariance matrix logarithm for human action recognition in videos. In: International conference on artificial intelligence, Buenos Aires (2015)
- Blank, M., Gorelick, L., Shechtman, E., Irani, M., Basri, R.: Actions as space-time shapes. In: Proceedings of international conference on computer vision, pp. 1395–1402 (2005)
- Can, E., Manmatha, R.: Formulating action recognition as a ranking problem. In: International workshop on action similarity in unconstrained videos (2013)
- Chang, C.C., Lin, C.J.: LIBSVM: a library for support vector machines. *ACM Trans Intell Syst Technol* **2**, 1–27 (2011)
- Dollar, P., Rabaud, V., Cottrell, G., Belongie, S.: Behavior recognition via sparse spatio-temporal features. In: IEEE international workshop on visual surveillance and performance evaluation of tracking and surveillance, pp. 65–72 (2005)
- Fan, H., Cao, Z., Jiang, Y., Yin, Q., Doudou, C.: Learning deep face representation. *Comput. Res. Repos.* [arxiv:1403.2802](https://arxiv.org/abs/1403.2802) (2014)
- Fan, R.E., Chang, K.W., Hsieh, C.J., Wang, X.R., Lin, C.J.: LIBLINEAR: a library for large linear classification. *J. Mach. Learn. Res.* **9**, 1871–1874 (2008)
- Gorelick, L., Blank, M., Shechtman, E., Irani, M., Basri, R.: Actions as space-time shapes. *IEEE Trans. Pattern Anal. Mach. Intell.* **29**(12), 2247–2253 (2007)
- Harris, C., Stephens, M.: A combined corner and edge detector. In: Proceedings of fourth alvey vision conference, pp. 147–151 (1988)
- Hasan, M., Roy-Chowdhury, A.: A continuous learning framework for activity recognition using deep hybrid feature models. *IEEE Trans. Multimed.* **99**, 1 (2015). doi:[10.1109/TMM.2015.2477242](https://doi.org/10.1109/TMM.2015.2477242)
- Hastie, T., Rosset, S., Zhu, J., Zou, H.: Multi-class AdaBoost. *Stat. Interface* **2**(3), 349–360 (2009)
- Jain, M., Jégou, H., Bouthemy, P.: Better exploiting motion for better action recognition. In: Proceedings of conference on computer vision pattern recognition, Portland (2013). <http://hal.inria.fr/hal-00813014>
- Jiang, Y.G., Dai, Q., Xue, X., Liu, W., Ngo, C.W.: Trajectory-based modeling of human actions with motion reference points. In: Proceedings of the 12th European conference on computer vision, vol. part V, ECCV'12, pp. 425–438 (2012)
- Kantorov, V., Laptev, I.: Efficient feature extraction, encoding and classification for action recognition. In: Proceedings of conference on computer vision and pattern recognition (2014)
- Karpathy, A., Toderici, G., Shetty, S., Leung, T., Sukthankar, R., Fei-Fei, L.: Large-scale video classification with convolutional neural networks. In: CVPR (2014)
- Khosla, A., Zhou, T., Malisiewicz, T., Efros, A.A., Torralba, A.: Undoing the damage of dataset bias. In: Proceedings of European conference on computer vision, pp. 158–171 (2012)
- Kliper-Gross, O., Gurovich, Y., Hassner, T., Wolf, L.: Motion interchange patterns for action recognition in unconstrained videos. In: Proceedings of European conference on computer vision, ECCV'12, pp. 256–269. Springer-Verlag, Berlin (2012)
- Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. *Adv. Neural Inf. Process. Syst.* **25**, 1097–1105 (2012)
- Kuehne, H., Jhuang, H., Garrote, E., Poggio, T., Serre, T.: HMDB: a large video database for human motion recognition. In: Proceedings of international conference on computer vision (2011)
- Lan, Z., Li, X., Lin, M., Hauptmann, A.G.: Long-short term motion feature for action classification and retrieval. *CoRR* (2015). [arxiv:1502.04132](https://arxiv.org/abs/1502.04132)
- Laptev, I.: On space-time interest points. *Int. J. Comput. Vis.* **64**(2–3), 107–123 (2005)
- Laptev, I., Marszalek, M., Schmid, C., Rozenfeld, B.: Learning realistic human actions from movies. In: Proceedings of conference on computer vision and pattern recognition, pp. 1–8 (2008)
- Lazebnik, S., Schmid, C., Ponce, J.: Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. *Proc. Conf. Comput. Vis. Pattern Recogn.* **2**, 2169–2178 (2006). doi:[10.1109/CVPR.2006.68](https://doi.org/10.1109/CVPR.2006.68)
- Liu, J., Luo, J., Shah, M.: Recognizing realistic actions from videos “in the wild”. In: Proceedings of conference on computer vision and pattern recognition, pp. 1996–2003 (2009)
- Murthy, O., Goecke, R.: Ordered trajectories for large scale human action recognition. In: Proceedings of international conference on computer vision and pattern recognition, pp. 412–419 (2013)
- Nasrollahi, K., Guerrero, S., Rasti, P., Anbarjafari, G., Baro, X., Escalante, H.J., Moeslund, T.: Deep learning based super-resolution for improved action recognition (2015)
- Niebles, J., Chen, C.W., Fei-Fei, L.: Modeling temporal structure of decomposable motion segments for activity classification. *Proc. Eur. Conf. Comput. Vis.* **6312**, 392–405 (2010)
- Peng, X., Wang, L., Wang, X., Qiao, Y.: Bag of visual words and fusion methods for action recognition: comprehensive study and good practice. *Comput. Res. Repos.* (2014). [arxiv:1405.4506](https://arxiv.org/abs/1405.4506)
- Perronnin, F., Sánchez, J., Mensink, T.: Improving the Fisher kernel for large-scale image classification. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010—European conference on computer vision, vol. 6314, pp. 143–156 (2010)
- Péteri, R., Fazekas, S., Huiskes, M.J.: DynTex : a comprehensive database of dynamic textures. *Pattern Recogn. Lett.* (2010)
- Ramana Murthy, O., Goecke, R.: Ordered trajectories for large scale human action recognition. In: Proceedings of international conference on computer vision (2013)
- Raptis, M., Soatto, S.: Tracklet descriptors for action modeling and video analysis. In: Proceedings of European conference on computer vision, pp. 577–590. Berlin, Heidelberg (2010)
- Reddy, K.K., Shah, M.: Recognizing 50 human action categories of web videos. *Mach. Vis. Appl.* **24**(5), 971–981 (2013)

35. Sadanand, S., Corso, J.J.: Action bank: A high-level representation of activity in video. In: CVPR, pp. 1234–1241. IEEE Computer Society (2012)
36. Schuldts, C., Laptev, I., Caputo, B.: Recognizing human actions: a local svm approach. In: Proceedings of conference on pattern recognition, vol. 3, pp. 32–36 (2004)
37. Shi, F., Petriu, E., Laganriere, R.: Sampling strategies for real-time action recognition. In: Proceedings of conference on computer vision and pattern recognition (2013)
38. Simonyan, K., Vedaldi, A., Zisserman, A.: Deep fisher networks for large-scale image classification. *Adv. Neural Inf. Process. Syst.* **26**, 163–171 (2013)
39. Simonyan, K., Zisserman, A.: Two-stream convolutional networks for action recognition in videos. *Adv. Neural Inf. Process. Syst.* **27**, 568–576 (2014)
40. Solmaz, B., Assari, S.M., Shah, M.: Classifying web videos using a global video descriptor. *Mach. Vis. Appl.* **24**(7), 1473–1485 (2013). doi:[10.1007/s00138-012-0449-x](https://doi.org/10.1007/s00138-012-0449-x)
41. Soomro, K., Zamir, A.R., Shah, M.: Ucf101: A dataset of 101 human actions classes from videos in the wild. *Comput. Res. Repos.* (2012). [arxiv:1212.0402](https://arxiv.org/abs/1212.0402)
42. Srivastava, N., Mansimov, E., Salakhutdinov, R.: Unsupervised learning of video representations using lstms. *CoRR* (2015). [arxiv:1502.04681](https://arxiv.org/abs/1502.04681)
43. Sultani, W., Saleemi, I.: Human action recognition across datasets by foreground-weighted histogram decomposition. *Proc. Conf. Comput. Vis. Pattern Recogn.*, pp. 764–771 (2014). doi:[10.1109/CVPR.2014.103](https://doi.org/10.1109/CVPR.2014.103)
44. Sun, D., Roth, S., Black, M.: Secrets of optical flow estimation and their principles. In: Proceedings of conference on computer vision and pattern recognition, pp. 2432–2439 (2010)
45. Tang, K., Fei-Fei, L., Koller, D.: Learning latent temporal structure for complex event detection. *IEEE Conf. Comput. Vis. Pattern Recogn. (CVPR)*, pp. 1250–1257 (2012). doi:[10.1109/CVPR.2012.6247808](https://doi.org/10.1109/CVPR.2012.6247808)
46. Torralba, A., Efros, A.A.: Unbiased look at dataset bias. In: Proceedings of conference on computer vision and pattern recognition (2011)
47. Ullah, M.M., Laptev, I.: Actlets: a novel local representation for human action recognition in video. In: Proceedings of IEEE international conference on image processing, pp. 777–780 (2012)
48. Vrigkas, M., Karavasilis, V., Nikou, C., Kakadiaris, A.: Matching mixtures of curves for human action recognition. *Comput. Vis. Image Underst.* **119**, 27–40 (2014)
49. Wang, H., Klaser, A., Schmid, C., Liu, C.L.: Action recognition by dense trajectories. In: Proceedings of conference on computer vision and pattern recognition, pp. 3169–3176 (2011)
50. Wang, H., Kläser, A., Schmid, C., Liu, C.L.: Dense trajectories and motion boundary descriptors for action recognition. *Int. J. Comput. Vis.* **103**(1), 60–79 (2013)
51. Wang, H., Muneeb Ullah, M., Kläser, A., Laptev, I., Schmid, C.: Evaluation of local spatio-temporal features for action recognition. University of Central Florida, Florida (2009)
52. Wang, H., Schmid, C.: Action Recognition with Improved Trajectories. In: Proceedings of international conference on computer vision, Sydney, pp. 3551–3558 (2013). doi:[10.1109/ICCV.2013.441](https://doi.org/10.1109/ICCV.2013.441). <http://hal.inria.fr/hal-00873267>
53. Wang, L., Qiao, Y., Tang, X.: Action recognition with trajectory-pooled deep-convolutional descriptors. *Comput. Res. Repos.* (2015)
54. Weinzaepfel, P., Revaud, J., Harchaoui, Z., Schmid, C.: Deepflow: large displacement optical flow with deep matching. *Proc. Int. Conf. Comput. Vis.*, pp. 1385–1392 (2013). doi:[10.1109/ICCV.2013.175](https://doi.org/10.1109/ICCV.2013.175)
55. Willems, G., Tuytelaars, T., Gool, L.: An efficient dense and scale-invariant spatio-temporal interest point detector. In: Proceedings of European conference on computer vision, Berlin, pp. 650–663 (2008)
56. Wulff, J., Butler, D.J., Stanley, G.B., Black, M.J.: Lessons and insights from creating a synthetic optical flow benchmark. In: Proceedings of European conference on computer vision, pp. 168–177 (2012)
57. Zhang, J., Marszalek, M., Lazebnik, S., Schmid, C.: Local features and kernels for classification of texture and object categories: a comprehensive study. *Proc. Conf. Comput. Vis. Pattern Recogn.*, p. 13 (2006). doi:[10.1109/CVPRW.2006.121](https://doi.org/10.1109/CVPRW.2006.121)



**Cyrille Beaudry** received the M.S. degree in image processing and mathematics from the University of La Rochelle, France (2012) and the Ph.D. degree from the same university in 2015. He is currently a post-doctoral fellow in the Mathematics, Image and Application laboratory, University of La Rochelle. His research interests include video analysis, pattern recognition and human activity recognition.



**Renaud Péteri** received the engineering degree in physics and image processing from Telecom Physique Strasbourg, France, the M.S. degree in photonics and image processing from the University of Strasbourg, Strasbourg, in 2000, and the Ph.D. degree in image and signal processing from MINES Paris-Tech, Paris, France, in 2003. Since 2005, he has been an Associate Professor with the University of La Rochelle, La Rochelle, France, and a member of the Mathematics, Image and Applications Laboratory. His current research interests include signal and image processing, dynamic textures, video analysis and computer vision. Dr. Péteri received the ERCIM Post-Doctoral Research Grant, spent at the Hungarian Academy of Sciences in 2004 and at the Mathematics and Computer Science Institute, Amsterdam, The Netherlands, in 2005. He was also an invited scholar at the University of California, San Diego in the SVCL laboratory in 2013.



**Laurent Mascarilla** is currently an Associate Professor in the Mathématiques, Image et Applications Laboratory at the University of La Rochelle in France. He received his Ph.D. in Computer Science at the University of Toulouse in France in 1996 and obtained his habilitation in 2008. He is a member of the Computer Science Department, University of La Rochelle. His current research projects focus on video sequence analysis. Dr. Mascarilla's areas of expertise

include color image processing and analysis and pattern recognition applications.