Histogram of Oriented Displacements (HOD): Describing Trajectories of Human Joints for Action Recognition

Marwan Torki
Alexandria University
42 PUBLICATIONS **571** CITATIONS
SEE PROFILE

Mohammad A. Gowayyed
Alexandria University
7 PUBLICATIONS **603** CITATIONS
SEE PROFILE

Mohamed E Hussein
Egypt-Japan University of Science and Technology
39 PUBLICATIONS **535** CITATIONS
SEE PROFILE

Motaz El Saban
Microsoft
45 PUBLICATIONS **642** CITATIONS
SEE PROFILE

Histogram of Oriented Displacements (HOD): Describing Trajectories of Human Joints for Action Recognition

Mohammad A. Gowayyed[1], **Marwan Torki**[1], **Mohamed E. Hussein**[1], **Motaz El-Saban**[2]
[1]Department of Computer and Systems Engineering, Alexandria University, Alexandria, Egypt
{m.gowayyed, mtorki, mehussein}@alexu.edu.eg
[2]Microsoft Research Advanced Technology Lab Cairo, Cairo, Egypt
motazel@microsoft.com

Abstract

Creating descriptors for trajectories has many applications in robotics/human motion analysis and video copy detection. Here, we propose a novel descriptor for 2D trajectories: Histogram of Oriented Displacements (HOD). Each displacement in the trajectory votes with its length in a histogram of orientation angles. 3D trajectories are described by the HOD of their three projections. We use HOD to describe the 3D trajectories of body joints to recognize human actions, which is a challenging machine vision task, with applications in human-robot/machine interaction, interactive entertainment, multimedia information retrieval, and surveillance. The descriptor is fixed-length, scale-invariant and speed-invariant. Experiments on MSR-Action3D and HDM05 datasets show that the descriptor outperforms the state-of-the-art when using off-the-shelf classification tools.

1 Introduction

A trajectory is defined as the path of a point moving with time. For videos, these points can be low level visual features, or high level visual features like human skeletal joints. If we can extract these features with acceptable accuracy and adequately describe their trajectories, we can perform better video classification. One of the most popular video classification applications is human action recognition.

Human action recognition in videos has been an active research topic in computer vision. One of the problems in the recognition is the availability of data. Accurate data like MoCap, such as CMU MoCap [1] and HDM05 [Müller *et al.*, 2007], is expensive to acquire. Recently, Microsoft Kinect and other low cost sensors provided the depth data with acceptable accuracy. [Shotton *et al.*, 2011] developed a real-time approach to extract 3D joints positions from a single depth image. Despite requiring extensive training on synthetic data, extracting the joint locations in real-time became a doable task. With all these available data, interactive touchless games became tractable. However, researchers still have a lot to do to enhance current recognition approaches and provide better gaming experience.

In this paper, we exploit these skeletal joint 3D locations to develop an efficient action recognition approach. We introduce a novel 2D trajectory descriptor: Histogram of Oriented Displacements (HOD). To construct HOD, each displacement in the trajectory votes with its length in an orientation angles histogram. We use HOD to describe the three projections of each 3D trajectory for each body joint. For temporal modeling, we apply a temporal pyramid that describes the trajectory as whole, halves and then quarters. We show how the proposed descriptor is efficient and discriminative on two popular datasets: MSR-Action3D [Li *et al.*, 2010] and HDM05 [Müller *et al.*, 2007]. For the two datasets, our descriptor outperforms the state-of-the-art using off-the-shelf classification approaches.

The paper is organized as follows: In Section 2, we review the related work in both trajectory description and activity recognition. Our approach is described in Section 3. Section 4 introduces the datasets used with the experimental results. Section 5 concludes the paper.

2 Related Work

2.1 Trajectory Description

The problem of finding efficient and representative trajectory descriptors has many applications in robotics/human motion analysis and video copy detection. Most work in literature that tried to create a descriptor for the trajectories did not target a fixed-length descriptor.

[Wu *et al.*, 2008] and [Yang and Li, 2010] introduced a variable-length descriptor using the curvature and torsion of each point on the trajectory and their derivatives. They developed a similarity measure to match their trajectory descriptors. In [Wang *et al.*, 2011], they described each trajectory using the normalized displacement vector, but limited the trajectory length to a fixed value, instead of handling different lengths.

In their work on activity recognition in unconstrained videos, [Sun *et al.*, 2009] used a fixed-length trajectory descriptor. They normalize and quantize the displacement vector D in terms of magnitude and orientation. Each displacement has a state between 1 to 25 (3 magnitude levels $\times$ 8 orientation levels $+$ 1 for the zero level). After quantizing all

[1] http://mocap.cs.cmu.edu/

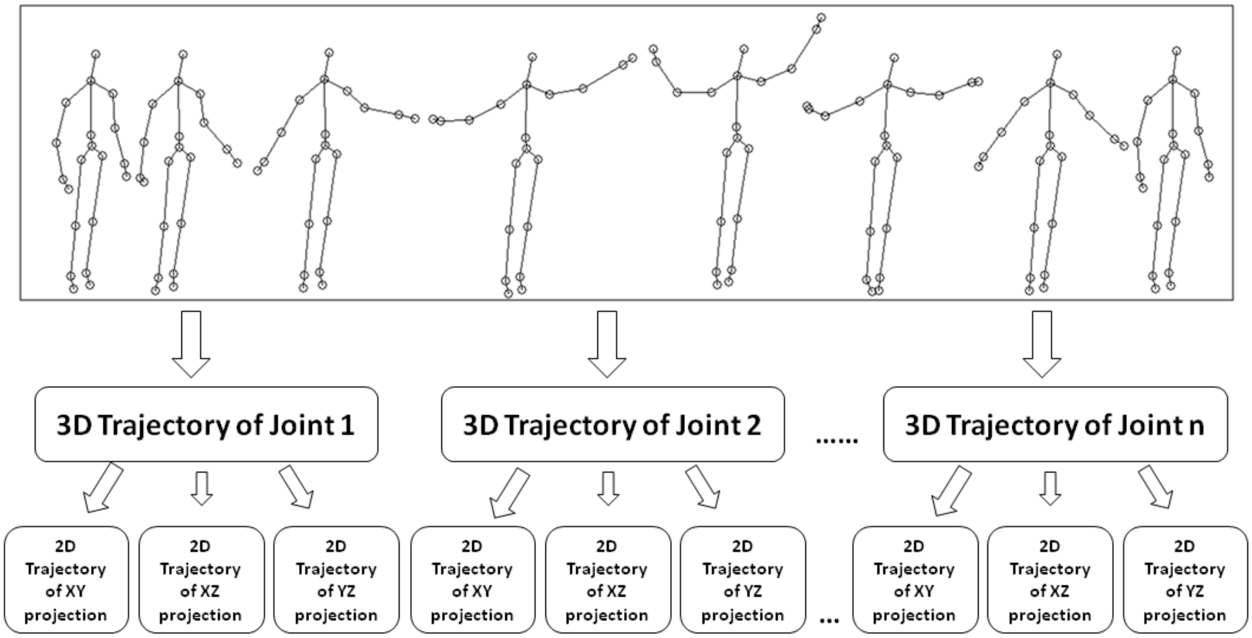
3D Trajectory of Joint 1
3D Trajectory of Joint 2
......
3D Trajectory of Joint n
2D Trajectory of XY projection
2D Trajectory of XZ projection
2D Trajectory of YZ projection
2D Trajectory of XY projection
2D Trajectory of XZ projection
2D Trajectory of YZ projection
...
2D Trajectory of XY projection
2D Trajectory of XZ projection
2D Trajectory of YZ projection

Figure 1: Given a sequence of body joints locations of a human performing an action in n frames, our goal is to provide a discriminative descriptor for this sequence. We describe the 3D trajectory of each individual joint, then concatenate the descriptors of all joints to form the final descriptor. Each 3D trajectory is represented by the HOD of its three 2D projections (xy, xz and yz).

displacement vectors along a trajectory, they translate the sequential relations between these vectors into a directed graph, which is similar to the state diagram of a Markov chain. Finally, they calculate the transition matrix P and the descriptor is the stationary distribution row vector π of P.

In their work on video copy detection, [Chen *et al.*, 2010] used a fixed-length descriptor for a 2D trajectory. The descriptor was used to classify the trajectories into four categories: *static, horizontal motion, vertical motion and complex motion*. They quantize the displacement vector D and build a 5-bin histogram for them. They used 2 quantization levels for magnitude and 4 for orientation. Each histogram bin counts the number of displacements in the quantization range.

2.2 Action Recognition

The availability of skeleton joints locations at real-time [Shotton *et al.*, 2011] was a great motivation for researchers to develop more efficient and powerful recognition approaches. [Yao *et al.*, 2011] showed that pose-based features outperform low-level appearance features, even when heavily corrupted by noise, suggesting that pose estimation is beneficial for the action recognition task.

Researchers used generative models like Hidden Markov Model (HMM), or discriminative models, such as Conditional Random Fields (CRF) [Han *et al.*, 2010]. These methods use the joint positions or histograms of the joint positions as observations. In [Xia *et al.*, 2012] a histogram of 3D joints descriptor in a frame is computed, a dictionary is built and the temporal modeling is done via HMM. However, these complex generative models are easy to overfit due to the limited amount of training data. Moreover, the 3D joint positions that are generated from depth map sequences are noisy (compared to that of the MoCap data). That makes determining the accurate states from the observations very difficult, especially in similar actions.

[Wang *et al.*, 2012] showed that the complex and nonlinear dynamics can be characterized by a Recurrent Neural Network [Martens and Sutskever, 2011]. However, the performance was not promising. Another applicable neural network method is the Conditional Restricted Boltzman Machines [Mnih *et al.*, 2012]. Due to the large number of parameters to tune, the latter two models need lots of data and epochs to be able to estimate model parameters accurately.

For MSR-Action3D dataset, the state-of-the-art classification accuracy using joint positions was done by [Wang *et al.*, 2012]. They used the 3D joints positions to construct a descriptor of relative positions between joints. The descriptor is then concatenated with Local Occupancy Patterns to capture the human-object interaction. To capture the temporal behavior, a Fourier Pyramid of this descriptor was introduced. Finally, they used this descriptor to mine a set of actionlets for each class. The weights of the actionlets are then learnt via Multiple Kernel Learning to perform the classification. We have proposed another descriptor in [Hussein *et al.*, 2013], which describes the action using covariance of joint features. However, our simple descriptor here with linear SVM outperforms [Wang *et al.*, 2012] sophisticated classification procedure. It also outperforms our other approach [Hussein *et al.*, 2013] on the datasets tested here.

3 Approach

Our approach is to describe the 3D trajectory of each joint separately, as shown in figure 1. First, we replace the 3D trajectory of each joint with three 2D trajectories representing the three projections on the three orthogonal Cartesian planes (xy, yz, and xz). The 3D trajectory descriptor is the concatenation of the three. Temporal pyramid is built for each 2D trajectory to capture the temporal information. We describe each 2D trajectory using a Histogram of Oriented Displacements.

In the next subsections, we explain the 2D descriptor: HOD, temporal pyramid and the final descriptor for each joint. The final descriptor, for each video sequence is the concatenation of this descriptor among all available joints.

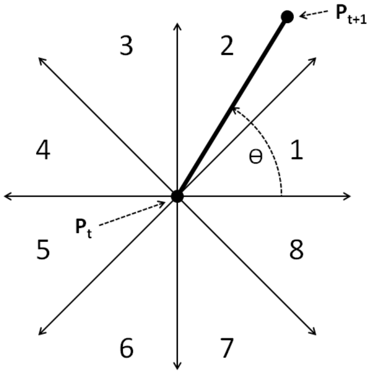

Figure 2: For a 2D trajectory where P_t is the position of the joint at time t. This figure shows a general displacement between P_t and P_{t+1}. In this example, the trajectory is described by a histogram of 8 bins. For each displacement, the angle θ and the length of the displacement are calculated. The length is added to the appropriate histogram bin. For the displacement shown in this figure, the length of (P_t, P_{t+1}) will be added to the second histogram bin.

3.1 Histogram of Oriented Displacements (HOD)

We introduce a novel 2D trajectory descriptor, we call it HOD. The trajectory is described using a histogram of the directions between each two consecutive points. Given a trajectory $T = \{P_1, P_2, P_3, ..., P_n\}$, where P_t is the 2D position at time t. For each pair of positions P_t and P_{t+1}, calculate the direction angle $\theta(t, t+1)$, as the angle of the line with slope in equation 1.

$$slope = \frac{P_{t+1}.y - P_t.y}{P_{t+1}.x - P_t.x} \quad (1)$$

Value of θ is between 0 and 360. A histogram of the quantized values of θ is created. If the histogram is of 8 bins, the first bin represents all θs between 0 and 45.

The histogram accumulates the lengths of the consecutive moves. For each θ, a specific histogram bin is determined using equation 2. The length of the line between P_t and P_{t+1} is then added to the specific histogram bin. Figure 2 shows an example of a displacement when using a histogram of length 8.

$$histogram\_bin = \frac{angle \times histogram\_length}{360} \quad (2)$$

To show the intuition behind the descriptor, consider the action of waving hands. At the end of the action, the hand falls down. When describing this down movement, the descriptor does not care about the position from which the hand started to fall. This fall will affect the histogram with the appropriate angles and lengths, regardless of the position where the hand started to fall.

HOD records for each moving point: how much it moves in each range of directions. HOD has a clear physical interpretation. It proposes that, a simple way to describe the motion of an object, is to indicate how much distance it moves in each direction. If the movement in all directions are saved accurately, the movement can be repeated from the initial position to the final destination regardless of the displacements order. However, the temporal information will be lost, as the order of movements is not stored-this is what we solve by applying the temporal pyramid, as shown in section 3.2. If the angles quantization range is small, classifiers that use the descriptor will overfit. Generalization needs some slack in directions-which can be done by increasing the quantization range.

3.2 Temporal Pyramid

Dealing with the trajectory as whole misses the temporal information. In order to capture the temporal evolution, we apply a temporal pyramid approach. In the first level, the whole trajectory is used to construct a part of the descriptor. At the second level, the trajectory is divided into two halves and each one of them is used separately to obtain the second two parts of the descriptor. If the number of levels is 3, the descriptor is made of 7 parts: 1 for the first level, 2 for the second level and 4 for the third. The final descriptor is the concatenation of the seven parts. In other words, a histogram at a specific level will be split into two histograms in the next level. Similar spatial and temporal pyramids were used in [Lazebnik *et al.*, 2009] and [Wang *et al.*, 2012]. The values of the histogram at a specific level to be the summation of the two histograms that result from its splitting.

Figure 3 shows an example of the pyramid using 3 levels for the xy projection in figure 4. The figure shows how the pyramid catches the temporal variation. Most moves happened in the second part of the video as clear from second level. The third level captured the difference in movement between the third quarter and the forth.

3.3 3D Trajectory Descriptor

In order to describe the 3D trajectory of one joint, we use the HOD of the three 2D projections (xy, yz and xz). The final descriptor for the joint is the concatenation of the three 2D descriptors.

Figure 4 shows the projections of the Right Hand joint when the first subject int the MSR-Action3D dataset performs the action of High Arm Waving.

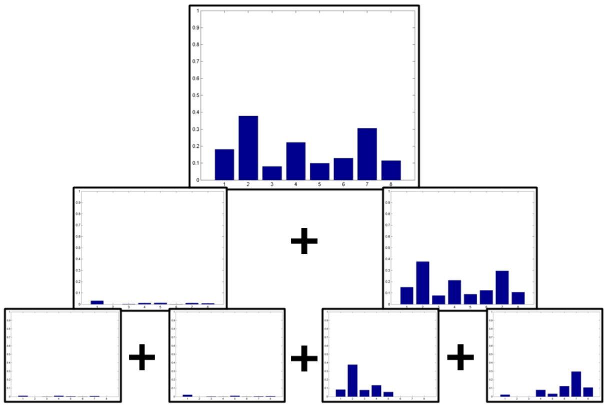

Figure 3: The figure shows the 3-level temporal pyramid of the xy projection at figure 4. At the first level, the whole trajectory is considered. At the second level, the trajectory is divided into 2 halves. At the third level, it is divided into 4 quarters. The final descriptor of this trajectory is the concatenation of the 7 histograms. It can be seen that each histogram at a level is a summation of the two histograms that result from its splitting.

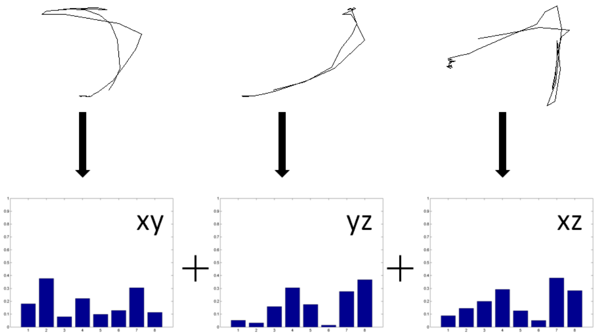

Figure 4: From the Action3D dataset, this figure shows the three projections of the Right Hand joint when a subject is performing the action of High Arm Wave. For each projection, HOD is used to describe the movement. The figure shows a pyramid of only one level. The final descriptor of the joint is the concatenation of the 3 HODs.

3.4 Discussion

The descriptor does not require any pre-processing for the joints positions. It is scale-invariant, given that the histograms are normalized. Our current implementation is not rotation-invariant. However, it can be made rotation-invariant by mapping all the positions such that two specific sticks become the x and y axes. For example, the stick between the two shoulders and the stick between head and spine.

Using the magnitude of the displacement to update the histogram makes the descriptor speed-invariant. If the linear movement between P_t and P_{t+1} is performed between 2 frames, it will affect the histogram by the same amount as if it is performed over 20 frames.

One important advantage of HOD is the efficiency in computations. For each frame, the only computation required is: 1) calculate the displacements of positions. 2) calculate the angles. 3) update the appropriate histogram bins. Even when applying multiple levels for the pyramid, we do not need to compute all levels. Due to the dependency between levels, we only need to compute the deepest level, then the higher levels are computed in constant time(for a specific number of bins).

One of the main reasons why the proposed descriptor has the potential to be applicable to real world is the low number of parameters to tune. Section 4.2 shows that our weakest configuration competes favorably with some of the popular approaches in literature.

4 Experiments

In this section, we explain our experimental setup and show our experiments and experimental results. We used two datasets: MSR-Action3D and a subset of the HDM05 dataset used in [Ofli *et al.*, 2012]. All experiments are done on the original raw data without any normalization before calculating the descriptor. After creating the descriptor, we apply L2 normalization on the descriptor to achieve scale-invariance. The used classification algorithm is a linear SVM using LIBSVM [Chang and Lin, 2011].

4.1 Datasets

MSR- Action3D

We use the MSR-Action3D [Li *et al.*, 2010] which has 20 action types, 10 subjects, each subject performs each action 2 or 3 times. There are 567 depth map sequences in total. We use the standard setup, which divides the data into three action sets. Each set has 8 actions with some overlap between action sets. The same setting has been used recently in [Wang *et al.*, 2012]. We used the 20 joints locations as extracted in [Shotton *et al.*, 2011]. The actions are *high arm wave, horizontal arm wave, hammer, hand catch, forward punch, high throw, draw x, draw tick, draw circle, hand clap, two hand wave, side-boxing,bend, forward kick, side kick, jogging, tennis swing, tennis serve, golf swing, pickup & throw*. The actions are divided into 3 action sets as shown in table 1. The available 20 joints are shown in figure 5.

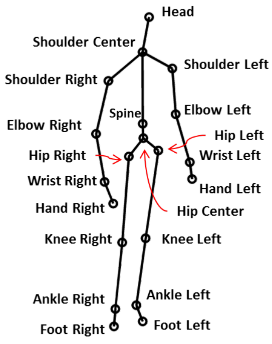

Figure 5: Skeleton joint locations and names as captured by the Kinect sensor.

ActionSet1 (AS1)	ActionSet2 (AS2)	ActionSet3 (AS3)
Horizontal Wave	High Wave	High Throw
Hammer	Hand Catch	Forward Kick
Forward Punch	Draw X	Side Kick
High Throw	Draw Tick	Jogging
Hand Clap	Draw Circle	Tennis Swing
Bend	Hands Wave	Tennis Serve
Tennis Serve	Forward Kick	Golf Swing
Pickup & Throw	Side Boxing	Pickup & Throw

Table 1: Action sets of MSR-Action3D

Set	1 level	2 levels	3 levels
Action Set 1	91.3	92.39	91.3
Action Set 2	89.29	90.18	90.18
Action Set 3	91.43	91.43	91.43
Mean	90.61	**91.26**	90.94

Table 2: Classification accuracy (%) on the MSR-Action3D dataset using 50% subjects split and linear SVM applied to the descriptor directly, using different number of levels for the temporal pyramid and 16-bin HOD.

HDM05

Similar to [Ofli *et al.*, 2012], we applied our approach to a Motion Capture dataset, namely the HDM05 database [Müller *et al.*, 2007]. There are three main differences between this dataset and the preceding dataset: First, it is captured using motion-capture sensors, which leads to much less noise than in the data acquired by a Kinect sensor. Second, the number of joints recorded is 31 instead of 20. This leads to a longer descriptor. Third, the frame rate is much higher, 120 fps instead of 15 fps as in the preceding dataset.

We used the same setup in [Ofli *et al.*, 2012] with the same 11 actions performed by 5 subjects. We had 249 sequences in total. We used 3 subjects (140 action instances) for training, and 2 subjects (109 action instances) for testing. The set of actions used in this experiment is: *deposit floor, elbow to knee, grab high, hop both legs, jog, kick forward, lie down floor, rotate both arms backward, sneak, squat, and throw basketball*

4.2 Results and Discussion

MSR-Action3D

For the MSR-Action3D dataset, we compare our approach to the state-of-the-art classification results using skeletal joint positions at [Wang *et al.*, 2012]. They use Multiple Kernel Learning to classify a testing example using different actionlets for each action.

Method	Accuracy(%)
Recurrent Neural Network	42.5
Hidden Markov Model	78.97
Action Graph on Bag of 3D Points	74.7
Random Occupancy Patterns	86.5
Actionlets Ensemble	88.2
Proposed Descriptor	**91.26**

Table 3: Classification Accuracy Comparison for MSR-Action3D dataset.

The best classification accuracy they obtain is **88.2**% using the ensemble and the accuracy is lower when they apply their descriptor directly to a linear SVM, they did not report the results of the latter case. However, they reported that removing ensemble from experiments on MSR-DailyActivity3D dataset, decreased their accuracy from 85.75% to 78%. This shows how the accuracy decreases considerably when the ensemble is removed.

Using 16-bin HOD and 2-level temporal pyramid, we get classification accuracy of **91.26**% without using boosting nor ensemble methods.[2]

Table 2 shows the classification accuracy using linear SVM and when using temporal pyramids of different number of levels and using 16-bin HOD. Results show that adding more levels enhances the classification accuracy. When the accuracy decreases upon increasing the levels, it means that the available frames in each histogram of the new level are too few to make a meaningful histogram. Even with only one level, our descriptor still outperforms the state-of-the-art. Table 3 shows the results reported in [Wang *et al.*, 2012] in addition to our obtained accuracy.

The number of bins of the histograms affects the results. We studied this effect by changing the length of the histogram and repeating the experiments of table 2. Figure 6 shows the classification accuracy when using different lengths of histograms. We tried 4 histogram configurations: 4, 8, 12 and 16. The 4-bin histogram is the least accurate but still a lot better than a random classifier.

One of the interesting observations is that our weakest configuration (4-bin HOD with only 1-level temporal pyramid) outperforms Recurrent Neural Networks [Martens and Sutskever, 2011], Hidden Markov Model [Xia *et al.*, 2012] and Action Graph on Bag of 3D Points [Li *et al.*, 2010] as shown in table 3. This superiority is due to the physical meaning of the descriptor that we illustrated in section 3. It contains more information about the sequence, that is you may be able to recover more of the original data than from other descriptors. Moreover, the descriptor for this configuration is only 240 values for the complete video (3 projections × 20 joints × 4 bins of histogram × 1 histogram for the only level). This means that our descriptor is still discriminative even without careful parameter tuning. Requiring careful parameter tuning is a problem that decreases the applicability of any recognition method. For instance, [Wang *et al.*, 2012] needs to choose the number of Fourier coefficients to use, confidence and ambiguity values to mine the actionlets.

One way to show the strength of the descriptor is to study the classification accuracy when using only one joint at a time. Table 4 shows that using the information of only one joint is much better than a random classifier. The descriptor's length is only 144 (3 projections × 1 joint × 16 bins of histogram × 3 histograms for the two levels). When using the

[2] We excluded 13 sequences with around 1/3 of each sequence with zero-measurements in actions *bend* and *pickup & throw*. When we include the corrupted sequences but remove the corrupted frames, we get 90.24%. In our method, removing the zero-measurement frames means a linear interpolation for these frames. With the corrupted sequences and the corrupted frames, we get 88.11%.

Joint	Accuracy(%)	Joint	Accuracy(%)
Left Shoulder	39.38	Right Wrist	70.32
Right Shoulder	48.67	Left Hand	56.49
Neck	44.25	**Right Hand**	**74.07**
Torso	43.22	Left Knee	34.7
Left Hip	43.26	Right Knee	41.55
Right Hip	37.84	Left Ankle	24.87
Center Hip	41.79	Right Ankle	32.63
Left Elbow	57.44	Left Foot	32.86
Right Elbow	64.17	Right Foot	36.66
Left Wrist	52.6	Head	42.34

Table 4: Results on MSR-Action3D when using each joint separately using 50% subjects split. Each video is represented by only 144 values (3 for the three projections × 3 for the 2-level temporal pyramid × 16 for each histogram).

Histogram length	1 level	2 levels	3 levels
4-bin HOD	80.9	95.45	**97.27**
8-bin HOD	86.36	94.55	**97.27**
12-bin HOD	87.28	92.73	94.55
16-bin HOD	88.18	93.64	94.55

Table 5: Classification accuracy (%) on the HDM05 dataset using linear SVM on the descriptor directly, using different number of levels and different number of bins for the histograms.

right hand joint only the accuracy is **74.07**%, this outperforms RNN and is very close to HMM and Action Graph on Bag of 3D Points, as shown in table 3.[3]

These interesting results propose that, we may not need to detect all joints to recognize actions. Only a subset of these joints (possibly one joint only) can discriminate actions well. If the joint movement is similar in two actions, this joint won't be enough for discrimination. This is clear when observing right hand joint results. The right hand joint mainly failed to discriminate between *high arm wave* and *two hand wave*, because its movement is similar in the two actions. Only 30% of the testing examples of the *high arm wave* action are classified right, while 58% of them are mis-classified as *two hand wave*.

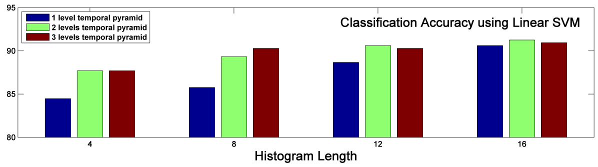

Figure 6: Classification accuracy when using different histogram lengths. The three lines represent the three levels configurations(one, two and three levels)

HDM05

For the HDM05 dataset, we compare our approach to [Ofli *et al.*, 2012]. The best classification accuracy they get is **84.4**%.

Joint	Accuracy(%)	Joint	Accuracy(%)
root	60	headtop	68.18
lhip	63.63	lshoulder	67.27
lknee	58.18	lelbow	62.73
lankle	53.64	lradius	60
lfoot	43.64	lwrist	55.45
ltoes	53.64	lhand	60.91
rhip	60	lfingers	64.55
rknee	50	lthumb	61.82
rankle	55.45	rshoulder	58.18
rfoot	48.18	relbow	**82.72**
rtoes	46.36	rradius	80.91
belly	68.18	rwrist	79.09
chest	58.18	rhand	75.45
thorax	62.72	rfingers	66.36
lowerneck	69.09	rthumb	77.27
head	68.18		

Table 6: Results on HDM05 using each joint separately. Each video is represented by only 168 values (3 for the three projections × 7 for the 3-level temporal pyramid × 8 for each histogram).

We get **97.27**% using 4-bin HOD and 3-level temporal pyramid. As shown in table 5, all our configurations outperform them except for the weakest one.

Using only the right elbow positions, we get classification accuracy of **82.72%**. This is very close to the prior work of [Ofli *et al.*, 2012] and with descriptor of length 168 (3 for the three projections × 7 for temporal pyramid × 8 for each histogram). The classification accuracy when using each joint separately are shown in table 6. We use the 31 available joints.

5 Conclusion

This paper addressed the problem of human action recognition using the skeletal joint locations. We introduced a novel 2D trajectory descriptor: Histogram of Oriented Displacements (HOD). We used HOD to efficiently describe the 3D joints trajectories. HOD is scale-invariant and speed-invariant. Recognition using this descriptor directly with linear SVM outperforms the best published results that use more elaborate classification procedures on two public datasets: MSR-Action3D and HDM05. The discrimination power of the descriptor allowed our weakest configuration to outperform three popular methods in literature: Hidden Markov Model, Recurrent Neural Networks and Action Graph on Bag of 3D Points.

Acknowledgment

This research was sponsored by a grant from the Microsoft Research Advanced Technology Lab Cairo. The project was hosted by Virginia-Tech–Middle East and North Africa (VT-MENA) Research Center in Alexandria University, Alexandria, Egypt.

[3] Subjects who recorded this dataset were informed to perform the action using their right hand, if it is done using only one hand.

References

[Chang and Lin, 2011] Chih-Chung Chang and Chih-Jen Lin. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27, 2011. Software available at http://www.csie.ntu.edu.tw/ cjlin/libsvm.

[Chen *et al.*, 2010] S. Chen, J. Wang, Y. Ouyang, B. Wang, Q. Tian, and H. Lu. Multi-level trajectory modeling for video copy detection. In *Acoustics Speech and Signal Processing (ICASSP), 2010 IEEE International Conference on*, pages 2378–2381. IEEE, 2010.

[Han *et al.*, 2010] L. Han, X. Wu, W. Liang, G. Hou, and Y. Jia. Discriminative human action recognition in the learned hierarchical manifold space. *Image and Vision Computing*, 28(5):836–849, 2010.

[Hussein *et al.*, 2013] 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 23rd International Joint Conference on Artificial Intelligence*, 2013.

[Lazebnik *et al.*, 2009] Svetlana Lazebnik, Cordelia Schmid, Jean Ponce, et al. Spatial pyramid matching. *Object Categorization: Computer and Human Vision Perspectives*, pages 401–415, 2009.

[Li *et al.*, 2010] W. Li, Z. Zhang, and Z. 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.

[Martens and Sutskever, 2011] J. Martens and I. Sutskever. Learning recurrent neural networks with hessian-free optimization. In *Proc. 28th Int. Conf. on Machine Learning*, 2011.

[Mnih *et al.*, 2012] V. Mnih, H. Larochelle, and G.E. Hinton. Conditional restricted boltzmann machines for structured output prediction. *arXiv preprint arXiv:1202.3748*, 2012.

[Müller *et al.*, 2007] M. Müller, T. Röder, M. Clausen, B. Eberhardt, B. Krüger, and A. Weber. Documentation mocap database hdm05. 2007.

[Ofli *et al.*, 2012] F. Ofli, R. Chaudhry, G. Kurillo, R. Vidal, and R. Bajcsy. Sequence of the most informative joints (smij): A new representation for human skeletal action recognition. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2012 IEEE Computer Society Conference on*, pages 8–13. IEEE, 2012.

[Shotton *et al.*, 2011] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake. Real-time human pose recognition in parts from single depth images. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 1297–1304. IEEE, 2011.

[Sun *et al.*, 2009] J. Sun, X. Wu, S. Yan, L.F. Cheong, T.S. Chua, and J. Li. Hierarchical spatio-temporal context modeling for action recognition. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 2004–2011. IEEE, 2009.

[Wang *et al.*, 2011] H. Wang, A. Klaser, C. Schmid, and C.L. Liu. Action recognition by dense trajectories. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 3169–3176. IEEE, 2011.

[Wang *et al.*, 2012] J. Wang, Z. Liu, Y. Wu, and J. Yuan. Mining actionlet ensemble for action recognition with depth cameras. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 1290–1297. IEEE, 2012.

[Wu *et al.*, 2008] S. Wu, YF Li, and J. Zhang. A hierarchical motion trajectory signature descriptor. In *Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on*, pages 3070–3075. IEEE, 2008.

[Xia *et al.*, 2012] L. Xia, C.C. Chen, and JK Aggarwal. View invariant human action recognition using histograms of 3d joints. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2012 IEEE Computer Society Conference on*, pages 20–27. IEEE, 2012.

[Yang and Li, 2010] JY Yang and YF Li. A new descriptor for 3d trajectory recognition. *The Ninth International Symposium on Operations Research and Its Applications*, 2010.

[Yao *et al.*, 2011] Angela Yao, Juergen Gall, Gabriele Fanelli, and Luc Van Gool. Does human action recognition benefit from pose estimation?". In *Proceedings of the 22nd British machine vision conference-BMVC 2011*, 2011.