Moving Poselets: A Discriminative and Interpretable Skeletal Motion Representation for Action Recognition

Lingling Tao and René Vidal
Center for Imaging Science, Johns Hopkins University
ltao4, rvidal@jhu.edu

Abstract

Given a video or time series of skeleton data, action recognition systems perform classification using cues such as motion, appearance, and pose. For the past decade, actions have been modeled using low-level feature representations such as Bag of Features. More recent work has shown that mid-level representations that model body part movements (e.g., hand moving forward) can be very effective. However, these mid-level features are usually hand-crafted and the dictionary of representative features is learned using ad-hoc heuristics. While automatic feature learning methods such as supervised sparse dictionary learning or neural networks can be applied to learn feature representation and action classifiers jointly, the resulting features are usually uninterpretable. In contrast, our goal is to develop a principled feature learning framework to learn discriminative and interpretable skeletal motion patterns for action recognition. For this purpose, we propose a novel body-part motion based feature called Moving Poselet, which corresponds to a specific body part configuration undergoing a specific movement. We also propose a simple algorithm for jointly learning Moving Poselets and action classifiers. Experiments on MSR Action3D, MSR DailyActivity3D and Berkeley MHAD datasets show that our two-layer model outperforms other two-layer models using hand-crafted features, and achieves results comparable to those of recent multi-layer Hierarchical Recurrent Neural Network (HRNN) models, which use multiple layers of RNN to model the human body hierarchy.

1. Introduction

Action recognition from video data has become an important topic in the computer vision community in recent years. In contrast to action recognition from 2D images, action recognition from video data usually involves processing sequential visual data that contains temporal movement information. While 2D images only provide appearance and pose information at one single frame, videos contain temporal dynamics of the entire sequence, and are thus much more informative than static images. However, recognizing actions in videos is still a difficult problem due to various challenges such as occlusions, view point changes and variation in appearance.

Recent developments in depth sensors (e.g. Microsoft Kinect) and pose estimation algorithms [20], have enabled efficient and relatively accurate prediction of human skeletons, with robustness to view point changes or appearance variations. This has motivated the interesting question of how to extract discriminative features from this kind of sequential data. A frequently-used method is Bag of Features (BoF) [10], which is based on extracting local spatial-temporal features, and computing the distribution of feature descriptors to represent each action instance. Recent work has shown that mid-level features can be more effective at recognizing actions. Unlike early BoF models, which only use local information, mid-level features can capture discriminative body part pose or movement (e.g., hand moving forward) for different actions. These features are usually interpretable, but they are typically generated with ad-hoc heuristics (e.g., selecting the set of mid-level descriptors that has a high ratio of in-class neighbors).

Our work aims at learning discriminative mid-level features based on body part movement. We use an automatic feature learning framework inspired by recent mid-level representations and neural networks models for object recognition. To capture information from different body parts, we learn one dictionary for each body part configuration. Specifically, our model extracts mid-level descriptors at every frame for each body part. These features are then represented in terms of their corresponding dictionaries to generate a set of response maps. Finally a high-level feature representation is computed based on the response maps and used for action classification.

Our proposed model is capable of learning **interpretable** and **discriminative mid-level** feature representation with an **efficient** feature learning scheme. Specifically, our contributions are three-fold:

1. *Body-part Motion Pattern Based Feature*. We design a body-part based feature descriptor to capture spatial-temporal movement of human body parts, which is defined as position and velocity values associated with specific body part in a temporal window. While prior work only uses mid-level features to capture the motion dynamics of single joint, or body pose information at one frame, we show that the movement of the human body part as a whole, is more informative.

2. *Discriminative and Efficient Learning*. Most prior work on learning mid-level features uses a dictionary that is pre-learned by complex data mining techniques. In sharp contrast, we present a framework for jointly learning the feature representations and action classifiers. In our model, each column of the mid-level dictionary acts as a linear feature classifier and the response maps to these classifiers are used to aggregate histograms for classification. Our model can be viewed as a modified two-layer Convolutional Neural Networks [12] model that is adapted to the human body structure.

3. *Interpretability*. The features learned by generic CNN models are usually hard to interpret. In sharp contrast, our mid-level feature classifiers are descriptive of body part configuration undergoing a certain movement, which is named as *Moving Poselet*. Thus the features are interpretable and can be visualized to help understand the discriminative body part movement (e.g., hand moving up) for each action.

2. Related Work

There is much related work on designing feature representation for action classification. For low-level features, most state-of-the-art methods are based on the popular Bag-of-Features (BoF) approach. A common first step of the BoF approach is to extract a set of spatial-temporal interest points using a Harris3D detector [10], densely sampled trajectories [23, 24], or other interest point detectors. Each interest point is then described using a spatio-temporal descriptor. Unsupervised learning techniques such as k-means are adopted to build a dictionary of motion words. A video is then represented by a histogram of these motion words [11, 23], and classifiers are trained on top of these histograms for recognizing actions. For skeleton data, the interest points are usually skeleton joints, and a sequence is represented by a histogram of 3D joint positions [27]. The main advantages of the BoF approach are its simplicity and empirical success. Nevertheless, the key drawbacks of the BoF approach are that (1) motion words depict only local information and that (2) motion words are neither interpretable nor discriminative of the action.

To overcome these shortcomings, there are several related studies in the direction of mid-level feature modeling. For videos, mid-level features are designed based on 3D regions, poselets, tracklets, and so on. Such examples are acteme [30], acton [31], motionlet [26], group of tracklet [19], etc. In [30], an acteme is defined as a volume of random size that captures a salient spatiotemporal visual pattern, represented by HOG/HOF features. In [31], different from actemes, actons are built on top of the BoF representation of each volume of interest, forming a mid-level dictionary of intermediate concepts to characterize the semantic properties. Similar to actemes, an activation vector is computed for the final classification. In [26], a greedy method is used to select the discriminative 3D regions with high motion saliency, and a spatio-temporal pyramid representation of the activation scores is used for final classification. In [19], groups of trajectories are employed to define mid-level primitives.

Similar mid-level features have also been developed for motion capture data. In [25], Wang et al. use actionlet and actionlet ensemble to represent actions. Each joint is described by the Fourier coefficients of its position values at different temporal scales. A mining algorithm is adopted to discover conjunctive structure on these joint features, which is defined as actionlet. The actionlet ensemble is then computed with Multiple Kernel Learning (MKL) [1]. Another work along this line is the pose based approach [22] by Wang et al. In that work, a skeleton sequence is first quantized using pre-learned pose dictionaries. Discriminative spatial and temporal part sets are then generated using contrast mining techniques. Actions are represented with a BoW histogram and classified by one-vs-one linear SVMs. In [4], a set of Linear Dynamic Systems (LDS) is fit to subsequences of the time series data at different spatial and temporal scales. MKL is then employed to compute the weight of the LDS representation associated with different spatial and temporal scales. In [29], the position, velocity and acceleration feature at one frame is defined as a Moving Pose feature, and a mining algorithm is adopted to compute the most discriminative Moving Pose frames. A voting scheme based on k nearest neighbors is utilized to predict the label of a test sequence. In [14], each action is modeled by a sequence of latent poses, where the pose dictionary and action/activity classifiers are jointly learned via Latent Structural SVM [28].

An important disadvantage of all these methods for building mid-level representations for action classification is that, except for [19, 4, 14], the mid-level codebook/classifier is learned separately from the action classifiers using clustering/mining techniques, which might not be discriminative for specific actions.

On the other hand, joint learning of mid-level features and classifiers has shown good performance in other visual recognition tasks. For example, Mairal et al. [16] and Boureau et al. [3] learn a sparse representation based dictionary together with the classifier for image classification. Lobel et al. [15] introduce a two-layer feature representation for image classification, in which the feature classifiers are learned jointly with object classifiers using Latent Structural SVMs [28]. Jain et al. [7] use top-down feature representation for semantic segmentation, and jointly learn the top-down feature dictionary with a Conditional Random Field model. Moreover, CNN [12] based techniques, which learn multi-layer features jointly, have been applied to video-based action recognition. Ji et al. [8] propose using a 3D CNN for video classifications. Karpathy et al. [9] apply CNNs to action recognition using 1 million videos and out-performs one-layer histogram based classification. Nonetheless, to the best of our knowledge, there is not much automatic feature learning method developed for action recognition based on motion capture data. The only one we are aware of is [5], which uses a Hierarchical Recurrent Neural Network (RNN) model. In this model, the data from each body part is used as input to its corresponding RNN model, and the generated hidden state series is used as input to the RNN model at the next layer (e.g. upper body and lower body layer, or full body layer). The output sequence at the full body layer is then fed to a fully connected layer following a softmax layer. This model shows good performance, but it is very complicated and still lacks interpretability since it contains multiple layers.

3. Our Framework

3.1. Body-Part Based Feature

Many actions can be differentiated by looking at the movement patterns associated with parts of the body. For example, a hand waving action can be recognized by detecting the 'waving' movement of the hand; a walking action can be recognized by detecting the right and left leg moving forward alternately. Furthermore, these discriminative patterns can be observed within a small temporal window, rather than the whole time series data. We thus propose to use dynamic motion features associated with a set of joints from short temporal segments as our mid-level feature descriptor.

More specifically, given a set of m_k body joints $J^k = \{j_1^k, j_2^k, \ldots, j_{m_k}^k\}$ corresponding to the kth body part, we compute the position and velocity of these joints for L consecutive frames $\{t, t+1, \ldots, t+L-1\}$, and concatenate them to form a feature $\mathbf{x}_t^k \in \mathbb{R}^{6m_kL}$ for body part k at frame t,

$$\mathbf{x}_t^k = [\mathbf{p}_{J^k}(t), \mathbf{v}_{J^k}(t), \ldots, \mathbf{p}_{J^k}(t+L-1), \mathbf{v}_{J^k}(t+L-1)], \tag{1}$$

where $\mathbf{p}_{J^k}(t)$ and $\mathbf{v}_{J^k}(t)$ denote the position and velocity for the set of joint J_k at frame t respectively, and $\mathbf{x}_t^k$ has dimension $6m_kL$ since there are m_k joints in the kth body part J^k.

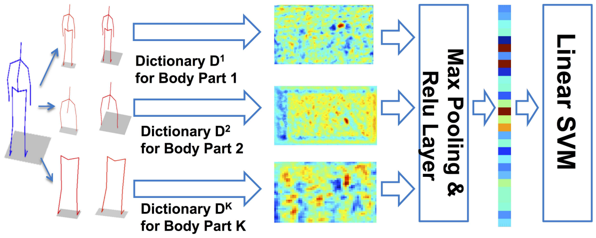


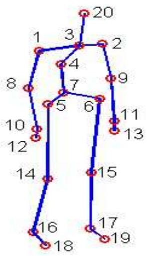
Figure 2: The feature descriptors extracted from each body part are fed into a set of dictionaries respectively to generate a set of response maps. A global feature is computed based on the response maps as input to a linear SVM for action classification.

In this work, we are interested in exploring the benefit of introducing body-part specific features. We manually select 10 body parts $\{J^k\}_{k=1}^K$, as shown in Figure 1. These body parts are selected to represent the human body hierarchy from limbs level to full body level. Ideally, one could choose to have parts at more granular levels. However, inspired by the analysis in [4], we choose to not include smaller parts to reduce the number of parameters and avoid redundancy in representation. [4] automatically learned a set of weights on the LDSs extracted from a larger set of body parts, and showed that most of the weights from smaller body parts are zero, suggesting that smaller parts might be redundant in representing body-level actions.

3.2. Action Classification with Mid-level Feature Representation

In [22], after extracting mid-level features from part-sets, a complex data mining technique is adopted to find discriminative features. In our work, instead of applying a Bag-of-Features scheme, we learn one dictionary for each body part configuration. Each dictionary atom is treated as a linear classifier for a specific body part movement pattern. Inspired by the 2D poselet work [2], we call such classifiers as *Moving Poselets* (MP), as they are descriptive of a body part configuration undergoing a certain movement. The response to these mid-level classifiers shows the similarity of the motion segment to learned feature patterns. After all response maps are computed, a max pooling step is performed to compute the final representation. Moreover, these mid-level feature classifiers are trained jointly with action classifiers to find discriminative mid-level motion patterns. This process is also shown in Figure 2.

Mathematically, given a sequence of skeleton data with T frames, we first extract the series of body-part based feature $\mathbf{X}^k = [\mathbf{x}_1^k, \mathbf{x}_2^k, \ldots, \mathbf{x}_T^k], k \in \{1, \ldots, K\}$ for each body part k. A set of mid-level feature classifiers $\mathbf{D}^k \in$



Body Part	Joint Set
Back	$J^1 = \{4:7\}$
Left Arm	$J^2 = \{3,1,8,10,12\}$
Right Arm	$J^3 = \{3,2,9,11,13\}$
Left Leg	$J^4 = \{5,14,16,18\}$
Right Leg	$J^5 = \{6,15,17,19\}$

Body Part	Joint Set
Torso	$J^6 = \{20,1:7\}$
Upper Body	$J^7 = \{20,1:4,8:13\}$
Lower Body	$J^8 = \{7,5,6,14:19\}$
Full Upper Body	$J^9 = \{20,1:13\}$
Full Body	$J^{10} = \{1:20\}$

Figure 1: Left: The skeleton model for MSR Action3D dataset [25]. Middle and Right: Mannually defined body parts for Moving Poselet feature.

$\mathbb{R}^{6m_k L \times c_k}$, c_k being the number of classifiers, is then applied to the motion pattern features to generate a response map,

$$\mathbf{h}_t^k = \mathbf{D}^{k\top}\mathbf{x}_t^k. \tag{2}$$

Notice that the features $\{\mathbf{x}_t^k\}$ have different dimensions for different body part k, thus the dictionaries also have different sizes. To compute the global representation, a max pooling step is performed over the response maps. For longer sequences, we adopt a temporal pyramid pooling structure, which decomposes a sequence into I subsequences at multiple scales. For each subsequence S_i of S, the pooled feature corresponding to body part k can be written as

$$f^{(i)}(\mathbf{X}^k;\mathbf{D}^k)[j] = \max_{t\in S_i} \mathbf{h}_t^k[j] = \max_{t\in S_i} \mathbf{D}_j^{k\top}\mathbf{x}_t^k, \tag{3}$$

where j means the jth entry, h_t^k denotes the response of the feature at frame t for body part k to its corresponding dictionary D^k.

These pooled features for different body parts and different subsequences are then concatenated to form the final global representation $F(\mathbf{X},\mathbf{D})$ of the sequence,

$$\begin{aligned} F(\mathbf{X},\mathbf{D}) = [&f^{(1)}(\mathbf{X}^1;\mathbf{D}^1),...,f^{(1)}(\mathbf{X}^K;\mathbf{D}^K),\\ &\ldots,\\ &f^{(I)}(\mathbf{X}^1;\mathbf{D}^1),...,f^{(I)}(\mathbf{X}^K;\mathbf{D}^K)],\\ &\mathbf{X} = \{\mathbf{X}^k\}_{k=1}^K, \mathbf{D} = \{\mathbf{D}^k\}_{k=1}^K. \end{aligned} \tag{4}$$

To classify the action label, this vector $F(\mathbf{X},\mathbf{D})$ is first passed through a rectified linear unit (ReLU) and then fed to action classifiers $\{W_q, b_q\}_{q=1}^Q$. The classification result y is given by:

$$\hat{F}(\mathbf{X},\mathbf{D}) = ReLU(F(\mathbf{X},\mathbf{D})) = \max(F(\mathbf{X},\mathbf{D}),0), \tag{5}$$

$$y = \arg\max_q W_q^\top \hat{F}(\mathbf{X},\mathbf{D}) + b_q, \tag{6}$$

where $\{W_q, b_q\}$ is the linear classifier corresponding to label q.

3.3. Relation with CNN

Our proposed model can be viewed as a variation of a two-layer CNN model. However, there are three major differences. First, we don't assume that the input time series are of fixed size. Instead, we use max pooling at the top layer to generate a fixed dimensional feature to represent each action. This gives more flexibility to process time series data. Secondly, we have one set of feature classifier per body part configuration. This helps us to mine the discriminative movements associated with each body part. Thirdly, we use temporal pyramid representation for long sequences. Our model is thus more specifically designed for modeling action with human skeleton data.

4. Learning

Given training data of N sequences $\{\mathbf{X}^{(n)}\}_{n=1}^N$ and their action labels $\{y^{(n)} \in \{1,\ldots,Q\}\}_{n=1}^N$, we aim to learn the set of dictionaries $\mathbf{D}$ jointly with the action classifiers W and b. The optimization problem is formulated as follows,

$$\begin{aligned} \min_{\mathbf{D},\{W_q\}_{q=1}^Q,\{b_q\}_{q=1}^Q} \sum_{q=1}^Q\sum_{n=1}^N L(Y_{qn}, W_q^\top \hat{F}(\mathbf{X}^{(n)},\mathbf{D}) + b_q) +\\ \frac{\lambda}{2}(\sum_{k=1}^K \|\mathbf{D}^k\|_F^2 + \sum_{q=1}^Q \|W_q\|_F^2), \end{aligned} \tag{7}$$

where the loss function

$$\begin{aligned} L(Y,&W_q^\top \hat{F}(\mathbf{X}^{(n)},\mathbf{D}) + b_q) =\\ &\max(0, 1 - Y_{qn}(W_q^\top \hat{F}(\mathbf{X}^{(n)},\mathbf{D}) + b_q)). \end{aligned} \tag{8}$$

The loss function $L(\cdot)$ is the standard hinge loss function, with Y_{qn} denoting the binary indicator of sample $\mathbf{X}^{(n)}$ having label q. The regularization term contains both regularization for action classifiers W and mid-level dictionaries $\mathbf{D}$.

We adopt a mini-batch stochastic gradient descent algorithm to solve the optimization problem. The gradients of action classifiers can be computed similarly as in standard SVM training, while the gradients with respect to $\mathbf{D}$ can be achieved by the back-propagation algorithm commonly used in CNN learning [12]. More specifically, the gradient with respect to the jth classifier for kth body part, i.e. $\mathbf{D}_j^k$, for a mini-batch B can be written as:

$$\begin{aligned}\mathbf{g}_{\mathbf{D_j^k}}(B) &= \sum_{n \in B} \sum_{q=1}^{Q} \frac{\partial L}{\partial \hat{F}(\mathbf{X}^{(n)}, \mathbf{D})} \cdot \frac{\partial \hat{F}(\mathbf{X}^{(n)}, \mathbf{D})}{\partial \mathbf{D}_j^k} + \lambda \mathbf{D}_j^k \\ &= \sum_{n \in B} \sum_{q=1}^{Q} \delta(1 - Y_{qn}(W_q^\top \hat{F}(\mathbf{X}^{(n)}, \mathbf{D}) + b_q) > 0) \\ &\quad * \delta(\hat{F}(\mathbf{X}^{(n)}, \mathbf{D})[z_{kj}] > 0) \cdot W_q[z_{kj}] \cdot \mathbf{x}_{t_{kj}^{(n)}}^{k(n)} + \lambda \mathbf{D}_j^k, \end{aligned} \tag{9}$$

where z_{kj} denotes the corresponding entry of the classifier response of $\mathbf{D_j^k}$ in the global feature $\hat{F}(\mathbf{X}, \mathbf{D})$, and $t_{kj}^{(n)}$ denotes the frame index of the MP feature in nth sample that gives the max value at entry j for kth body part. During training, we use a step decay strategy to anneal the learning rate. We start from a small learning rate τ_0 and then reduce it by factor γ for every T_e epochs.

5. Experiments

5.1. Datasets

We validate our algorithm on the MSR Action3D [13], MSR DailyActivity3D [25] and Berkeley MHAD [17] datasets, which are commonly used datasets for action recognition from skeleton data. The MSR Action3D dataset consists of skeleton data sequences of 20 actions such as *hand waving* and *clapping*. Each action is performed 2-3 times by 10 subjects, and the 3D body joint positions of 20 joints are extracted from RGB-D videos. These action sequences are relatively short sequences with 30-50 frames, and the frame rate is 15 frames per second. We conduct two set of experiments, following the experimental setup in [25] and [13] respectively. In Setup 1, all sequences from subjects 1, 3, 5, 7 and 9 are used for training and the remaining ones for testing. In Setup 2, the dataset is divided into three action sets, AS1, AS2 and AS3, and the same algorithm is tested on each of the three sets.

The MSR DailyActivity3D dataset consists of 16 daily activities such as *drinking* and *reading books*. Each action is performed twice by 10 subjects, making up 320 sequences in total. This dataset has longer sequences, with 100-300 frames. The skeleton data also contains 3D positions of the same 20 joints extracted from RGB-D videos. It is more challenging than MSR Action3D, since the actions are more complex, and contain human-object interactions. Also following [25], we use the sequences from subject 1, 3, 5, 7 and 9 for training, and remaining ones for testing.

The Berkeley MHAD dataset consists of 11 actions such as *jumping* and *clapping*. Each action is performed by 12 subjects with 5 repetitions, making up 659 sequences in total. The skeleton data is obtained via a motion capture system. It contains 3D positions of 35 joints and has a frame rate of 480 fps. Following [4], we use sequences from the first 7 subjects for training, and the remaining ones for testing.

5.2. Implementation Details

Data Preprocessing. Before computing MP features, we first normalize the skeleton data according to Algorithm 1 described in [29]. The raw skeleton joint positions are normalized so that the limbs (skeleton segments) have same lengths as a template skeleton model, while the joint angles are not modified. The hip center joint position is then subtracted from the skeleton data so that all sequences are centered at the origin. Following [29], after extracting velocity features at every frame, we normalize them to unit norm and scale them by a weight α. This weight is set according to the best value in [29], which is 0.75 for MSR Action3D, and 0.6 for MSR DailyActivity3D, and we also choose 0.6 for Berkeley MHAD. For Berkeley MHAD dataset, since the data has a high frame rate, we subsample each sequence at every 16 frames.

Temporal Pyramid. Since MSR Action3D consists of shorter sequences with simple actions, we set the pyramid level to be 1, i.e., the feature is max-pooled over the whole sequence. For MSR DailyActivity3D, which contains more complex actions, we set the pyramid level to be 3, and compute features pooled from 7 subsequences. For Berkeley MHAD dataset, we also set pyramid level to 1.

Optimization. In the stochastic gradient descent (SGD) algorithm, we use a mini-batch of size 10. The initial learning rate τ_0 is set to be 0.05 and it is reduced by a factor $\gamma = 0.5$ for every $T_e = 50$ epochs. The regularization term λ is set to $1e^{-4}$. To initialize $\mathbf{D}$ and $\mathbf{W}$, each entry is randomly sampled from a uniform distribution of $[-1, 1]$, and each atom/classifier is then scaled by a factor of $\frac{1}{\sqrt{d}}$, where d is the dimension of the corresponding vector. The bias term is initialized as 0. Due to the randomness in SGD optimization, for each set of parameters, we run 10 repetitions of the same experiment and report the mean accuracy and standard deviation in our results section.

5.3. Results

We first compare our approach with other state-of-the-art skeleton-based action recognition methods. In this case, we use 10 body parts, and 50 mid-level feature classifiers for each body part.

The performance on MSR Action3D under two experiment setups is shown in Table 1 and Table 2. The number in bracket is the standard deviation. We can see that our

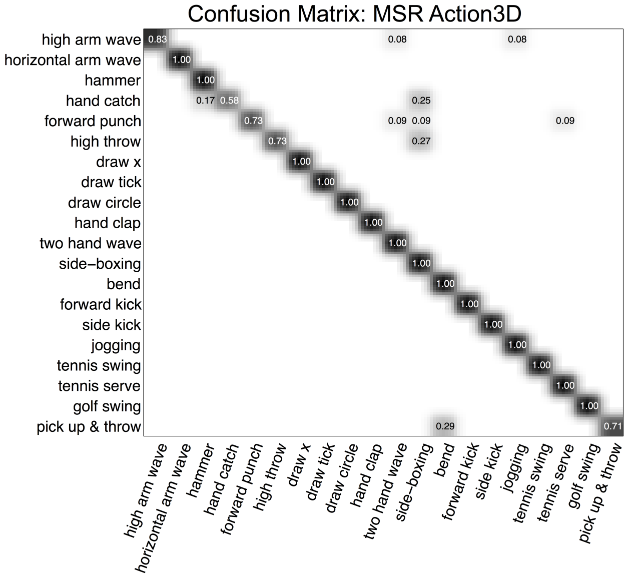


Figure 3: Confusion Matrix for MSR Action3D Dataset

Table 1: Action Classification Accuracy on MSR Action3D (setup 1)

Method	Accuracy
Actionlet Ensemble[25]	88.2
Lie Group [21]	89.5
Hierarchical LDS [4]	90.2
Pose Base Approach[22]	90.2
Moving Pose [29]	91.7
[6]	91.5
Moving Poselets (Ours)	**93.6** (0.24)

Table 2: Action Classification Accuracy on MSR Action3D (setup 2)

Method	AS1	AS2	AS3	avg
Bag of 3D Points [13]	72.9	71.9	79.2	74.7
Lie Group[21]	95.29	83.87	98.22	92.46
HRNN (HURNN-L) [5]	92.38	93.75	94.59	93.57
HRNN (HBRNN-L) [5]	93.33	94.64	95.50	94.49
Moving Poselets (Ourts)	89.81	93.57	97.03	93.50

Table 3: Action Classification Accuracy on MSR DailyActivity3D

Method	Accuracy
Actionlet Ensemble [25]	68.0
Moving Pose [29]	73.8
[6]	73.1
Ours	**74.5** (1.43)

Table 4: Action Classification Accuracy on Berkeley MHAD

Method	Accuracy
SMIJ [18]	95.37
Hierarchical LDS [4]	100
HURNN-L [5]	99.64
HBRNN-L [5]	100
Moving Poselets (Ours)	**100**

Moving Poselet approach achieves 93.6% mean accuracy for Setup 1 on this dataset, while the Moving Pose [29], which uses similar features (position, velocity and acceleration of the full body at single frame) only gives 91.7%. This suggests that the feature representation learned from our method is more discriminative. Figure 3 presents the confusion matrix from one repetition of the experiments under this setup. We can observe that this approach achieves 100% accuracy on 15 out of 20 action classes. There is confusion between *hand catch, forward punch, high throw* and *side boxing*. This is expected, since they all involve hand movement. Also, the action *pick up* is confused with *bending* since *pick up* also involves a bending action.

For Setup 2, our method achieves comparable results to the state-of-art HRNN based method [5]. Note that the HURNN-L version is the HRNN model with unidirectional RNNs, while the HBRNN-L version uses a hierarchy of bidirectional RNNs. We can see that our simple two-layer model generates similar result as the very complicated HURNN-L model, while its performance is only 1% less than that of the HBRNN-L model.

On the MSR DailyActivity3D dataset (see Table 3), our Moving Poselets approach achieves a mean accuracy of 74.5%, outperforming other state-of-the-art methods. However, only 5 out of 16 action classes are classified with 100% accuracy on this dataset, as it's more challenging then MSR Action3D. Another observation is that the actions *eat, read book, call cellphone, write on a paper, use laptop* are usually confused with each other, since they all involve human manipulating some object that is close to his face.

On the Berkeley MHAD dataset (see Table. 4), our Moving Posetlets approach achieves 100% accuracy, which is much higher than the performance on the previous two datasets. Our conjecture is that since the skeleton data in this dataset is obtained through motion capture system, it is less noisy than the skeleton data extracted from RGB-D videos, and thus easier to be classified. For comparison with other methods, the HURNN-L model gives 99.64% accuracy, and the Hierachical LDS and HBRNN-L models both give 100% accuracy. This suggests that our two-layer feature learning framework works as well as or better than multilayer HRNN models.

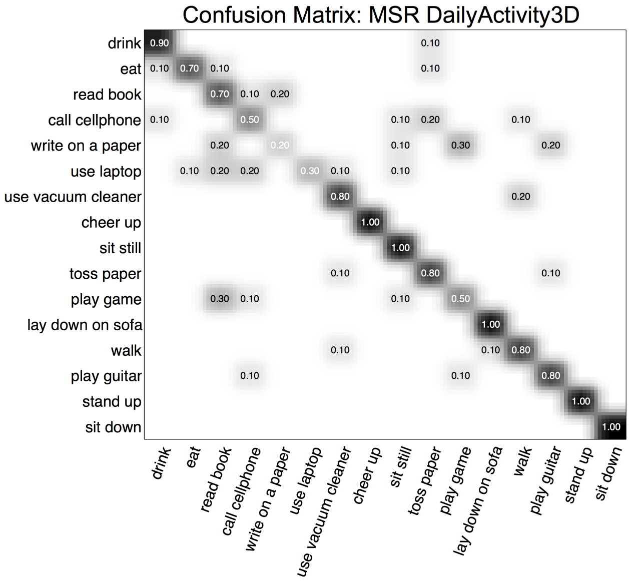


Figure 4: Confusion Matrix for MSR DailyActivity3D Dataset

5.4. Analysis

Importance of Body Part Structure. The intuition behind our Moving Poselet feature is that the movement patterns associated with a specific body part are discriminative for recognizing actions. To validate this, we run experiments using only the Moving Poselet feature from the full body for comparison. More specifically, instead of having 50 mid-level classifiers per body part for 10 body parts, we use 500 classifiers for features only extracted from the full body. The performance does not change much on the less challenging MSR Action3D (93.6%) and Berkeley MHAD (100%) datasets, but it drops significantly on the more challenging MSR DailyActivity3D dataset (70.8%). This suggests that exploring the discriminative body parts is very important for recognizing human actions.

Comparison with Bag-of-Words Based Models. To show the importance of jointly learning mid-level features and action classifiers, we run experiments that compare with the Bag-of-Words model. In this Bag-of-Words model, the dictionary is trained via K-means using the Moving Poselet features extracted at every frame. We use a dictionary of size 500 and each video is represented by the aggregated histogram with the same temporal pyramid. The performance of this model on MSR DailyActivity3D is 60.6% with linear SVM, while our method gives 74.5% average accuracy. This suggests that performing feature learning can help improve classification performance.

Size of Mid-level Classifiers. To understand the effect of the size of mid-level classifiers, we run the same experiment with the number of mid-level classifiers set to 100, 250, 500, and 800 (10, 25, 50, 80 per body part, respectively). The performance is given in Table 5. The results suggest that for Berkeley MHAD dataset, the performance does not change and the accuracy is always 100%. Our conjecture is that since this dataset is obtained from motion capture system, it is less noisy and easier to classify comparing with the other two datasets, and thus gives 100% accuracy using our model. For the other two datasets, the performance is better with larger size of mid-level classifiers. But the improvement starts to converge when the number of classifiers reaches 500. For MSR DailyActivity3D, the accuracy even starts to go down when the size is bigger than 500. Our conjecture is that since this dataset only contains 320 samples, using large number of classifiers leads to overfitting and could affect the performance.

Table 5: Performance Using Different Number of Mid-level Classifiers on Three Datasets

	MSR Action3D	MSR DailyActivity3D	MHAD
100	92.2 (0.69)	72.2 (1.85)	100
250	93.0 (0.48)	73.3 (1.37)	100
500	**93.6** (0.24)	**74.5** (1.43)	100
800	**93.6** (0.17)	73.3 (1.25)	100

Effect of ReLU. To evaluate the contribution of the ReLU layer, we perform the same experiment using 500 mid-level classifiers, but with the ReLU layer removed. Similarly, for the less challenging Berkeley MHAD dataset, the performance is the same (100%). For MSR Action3D dataset, the accuracy is slightly worse (92.8% versus 93.6%). For MSR DailyActivity3D, removing the ReLU layer leads to a dramatic decrease in performance (66.2% versus 74.5%). This suggests that for challenging datasets with complex structures and small amount of training data, adding a ReLU layer leads to better performance.

Visualization of Discriminative Moving Poselets. To visualize the discriminative features, we first select the mid-level classifiers corresponding to the top 5 highest weights in W_q for each action class q. From training data, we find the $L-$frame segment that gives the highest response to each classifier. The selected Moving Poselet segments for the top 5 mid-level classifiers per action are visualized in Figure 5. We can see that our algorithm is able to automatically select the discriminative body parts and their movements. For example, for the *high arm wave* action, the algorithm selects movements corresponding to upper body or right arm; for the *side kick* action, the algorithm selects movements corresponding to the right leg.

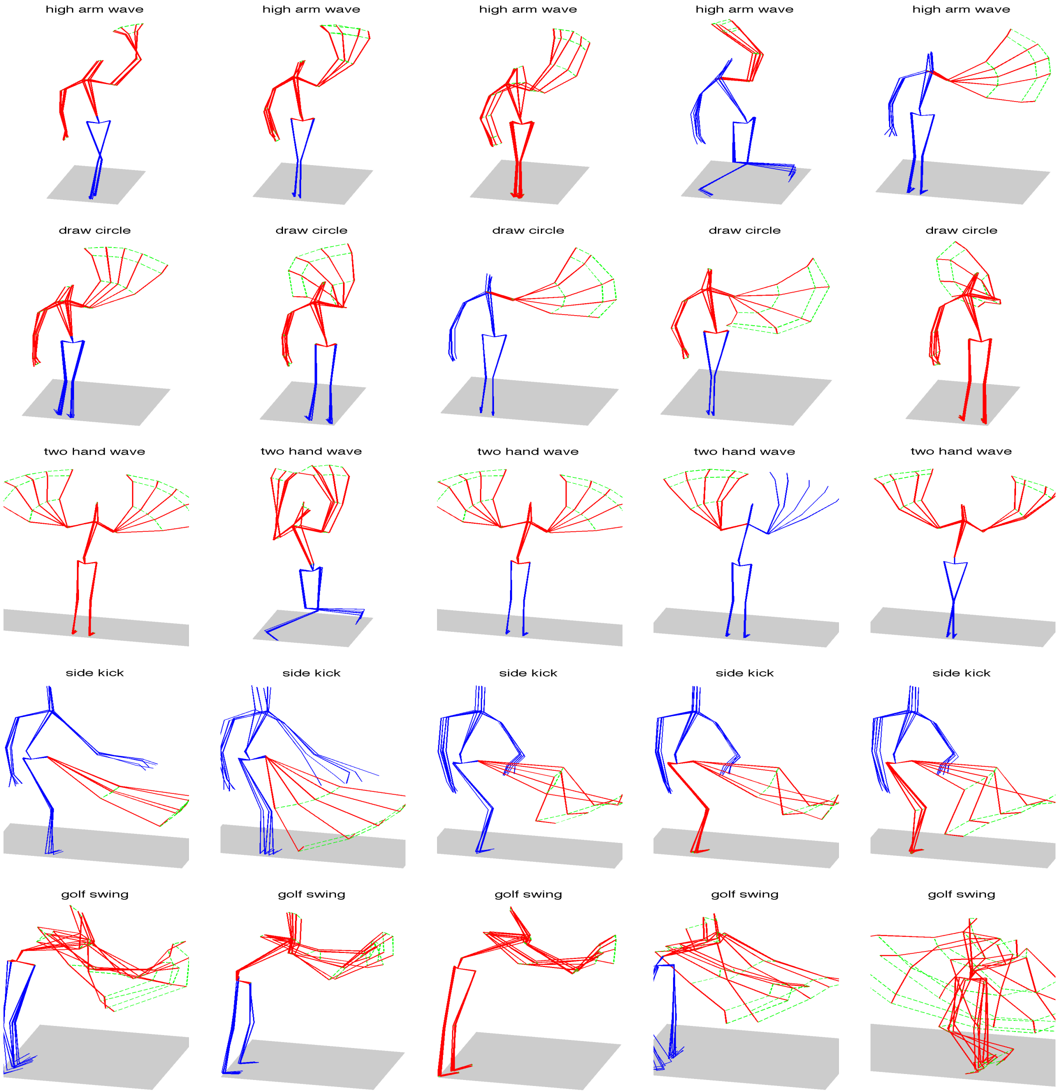


Figure 5: Moving Poselet segments that give highest response to the top 5 mid-level classifiers for each action. The actions (from top to bottom) are: *high arm wave, draw circle, two hand wave, side kick, golf swing*. The red line indicates the corresponding body parts, while the green dashed line shows the trajectory of the joints from selected body parts.

6. Conclusion and Future Work

We have proposed a novel Moving Poselet based mid-level feature learning method for action recognition using skeleton data. The results showed that by jointly learning the feature representation and action classifiers, and exploring discriminative body part movement for actions, our algorithm outperformed state-of-the-art methods. Our current work uses manually selected body parts for a fixed temporal scale (a few frames). For future work, we are interested in extending our work to automatically select body part configurations and temporal scales. We are also interested in its extension in video data.

7. Acknowledgement

We thank Colin Lea for useful discussions. This work is supported by NSF 1218709 and ONR N000141310116.

References

[1] F. Bach, G. Lanckriet, and M. Jordan. Multiple kernel learning, conic duality, and the SMO algorithm. In *Proceedings of International Conference on Machine Learning*, 2004. 2

[2] L. Bourdev and J. Malik. Poselets: Body part detectors trained using 3d human pose annotations. In *IEEE International Conference on Computer Vision*, pages 1365–1372, 2009. 3

[3] Y.-L. Boureau, F. Bach, Y. LeCun, and J. Ponce. Learning mid-level features for recognition. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2010. 3

[4] R. Chaudhry, F. Ofli, G. Kurillo, R. Bajcsy, and R. Vidal. Bio-inspired dynamic 3D discriminative skeletal features for human action recognition. In *International Workshop on Human Activity Understanding from 3D Data*, 2013. 2, 3, 5, 6

[5] Y. Du, W. Wang, and L. Wang. Hierarchical recurrent neural network for skeleton based action recognition. June 2015. 3, 6

[6] A. Eweiwi, F. Cheema, C. Bauckhage, and J. Gall. Efficient pose-based action recognition. In *Asian Conference on Computer Vision*, 2014. 6

[7] A. Jain, L. Zappella, P. McClure, and R. Vidal. Visual dictionary learning for joint object categorization and segmentation. In *European Conference on Computer Vision*, 2012. 3

[8] S. Ji, W. Xu, M. Yang, and K. Yu. 3d convolutional neural networks for human action recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(1):221–231, Jan 2013. 3

[9] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei. Large-scale video classification with convolutional neural networks. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2014. 3

[10] I. Laptev. On space-time interest points. *International Journal of Computer Vision*, 64(2-3):107–123, 2005. 1, 2

[11] I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld. Learning realistic human actions from movies. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2008. 2

[12] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. In *Proceedings of the IEEE*, pages 2278 – 2324, 1998. 2, 3, 5

[13] 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, June 2010. 5, 6

[14] I. Lillo, A.Soto, and J.C.Niebles. Discriminative hierarchical modeling of spatio-temporally composable human activities. 2014. 2

[15] H.-A. Lobel, A. Soto, and R. Vidal. Hierarchical joint max-margin learning of mid and top level representations for visual recognition. In *IEEE International Conference on Computer Vision*, 2013. 3

[16] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman. Discriminative learned dictionaries for local image analysis. *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2008. 3

[17] F. Ofli, R. Chaudhry, G. Kurillo, R. Vidal, and R. Bajcsy. Berkeley MHAD: A comprehensive multimodal human action database. In *IEEE Workshop on Applications of Computer Vision*, 2013. 5

[18] 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. *Journal of Visual Communication and Image Representation*, 25(1):24–38, 2014. 6

[19] M. Raptis, I. Kokkinos, and S. Soatto. Discovering discriminative action parts from mid-level video representations. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2012. 2

[20] 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 *IEEE Conference on Computer Vision and Pattern Recognition*, 2011. 1

[21] R. Vemulapalli, F. Arrate, and R. Chellappa. Human action recognition by representing 3d skeletons as points in a lie group. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2014. 6

[22] C. Wang, Y. Wang, and A. Yuille. An approach to pose-based action recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2013. 2, 3, 6

[23] H. Wang, A. Klaser, C. Schmid, and C. Liu. Action recognition by dense trajectories. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2011. 2

[24] H. Wang, A. Klaser, C. Schmid, and C. Liu. Dense trajectories and motion boundary descriptors for action recognition. *International Journal of Computer Vision*, 103(1):60–79, 2013. 2

[25] J. Wang, Z. Liu, Y. Wu, and J. Yuan. Mining actionlet ensemble for action recognition with depth cameras. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2012. 2, 4, 5, 6

[26] L. Wang, Y. Qiao, and X. Tang. Motionlets: Mid-level 3d parts for human motion recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2013. 2

[27] L. Xia, C.-C. Chen, and J. Aggarwal. View invariant human action recognition using histograms of 3d joints. In *International Workshop on Human Activity Understanding from 3D Data*, 2012. 2

[28] C.-N. J. Yu and T. Joachims. Learning structural svms with latent variables. In *International Conference on Machine Learning (ICML)*, 2009. 2, 3

[29] M. Zanfir, M. Leordeanu, and C. Sminchisescu. The moving pose: An efficient 3d kinematics descriptor for low-latency action recognition and detection. In *IEEE International Conference on Computer Vision*, 2013. 2, 5, 6

[30] W. Zhang, M. Zhu, and K. Derpanis. From actemes to action: A strongly-supervised representation for detailed action understanding. In *IEEE International Conference on Computer Vision*, 2013. 2

[31] J. Zhu, B. Wang, X. Yang, W. Zhang, and Z. Tu. Action recognition with actons. In *IEEE International Conference on Computer Vision*, 2013. 2