Pattern Anal Applic (2014) 17:17–36
DOI 10.1007/s10044-012-0292-8

THEORETICAL ADVANCES

Human action recognition from simple feature pooling

Manuel J. Marín-Jiménez · Nicolás Pérez de la Blanca · M. Ángeles Mendoza

Received: 22 February 2011 / Accepted: 1 August 2012 / Published online: 2 September 2012

Abstract Human action recognition (HAR) from images is an important and challenging task for many current applications. In this context, designing discriminative action descriptors from simple features is a relevant task. In this paper we show that very good descriptors can be build from simple filter outputs when multilevel architectures and non-linear transformations are used. We propose a new multiscale descriptor for HAR from a Pyramid of Accumulated Histograms of Optical Flow. We also show that, in this case, space–time gradients provide sufficient information for the recognition task. Our descriptor is evaluated on three standard databases of human actions: KTH, Weizmann and IXMAS. We compare very favorably the results of our descriptor with the current results for these three databases from other algorithms. In particular, our descriptor is directly comparable to the state-of-the-art on KTH database with an average of 96 % of correct recognition.

Keywords Human motion · Action description and recognition · Feature pooling · Optical Flow

M. J. Marín-Jiménez (✉)
Department of Computer Science and Numerical Analysis, University of Córdoba, Edif. C3, bajo. Campus de Rabanales, 14071 Córdoba, Spain
e-mail: mjmarin@uco.es

N. Pérez de la Blanca · M. Á. Mendoza
Department of Computer Science and Artificial Intelligence, University of Granada, 18071 Granada, Spain
e-mail: nicolas@ugr.es

M. Á. Mendoza
e-mail: nines@decsai.ugr.es

1 Introduction

A lot of attention has been given to the task of human action recognition (HAR) from image sequences in the last years [32, 36]. Due to the importance of human behavior in many aspects of life, this task has shown to be very relevant in very important multimedia applications such as video surveillance, advanced computer interfaces, video annotation, indexing and browsing [6, 11]. Many different approaches have been proposed to characterize human actions [32], ranging from simple holistic ones where the appearance information of the full image is used to characterize the action, to the most complex ones fitting sophisticated human-body models to get a sequence of body poses from which to identify the action. Nevertheless, the challenge remains active, mainly due to the difficulty of modelling the 3D variability associated to human actions from 2D images.

More recent approaches combine information from appearance with structural information provided by pose-based statistical models [46] or directly with pose-based geometrical models [38]. These approaches, in addition to the difficult template initialization step, carry out an expensive computational effort in the model fitting process. As reward, an increase in accuracy in the motion estimation of the human-body parts is obtained. However, to recognize global human-body behaviors, the approach appears to be very costly. A simpler alternative is to consider an improved holistic model extracting features from non-overlapping regions of the images [12, 24, 45, 54]. However, a relevant problem in these approaches is the loss of discriminative power due to the correlations between the feature vector components when complex features are created.

A common technique in HAR is to stack the image sequence, along the time axis, generating a 3D block containing 3D local features defined by the gray-level

changes associated to the human-body motion. Nevertheless, the high variability of the local deformations associated to the subject's performance, in addition to the global changes from different camera viewpoint, makes difficult to get robust training information from local space–time events of the 3D volume [10, 21, 23, 40–42, 54].

In the last years, important improvements on descriptor generation, feature combination and multilayer architectures for object recognition in still images have been reported [19, 34]. One of the most interesting findings regarding the new descriptors is the relevant role played on classification tasks by simple non-linear transformations (NLTs). In Refs. [19, 37, 44], it is shown that the use of some kind of NLTs on the output of low-level filters improves the recognition score of simple models to the state-of-the-art on very well-known databases. This points out that NLTs are needed as a mechanism of selecting the relevant information eliminating the correlation between the feature values and improving their discriminative power.

In the HAR framework, these findings are relevant in two ways. Firstly, the search of discriminant descriptors from simpler features is an important goal by itself to maintain a low processing amount by each image. Secondly, the identification of good middle-level features from NLTs of filter outputs precludes the need of searching complex features fixing the emphasis on the set of transformations.

In this work, we mainly focus on the discriminative power of the image motion to recognize 3D motion. In this way we approach the HAR task under the following conditions: (1) the camera viewpoint is fixed but unrestricted and the person performs the action freely; (2) the actions are performed by an isolated actor. The effect of occlusions has been considered in simulated experiments.

1.1 Contributions of this work

We approach the HAR task from a new action descriptor based on several steps including normalization and local NLTs of simple filter outputs. We consider temporal information as the output of the low-level filters. The gradients along the time are calculated from the optical flow (OF). Therefore, dense maps of optical flow from every two adjacent frames are used in our experiments.

The descriptor is defined as an architecture of five steps (see Fig. 1): (1) initial transformation [i.e., bounding-box (BB) cropping and interpolation to fix the image size]; (2) low-level filtering (e.g., spatial gradient, optical flow); (3) NLT of the filter output (i.e., normalized histograms of quantized orientations and magnitudes); (4) aggregation of the temporal information to extract stable estimations (i.e., histograms accumulated along time); (5) multiscale descriptor.

A wide experimental study of our descriptor in terms of classification performance is carried out on three well-known databases of video sequences: KTH [42], Weizmann [3] and IXMAS [52]. Comparative results with the state of the art on these databases are shown in this paper.

The main contributions presented in this work are: (1) a new focus on the relevance of simple NLTs instead of complex functions from simple features; (2) a new multi-scale action descriptor (PaHOF) based on pooling and NLTs of the low-level filter outputs; (3) an experimental study evaluating Pyramid of Accumulated Histograms of Optical Flow (PaHOF) under different assumptions and comparing its performance with the state-of-the-art results on the three mentioned databases.

1.2 Outline of the paper

In Sect. 2, we discuss the related works. In Sect. 3, we introduce a single-scale motion descriptor, Accumulated Histograms of Optical Flow (aHOF). In Sect. 4, we extend it to multiple spatial and temporal scales, resulting in a richer descriptor named PaHOF. In Sect. 5, diverse experiments of action recognition with aHOF and PaHOF are carried out on standard databases. Finally, the conclusions are presented in Sect. 6.

2 Related works

In the last decade, different parametric and non-parametric approaches, to obtain good descriptors for the HAR task, have been proposed (see for instance [32, 36, 48]). Nevertheless, as it has already been mentioned, video sequence classification of human motion is a challenging and open problem, at the root of which is the need of searching for invariant characterizations of 3D human motions from 2D image features. In particular, for the camera viewpoint, motion and type, subject performance, lighting, clothing, occlusion and background. In the recent years, different approaches building *middle-level* (informative) descriptors from spatial gradients or shape descriptors have been proposed (see [10, 12, 14, 25, 27, 39, 43]).

Sun et al. [47] present a method to fuse local (i.e., SIFT-based descriptors) and holistic (i.e., Zernike moments) information based on frame differencing for HAR. Their experiments show that whereas local descriptors are more suitable for some datasets (e.g., KTH dataset), holistic descriptors are better for other datasets (e.g., Weizmann dataset). The main drawback of their approach is that it relies on the computation of frame differencing and it is not suitable for dynamic backgrounds. Wang et al. [50] carry out a comparison of different local spatio-temporal features on three action datasets. There is not a clear winner among

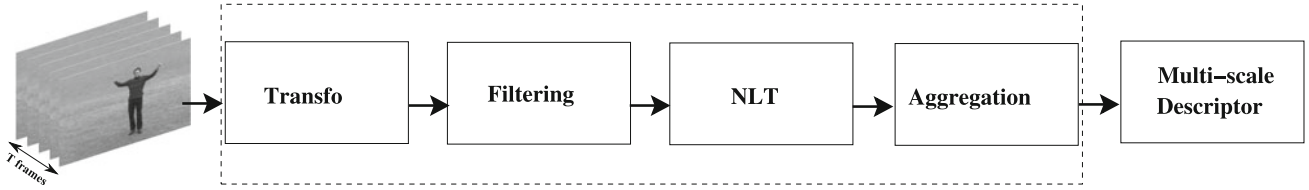

Fig. 1 Processing stages of the proposed model. The input data is a video sequence and the output is a multiscale descriptor of the video sequence: simple transformation (e.g., frame cropping and scaling); image filtering (e.g., optical flow computation); non-linear transformation (NLT) of the data (e.g., histograms of discretized OF); aggregation of the data (e.g., temporal accumulation of the histograms)

the evaluated descriptors, but what seems clear for the authors is that the combination of spatial gradients and optical flow is a good choice on the evaluated datasets. They also state the importance of further research of OF-based descriptors.

Ballan et al. [2] introduce a bag-of-words framework, where the spatio-temporal volumes are represented by the combination of 3D spatial gradients and optical flow, which shows competitive results in HAR. Kovashka and Grauman [22] propose a method to learn hierarchies of space–time features to be used in combination with a bag-of-words model.

The use of the bounding-box covering the person helps to discard background clutter and allows to include, at some degree, some geometric information in the motion descriptors.

The main motivation of most of these approaches follows the bag of features scheme suggested for object recognition. That is, a set of space–time interest points are detected along the image sequence and one descriptor from a volume around each of them is extracted. Then this information is combined, using different approaches, to generate middle-level features which feed a classifier with.

Serre et al. [44], and Pinto et al. [37] inspired in the behavior of the human visual system, build image features from simple filter responses followed by non-linear and pooling operations for object-recognition tasks. In both cases, they got results in the state-of-the-art on the Caltech-101 database. In Jarret et al. [19], an experimental study with different multilayer architectures shows that non-linear rectifying transformations are the single most important factors for improving recognition.

Although the above-mentioned HAR-approaches include some of these steps, they mainly do address the search of potential *middle-level* features to feed a classifier with. In contrast with it, here we focus on the design of powerful action descriptors from the appropriate combination of a multilevel architecture with some NLTs. We start building up the 3D image block by cropping and scaling the bounding-box sequence. We consider this as a 3D image showing a deformable object seen from a specific viewpoint. It is worthy to note that this case shows higher variability than the 3D rigid-object-recognition case.

To provide our descriptor with multiscale capability, we adapt the Pyramid-Match approach [18, 26] to a framework to do matching at different resolutions for object recognition. To do this, we compute spatially localized descriptors inside a grid of non-overlapping image windows and accumulate them along the image sequence. Then, we extend the previous descriptor with multiresolution spatial grids.

3 Single-scale descriptor: aHOF

The first step in our design is to characterize a single-scale motion. We use previous ideas as inspiration [28, 30] but in a different way.

3.1 Computing aHOF

For each image, we focus our interest on the bounding-box area enclosing the actor performing the action. This BB can be computed by following different approaches, for example, by background segmentation [1, 3], with person detectors (i.e., full-body [9, 15], upper-body [16]) and/or person trackers [7, 51].

In our case, for each sequence, the BB is fixed in size to include all possible arm positions. Next, we interpolate each BB to a fixed size to get the final sequence.

The optical flow computed from each pair of consecutives frames is represented by a set of *orientation* × *magnitude* histograms (HOF) from non-overlapping regions (grid). Each optical flow vector votes into the bin associated to its magnitude and orientation. The sequence-descriptor, named *aHOF*, is a transformed version of the motion descriptors accumulated along the sequence.

The main steps of aHOF computation are summarized in Fig. 2: (1) spatial grid definition inside the BB (Fig. 2a); (2) HOF computation per cell (Fig. 2b); (3) accumulation of HOF along time and normalization (Fig. 2c).

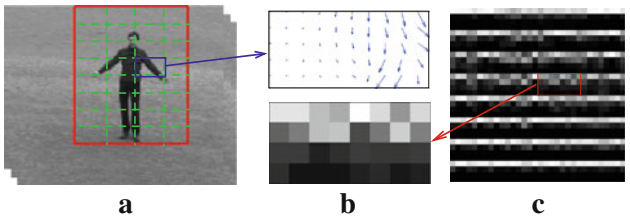

Fig. 2 How to compute aHOF descriptor. **a** BB enclosing person, with superimposed grid (8×4). **b** *Top*: optical flow inside the selected grid cell for the visible single frame. *Bottom*: for that spatial cell, each column (one per orientation) is a histogram of OF magnitudes (i.e., 8 orientations $\times$ 4 magnitudes). **c** aHOF computed from 20 frames around the visible one. Note that in the areas with low motion (e.g., *bottom half*) most of the vectors vote in the lowest magnitude bins (intensity coding: *white* = 1, *black* = 0)

Let $HOF(I_t)$ be the HOF descriptor of image I_t in the sequence, therefore, the (unnormalized) aHOF descriptor of Nth order (aHOF$_N$) for a sequence of images $\{I_1, \ldots, I_N\}$ is defined by the following expression:

$$aHOF_N(I_1, \ldots, I_N) = \sum_{t=1}^{N} HOF(I_t) \tag{1}$$

The dimensionality of an aHOF descriptor is the same that each HOF descriptor in the sequence. In particular, an aHOF descriptor with L spatial cells, O OF bin orientations and M OF bin magnitudes, is a compound of $L \times O \times M$ elements.

To get invariance on the vector magnitude, we have tried the following two different transformations.

(1) A normalization of the magnitude values associated to each orientation bin, independently for each window histogram (see Fig. 2b). This provides an estimation of the orientation distribution,

$$aHOF(l, o, m) \leftarrow aHOF(l, o, m)/C_{\text{lo}} \tag{2}$$

where $C_{\text{lo}} = \sum_k aHOF(l, o, k)$, and indexes l, o, m refer to location, orientation and magnitude, respectively.

(2) A non-linear rectification of each one of the accumulated histograms. We use a sigmoidal function with parameters tuned per each orientation.

The rectified value $\sigma(b_i)$ for each input bin-value b_i is defined as:

$$\sigma(b_i) = [1 + \exp^{-\beta \cdot (b_i - \mu)}]^{-1} \tag{3}$$

where β (steepness) and μ (mean) are parameters to be tuned. The output values will be in the interval $[0, 1]$, as done by the normalization.

In both cases, these transformations have shown to be very relevant to improve the recognition score.

The algorithm to compute an aHOF descriptor is summarized as follows:

ALGORITHM: aHOF

Input: A sequence $\mathbf{I_{N+1}}$ of images (same dimensions) where the person performing the action is centered.

Step 1. Compute OF for each pair of frames (I_t, I_{t+1}).

Step 2. Compute *HOF* descriptor (unnormalized) for each one of the N pairs of frames (I_t, I_{t+1}) based on the computed OF.

Step 3. Compute value of bin (i, j, k) in aHOF descriptor by adding all values (along the sequence) from bins (i, j, k) from the previously computed *HOF* descriptors:

$$aHOF_N(i, j, k; \mathbf{I_{N+1}}) \leftarrow 1 + \sum_{t=1}^{N} HOF(i, j, k; I_t)$$

Step 4. Normalize or rectify the bin values (see text, eq. 2 and eq. 3).

An aHOF descriptor represents a summary of the motion for a N-frames sequence. That is, we compute the aHOF descriptor of order N for a frame i, taking into account the N previous frames.

Figure 3 shows the aHOF representation for different actions in KTH database. The descriptor has been computed from a window of 20 frames previous to the

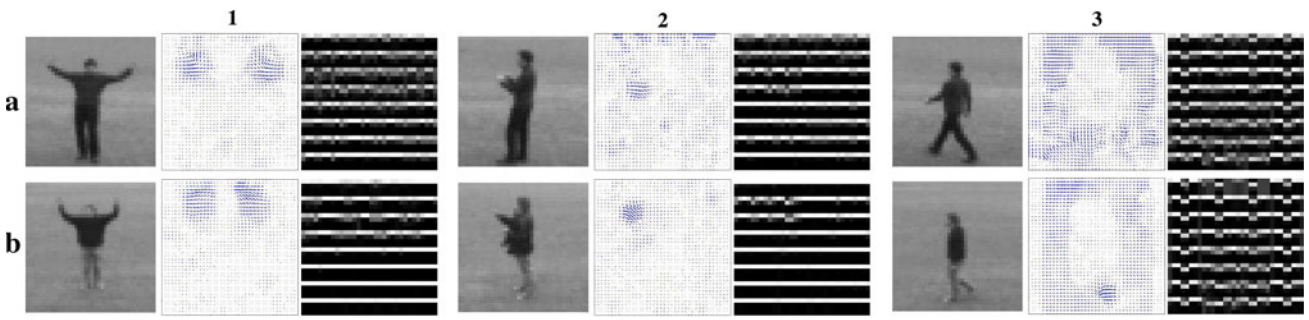

Fig. 3 Examples of aHOF for different actions. Along with the video frames, the estimated optical flow and the aHOF descriptor (order 20) are shown. Note that the descriptor looks different for the diverse class actions, but similar for the same class actions. *Columns from left to right* correspond to actions: *handwaving*, *boxing* and *walking*. Here, normalization has been used

displayed frame. Columns from left to right, correspond to actions: *handwaving*, *boxing* and *walking*.

3.2 Properties of aHOF

Our descriptor has been designed with the idea of being independent of the subject appearance and robust to (1) shifts in the location of the BB, (2) noise estimation of the OF, and (3) variations in the performance (i.e., velocity) of the same actions by different individuals. As an additional property, we aim to make it simple/quick to compute.

The main difference of our descriptor with others published in the literature is that aHOF is based on the aggregation, inside the region of interest, of the information along time. In this way, we take into account not only the relevant gray-level changes but also the regularities along time. This contrasts with the representations based on the extraction of interest points from the volume representation. In our case from a single pass on the sequence, we build the action descriptor. Moreover, since our descriptor is accumulative along time, it could be used in online classification processes by adding a stopping rule in the accumulation stage.

4 Multiscale descriptor

4.1 aHOF at multiple spatial resolutions

In Sect. 3, we have described how to build our motion descriptor at a single spatial grid resolution. However, the results reported in Refs. [18] and [5] showed the relevance of using a multiscale descriptor in order to get higher level of robustness to noise. In this section, we introduce a pyramidal representation of aHOF. That is, spatial grids of different resolutions are defined to describe the same image window.

Figure 4 represents a pyramid of HOF with three levels of spatial grid resolution: 2×1, 4×2 and 8×4. At each cell, a histogram of optical flow (*orientation* $\times$ *magnitude*) is computed.

4.1.1 Pyramids of aHOF: PaHOF

A Pyramid of aHOF descriptor can be defined as the concatenation of L aHOFs descriptors with different spatial grid resolutions. The histogram normalization is done as in aHOF, at each level l_i independently.

PaHOF can be computed in an efficient way if the grid configurations of the pyramid levels are related by an integer factor (e.g., $\times 2^n$, $\times 3^n, \ldots$). We firstly compute the HOF descriptors at the level with the finest resolution (e.g., level 8×4, in Fig. 4). Then, to compute a HOF at a coarser level (e.g., level 4×2 in Fig. 4), we can aggregate the contributions of the N cells at the previous (finer) level that fall into the current spatial cell.

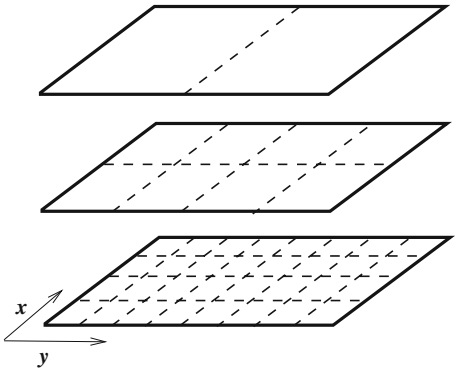

Fig. 4 Pyramidal representation of accumulated motion. In this representation, three levels of spatial grid resolution are used: 2×1, 4×2 and 8×4. At each cell, a histogram of optical flow (*orientation* $\times$ *magnitude*) is computed. In the case of PaHOF, levels are not weighted to define the feature vector, and normalization is done as in aHOF, at each level independently. In contrast, in the case of aPM, the different levels are weighted and the normalization is global to the pyramid

Note that these steps can be done either before or after the accumulation along time. However, the histogram normalization (or rectification) should be done as the final step.

4.1.2 Baseline: aPM

To compare our proposed descriptors (aHOF and PaHOF), as a baseline, we define a descriptor based in the pyramidal representation proposed by Lazebnik et al. [26]. This descriptor will, hereinafter, be referred as accumulated Pyramid-Match (aPM).

It contains multiple levels of spatially localized histograms of optical flow (orientation $\times$ magnitude bins). Since the representation in Ref. [26] is intented to be used with still images, and our aim is to deal with video sequences, we use the same underlying idea as in aHOF, the accumulation of the histograms along the time.

The computation of aPM descriptor with $L + 1$ levels is as follows: ALGORITHM aPM

Step 1. At level i, define a spatial grid with $M_i \times N_i$ cells.
Step 2. For each cell in the grid at level i, compute a 2D histogram of OF by quantizing orientation and magnitude.
Step 3. Apply different weights w_l to the bins depending on the level l. Let $l = 0, \ldots, L$ be the level id, and using the following equation: if level 0, then $w_0 = 1/2^L$, otherwise, $w_l = 1/2^{(L-l+1)}$.

Step 4. Concatenate all histograms from all levels to build a vector-wise representation.
Step 5. Accumulate bins along time.
Step 6. Normalize vector values by dividing each element by the total sum of the elements.

Here as in PaHOF, the image resolution is always the same, only the spatial grid configuration changes. The main differences with regard to PaHOF are the use of weights that depend on the pyramid level and the bin normalization with a global value.

4.2 Multiple temporal scales (MTS)

Here, we propose two ways of combining motion information at diverse temporal scales to analyze its discriminative strength.

A first approach is to concatenate N PaHOFs (with D dimensions) of different temporal orders to obtain a new feature vector with $N \cdot D$ dimensions. This descriptor will contain motion information at multiple spatial and temporal scales. To reduce the dimensionality of the descriptor, we tried out two different techniques: PCA and LSH. However the classification results did not show any improvement.

4.2.1 Bag of PaHOF: BOP

Inspired in the *Bag of Words* (BOW) representations [4, 31], we approach a video sequence as a *Bag of PaHOFs* (BOP). The idea is to compute PaHOFs of different temporal orders along the video sequence, and then, to build a histogram of PaHOFs by previously assigning a cluster identifier to each of them. The learning algorithm is as follows:

ALGORITHM BOP

Step 1. Compute PaHOF of L spatial levels and (temporal) order $N_1, N_2, \ldots$ (for example, 10, 15 and 20 frames) on the training sequences to get a set of F features.
Step 2. Apply K-means to the set F' to discover the K centroids that will be used as members of the *codebook* B.
Step 3. To represent (encode) a video sequence, assign a cluster-ID to each PaHOF in the sequence and build a histogram of cluster-IDs.
Step 4. Normalize the histogram.

We assign each PaHOF computed in the test sequence to the nearest codebook member using the Euclidean distance to the centroids. In contrast to our first alternative approach, the dimensionality of this representation is equal to the number of clusters.

5 Experiments

We perform a number of experiments to evaluate the quality of the proposed descriptors. Firstly, in Sect. 5.3, we carry out experiments to evaluate (aHOF) at a single spatial and temporal scale. Then, in Sects. 5.4 and 5.5, PaHOF is evaluated and compared to aHOF.

The set of parameters that defines our proposed descriptor are: (1) number of OF orientation bins, (2) OF magnitude intervals, (3) spatial grid configuration. In our case, these parameters have been empirically fixed using a grid of values on a validation set (see Sect. 5.2).

5.1 Databases

We test our approach on three publicly available databases that have been widely used in action recognition: KTH human motion dataset [42], Weizmann human action dataset [3] and INRIA Xmas Motion Acquisition Sequences (IXMAS) [52]. Different results are available on these databases, therefore, we contrast our results with them.

5.1.1 KTH database

This database contains a total of 2,391 sequences, where 25 actors perform six classes of actions (*walking, running, jogging, boxing, handclapping* and *handwaving*). The sequences were taken in four different scenarios: outdoors (s1), outdoors with scale variation (s2), outdoors with different clothes (s3) and indoors (s4). In general, each action is repeated by each actor four times in each scenario.

Some examples are shown in Fig. 5. As in Ref. [42], during the experiments, we split the database in 16 actors for training and 9 for test.

In our experiments, we consider KTH as five different datasets: each one of the four scenarios is a different dataset, and the mixture of the four scenarios is the fifth

Fig. 5 KTH dataset. Typical examples of actions included in KTH dataset. *From left to right*: *boxing, handclapping, handwaving, jogging, running, walking*

Fig. 6 Weizmann dataset. Typical examples of actions included in Weizmann dataset

one (hereafter *s1234*). In this way, we make our results comparable with others appeared in the literature.

5.1.2 Weizmann database

This database consists of 93 videos[1], where 9 people perform 10 different actions: *walking, running, jumping, jumping in place, galloping sideways, jumping jack, bending, skipping, one-handwaving* and *two-hands waving*. Some examples are shown in Fig. 6.

In our experiments, we report average results by performing *leave-one-out* on the actors.

5.1.3 IXMAS database

This database contains a total of 14 actions performed by 12 persons three times each and recorded using five cameras with diverse viewpoints. The action names are: *check-watch*, *cross-arms*, *scratch-head*, *sit-down*, *get-up*, *turn-around*, *walk*, *wave*, *punch*, *kick*, *point*, *pick-up*, *throw-over-head* and *throw-from-bottom-up*. Some examples are shown in Fig. 7.

In our experiments, we report average results by performing *leave-one-out* on the actors.

5.2 Experimental setup

On each frame, we estimate the region of interest fixing a BB. We use a simple thresholding method, based on the one proposed in Ref. [35], approximating size and mass center, and smoothed along the sequence. BBs proportional to the relative size of the object in the image, and large enough to enclose the entire person, regardless of his pose, have been used (Fig. 2a). All the cropped frames are scaled to the size of 40×40 pixels. Then, the Farnebäck's algorithm [13] is used to estimate the motion vector on each pixel. The estimated motion gives us an estimation of the local motion defining the action, the shifting motion has been already removed by the spatial alignment of the cropped frames.

For all the experiments, we have empirically determined the use of 8-bins for orientation and 4-bins for magnitude: $[0, 0.5], (0.5, 1.5], (1.5, 2.5], (2.5, +\infty)$. We consider as relevant the small motions between frames. We add 1 to all the bins to avoid zeros.

Different values for the spatial grid were tried out in the experiments, but eventually a grid with 8×4 cells was fixed. Then, the full descriptor for each frame is a 1024-vector with values in the interval $(0, 1)$.

We train and test with different discriminative classifiers: (1) SVM with radial basis [8], and (2) GentleBoost (512 rounds) with decision stumps [17]. We also compare both classifiers with kNN using euclidean distance. Since this is a multiclass problem, a *one-vs-all* approach is used on the binary classifiers. We assign a class label to a full video sequence by classifying multiple subsequences (same length) and using a *majority voting* criterion on the subsequences label.

5.3 Experiment: HAR with aHOF

In these experiments we evaluate the performance of aHOF (single scale) in terms of action classification. Firstly, we explore different configurations of the descriptor (i.e., spatial grid and accumulated frames). Then, in Sect. 5.3.2, we investigate whether just one motion descriptor is enough to classify a video sequence and what is the minimum number of frames needed for our descriptor. Finally, we evaluate the influence of histogram normalization and rectification (see Sect. 5.3.3) in terms of recognition.

[1] There are 93 videos instead of 90 since Weizmann's actor *Lena* repeats twice the actions *run*, *skip* and *walk*.

Fig. 7 IXMAS dataset. Typical examples of actions and camera viewpoints included in IXMAS dataset. *Top row*: cameras 0, 1, 2. *Bottom row*: cameras 3 and 4, plus example of BB

5.3.1 Grid configurations

For KTH dataset, all the results we show in this experiment obtained from averaging the results of 10 repetitions of the experiment with different pairs of training/test sets. For Weizmann dataset, results are computed by performing *leave-one-out* on the actors.

To define the best grid size for aHOF, we have carried out experiments with three different grid configurations: 2×1, 4×2 and 8×4. Table 1 shows the results on the KTH dataset and Table 2 the results on the Weizmann dataset. In both cases, the 8×4 grid provides good results, although on the Weizmann dataset, slight better results are obtained with a 4×2 configuration. Note that with the so simple configuration 2×1 (nearly upper body and lower body separation), it is able to classify correctly more than 87 % of the sequences of KTH.

Figure 8 shows the location of the features selected by GentleBoost from the original aHOFs for one of the training/test sets on KTH dataset. For actions implying displacement (e.g., *walking*, *jogging*), the most selected features are located at the bottom half of the grid. However, for those actions where the arm motion defines the action (e.g., *handwaving*), GentleBoost prefers features from the top half. This confirms that our descriptor captured the relevant information from the motion.

Fig. 8 Features selected by GentleBoost from raw aHOF on KTH. Spatial location of features selected by each class-specific GentleBoost classifier. The lighter the pixel the greater the contribution to the classification. *From left to right*: *boxing, handclapping, handwaving, jogging, running, walking*

Table 1 Grid configuration study on KTH

Grid	*1NN*	*5NN*	*9NN*
2×1	87.4	87.5	87.6
4×2	92.2	92.9	93.3
8×4	94.0	94.5	94.3

Classification results on KTH with different grid configurations, using aHOF with k-Nearest Neighbor (kNN)

Table 2 Grid configuration study on Weizmann

Grid	Subseqs	Seqs
2×1	87.7	86.9
4×2	95.3	95.8
8×4	94.3	91.9

Classification results on Weizmann with different grid configurations, using aHOF with SVM

5.3.2 Subsequence length study

Here, we are concerned with two related problems. Firstly, to determine the minimum number of frames needed to recognize an action using aHOF. Secondly, to determine the best length of the subsequences to classify the full sequence using a majority voting criterion. Subsequences are extracted every other frame from the full-length sequence and one feature vector is extracted from each one.

(A) *Minimum length* To determine the amount of consecutive frames that we have to accumulate to recognize a specific action, we extract blocks of N consecutive frames (henceforth *subsequences*) from the full-length sequences.

Experiments on Weizmann Table 3 shows mean classification performance (on the repetitions) and confidence interval (at 95 %) for action recognition on Weizmann database.

The reported results come from the average of nine repetitions, following a *leave-one-out* strategy on the actors.

Table 3 Recognition on Weizmann

N	5	10	15	20	25	30
SVM	71.1; 8.8	81.5; 24.3	**95.6**; 11.3	94.8; 10.9	92.6; 17.8	94.1; 19.0
GB	52.6; 26.5	64.44; 12.0	85.9; 22.1	**87.4**; 16.6	86.7; 23.6	85.2; 18.2

The best mean recognition performance per classifier is marked in bold

Mean and confidence interval (at 95 %) of the performance, using *N* central frames to compute aHOF

Table 4 Recognition on KTH

N	5	10	15	20	25	30
SVM	76.5; 2.3	84.1; 2.0	87.1; 1.6	89.5; 1.3	90.9; 2.0	89.6; 0.5
GB	78.7; 2.2	85.9; 2.3	88.8; 2.3	91.5; 1.8	92.7; 1.6	**93.5**; 1.7

The best mean recognition performance is marked in bold

Mean and confidence interval (at 95 %) of the performance, using *N* central frames to compute aHOF

The results indicate that, on average, nearly the 96 % of the video sequences are correctly classified by just computing a single aHOF descriptor on 15 frames blocks.

These results could explain the behavior of the majority voting scheme previously observed in the experiment summarized in Table 5. That is, the set of frames situated at the beginning and the end of the video sequence, are probably less-informative than the ones situated at the center of the sequence. And, therefore, they are contaminating the class labels.

Experiments on KTH This experiment is performed on the mixed scenarios dataset of KTH database. The reported results come from the average of ten repetitions, where 16 actors are used for training and 9 for testing.

Table 4 shows mean classification performance (on the repetitions) and confidence interval (at 95 %) for action recognition. These results indicate that, on average, nearly the 93.5 % of the video sequences can be correctly classified by just computing a single aHOF descriptor on 30 frames blocks.

The results on KTH and Weizmann show that this number of frames lies in the range $[15, 25]$. We can also see in the tables that the confidence intervals on Weizmann are larger than the ones on KTH. This is probably due to the fact there are less training samples on Weizmann than in KTH for each class action. Moreover, Weizmann contains 10-class actions versus the 6-class actions included in KTH.

(B) *Optimal subsequence length for majority voting* The results for the second problem are summarized in two tables.

Experiments on Weizmann For the Weizmann dataset, our best result is obtained using 30 frames subsequences (see Table 5).

Nevertheless, and in contrast to the results reported on KTH, the classification of the full-length sequences using a *majority voting* scheme worsens the results. This is due to several actions with very few training samples, which introduce a high number of false negatives.

Experiments on KTH In Table 6, we show the classification results both for the individual subsequences (i.e., classified with GentleBoost) and the full sequences on the KTH dataset. The results show that for this database, subsequence lengths between 15 and 25 frames provide the best classification scores.

Once we have estimated that 20 is an intermediate good length to compute the motion descriptor, we perform additional experiments on the different scenarios of KTH with different classifiers (see Table 7).

Scenario 3 results to be the hardest. In our opinion, it is due to the loose clothes worn by the actors, and whose movement creates a great amount of OF vectors irrelevant to the target action.

Table 8 represents the confusion matrix for the classification with SVM on the mixed scenarios *s1234* (see Table 7 for global performance). Note that the greatest confusion is located in action *jogging* with actions *walking* and *running*. Even for a human observer that action is easy to be confused with any of the other two.

5.3.3 Parametrized histogram value rectification

In this experiment, a parametrized rectification is proposed on the histogram bins. Our intuition suggests that different human motions can be better described using weights on the orientation vectors. In our case, this means to use different parameters for the orientation bins in Eq. 3. In this experiment, we consider two cases: (1) common parameters for all histogram bins; (2) different parameters for subsets of orientation bins.

Experiments on KTH Table 9 contains percentages of correct classification averaged on ten repetitions, as in the

Table 5 Evaluating different lengths of subsequences on Weizmann

Len	Subseqs		Seqs	
	GB	SVM	GB	SVM
15	92.0	93.4	92.6	93.3
20	92.8	94.3	91.9	91.9
25	93.9	95.5	91.9	93.3
30	95.8	**96.5**	94.1	**94.8**

The best mean recognition performances are marked in bold

Percentage of correct recognition with GentleBoost and SVM on Weizmann using aHOF. Each row shows the obtained scores for different length subsequences and full sequences using majority voting

Table 6 Evaluating different lengths of subsequences on KTH s1234

	10	15	20	25	30	Full
Seqs	94.4	94.8	94.6	95.0	94.4	93.7
Subseqs	86.2	89.6	91.9	93.0	93.9	93.7

Classification results with GentleBoost on aHOF vectors using subsequences of different lengths. The row *Seqs* shows the sequence classification score using majority voting on subsequences. The row *Subseqs* show the score for different length subsequences. Column *Full* indicates that all frames of the sequence have been used to compute a single aHOF descriptor

Table 7 Classifying full sequences (subseqs. len. 20)

Scenario	Subseqs		Seqs	
	GB	SVM	GB	SVM
s1	92.6	92.3	**95.6**	95.1
s2	92.0	90.5	**97.1**	96.3
s3	89.3	87.4	**89.8**	88.2
s4	94.2	94.3	97.1	**97.6**
s1234	91.9	92.1	94.6	**94.8**

The best mean recognition performance per scenario is marked in bold

GentleBoost and SVM on KTH using aHOF

Table 8 Confusion matrix on KTH: scenario s1234

	box	*hclap*	*hwave*	*jog*	*run*	*walk*
box	**98.6**	1.2	0.2	0.0	0.0	0.0
hclap	4.9	**92.2**	2.8	0.0	0.0	0.0
hwave	1.6	0.2	**98.2**	0.0	0.0	0.0
jog	0.0	0.5	0.0	**89.9**	6.0	3.5
run	0.0	0.0	0.1	8.3	**91.3**	0.3
walk	0.2	0.6	0.0	0.2	0.4	**98.6**

The elements of the main diagonal are marked in bold

Percentages corresponding to full-length sequences. SVM is used for classifying subsequences of length 20. The greatest confusion is located in *jogging* with *walking* and *running*. Even for a human observer that action is easy to be confused with any of the other two

Table 9 Parametrized histogram values rectification on KTH

S	*Len*	*Pars*	Subseqs	Seqs
s1	20	0.005, 125	90.8	**94.3**
s1	20	0.05, 125	87.4	91.6
s2	20	0.005, 125	91.0	**96.7**
s3	20	0.001, 125	88.4	**89.2**
s4	20	0.001, 125	95.7	**97.5**
s1234	20	0.001, 125	90.5	**93.5**

The best mean recognition performance per scenario is marked in bold

Percentage of correct recognition on KTH using aHOF and SVM

Table 10 Parametrized histogram values rectification on Weizmann

Len	*Pars*	Subseqs	Seqs
20	0.005, 125	93.9	92.6
20	0.09, 125	87.0	87.4
20	$0.010-0.005-0.005-0.010, 125$	**94.2**	**94.1**
30	0.005, 188	**96.0**	**94.1**
30	$0.010-0.005-0.005-0.010, 188$	96.0	93.3

The best mean recognition performances are marked in bold

Percentage of correct recognition on Weizmann using aHOF and SVM

previous experiments. SVM is used as classifier for the subsequences and majority voting is used to assign the final label to the full sequences.

Column *Len* refers to the number of frames used in the subsequences. Column *Pars* contains parameters of the sigmoidal function (β and μ).

Only the best combination of the parameters is shown in the table, although a coarse grid search on the parameter space has been performed to tune the parameters of the sigmoidal function. A finer grid search could eventually rise these results.

Experiments on Weizmann Table 10 contains percentages of correct classification by following a *leave-one-out* strategy on the actors, as in the previous experiments. SVM is used as classifier for the subsequences and majority voting is used to assign the final label to the full sequences.

Instead of just using a common β parameter for all the orientation bins, we have experimented with the use of different β values per subset of orientation bins. We consider different subsets of orientations. In particular, $0.010-0.005-0.005-0.010$ means that $\beta = 0.010$ has been used for the first and second orientation bins, $\beta = 0.005$ for the third and fourth ones, $\beta = 0.005$ for fifth and sixth ones, and $\beta = 0.001$ for seventh and eighth ones. Note that only the best combinations of the parameters are shown in the table.

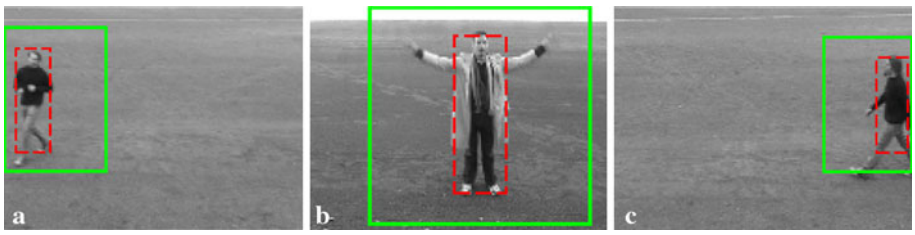

Fig. 9 Defining BBs for aHOF using a pedestrian detector. *Dashed red lines* are the detector outputs. *Solid green lines* are the enlarged regions used as input for aHOF. **a** Since the person is leaving the scene, part of the enlarged BB falls outside the image. **b** Person arms are outside the detection area. **c** The detector has returned a wrong BB scale and portions of body parts are outside the scene (best viewed in color) (color figure online)

The results show that the choice of individual pairs of parameters for the case of $Len = 20$ increases the richness of the motion descriptor in terms of classification performance. In the case of $Len = 30$, the results improve only in terms of subsequence classification and just a single β is valid for all the orientation bins. As in the previous experiments on this dataset, majority voting does not help to classify the full-length sequence. Moreover, note that these results are comparable to the ones reported in Table 5.

In this experiment, our aim was to study if the central part of the data distribution was more discriminative than the tails of it. The sigmoidal-based rectification did the task. In this case, the results indicate that the data located in the tails of the distribution resulted to be the most informative in terms of description. Although the results achieved by the rectification method are not superior to the ones offered by the histogram normalization described in Sect. 3, they lie in the same range and can be seen as an alternative.

5.3.4 Experiment: importance of the BB

The goal of this experiment is to evaluate the influence of the bounding-box tracking error on the quality of our descriptor. This experiment is carried out on KTH dataset with aHOF 8×4.

A well-known pretrained pedestrian detector [15] is run on every single frame of KTH dataset's videos. The detection score threshold is fixed to 0. Given a target video, for each frame, the detection window with the highest detection score is kept for the subsequent stage. To eliminate unexpected shifting in the BB coordinates $\{(x_i, y_i)\}$, a smoothing function on 5-frame windows is applied. Moreover, since the pedestrian detector returns a rectangular BB that typically does not cover the person arms in actions such as *handwaving* (see Fig. 9b), the detection windows are extended to a unified square shape (see solid green BBs in Fig. 9). Finally, the image areas defined by the extended BBs are crop and scaled to a 40×40 pixels size.

Note that the height (in pixels) of the actors in most of the KTH videos is lower than the minimum detection size of the pedestrian detector. Therefore, we rescaled (i.e., $3\times$) every single frame before running the named detector. Then, the BB coordinates were converted to the original coordinates frame space.

The same evaluation procedure has been followed as previously done (i.e., ten repetitions). To decouple the effect of the majority voting stage, we directly perform the study on the extracted subsequences. In particular, we report in Table 11 the classification results on subsequences of length 20 and 30 frames.

The results of this experiment suggest that the proposed descriptor is robust to inaccurate estimations of the BB. In comparison to the results reported in Table 7, in scenarios 1 and 4, the recognition rate on the subsequences is similar to the results offered by previously used BBs. However, in scenario 2, the classification performance decreases around 6 %. In our opinion, the hard scale changes widely present in that scenario have a negative effect on the person scale returned by the detector. Moreover, we can see in Table 11

Table 11 Influence of the bounding-box quality on classification performance

S	*Len*	SVM	GB
s1	20	91.0	92.5
s1	30	93.8	**94.8**
s2	20	82.7	85.9
s2	30	85.1	**89.0**
s3	20	81.9	85.8
s3	30	84.1	**88.5**
s4	20	94.3	96.6
s4	30	95.7	**97.7**

The best mean recognition performance per scenario is marked in bold

Subsequences of *Len* frames are classified by SVM and GB using aHOF as descriptor

that $aHOF_{30}$ deals with the wrong person detections better than $aHOF_{20}$, since the noise is probably better cancelled in a larger temporal range.

5.3.5 Experiment: effect of partial occlusions

To get an idea of the effect of partial occlusions in the motion description by using aHOF, we introduce random black bars in the scenes. We run this experiment on KTH with aHOF 8×4 using the previously computed BBs (see Sect. 5.3.4).

We place black bars at random positions with random width or height (in a given range). In particular, the procedure to generate the occlusions is as follows: (1) choose if the new bar is either vertical (i.e., whole frame height will be used) or horizontal (i.e., whole frame width will be used) with equal probability; (2) if the bar is vertical, choose the initial x_{ini} position at random, and, if it is horizontal, choose the initial y_{ini} position at random; (3) given a maximum percentage $P_{\max}$of the frame width or height (e.g., 20 %), and depending on if the bar is vertical or horizontal, generate a random bar width or height in the range $[1, w_{\max}]$ or $[1, h_{\max}]$, respectively.

In our experiments, we have chosen $P_{\max} = 20$ and, therefore, $w_{\max} = 0.2 \cdot 160$ and $h_{\max} = 0.2 \cdot 120$.

Figure 10 shows some examples of randomly generated occlusions on different actions. Note the different orientations, locations and sizes of the black bars. We have measured that, in the processed videos, around 78 % of the BBs are directly affected by occlusion, although in different degrees. There are cases, as the second one represented in Fig. 10, where the bar does not overlap even the person BB.

Videos with occlusions were used only during testing time. For training, clean videos (i.e., without occlusions) were used. We report results only on the scenarios where the persons where better localised in the previous experiment (Sect. 5.3.4), i.e., scenarios *s1* and *s4*, so the results of this experiment are not much biased by the person detector. We use the assumption that the person BB can be interpolated in the frames where the person is occluded by the black bar.

The recognition performances at subsequence level are reported in Table 12. They suggest that longer subsequences (i.e., $aHOF_{30}$) allow to minimise the effect of the occlusions. Note that the recognition performance decreases up to 9 % with regard to the results summarized in Table 11. This makes sense given some of the examples represented in Fig. 10 where the discriminative motion is mostly occluded (e.g., arms in boxing).

5.4 Experiment: Pyramids of aHOF

In the following experiments, we evaluate the performance of the pyramidal features (PaHOF) in terms of recognition. The reported results will be compared to the ones achieved by the single-scale descriptor (aHOF) evaluated in the previous experiments.

5.4.1 Experiments on KTH

Experiments are carried out on the different scenarios of KTH and the results are summarized in Table 13. The

Table 12 Effect of the occlusions on the classification performance

S	*Len*	SVM	GB
s1	20	80.8	83.7
s1	30	86.6	**87.0**
s4	20	82.0	84.8
s4	30	85.8	**89.3**

The best mean recognition performance per scenario is marked in bold

Subsequences of *Len* frames are classified by SVM and GB using aHOF as descriptor

Table 13 Pyramids of aHOF representation: full sequences classification results

Scenario	*Pyram*	*Single*
s1	**95.9**	95.6
s2	97.0	**97.1**
s3	**90.5**	89.8
s4	96.8	**97.1**
s1234	**96.0**	94.6

The best mean recognition performance per scenario is marked in bold

Subsequences of 20 frames are classified using GentleBoost and majority voting is used afterwards. Classification results in column *Pyram* are compared to single level representation in column *Single* (see Table 7)

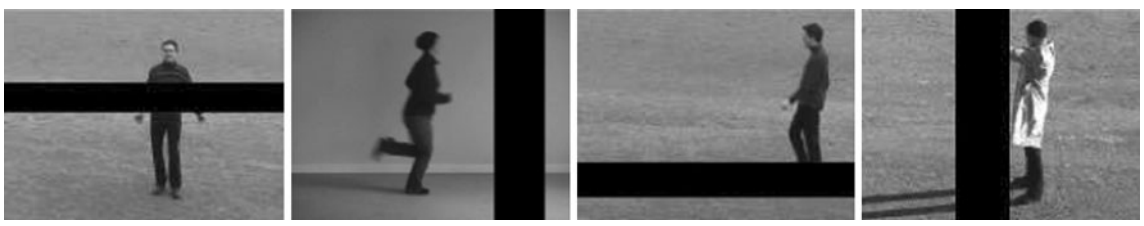

Fig. 10 Simulated occlusions on KTH videos

Table 14 Confusion matrix on KTH with PaHOF: scenario s1234.

	box	*hclap*	*hwave*	*jog*	*run*	*walk*
box	**99.3**	0.6	0.1	0.0	0.0	0.0
hclap	3.6	**95.1**	1.3	0.0	0.0	0.0
hwave	0.7	0.5	**98.8**	0.0	0.0	0.0
jog	0.0	0.4	0.0	**93.8**	3.0	2.8
run	0.0	0.0	0.0	9.8	**90.1**	0.1
walk	0.5	0.1	0.0	0.3	0.0	**99.1**

The elements of the main diagonal are marked in bold

Percentages corresponding to full-length sequences. GentleBoost is used for classifying subsequences of length 20. The greatest confusion is located in *running* with *jogging*

evaluated pyramid of features contains (2×1, 4×2, 8×4) spatial cells of histograms. Reported results are the average on ten repetitions (training/test).

Slight improvements in performance are detected in scenarios 1 and 3 (<1 %), in comparison with Table 7. However, the improvement in the mixed scenarios (s1234) is 1.4 %, raising our best recognition result up to 96 %. The confusion matrix for this scenario with PaHOF is shown in Table 14. Note that the major confusion is between *running* and *jogging*.

Comparison to accumulated Pyramid-Match representation [26] In this experiment, we compare our proposed descriptors with the baseline method described in Sect. 4.1.2. We use GentleBoost (512 rounds) for training/testing on subsequences of length 20 frames.

Table 15 summarizes the classification results achieved with the aPM representation. As in aHOF, eight orientations and four magnitudes are used for quantizing the OF vectors. As in the previous experiments, percentages reported in the table are the average on ten repetitions (training/test).

Two different pyramidal configurations are used in these experiments: pyramid A configuration is (1×1, 2×2, 4×4), therefore, vectors are 672 dimensional; pyramid B configuration is (2×1, 4×2, 8×4), therefore, vectors are 1344 dimensional.

Table 15 Accumulated Pyramid-Match representation: full sequences classification results

Scenario	*Pyr-A*	*Pyr-B*
s1	93.1	**93.6**
s2	**93.8**	93.6
s3	89.4	**89.4**
s4	95.9	**96.2**
s1234	92.1	**92.4**

The best mean recognition performance per scenario is marked in bold

Subsequences of 20 frames are classified using GentleBoost. Majority voting is used afterwards. Configuration of Pyr-A is (1×1, 2×2, 4×4), and Pyr-B is (2×1, 4×2, 8×4). On average, configuration of Pyr-B offers better results that Pyr-A

Note that the results reported using aHOF (see Table 7) are superior to the ones achieved with the aPM representation. If we focus in the mixed scenarios case (s1234), aHOF (single level) improves on Pyr-B the classification performance in around 2 %.

Comparison with the state-of-the-art on KTH DB A comparison of our method with the state-of-the-art performance, on KTH database, can be seen in Table 16. Note that our descriptors, just based on optical flow, offer a classification performance comparable to the best result published up to our knowledge [29] (i.e., 96 %), with the same experimental setup.

We report results for each scenario trained and tested independently, as well as the results for the mixed scenarios dataset. The result reported by Lui et al. [29] corresponds to the mixed scenarios dataset, directly comparable with our *s1234*. Unfortunately, only Jhuang et al. [20] publish the individual results per scenario (here their *Avg*. score is the mean of the separate scenarios). In our case, the mean of the four separate scenarios is 94.3 % with SVM and 94.9 % with GB. On the other hand, the mean of the four scenarios with PaHOF and GB is 95.1 %.

The bottom rows [27, 28, 41, 53, 55] of the Table 16 contain results on this database but using a different experimental setup. In particular, one of the best published results [27] (93.4 % on *s1234*) uses both shape and motion features, and also, they use more actors for training (i.e., *leave-one-out*) than us. On the other hand, although the method proposed by Lucena et al. [28] only uses HOF as well, their best recognition result, using 24 training actors, is lower than ours (i.e., 91.1 vs. 96.0 %). These facts highlight the benefits of using our approach just based on OF.

5.4.2 Experiments on Weizmann

In this experiment, we define pyramids with three levels and the following spatial grid configuration: $8 \times 4 - 4 \times 2 - 2 \times 1$. Table 17 contains the classification results obtained using PaHOF features. The results are the average returned by *leave-one-out* on the actors.

Note that with 15 frames, the mean recognition at sequence level is around 96 % (i.e., 4 out of 93 sequences are incorrectly classified), and with 30 frames, 98 % at subsequence level.

If we compare with the results obtained at a single level (see Table 5), the average improvement is >1 %.

Confusion matrix is shown in Table 18. Note that the greatest confusion is located in *run* with *skip*. Probably,

Table 16 Comparison with the state-of-the-art on KTH

Method	s1234	s1	s2	s3	s4	Avg.
aHOF + SVM	**94.8**	95.1	96.3	88.2	97.6	94.3
aHOF + GB	**94.6**	95.6	97.1	89.8	97.1	94.9
PaHOF + GB	**96.0**	95.9	97.0	90.5	96.8	95.1
Laptev et al. [24]	91.8	–	–	–	–	–
Jhuang et al. [20]	–	96.0	86.1	88.7	95.7	91.6
Fathi and Mori [14]	90.5	–	–	–	–	–
Kovashka and Grauman [22]	94.5	–	–	–	–	–
Lui et al. [29]	96.0	–	–	–	–	–
Lucena et al. [28]	91.1	–	–	–	–	–
Zhang et al. [55]	91.3	–	–	–	–	–
Schindler and van Gool [41]	92.7	–	–	–	–	–
Lin et al. [27]	93.4	98.8	94	94.8	95.5	95.8
Yu et al. [53]	95.7	–	–	–	–	–

The results of this paper for scenario s1234 are marked in bold

Column *s1234* corresponds to 'all-in-one' dataset, and columns *s1–s4* show the results per scenario. *Avg.* column shows the averaged result on the four scenarios. Symbol '–' indicates that such result is not available

Table 17 Classifying actions on Weizmann with PaHOF

Len	Subseqs	Seqs
10	93.7	93.6
15	95.4	**95.8**
20	96.1	95.8
25	96.8	94.7
30	**98.1**	**95.8**

The best mean recognition performances are marked in bold

Percentage of correct recognition with SVM on Weizmann using PaHOF

due to the fact that both actions imply fast displacement and the motion field is quite similar. The remaining confusions are between *jump* and *skip*, where in both actions, the actor is jumping forward but the main difference is that the second one is performed by using just one leg. In this case, OF is not enough to represent such subtle difference, which might be disambiguated by adding gradient-based features (e.g., HOG).

5.4.3 *Experiment: training on KTH and testing on Weizmann*

To evaluate the capability of generalization of the proposed descriptor, we have classified the subset of Weizmann's actions that are common to both datasets with classifiers trained on KTH-s1234. Those actions are: *handwaving with two hands*, *running* and *walking*.

In particular, we have extracted a total of 573 aHOF/PaHOF descriptors of order 20 from 29 video sequences (i.e., the 9 Weizmann's actors are used).

Each descriptor has been tested with a GentleBoost classifier trained in the previous experiments (see Sect. 5.3.2 and Sect. 5.4.1). Since the GentleBoost classifiers used in this paper are binaries, although trained in a *one-vs-all* framework, we have got a total of six classifiers (one per each KTH's action). Therefore, a subsequence could be classified as one of the not used classes (i.e., *boxing*, *handclapping* or *jogging*). Note that class *jogging* could be considered to be an action in between *walking* and *running*, although in our opinion it is more similar to *running*. To cope with this situation, we have considered three possible cases: (1) to use just the three relevant binary classifiers; (2) to use the six binary classifiers; (3) to use the six binary classifiers but fusing *jogging* and *running* labels in one.

Table 19 summarizes the classification results of this experiment.

Column *Features* indicates which descriptor has been used: aHOF or PaHOF. The remaining columns refer to the three possible situations previously disccussed: *3-classes-(a)*, *6-classes-(b)*, *6-classes-fusion-(c)*.

We can see that a perfect classification score at sequence level is achieved in the case of *3-classes* using PaHOF. However, in the case of *6-classes*, the recognition score decreases, since two *running* sequences are classified as *jogging* and one *walking* sequence is classified as *jogging* as well. Note that the KTH's classifiers were trained in a finer-grain manner with regard to the actions that imply displacement. However, the set of Weizmann's actions does not contain any in between *running* and *walking*.

Note that in this case we are using around 256 KTH's training sequences per action (i.e, 16 actors $\times$ 4 scenarios $\times$ 4 repetitions) in contrast to the eight training sequences

Table 18 Confusion matrix on Weizmann

	wave1	*wave2*	*jump*	*pjump*	*side*	*walk*	*bend*	*jack*	*run*	*skip*
wave1	**100.0**	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
wave2	0.0	**100.0**	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
jump	0.0	0.0	**88.9**	0.0	0.0	0.0	0.0	0.0	0.0	11.1
pjump	0.0	0.0	0.0	**100.0**	0.0	0.0	0.0	0.0	0.0	0.0
side	0.0	0.0	0.0	0.0	**100.0**	0.0	0.0	0.0	0.0	0.0
walk	0.0	0.0	0.0	0.0	0.0	**100.0**	0.0	0.0	0.0	0.0
bend	0.0	0.0	0.0	0.0	0.0	0.0	**100.0**	0.0	0.0	0.0
jack	0.0	0.0	0.0	0.0	0.0	0.0	0.0	**100.0**	0.0	0.0
run	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	**90.0**	10.0
skip	0.0	0.0	10.0	0.0	0.0	0.0	0.0	0.0	10.0	**80.0**

The elements of the main diagonal are marked in bold

Percentages corresponding to full-length sequences. SVM is used for classifying subsequences of length 30 described by PaHOF (3 levels). The greatest confusion is located in *run* with *skip*. Both actions imply fast displacement

Table 19 Training on KTH and testing on Weizmann: recognition results with GentleBoost

Features	3-classes		6-classes		6-classes-fusion	
	Subseqs	Seqs	Subseqs	Seqs	Subseqs	Seqs
HOF_{20}	96.7	**96.6**	95.1	96.6	95.1	96.6
$aHOF_{20}$	97.9	**100**	87.4	89.7	91.8	96.6

The best mean recognition performances are marked in bold

The three common classes to both datasets (i.e., *handwaving with two hands*, *running* and *walking*) are used in these experiment to evaluate the capacity of generalization of the proposed descriptor. Note that a 100 % of recognition is achieved on Weizmann's sequences using GentleBoost classifiers trained with KTH's samples

used in the previous Weizmann's experiments. In our opinion, the results of the current experiment support our intuition that the low amount of Weizmann's training samples makes difficult for our descriptor to achieve a 100 % of recognition in the experiments of the previous sections (i.e., Sect. 5.4.2).

5.5 Experiment: PaHOF at multiple temporal scales

In the following experiments, our aim is to evaluate different strategies that can be used to combine motion information extracted at diverse temporal scales.

5.5.1 Bag of PaHOF (BOP)

In Sect. 4.2.1 we introduced a model for combining PaHOF at MTS. Such model is inspired in the BOW one. In these experiments, we evaluate such approach.

Diverse variants for cluster assignment have been tried out (e.g., soft assignment) in the following experiments. However, since the *hard* assignment has reported the best results, only those are included in the tables.

Experiments on KTH Table 20 shows the classification results (avg. on 10 repetitions) on KTH dataset. Column *Len* refers to the number of frames used to compute the descriptors; column *Pyr* indicates the spatial pyramid configuration; and, column *K-means* is the number of clusters used to define the BOP.

Table 20 BOP on KTH

Scenario	*Len*	*Pyr*	*K-means*	SVM
s1234	10	8×4	300	**93.3**
s1234	10, 15, 20	2×1 4×2 8×4	100	91.9
s1234	15	2×1 4×2 8×4	600	92.7
s1234	20	8×4	100	90.5
s1234	20	2×1 4×2 8×4	100	92.1

The best mean recognition performance is marked in bold

Classification results with BOP and SVM. Column *Len* refers to different combinations of subsequence lengths, column *Pyr* indicates the combination of spatial grid configurations, and column *K-means* shows the number of clusters used

In general, the combination the PaHOFs of different temporal order in this way does not improve the classification accuracy on this dataset. In our opinion, this indicates that the data information is correlated. However, what seems to be clear is that the number of clusters used to build the BOP is a crucial point.

Table 21 BOP on Weizmann

Len	*Pyr*	*K-means*	*SVM*
5	2×1 4×2 8×4	600	91.6
10	2×1 4×2 8×4	300	**94.1**
5,10	2×1 4×2 8×4	300	91.4
5,10,20	2×1 4×2 8×4	300	91.4
10,15,20	2×1 4×2 8×4	300	93.8
15,20,25	2×1 4×2 8×4	400	93.6

The best mean recognition performance is marked in bold

Classification results with BOP and SVM. Column *Len* refers to different combinations of subsequence lengths, column *Pyr* indicates the combination of spatial grid configurations, and column *K-means* shows the number of clusters used

Experiments on Weizmann Table 21 shows the classification results (*leave-one-out* evaluation) using BOP as feature vector.

We can see in the table that the best result is 94.1 %, which is lower than the 95.8 % (see Table 17) previously achieved with majority voting on individually classified subsequences.

Discussion This BOW-based approach allows to combine descriptors computed at different temporal and spatial scales in a simple way, but at the cost of introducing a new parameter in the model—the number of clusters. Although the classification results achieved with this model are not better than the previous ones, they are not very inferior. As a side effect, with the use of clusters, the final dimensionality of the BOP descriptor is lower than the aHOF and the PaHOF ones, what could be useful in some applications for storage purposes and classification velocity.

5.5.2 Joining multiple time resolution PaHOFs

These experiments refer to the approach defined in Sect. 4.2. In this case, several motion descriptors, computed at different temporal scales, are concatenated to build a single feature vector.

Experiments on Weizmann Classification results on Weizmann DB (*leave-one-out* averaging) are reported in Table 22.

Table 22 PaHOF-MTS on Weizmann: recognition results per sequence with SVM

Len	*Pyr*	*SVM*
5–10–20	8×4	94.1
5–10–20	2×1 4×2 8×4	**95.8**
10–20–30	8×4	94.8

The best mean recognition performance is marked in bold

In this experiment, the feature vectors are built from the concatenation of three PaHOF descriptors of different orders

Table 23 PaHOF-MTS on KTH: recognition results per sequence with SVM

Scenario	*Len*	*Pyr*	*SVM*
s1234	5–10–20	2×1 4×2 8×4	**94.7**

The achieved recognition performance is marked in bold

In this experiment, the feature vectors are built from the concatenation of three PaHOF descriptors of order 5, 10 and 20

Column *Len* indicates the length of the subsequences (e.g., 5–10–20) concatenated to build the feature vectors, and column *Pyr* refers to the spatial pyramid setup.

This model achieves 95.8 % of correct recognition (at sequence level), what is equal to the best result achieved by just using PaHOF at a single temporal scale (see Table 17). This indicates that the proper selection of the temporal scale is enough to describe the actions in this dataset.

Experiments on KTH Classification results on KTH DB (averaged on 10 runs) are reported in Table 23.

Column *Len* indicates the length of the subsequences (e.g., 5–10–20) concatenated to build the feature vectors, and column *Pyr* refers to the spatial pyramid setup.

Note that this result (94.7 %) does not improve the best one achieved by PaHOF at a single temporal scale (96 %). This may indicate that this simple information fusion approach is not suitable for these feature vectors.

5.6 Experiment: action recognition in a multiple view scenario

This experiment is carried out on IXMAS dataset. In some papers as [33], actions *throw-over-head* and *throw-from-bottom-up* are considered as a single action class. We follow the previous criterion as well. As previously done on Weizmann dataset, the evaluation on this one is performed by following a leave-one-out strategy on the actors. Therefore, the reported results are the outcome of 12 repetitions.

Since the resolution of these videos (i.e., 390×291) is much better than the Weizmann and KTH's ones, we have experimented with cropped BB areas of sizes 48×48 and 80×80 pixels. We use the per-frame person segmentations provided by the authors of the dataset to define the BB needed by our descriptor.

The results are shown in Table 24. As indicated by column *Camera*, we have trained and tested on the same camera. The rows that begin with *avg* in that column, reports averaged results on the given cameras. Column *Config* indicates the configuration of the PaHOF and the subsequences length.

We have trained a one-vs-all classifier per class action, in particular, a SVM with χ^2-kernel [49].

Table 24 IXMAS dataset: full sequences classification results

Camera	*Config*	*SVM-χ^2*
cam0	*s*48 × 48_*len*20_8 × 4	71.3
cam0	*s*48 × 48_*len*30_8 × 4	73.7
cam0	*s*80 × 80_*len*30_8 × 4	73.7
cam1	*s*48 × 48_*len*20_8 × 4	67.6
cam1	*s*48 × 48_*len*30_8 × 4	70.9
cam1	*s*80 × 80_*len*30_8 × 4	68.0
cam2	*s*48 × 48_*len*20_8 × 4	76.7
cam2	*s*48 × 48_*len*30_8 × 4	73.6
cam2	*s*80 × 80_*len*30_8 × 4	74.2
cam3	*s*48 × 48_*len*20_8 × 4	71.6
cam3	*s*48 × 48_*len*30_8 × 4	71.8
cam3	*s*80 × 80_*len*30_8 × 4	76.7
cam4	*s*48 × 48_*len*20_8 × 4	65.5
cam4	*s*48 × 48_*len*30_8 × 4	65.7
cam4	*s*80 × 80_*len*30_8 × 4	62.5
avg:0123	*s*48 × 48_*len*30_8 × 4	72.5
avg:all	*s*48 × 48_*len*30_8 × 4	71.1
avg:0123	*s*80 × 80_*len*30_8 × 4	**73.2**
avg:all	*s*80 × 80_*len*30_8 × 4	71.0

The best mean recognition performance for the combined cameras is marked in bold

Trained and tested on the same camera view. Column *Config* indicates the enlarged BB size, subsequence length and spatial grid used for aHOF. Bottom rows contain averaged results on the indicated cameras

To understand better the achieved results, we show in Table 25 the confusion matrix associated to experiment *s*48 × 48_*len*20_8 × 4 on camera #2 as an example. On one hand, the worst recognition rate with camera #2 is located in *point* action, which is mostly confused with actions like *cross-arms* or *check-watch*. Such actions imply subtle movements of the arms. On the other hand, the best discriminated actions are *walk* and *sit-down*.

State-of-the-art on IXMAS Nebel et al. [33] present recent results on this dataset following the same experimental setup as we do. Three of the evaluated methods in that paper rely on BBs, as our proposed method does. Those methods offer an average recognition performance (i.e., discarding camera #4) of 54, 63.9 and 85 % respectively. In our case, our best result is 73.2 %, which is the second best comparing with those ones.

5.7 About processing time

To have an idea of the computational performance of the proposed descriptor, on a state-of-the-art computer[2] we have measured the following times per video frame: (1) resize of the image enlarged area (around 120 × 120 pixels on KTH) to 40 × 40 pixels: 3.1 ms; (2) Farneback's OF computation: 6.5 ms; c) HOF 8 × 4: 15.9 ms. In addition, the final step (accumulation plus normalization) to compute $aHOF_{20}$ requires 2 ms, what makes a total of $21 \times 3.1 + 20 \times (6.5 + 15.9) + 2 = 515.1$ ms for the first $aHOF_{20}$. The subsequent ones can reuse information already computed for previous ones. Although the computation of the BB is not included in the total time, this timing can be considered real-time. Moreover, since our implementation is neither carefully optimized nor parallelized, and written for © Matlab, it should be possible to reduce drastically this computational time if needed.

[2] Intel Core i3 M350@2.27GHz, 4 GB RAM, Matlab 2009b on a single CPU core.

5.8 Summary of the experiments

A very complete setup of experiments using the proposed descriptors has been given. We have shown results on direct pooling of features at a single scale (aHOF), where spatial pyramids have been used to introduce spatial scale (PaHOF). Moreover, PaHOF at MTS have been tested.

Apart from the very good score obtained on the KTH database (PaHOF on mixed scenarios 96 %, state-of-the-art at sequence level) and the other databases (i.e., Weizmann and IXMAS), new important questions have come up. In our opinion, one of the most interesting ones is the lack of improvement when using different combinations of subsequence lengths. This indicates a smoothing effect on the discriminative information, due to the existence of strong correlations between features from different levels.

In addition, the results on IXMAS indicate that the proposed descriptor behaves well with different camera viewpoints, as long as, such viewpoints have been *shown* during training. And surprisingly, the results achieved with camera #4 (i.e., bird's eye shot) are not so low taking into account the small image region that person actually occupies in the scene in that view.

6 Discussion and conclusions

In this paper, we have presented and evaluated a new model to compute multiscale motion descriptors from video sequences. In particular, this paper presents motion descriptors based on histograms of Optical Flow: PaHOF and its variants.

These descriptors have been extensively evaluated, with state-of-the-art classifiers (SVM and GentleBoost), on three public databases: KTH, Weizmann and IXMAS. These datasets are widely used in the evaluation of systems designed for HAR.

Table 25 Confusion matrix on IXMAS: cam2

	cw	*ca*	*sh*	*sd*	*gu*	*ta*	*wa*	*wv*	*pu*	*ki*	*po*	*pi*	*th*
cw	**74.3**	11.4	2.9	0.0	0.0	0.0	0.0	2.9	0.0	0.0	8.6	0.0	0.0
ca	8.6	**77.1**	5.7	0.0	0.0	0.0	0.0	0.0	2.9	0.0	5.7	0.0	0.0
sh	5.9	2.9	**67.6**	0.0	0.0	2.9	0.0	11.8	0.0	0.0	8.8	0.0	0.0
sd	0.0	0.0	0.0	**97.1**	2.9	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
gu	0.0	0.0	0.0	12.1	**75.8**	0.0	0.0	0.0	0.0	6.1	0.0	6.1	0.0
ta	0.0	0.0	0.0	0.0	0.0	**94.1**	5.9	0.0	0.0	0.0	0.0	0.0	0.0
wa	0.0	0.0	0.0	0.0	0.0	0.0	**100.0**	0.0	0.0	0.0	0.0	0.0	0.0
wv	8.6	2.9	11.4	2.9	2.9	0.0	0.0	**57.1**	0.0	0.0	5.7	2.9	5.7
pu	2.9	2.9	0.0	2.9	2.9	0.0	0.0	0.0	**60.0**	11.4	11.4	2.9	2.9
ki	0.0	0.0	0.0	0.0	0.0	0.0	17.6	0.0	0.0	**79.4**	2.9	0.0	0.0
po	8.6	14.3	2.9	0.0	0.0	8.6	2.9	2.9	0.0	5.7	**48.6**	5.7	0.0
pi	0.0	0.0	0.0	8.6	5.7	0.0	5.7	0.0	0.0	0.0	0.0	**80.0**	0.0
th	0.0	0.0	0.0	0.0	0.0	0.0	2.9	0.0	0.0	2.9	2.9	2.9	**88.6**

The elements of the main diagonal are marked in bold

aHOF configuration: *s*48 × 48_*len*20_8 × 4. Actions: *cw* checkwatch, *ca* crossarms, *sh* scratchhead, *sd* sitdown, *gu* getup, *ta* turnaround, *wa* walk, *wv* wave, *pu* punch, *ki* kick, *po* point, *pi* pickup, *th* throw

Firstly, aHOF is evaluated as a single-scale motion descriptor. After studying and tuning its parameters (i.e., OF quantization, spatial grid configuration and subsequence length), two kinds of NLTs of the data are compared: (1) normalization of the OF magnitude bins per orientation at each spatial cell, and, (2) rectification of the data using a sigmoidal function. The results of the experiments indicate that the first transformation offers the best results in terms of classification. Although it is not included in the experiments, as a first approach, we also evaluated the performance of the global normalization of the descriptor regardless of the location, orientation and magnitude bins, but the achieved results were inferior. The main conclusion of this block of experiments is that the sole use of Optical Flow as basis of a feature vector is enough in terms of description of the evaluated human actions.

Note that this descriptor achieves a 94.8 % of recognition on KTH-s1234, comparable to the top results on that dataset (e.g., 94.5 % [22], see Table 16), without the need of using additional kind of features (e.g., spatial gradients) as other works described in Sect. 2 do.

Afterwards, we have studied both the importance of the person BB in the aHOF descriptor as well as the effect of partial occlusions in the recognition performance. The results show that aHOF descriptor absorbs well some amounts of noise introduced by the person detection stage. In addition, although the descriptor is affected by some types of person occlusions, it behaves reasonably well in those situations.

Our first attempt of enriching our motion descriptor was to combine information of different spatial scales. In this sense, we proposed and compared a PaHOF with a baseline method inspired in the Pyramid-Match representation and adapted to our problem. The experimental results give support to conclude that the addition of new scales of information are useful in terms of description of the action, improving on the single-scale results. Moreover, PaHOF features performed better than the baseline approach (i.e., aPM). At this point, PaHOF achieves a 96 % of recognition on KTH-s1234, comparable to the state-of-the-art [29] in that dataset.

In addition, we showed experimentally that our descriptor has good generalization properties since we could effectively recognize Weizmann's video sequences of actions (common to both datasets) trained on KTH DB (see Sect. 5.4.3).

Our next step was to study what is the best way to include in our descriptor temporal information at different scales. In particular, we proposed and evaluated two alternatives: (1) a combination of PaHOF of different temporal scales through a BOW approach (i.e., BOP), and (2) the direct concatenation of PaHOF vectors computed at different temporal scales. Although the second alternative offered better results than the first one, neither of these approaches helped to improve the results achieved up to that moment. That behavior might suggest that such information is highly correlated. In our opinion, this result along with the previous ones indicates that the combination of information at multiple spatial scales is more relevant

than the one provided by the temporal scales in these classes of human actions.

The design of these descriptors could allow the development of an online action recognition system. That is, once a new frame comes into the system, the PaHOF descriptor (unnormalized) could be updated by simply adding the contribution of the new frame, and by subtracting the contribution of the oldest frame. Therefore, the named classification of subsequences could be also seen as the classification of single frames, but taking into account the history of the N (subsequence length) previously seen frames.

In Sect. 5.7 we include some brief information about processing time, showing that could be easily used in real-time problems.

The experiments with temporal multiscale motion descriptors do not report improvements in comparison with the single-scale ones. We believe that the main reason for this fact is the existing high correlation between features from different scales. In this way, the introduction of a new decorrelating feature stage in our model will be the goal of future research.

The results of the final experiments, carried out on a multiple view dataset (i.e., IXMAS) show that the proposed method is able to deal with actions viewed from different cameras as long as those views have been included in the training data. Fair results are even achieved on the hard bird's-eye camera (i.e., *cam4*) suggesting the good adaptability of the descriptor to free viewpoints.

Acknowledgments This work has been granted by the Project CSD2007-00018 (MIPRCV) from the Spanish Minister of Science and Technology.

References

1. Balcells M, DeMenthon D, Doermann D (2004) An appearance-based approach for consistent labeling of humans and objects in video. Pattern Anal Appl 7:373–385
2. Ballan L, Bertini M, Del Bimbo A, Seidenari L, Serra G (2009) Recognizing human actions by fusing spatio-temporal appearance and motion descriptors. In: Proceedings of the IEEE international conference on image processing, pp 3569–3572
3. Blank M, Gorelick L, Shechtman E, Irani M, Basri R (2005) Actions as space–time shapes. In: International conference on computer vision, vol 2, pp 1395–1402
4. Blei D, Ng A, Jordan M (2003) Latent Dirichlet allocation. J Mach Learn Res 3:993–1022
5. Bosch A, Zisserman A, Muñoz X (2007) Representing shape with a spatial pyramid kernel. In: Proceedings of CIVR
6. Boukir S, CheneviFre F (2004) Compression and recognition of dance gestures using a deformable model. Pattern Anal Appl 7:308–316
7. Breitenstein MD, Reichlin F, Leibe B, Koller-Meier E, Gool LV (2009) Robust tracking-by-detection using a detector confidence particle filter. In: IEEE international conference on computer vision (ICCV'09)
8. Chang CC, Lin CJ (2001) LIBSVM: a library for support vector machines. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm/. Accessed 1 Aug 2012
9. Dalal N, Triggs B, Schmid C (2006) Human detection using oriented histograms of flow and appearance. In: European conference on computer vision
10. Dollar P, Rabaud V, Cottrell G, Belongie S (2005) Behavior recognition via sparse spatio-temporal features. In: 2nd IEEE workshop VS-PETS, pp 65–72
11. Duchenne O, Laptev I, Sivic J, Bach F, Ponce J (2009) Automatic annotation of human actions in video. In: International conference on computer vision
12. Efros A, Berg A, Mori G, Malik J (2003) Recognizing action at a distance. In: International conference on computer vision, vol 2, pp 726–733
13. Farnebäck G (2003) Two-frame motion estimation based on polynomial expansion. In: Proceedings of the 13th Scandinavian conference on image analysis, LNCS, vol 2749, pp 363–370
14. Fathi A, Mori G (2008) Action recognition by learning mid-level motion features. In: CVPR
15. Felzenszwalb P, McAllester D, Ramanan D (2008) A discriminatively trained, multiscale, deformable part model. In: IEEE conference on computer vision and pattern recognition
16. Ferrari V, Marin-Jimenez M, Zisserman A (2008) Progressive search space reduction for human pose estimation. In: IEEE conference on computer vision and pattern recognition
17. Friedman J, Hastie T, Tibshirani R (1998) Additive logistic regression: a statistical view of boosting: technical report. Department of Statistics, Stanford University, California
18. Grauman K, Darrell T (2005) The pyramid match kernel: discriminative classification with sets of image features. In: Proceedings of the IEEE ICCV
19. Jarrett K, Kavukcuoglu K, Ranzato M, LeCun Y (2009) What is the best multi-stage architecture for object recognition? In: International conference on computer vision
20. Jhuang H, Serre T, Wolf L, Poggio T (2007) A biologically inspired system for action recognition. In: Proceedings of ICCV'07, pp 1–8
21. Ke Y, Sukthankar R, Hebert M (2005) Efficient visual event detection using volumetric features. In: Proceedings of IEEE international conference on computer vision (ICCV '05), pp 166–173
22. Kovashka A, Grauman K (2010) Learning a hierarchy of discriminative space–time neighborhood features for human action recognition. In: IEEE conference on computer vision and pattern recognition
23. Laptev I (2005) On space–time interest points. Int J Comput Vis 64(2/3):107–123
24. Laptev I, Marszalek M, Schmid C, Rozenfeld B (2008a) Learning realistic human actions from movies. In: Proceedings on CVPR
25. Laptev I, Marszalek M, Schmid C, Rozenfeld B (2008b) Learning realistic human actions from movies. In: International conference on computer vision and pattern recognition
26. Lazebnik S, Schmid C, Ponce J (2006) Beyond bags of features: spatial pyramid matching for recognizing natural scene categories. CVPR 2:2169–2178
27. Lin Z, Jiang Z, Davis LS (2009) Recognizing actions by shape-motion prototype trees. In: International conference on computer vision
28. Lucena M, de la Blanca NP, Fuertes J (2012) Human action recognition based on aggregated local motion estimates. Mach Vis Appl 23:135–150
29. Lui YM, Beveridge J, Kirby M (2010) Action classification on product manifolds. In: IEEE conference on computer vision and pattern recognition, pp 833–839

30. Marín-Jiménez M, de la Blanca NP, Mendoza M, Lucena M, Fuertes J (2009) Learning action descriptors for recognition. In: IEEE (ed) WIAMIS 2009, London, UK. IEEE Computer Society, New York, pp 5–8
31. Mitchell TM (1997) Machine learning. McGraw-Hill, New York
32. Moeslund TB, Hilton A, Kruger V (2006) A survey of advances in vision-based human motion capture and analysis. Comput Vis Image Underst 104:90–126
33. Nebel JC, Lewandowski M, Thévenon J, Martínez-Contreras F, Velastin S (2011) Are current monocular computer vision systems for human action recognition suitable for visual surveillance applications? ISVC 2:290–299
34. Norouzi M, Ranjbar M, Mori G (2009) Stacks of convolutional restricted Boltzmann machines for shift-invariant feature learning. In: IEEE conference on computer vision and pattern recognition
35. Otsu N (1979) A threshold selection method from gray level histograms. IEEE Trans Syst Man Cybern 9:62–66
36. Pantic M, Pentland A, Nijholt A, Huang T (2007) Human computing and machine understanding of human behavior: a survey. Artif Intell Human Comput 4451:47–71
37. Pinto N, Cox DD, Dicarlo JJ (2008) Why is real-world visual object recognition hard? PLoS Comput Biol 4(1):e27
38. Ramanan D, Forsyth D, Zisserman A (2007) Tracking people by learning their appearance. IEEE Trans Pattern Anal Mach Intell 29(1):65–81
39. Reddy KK, Liu J, Shah M (2009) Incremental action recognition using feature-tree. In: International conference on computer vision
40. Schindler K, van Gool L (2008) Action snippets: how many frames does human action recognition require? In: IEEE conference on computer vision and pattern recognition
41. Schindler K, van Gool L (2008) Combining densely sampled form and motion for human action recognition. In: DAGM08, pp 122–131
42. Schüldt C, Laptev I, Caputo B: Recognizing human actions: a local SVM approach. In: International conference on pattern recognition, Cambridge, UK, vol 3, pp 32–36
43. Seo HJ, Milanfar P (2009) Detection of human actions from a single example. In: International conference on computer vision
44. Serre T, Wolf L, Bileschi S, Riesenhuber M, Poggio T (2007) Robust object recognition with cortex-like mechanisms. IEEE Trans Pattern Anal Mach Intell 29(3):411–426
45. Sminchisescu C, Kanaujia A, Li Z, Metaxas D (2005) Conditional models for contextual human motion recognition. In: Proceedings of ICCV'05, IEEE
46. Song Y, Goncalves L, Perona P (2003) Unsupervised learning of human motion. IEEE Trans Patt Anal and Mach Intell 25(7):1–14
47. Sun X, Chen MY, Hauptmann A (2009) Action recognition via local descriptors and holistic features. International workshop on human communicative behaviour analysis-CVPR
48. Turaga P, Chellappa R, Subrahmanian VS, Udrea O (2008) Machine recognition of human activities: a survey. Circuits Syst Video Technol IEEE Trans 18(11):1473–1488
49. Vedaldi A, Zisserman A (2012) Efficient additive kernels via explicit feature maps. IEEE PAMI 34(3):480–492
50. Wang H, Ullah MM, KlSser A, Laptev I, Schmid C (2009) Evaluation of local spatio-temporal features for action cognition. In: Proeedings of BMVC
51. Wang RR, Huang T (2004) A framework of joint object tracking and event detection. Pattern Anal Appl 7:343–355
52. Weinland D, Ronfard R, Boyer E (2006) Free viewpoint action recognition using motion history volumes. In: CVIU
53. Yu T, Kim T, Cipolla R (2010) Real-time action recognition by spatiotemporal semantic and structural forests. In: Proceedings of BMVC, pp 1–12
54. Zelnik-Manor L, Irani Michal (2006) Statistical analysis of dynamic actions. IEEE Trans Pattern Anal Mach Intell 28(9): 1530–1535
55. Zhang Z, Hu Y, Chan S, Chia L (2008) Motion context: a new representation for human action recognition. In: ECCV 2008, pp 817–829