Front.Comput.Sci.
DOI

RESEARCH ARTICLE

Frontiers of Computer Science

Distribution of Action Movements (DAM): A Descriptor for Human Action Recognition

Franco RONCHETTI (✉), Facundo QUIROGA, Laura LANZARINI, Cesar ESTREBOU

Instituto de Investigacion en Informatica III-LIDI, Facultad de Informatica, Universidad Nacional de La Plata, Argentina

Abstract

Human action recognition from skeletal data is an important and active area of research in which the state of the art has not yet achieved near-perfect accuracy on many well-known datasets. In this paper, we introduce the Distribution of Action Movements Descriptor, a novel action descriptor based on the distribution of the directions of the motions of the joints between frames, over the set of all possible motions in the dataset. The descriptor is computed as a normalized histogram over a set of representative directions of the joints, which are in turn obtained via clustering. While the descriptor is global in the sense that it represents the overall distribution of movement directions of an action, it is able to partially retain its temporal structure by applying a windowing scheme.

The descriptor, together with a standard classifier, outperforms several state-of-the-art techniques on many well-known datasets.

Keywords human action recognition, descriptor, prob som, msrc12, action3d

1 Introduction

1.1 Motivation

Human Action Recognition remains an unsolved problem in the fields of computer vision and pattern recognition. It can be employed to monitor patients, detect suspicious activities, classify work situations as dangerous or safe, recognize player actions in games, or to provide touch-less human computer interactions. Traditionally, vision-based human action recognition techniques used low-level image features as input to action models. In recent years, researchers have been increasingly using Mocap systems or sensors such as the Microsoft Kinect to obtain skeletal information of users performing actions. This trend has been motivated by the low cost and wide availability of this types of capture devices, and growing evidence supporting the hypothesis that skeletal data provides a superior representation for human action recognition [1].

1.2 Overview

This paper presents the Distribution Action Movements Descriptor (DAM), a novel descriptor for human actions based on the distribution of the movements of each joint in a skeletal representation of the user.

We extend and adapt ProbSom's approach to person identification by voice, described in [2], to the case of human action recognition using skeletal information. The following is an outline of the computation of the resulting descriptor, explained in full in section 2.

We have a set of action recordings performed by persons, which are composed of a variable number of frames, where for each frame we know the 3D position of a given number of joints of the body. We first transform each action, replacing the absolute 3D position of each joint by the direction of its motion between frames. Thus, we obtain $n-1$ *direction*

Received month dd, yyyy; accepted month dd, yyyy

E-mail: fronchetti at lidi.info.unlp.edu.ar

frames from the original *n absolute position frames* of the action. Note that the direction frames are translation invariant. Previously, each joint of each action was smoothed and resampled individually; the first to remove spurious directions, the second to ensure each action has the same number of frames. The resulting directions of motion are 3D vectors, one for each frame and joint. We apply a windowing scheme to preserve temporal information, generating windowed direction frames (WDFs) from the direction frames. Then, we perform a clustering of the set of all the WDFs of the set **A** of all actions to obtain a reduced set C_w of direction frames that efficiently represents the distribution of directions of **A**.

The descriptor of a new action **A** is then computed as the normalized histogram of its direction frames over the set C_w, where each direction frame in C_w is a bin, and each direction frame of the action is assigned to the bins according to the same distance metric employed in the generation of C_w.

This novel descriptor achieves excellent results on two well known datasets, MSRC12 and MSR Action3D, described in section 3. To the best of our knowledge, it surpasses the classification accuracy of almost all other state-of-the-art techniques on standard datasets to date (see also section 3).

1.3 Related work

As described in [3], there are three main challenges in a Human Action Recognition System: data capture, feature descriptors and action modeling. Our method does not deal with the first challenge, assuming the 3D skeletal data is somehow available, captured for example by employing a Mocap device or the Kinect as is the case with the datasets used in our experiments. The second and third challenges are very much intertwined, and are, in general, solved jointly.

Feature descriptors can be global, in the sense that they represent the action as a whole, or they can be local, in the sense that a separate descriptor is computed for each frame or interest point of the action. In the latter case, additional steps are needed to create an effective descriptor, for example, by modeling the temporal dependence with a Hidden Markov Model, or aggregating each local descriptor to generate a global one.

In recent years, there has been much interest in finding appropriate descriptors for action recognition.

In [4], Barnachon et al. apply a clustering algorithm to all the poses of every action sample, and then define a pose-based integral histogram for actions, computed over the resulting clusters, as descriptor. Actions can thus be encoded with this representation, and then compared using the Bhattacharyya distance. Since the descriptor clusters the poses directly, it is not translation-invariant.

Hussein et al. [3] utilize a covariance descriptor for each action, which is computed as the covariance matrix of the set of poses that comprise an action and contains as information the variance of movement of each joint (diagonal elements) and the relative movement between joints (non-diagonal elements). The descriptor has the disadvantage of being high dimensional. To build a model, they compute the descriptor for all training actions and then train an off-the-shelf support-vector machine (SVM) classifier with a linear kernel.

The approach in Gowayyed et al. [5] called Histogram of Oriented Directions, is similar to ours, in that it employs a histogram of directions, but their approach calculates histograms independently for each joint, and uses, not the 3D positions of the joint themselves, but the three 2D projections to planes *xy*, *xz* and *yz* of their position in the virtual coordinate space. The set of bins is defined arbitrarily as a regularly spaced grid. By calculating histograms on each joint separately, their descriptors lacks synchronization information between the joints in multi-joint actions. As in the method of Hussein et al., the classification is performed by combining the descriptor with a SVM.

Since this three descriptors all lack temporal information, a sub-action decomposition or temporal pyramid scheme was applied to the descriptors to generate more complex ones that retain some of the sequence information of the original actions.

The sequence of most informative joints (SMIJ) representation introduced by Ofli et al. [6] divides each action into a number of segments, and for each segment computes a ranking of the most informative joints according to an entropy-based measure, and chooses the *k* most relevant ones to describe the segment. These are concatenated into a descriptor, and standard SVM and k-nearest neighbours classifiers were trained using it as input. The representation completely disregards any trajectory information contained in the action, since it only measures the amount of activity in the joints.

In [7], Cho et al. train a hybrid Multilayer Perceptron/Deep auto-encoder on a set of features composed of the 3D direction vector between joints for each frame, rotated and scaled to canonical angles and sizes to obtain invariances.

Li et al. [8] do not employ a full pose estimation technique, but instead sample 3D points belonging to the person's body from depth maps. They quantize the set of all poses from the training samples via clustering, and build an action graph using the cluster centers or *salient poses* as nodes, inferring the transition probabilities from the training samples as well. Afterwards, they apply a decoding scheme based on Bi-gram with Maximum Likelihood Decoding (BMLD) to classify new samples according to the graph information.

None of these techniques achieves perfect accuracy on cross-subject cross-validation experiments in the datasets used to test them.

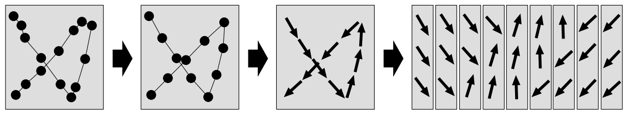


Fig. 1 Preprocessing steps for a single joint of the action. The original positions (a) are smoothed and resampled obtaining a more uniform sequence (b). Then, direction vectors are computed (c) and concatenated into overlapping WDFs with $W = 3$ (d).

2 Descriptor

In this section we develop the formulation for the descriptor and classification model. To ease the explanation we begin by introducing a notation for actions and their constituent parts.

2.1 Notation

Let $\mathcal{A} = \{\mathbf{A}^1, \ldots, \mathbf{A}^N\}$ be a set of N actions, F_i the number of frames of action $\mathbf{A}^i$ and J the number of joints (the same for every action).

Then we have:

1. $\mathbf{A}^i$: *ith* action, with F_i frames, where each frame contains the position of the J joints.
2. $\mathbf{A}^i_j$: the joint positions for all F_i frames of action i.
3. $\mathbf{A}^i_f$: the position of the J joints of action i in frame f, ie, a *position frame*.
4. $\mathbf{A}^i_{f,j}$: the position of joint j of action i in frame f.

2.2 Preprocessing

The joint positions of each action $\mathbf{A}^i_{f,j}$ are smoothed with a gaussian-distributed weighted moving average (WMA). While simple, this step is essential to the effectiveness of the descriptor, since otherwise sensing errors or short spurious movements distort long term trends in the direction of movement of the joints on which we rely for classification.

Afterwards, the same joint positions $\mathbf{A}^i_{f,j}$ are re-sampled to a constant number of frames F, which is the same for all actions and joints, so that at the end of this process $F_i = F$ for all i. The re-sampling is done at uniformly spaced points along the arc-length of the path traced by the joint through all the frames of the action. The position of the joints in the new frames are estimated by cubic splines interpolation.

Since each joint is re-sampled individually and with an arc-length parametrization, this step could distort the temporal synchronization between joints, but visual inspection in the datasets used for the experiments revealed no noticeable warping.

To obtain a translation invariant descriptor, we calculate for each joint and frame a direction vector $\mathbf{D}^i_{f,j} \in \mathbb{R}^3, f = 1, \ldots, F-1$, that represents the direction vector between positions $\mathbf{A}^i_{f+1,j}$ and $\mathbf{A}^i_{f,j}$, such that:

$$\mathbf{D}^i_{f,j} = \mathbf{A}^i_{f+1,j} - \mathbf{A}^i_{f,j}, \quad f = 1, \ldots, F-1 \tag{1}$$

By analogy, we call each $\mathbf{D}^i_{f,j}$ a *direction frame*, since it contains the direction of motion between the position frames $\mathbf{A}^i_{f+1,j}$ and $\mathbf{A}^i_{f,j}$.

Since afterwards we intend to apply a clustering of all direction frames, losing all sequence information, we apply a windowing scheme to preserve temporal information.

From the $F-1$ direction frames, we generate $F^W = (F-1)-(W-1) = F-W$ *windowed direction frames* (WDFs) by concatenating W direction frames together in an overlapping fashion, such that:

$$\mathbf{W}^i_f = \left[\mathbf{D}^i_f, \mathbf{D}^i_{f+1}, \ldots, \mathbf{D}^i_{f+(w-1)}\right], \quad f = 1, \ldots, F^W \tag{2}$$

where W, the window size, is a parameter; the greater W is, more temporal information is retained. In an extreme case, if $W = F-1$, then $F^W = 1$, and there is just one WDF which is exactly $\mathbf{D}^i$. On the other hand, if $W = 1$, then $F^W = F-1$ and $\mathbf{W}^i_f = \mathbf{D}^i_f$, ie, the WDFs are simply the direction frames.

Note that the notation for WDFs and direction frames is analogous to that of actions described earlier.

2.3 Descriptor

To calculate the descriptor, we first obtain the K cluster centers $C_w = \{\mathbf{C}^1, \mathbf{C}^2, \ldots, \mathbf{C}^K\}$ of a clustering of the set of all WDFs of all actions $\mathcal{W}^{all} = \{\mathbf{W}^i_f | i = 1, \ldots, N, f = 1, \ldots, F-W\}$ contained in the training set. The clustering C_w is built with a Self Organizing Map (SOM) algorithm [9].

We use the clustering to quantize the space of all possible WDFs, so as to compute the distribution of movements over that set. For any WDF $\mathbf{W}_f$, we can assign it to a cluster with center $\mathbf{C}^l$, according to the same distance metric employed in the generation of the cluster. Denoting the center corresponding to $\mathbf{W}_f$ as $BMU(\mathbf{W}_f)$, we calculate a normalized histogram $\mathbf{H}$ with K bins given by the cluster centers $\mathbf{C}^l$, so that:

$$H_l = \frac{|\{\mathbf{W}_f \mid BMU(\mathbf{W}_f) = \mathbf{C}^l\}|}{F^W}, \quad l = 1, \ldots, K \qquad (3)$$

The resulting descriptor, $\mathbf{H}$, measures the distribution of WDFs of the original action $\mathbf{A}$. The histogram elements H_l can be interpreted as the frequentist estimation of the probability that an action $\mathbf{A}$, transformed into $\mathbf{H}$, contains WDF $\mathbf{C}^l$ as a composing element, ie, $H_l = P(\mathbf{C}^l|\mathbf{A})$.

2.4 Classification Model

We have employed the classification model described in [2]. Using the same training actions $\mathcal{A}$ we employed to build the clustering, we will build a model of $P(c|\mathbf{C}^l)$, ie, the probability that an action $\mathbf{A}$, with the WDF $\mathbf{C}^l$ as composing element, is of class c.

With that, we can compute the probability of an action $\mathbf{A}$ belonging to a certain class c as the average of $P(c|\mathbf{C}^l)$ over the K cluster centers $\mathbf{C}^l$:

$$P(c|\mathbf{A}) = \sum_{l=1}^{K} P(c|\mathbf{C}^l) * P(\mathbf{C}^l|\mathbf{A}) = \sum_{l=1}^{K} P(c|\mathbf{C}^l) * H_l \qquad (4)$$

Having built the cluster, we are thus able to compute descriptors for the set of training actions $\mathcal{A} = \{\mathbf{A}^1, \ldots, \mathbf{A}^N\}$, for which we know the corresponding classes c_i. We can therefore build a model of $P(c|\mathbf{C}^l)$ as:

$$P(c|\mathbf{C}^l) = \frac{|assigned(\mathbf{C}^l) \cap ofClass(c)|}{|assigned(\mathbf{C}^l)|} \qquad (5)$$

Where $ofClass(c)$ is the set of WDFs of class c, and $assigned(\mathbf{C}^l)$ is the set of WDFs assigned to center $\mathbf{C}^l$ (over all the training actions, in both cases):

$$\begin{aligned} assigned(\mathbf{C}^l) = \{\mathbf{W}^i_f | BMU(\mathbf{W}^i_f) = \mathbf{C}^l, \\ i = 1, \ldots, N,\ f = 1, \ldots, F^W\} \end{aligned}$$

$$ofClass(c) = \{\mathbf{W}^i_f | c_i = c,\ f = 1, \ldots, F^W\}$$

To assign a class label c^* to an action $\mathbf{A}$ we therefore compute the descriptor $\mathbf{H}$ for the action, and then calculate:

$$c^* = \max_c P(c|\mathbf{A}), \quad c = 1, \ldots, C \qquad (6)$$

where C is the number of classes, and $P(c|\mathbf{A})$ is defined as in eq. 4.

3 Experiments

We evaluated the descriptor on two well known and publicly available datasets, MSRC12 [10] and MSR Action3D [8], both acquired using a Kinect Sensor, which captures 20 joints of the human body (Figure 2). [1)]

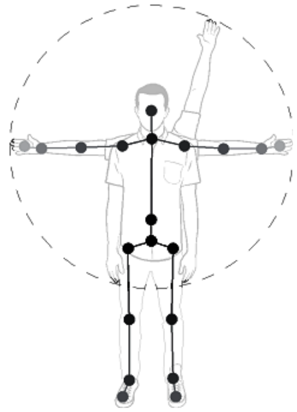


Fig. 2 Joints detected by Kinect

For both datasets, we tested various values for W, the window size, to evaluate the importance of including temporal information in the descriptor and K, the number of clusters, to establish if the classifier was robust in a wide variety of settings of this parameter. The amount of frames the actions are resampled to, F, was set to 25 for all experiments with Action3D. For the MSRC12 dataset, since the actions are easier to separate by means of joint activation patterns instead of trajectory information, the actions were resampled to $F = 16$ frames without losing accuracy. After computing the WDFs, they were rescaled to euclidean norm 1 so as to provide scale invariance to the descriptor.

3.1 MSR Action3D Dataset

We used the MSR Action3D dataset [8] which consists of action sequences of 20 different types of actions performed by 10 subjects. Each sequence consists of an action, segmented so that the start of the sequence roughly corresponds to that of the action, and the same for the end. For each sequence the dataset contains the position of the 20 joints for all the frames captured, which are time-labeled.

There are 567 sequences in total, of which we used 526. The original authors had already left out 10 sequences out

[1)] A detailed description of how we prepared the datsets and which action instances were corrected/left out, scripts to read the files, and code to run the experiments can be found in https://sites.google.com/site/damdescriptor.

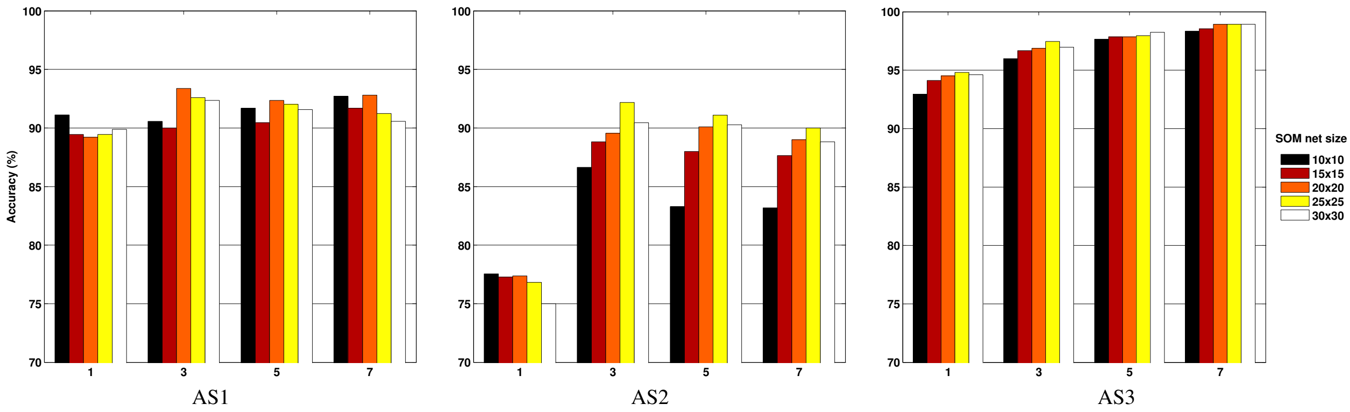


Fig. 3 Results obtained with our descriptor for MSR-Action3D, with different values for W, the window size, and K, the amount of clusters. Each result is the average of 30 independent runs.

Table 1 Comparison of results on the MSR-Action 3D dataset.

Method	Accuracy (%)
Ellis (logistic regression) [11]	65.70
Li (Action Graph) [8]	74.70
Wang (Random Occupancy Pattern) [12]	86.50
Wang (Actionlets Ensemble) [13]	88.20
Hussein (Cov3DJ) [3]	90.53
Proposed descriptor	**94.07**

of the 567; we excluded 31 more that we found heavily corrupted and corrected 48 in which some joints positions were exchanged [2] [8].

3.1.1 Experimental setup

We employed the typical experimental setup on this dataset [8], which divides the action classes into three action sets (AS1, AS2 and AS3), each composed of the actions of 8 classes, with some overlap between action sets.

We performed experiments on each set separately, following a cross subject protocol with a 50/50 subject split [8]. We performed 30 cross validation runs for each parameter configuration, and computed the average classification rate for each over the 30 runs.

Our best results were obtained with a window size $W = 3$ and $K = 625$ clusters obtained after training a 25×25 SOM network.

3.1.2 Results

Table 1 compares the results obtained with the proposed method against other state-of-the-art models. Our descriptor achieves an accuracy of 94.07%, reducing the error in 1/3 with respect to the best method.

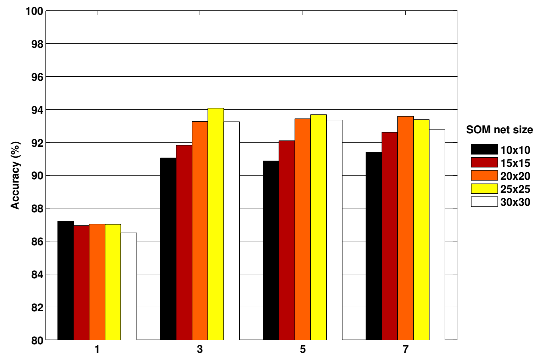


Fig. 4 Average accuracy (*y axis*) on MSR-Action3D for different window sizes W (*x axis*) and number of clusters, K (colors). The average is computed over the accuracies of the 3 subsets (AS1, AS2, AS3).

Figure 3 displays the different accuracies obtained by varying the parameters of the method for each subset of the dataset. It is evident that the effect of the windowing scheme is beneficial in almost all cases. On the other hand, the effect of the number of clusters in the model is mostly insignificant.

Figure 4 summarizes the results showing the average accuracy over the three datasets for all the parameter configurations evaluated . Lastly, table 2 shows the results obtained in the each subset of the dataset separately with our best configuration ($K = 25x25$ neurons in the SOM network, $W = 3$ window size). Note that results for our descriptor with $W = 5$ and $W = 7$ also outperform the second best known method listed in table 1.

3.1.3 Discussion

In this section we perform a detailed analysis of the proposed method, exploring the results in each subset of the dataset separately.

[2] The site https://sites.google.com/site/damdescriptor contains a full list of excluded and corrected actions.

Table 2 Accuracy of the model on each subset of the MSR-Action 3D dataset, for different values of W and a fixed number of clusters $K = 25 \times 25$. Standard deviation shown in parenthesis.

Window	1	3	5	7
AS1	89.4 (±0.9)	92.6 (±1.4)	92.0 (±1.5)	91.2 (±1.5)
AS2	76.8 (±1.9)	92.2 (±1.6)	91.1 (±1.3)	90.7 (±1.0)
AS3	94.8 (±0.5)	97.4 (±0.7)	97.9 (±1.0)	98.9 (±0.7)
Mean	**87.02**	**94.07**	**93.68**	**93.61**

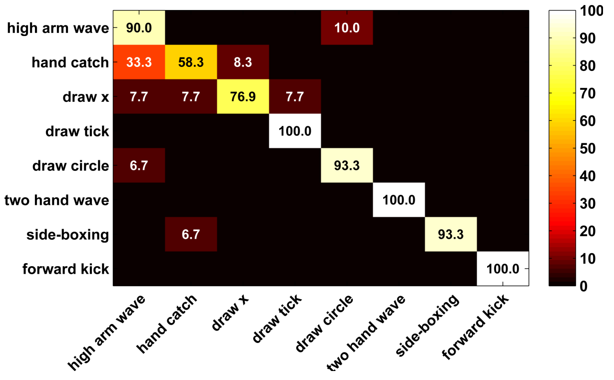


Fig. 5 Confusion matrix for subset AS2 of MSR-Action3D.

Figure 3 shows a significant difference in accuracy between subsets. The method consistently reaches very high accuracies on Subset AS3, as high as 99% in occasions. On the other hand, the method achieves much worse accuracies on AS2, in which most configurations are below 90%.

Figure 5 shows the confusion matrix for a single run of the experiment on AS2, from which it can be deduced which actions affect the recognition rate the most. For example, actions *high arm wave* and *hand catch* are performed with the same set of joints and are very similar in their motions, thus making our model less likely to discriminate between them. Other actions such as *forward kick* or *two hand wave* are naturally easier to distinguish since they involve the use of different joints altogether. This pattern in the difficulty of the dataset was already described in previous works [13] [12].

Figure 6 shows the probability matrix employed by the ProbSOM classification rule for the testing actions. The matrix shows the average estimated probabilities that an action of a certain class C belongs to another class C' (where possibly $C = C'$) as computed by our classification rule. It is remarkable that for some actions (*forward kick*,*two hand wave*) not only the class prediction was perfect (*forward kick*,*two hand wave*) but also that the probability of the action $\mathbf{A}^i$ belonging to a class c was 0 for classes other than c_i (the true class of the action $\mathbf{A}^i$), ie, $P(c|\mathbf{A}^i) = 0$ if $c \neq c_i$. However, some actions which can be readily identified as similar from just their general trajectory did not have such a perfect probability assignment.

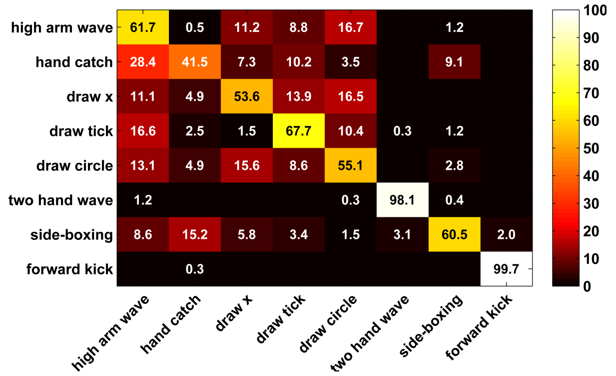


Fig. 6 Average probability of assigning an action of class c (rows) to every other class (columns) for subset AS2 of MSR-Action3D.

3.2 MSRC12 Dataset

The MSRC12 dataset is described in [10] by Fothergill et al. and consists of 594 sequences of 12 different types of actions performed by 30 subjects. Each sequence contains the recorded positions of several instances or performances of the same action by the same person. The action types are described as: *Crouch or hide, Shoot a pistol, Throw an object, Change weapon, Kick, Put on night vision goggles, Start Music/Raise Volume (of music), Navigate to next menu, Wind up the music, Take a Bow to end music session, Protest the music,* and *Move up the tempo of the song*. The data was captured using a Kinect device.

3.2.1 Experimental setup

Following the experimental setup by Hussein et al [3], we segmented the sequences to isolate each action instance using their annotation, obtaining a set of 6244 action instances.

By inspecting the actions visually we were able to detect instances in which the subject did not follow the instructions as intended by the experimenters. For example, subject 10 performs *Kick* with his left leg, instead of the right one; for the action *Wind it up*, subjects must trace a circle with both hands in front of them, but subject 9 makes it with only one hand (alternating right and left), and subject 21 traces only half of the circle. This was indeed expected by the experimenters, given that the database was generated precisely to measure how different instruction modalities (video, pictures, text) affect the performance of the actions, and so those actions performed incorrectly were not excluded from the dataset [10]. The differences in execution are so wide that this outliers can be readily considered as instances of other

Table 3 Comparison of results on the MSRC-12 dataset with cross-subject validation.

Method	Acc. (%)
Ellis (Logistic reg.) [11]	88.7
Hussein (Cov3DJ) [3]	91.7
Proposed descriptor	**91.7**

Table 4 Comparison of results on the MSRC-12 dataset with leave-one-subject-out cross-validation. Note that in the case of Jiang et al's Hierarchical Model, a traditional leave-one-out cross-validation scheme was employed, training and testing with the same subject.

Method	Test Validation	Acc. (%)
Hussein et al. (Cov3DJ) [3]	leave-one-subject-out	93.6
Negin et al. (Decision Forest) [14]	leave-one-subject-out	93.2
Jiang et al. (Hierarchical Model) [15]	leave-one-out	94.6
Proposed descriptor	leave-one-subject-out	93.1

types of action classes, and we have therefore excluded those instances from our experiments [3)].

With the filtered dataset, we performed cross subject experiments with a 50/50 subject split, and 30 cross validation runs for each parameter configuration. We report the average classification rate for each configuration over the 30 runs. The window size for WDFs was set to $W = 5$ and $K = 900$ clusters where used in a 30×30 topography SOM network. We also performed leave-one-subject-out cross validation tests following the protocol described in [3].

3.2.2 Results

Table 3 shows the results obtained for MSRC12 with the proposed method, compared with the most recent ones of the state of the art. The experiments employ a cross-subject validation scheme, following [3], but with subjects chosen randomly for each subject split. The results show the average accuracy on the dataset over 30 independent runs.

[3)] The site https://sites.google.com/site/damdescriptor contains a full list of left out and corrected actions.

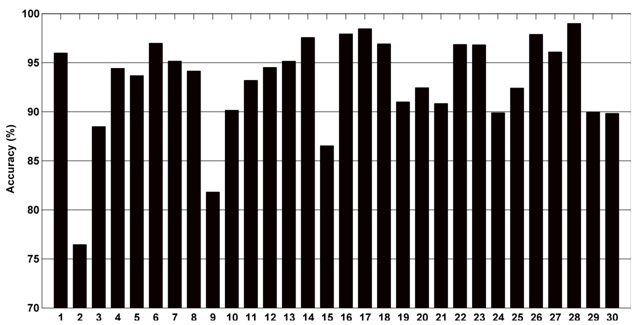


Fig. 7 Accuracy by subject on the MSRC-12 dataset for leave-one-subject-out cross-validation

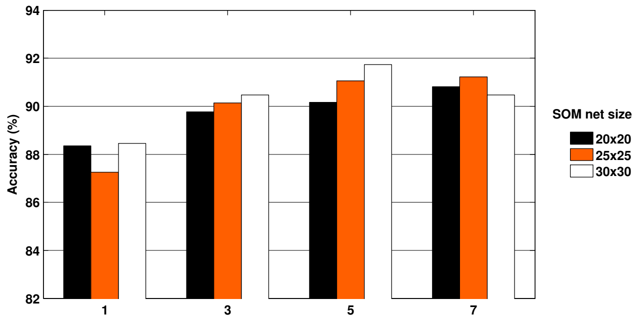


Fig. 8 Accuracy on the MSRC-12 dataset for different window and cluster sizes. $x - axis$ = window sizes, $y - axis$ = Method Accuracy

Table 4 shows results for a leave-one-subject-out validation scheme (with 30 runs), which allows to evaluate the descriptor's performance by simulating a classic scenario where a new subject tries to use a system trained by other people. In this setting, our method obtains an accuracy of 93%, a similar result to those obtained by Negin et al., Hussein et al. and Jiang et al.

In both experimental settings, the variances for the average accuracy of our classification method were low ($\sigma^2 = 5$), but standard hypothesis tests for the differences in accuracies between classifiers could not be performed because neither the variances for the classification accuracy nor the code for performing the tests was available for any of the other methods considered. Nonetheless, the variances obtained seem appropriate given that the test results depend highly on the subjects chosen for testing, as explained in the next paragraph.

Figure 7 shows the accuracy obtained for each subject separately in the leave-one-subject-out cross validation experiments. The graph shows that there is a high amount of variation in the accuracies for each subject, which can be attributed to the different methods the subjects were instructed with (video, pictures, text). For example, subject 28 achieves an accuracy of nearly 100%, while on the other hand for subject 2 it is only 76%. This data suggests that the observed differences in classification accuracy between our descriptor and other models should not be statistically significant.

Figure 8 shows the behaviour of the method for different window and cluster sizes. As with MSRAction3D (subsection 1), the bar graphs clearly show the advantage of using WDFs instead of simple direction frames. On the other hand, the difference in accuracy from varying the number of clusters is much less significant. Given that the success of the ProbSOM classification rule is ultimately determined by how

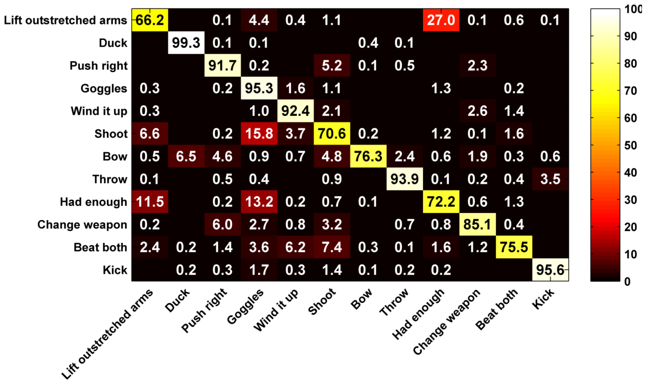


Fig. 9 Average probability of assigning an action of class *c* to every other class for MSRC-12 dataset with leave-one-subject-out validation.

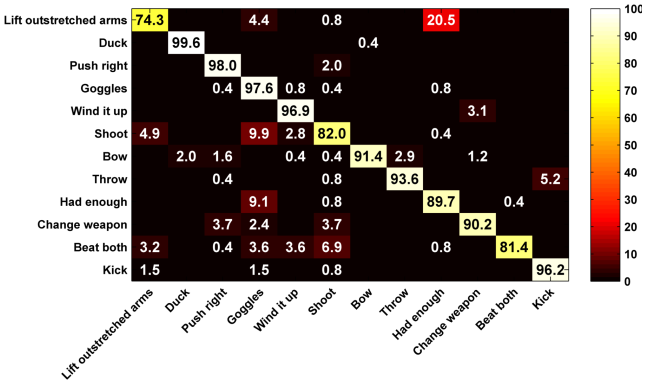


Fig. 10 Confusion matrix for MSRC-12 dataset with leave-one-subject-out validation.

well the estimation of the distribution of WDF for each class was performed, and that estimation on the diversity in performing a certain action, the amount of clusters actually depends highly on the amount of samples in the dataset.

3.2.3 Discussion

The method compares favorably with other state-of-the-art classification techniques for the MSRC12 dataset using traditional cross-subject validation.

Although the accuracy our method achieves is the same as Hussein's [3], their descriptor has a very high dimensionality; the best results they obtain are with a descriptor with $L = 2$ overlapping levels, for which a skeleton represented with 20 joints results in a descriptor of length 7320 [3]. Our descriptor is of length at most 900 in the case of a 30×30 SOM network. Coupled with the fact that the computation of the descriptor involves only simple arithmetic operations, it should provide better real-time performance in terms of efficiency and frame rate.

The method also shows state-of-the art performance when tested with a leave-one-subject-out cross-validation scheme, making it clear that new users can expect a robust response when using a human action recognition system that employs the descriptor.

As with the Action3D dataset, figure 9 shows the distribution of probabilities used by the method to classify actions in the MSRC12 dataset. This is the average of the probabilities for all actions, with the average for each class computed with all the test actions of the class. We also present a confusion matrix for the same dataset and testing scheme in figure 10.

Some actions(i.e *Duck* or *Push Right*) were classified almost perfectly, reaching an great probability distribution too, which presents a strong case for the robustness of the model. Certain actions, like *Lift outstretched arms* or *Shoot* present more difficulties to the descriptor, presumably because they employ similar joints and direction movements. However, the confusion matrix in figure 10 also shows that the misclassification errors mostly involve two or three classes at a time, and so a second layer of classifiers could be trained to separate those specific classes which when together raise the classification error.

4 Conclusion

We have defined a descriptor for human action recognition based on the distribution of action movements (DAM descriptor).

The descriptor is computed as a histogram over a set of canonical or representative movements windowed directions, which are in turn obtained by performing a clustering with a SOM network, over the set of all windowed directions in the training set. We believe that a key step in the generation of the descriptor is the windowing scheme that, unlike other authors' approaches, is included as an intermediate step and not with a pyramidal scheme.

The descriptor is both scale and translation invariant. If desired, direction invariance of the trajectory of an action can be achieved by inverting the direction of each joint in the direction frames. Rotation invariance can be incorporated as well by rotating the user to a canonical rotation using his position and pose to set the axes of the joint positions instead of the camera's.

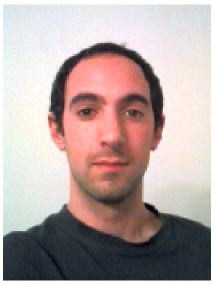
The descriptor shows state-of-the-art performance on standard cross subjects tests on two well known datasets: MSRC12 and MSRAction3D achieving a classification rate of 91.7% and 94% respectively. Additional leave-one-subject-out tests on MSRC12 have also shown state-of-the-art results with an accuracy of 93%.

Our future work will concentrate on including more datasets in our tests to evaluate the generalization performance of the method with other classes, further testing with pyramidal schemes to better include temporal information about the action, and evaluating its performance in online classification tasks.

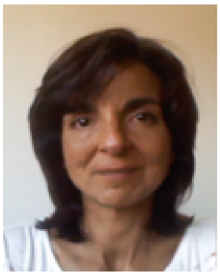
References

1. Angela Yao, Juergen Gall, Gabriele Fanelli, and Luc J Van Gool. Does human action recognition benefit from pose estimation?. In *BMVC*, volume 3, page 6, 2011.
2. Cesar Estrebou, Laura Lanzarini, and Waldo Hasperue. Voice recognition based on probabilistic som. In *Latinamerican Informatics Conference. CLEI 2010. Paraguay. October 2010.*, 2010.
3. Mohamed E Hussein, Marwan Torki, Mohammad A Gowayyed, and Motaz El-Saban. Human action recognition using a temporal hierarchy of covariance descriptors on 3d joint locations. In *Proceedings of the Twenty-Third international joint conference on Artificial Intelligence*, pages 2466–2472. AAAI Press, 2013.
4. Mathieu Barnachon, Saïda Bouakaz, Boubakeur Boufama, and Erwan Guillou. Ongoing human action recognition with motion capture. *Pattern Recognition*, 47(1):238–247, 2014.
5. Mohammad A Gowayyed, Marwan Torki, Mohamed E Hussein, and Motaz El-Saban. Histogram of oriented displacements (hod): describing trajectories of human joints for action recognition. In *Proceedings of the Twenty-Third international joint conference on Artificial Intelligence*, pages 1351–1357. AAAI Press, 2013.
6. Ferda Ofli, Rizwan Chaudhry, Gregorij Kurillo, René Vidal, and Ruzena Bajcsy. Sequence of the most informative joints (smij): A new representation for human skeletal action recognition. *Journal of Visual Communication and Image Representation*, 25(1):24–38, 2014.
7. Kyunghyun Cho and Xi Chen. Classifying and visualizing motion capture sequences using deep neural networks. *arXiv preprint arXiv:1306.3874*, 2013.
8. Wanqing Li, Zhengyou Zhang, and Zicheng Liu. Action recognition based on a bag of 3d points. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2010 IEEE Computer Society Conference on*, pages 9–14. IEEE, 2010.
9. Teuvo Kohonen. *Self-organizing maps*, volume 30. Springer, 2001.
10. Simon Fothergill, Helena Mentis, Pushmeet Kohli, and Sebastian Nowozin. Instructing people for training gestural interactive systems. In *Proceedings of the 2012 ACM annual conference on Human Factors in Computing Systems*, pages 1737–1746. ACM, 2012.
11. Chris Ellis, Syed Zain Masood, Marshall F. Tappen, Joseph J. Laviola, Jr., and Rahul Sukthankar. Exploring the trade-off between accuracy and observational latency in action recognition. *Int. J. Comput. Vision*, 101(3):420–436, February 2013.
12. Jiang Wang, Zicheng Liu, Jan Chorowski, Zhuoyuan Chen, and Ying Wu. Robust 3d action recognition with random occupancy patterns. In *Computer Vision–ECCV 2012*, pages 872–885. Springer, 2012.
13. Jiang Wang, Zicheng Liu, Ying Wu, and Junsong Yuan. Mining actionlet ensemble for action recognition with depth cameras. In *CVPR*, pages 1290–1297. IEEE, 2012.
14. Farhood Negin, Fırat Özdemir, Ceyhun Burak Akgül, Kamer Ali Yüksel, and Aytül Erçil. A decision forest based feature selection framework for action recognition from rgb-depth cameras. In *Image Analysis and Recognition*, pages 648–657. Springer, 2013.
15. Xinbo Jiang, Fan Zhong, Qunsheng Peng, and Xueying Qin. Robust action recognition based on a hierarchical model. *2013 International Conference on Cyberworlds*, 0:191–198, 2013.

Franco RONCHETTI is an advanced PhD student in Computer Science at the School of computer Science of the UNLP. Teaching Assistant for Grade Courses at the National University of La Plata. Between 2008-2011, recipient of a scholarship granted by the Ministry of Education of the Nation for the strengthening of human resources in the area of Information and Communication Technologies. Carrying out research activities since 2010 at the Institute of Research in Computer Science III-LIDI in areas related to soft-computing and intelligent systems, with focus on dynamic problems such as signal processing and pattern recognition, with various publications in the field.



Facundo QUIROGA received his BS in Computer Science in 2014 from the Faculty of Informatics of UNLP, Argentina. He is currently a PhD Student in the same Faculty, doing research work at the III-LIDI Institute. His main research fields comprise machine learning and signal processing, with applications in action recognition and biomedical data analysis. He is also interested in computational neuroscience and the interplay between artificial neural networks and biological ones.



Laura LANZARINI is a full-Time Head Professor since 2001 at the School of Computer Science of the UNLP, teaching in Data Mining topics. Guest Head Professor at the National University of Tierra del Fuego. Guest Professor of the Post-Grade Program at the University of Buenos Aires. Distinguished by the UNLP in December 2010 with the Award to Scientific, Technological, and Artistic Work for Senior Researchers. Director of the group "Intelligent Systems," part of the Institute of Research in Computer Science III-LIDI. Author of numerous publications in journals, congress proceedings and scientific workshops with international referees in subjects pertaining to the design and development of adaptive application based on neural networks, swarm intelligence, and other metaheuristics applicable to Data Mining and Text Mining problems.

Cesar ESTREBOU is a PhD student in Computer Vision, Image Processing and Graphic Computing at the School of Computer Science of the UNLP. Since 2008 is Teaching Assistant (Assignments Supervisor) for Grade Courses at the National University of La Plata. He has been carrying out research activities since 2005 at the Institute of Research in Computer Science III-LIDI in areas related to intelligent systems, real-time systems and software engenieering, with publications in journals, congress proceedings and scientific workshops with international referees in those fields.