Recognizing Human Actions as the Evolution of Pose Estimation Maps

Mengyuan Liu[1] Junsong Yuan[2,*]
[1] School of Electrical and Electronic Engineering
Nanyang Technological University, Singapore 639798
[2] Department of CSE, SUNY at Buffalo, Buffalo NY 14260
liumengyuan@ntu.edu.sg jsyuan@buffalo.edu

Abstract

Most video-based action recognition approaches choose to extract features from the whole video to recognize actions. The cluttered background and non-action motions limit the performances of these methods, since they lack the explicit modeling of human body movements. With recent advances of human pose estimation, this work presents a novel method to recognize human action as the evolution of pose estimation maps. Instead of relying on the inaccurate human poses estimated from videos, we observe that pose estimation maps, the byproduct of pose estimation, preserve richer cues of human body to benefit action recognition. Specifically, the evolution of pose estimation maps can be decomposed as an evolution of heatmaps, e.g., probabilistic maps, and an evolution of estimated 2D human poses, which denote the changes of body shape and body pose, respectively. Considering the sparse property of heatmap, we develop spatial rank pooling to aggregate the evolution of heatmaps as a body shape evolution image. As body shape evolution image does not differentiate body parts, we design body guided sampling to aggregate the evolution of poses as a body pose evolution image. The complementary properties between both types of images are explored by deep convolutional neural networks to predict action label. Experiments on NTU RGB+D, UTD-MHAD and PennAction datasets verify the effectiveness of our method, which outperforms most state-of-the-art methods.

1. Introduction

1.1. Motivation and Objective

Human action recognition from videos has been researched for decades, since this task enjoys various applications in intelligent surveillance, human-robot interaction and content-based video retrieval. The intrinsic property of existing methods [22, 43, 37, 24, 1] is to learn mapping functions which transform videos to action labels. Since they do not directly distinguish human body from videos, these methods are easily affected by clutters and non-action motions from backgrounds.

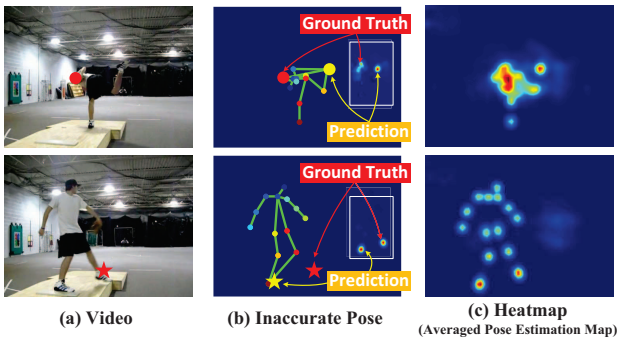

Figure 1: An illustration of the complementary property between poses and heatmaps (averaged pose estimation maps), which are both estimated from video frames. (a) An action "baseball pitch" from PennAction dataset [54] is simplified as two frames. The red circle and red star denote the hand and foot, respectively. (b) With inaccurate pose estimation, the estimated poses cannot accurately annotate human body parts. For example, we show the pose estimation map of the hand, where the multiple peaks lead to false prediction. (c) Although heatmaps cannot differentiate body parts, they provide richer information to reflect human body shape.

To address this limitation, an alternative solution is to detect human [39] and estimate the body pose in each frame. This approach works well in the field of human action recognition from depth videos, e.g., Microsoft Kinect [55, 27]. By detecting 3D pose from each depth frame with an accurate body pose estimation method [36], human movements in depth videos can be simplified as 3D pose sequences [52]. Recent deep learning models, e.g., CNN [17, 20], RNN [9] and LSTM [26, 25], have achieved high performances on the extracted 3D poses, which outperform methods [32, 50] that rely on raw depth video sequences.

The success of 3D human pose inspires us to estimate 2D human poses from videos for action recognition. However, despite the significant advances of 2D pose estima-

*Corresponding author

tion in images and videos [51, 5, 46, 2, 4], the performance is still inferior to the 3D pose estimation in depth videos. Fig. 1 illustrates the estimated poses from video frames by a state-of-the-art pose estimation method [4]. Due to complex background and self-occlusion of human body parts, the estimated poses are not fully reliable and may misinterpret the configuration of human body. In the first row of Fig. 1 (b), the multi-modal pose estimation map in the white bounding box indicates the location of the person's hand. The map contains two peaks, where the ground truth location does not correspond to the highest peak, thus provides a wrong estimation of the hand's location.

To better utilize the pose estimation maps, instead of relying on the inaccurate 2D pose estimated from the pose estimation maps, we propose to directly model the evolution of pose estimation maps for action recognition. In Fig. 1 (c), heatmaps (averaged pose estimation maps) provide richer information to reflect human body shape.

1.2. Method Overview and Contributions

Our method is shown in Fig. 2. Given each frame of a video, we use convolutional pose machines to predict pose estimation map for each body part. The goal of representing these pose estimation maps is to preserve both global cues, which reflect whole shapes that suffer less from the noise and local cues, which detail the locations of body parts.

To this end, we average pose estimation maps of all body parts to generate an averaged pose estimation map (heatmap) for each frame. The temporal evolution of heatmaps can reflect the movements of body shape. Different from the original RGB image, the heatmap is sparse. Considering the huge spatial redundancy, we develop a spatial rank pooling method to compress the heatmap as a compact yet informative feature vector. The merit of spatial rank pooling is that it can effectively suppress spatial redundancy, without significantly losing spatial distribution information of the heatmap. The temporal concatenation of feature vectors constructs a 2D body shape evolution image, which reflects the temporal evolution of body shapes.

As body shape evolution image cannot differentiate body parts, we further predict joint location from pose estimation map of each body part, generating a pose for each frame. Since the number of estimated pose joints is limited, we use body structure to guide the sampling of more abundant pose joints to represent human body. The temporal concatenation of all pose joints constructs a body pose evolution image, which reflects the temporal evolution of body parts. Intuitively, the body shape evolution image and body pose evolution image benefit the recognition of general movements of body shape and elaborate movements of body parts. Thereby, both images are explored by CNNs to generate discriminative features, which are late fused to predict action label. Generally, our contributions are three-fold.

- Given inaccurate 2D poses estimated from videos, we boost the performance of human action recognition by recognizing actions as the evolution of pose estimation maps instead of the unreliable 2D body poses.
- The evolution of pose estimation maps are described as body shape evolution image and body pose evolution image, which capture the movements of both whole body and specific body parts in a compact way.
- With CNNs and late fusion scheme, our method achieves state-of-the-art performances on NTU RGB+D, UTD-MHAD and PennAction datasets.

2. Related Work

2.1. 3D Pose-based Action Recognition

3D pose provides direct physical interpretation for human actions from depth videos. Hand-crafted features [42, 47, 13] were designed for describing evolution of 3D poses. Recently, deep neural networks were introduced to model the spatial structures and temporal dynamics of poses. For example, Du *et al.* [9] firstly used hierarchical RNN for pose-based action recognition. Liu *et al.* [25] extended this idea and proposed spatio-temporal LSTM to learning spatial and temporal domains. To enhance the attention capability of LSTM, Global Context-Aware Attention LSTM [26] was developed with the assistance of global context.

2.2. Video-based Action Recognition

Local features are motion-related and are robust to cluttered background to some extent. Spatial temporal interest points (STIPs) [22] and dense trajectory [43] were applied to extract and describe local spatial temporal patterns. Based on these basic features, multi-feature max-margin hierarchical Bayesian model [49] and a novel feature enhancing technique called Multi-skIp Feature Stacking [21] were proposed to learn more distinctive features. Since local features ignore global relationships, holistic features were encoded by two-stream convolutional network [37], which learns spatial-temporal features by fusing convolutional networks spatially and temporally. Based on this network, the relationships between the spatial and temporal structures were further explored [11, 45]. Different from two-stream network, the spatial and temporal information of actions can be fused before they are input to CNNs. Fernando *et al.* [12] proposed rank pooling method to aggregate all video frames to a compact representation. Bilen *et al.* [1] deeply merged rank pooling method with CNN to generate an efficient dynamic image network.

Human actions are inherently structured patterns of body movements. Recent studies [56, 31, 14, 38, 30] extracted whole human body or body parts instead of whole video for action analysis. Meanwhile, human action recognition and

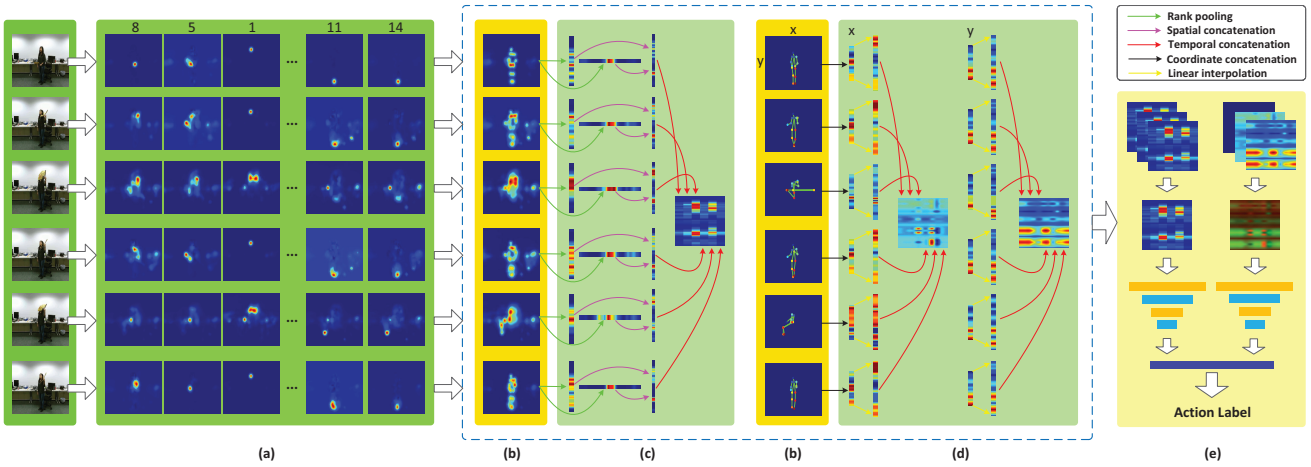

Figure 2: The overview of the proposed method. a) Convolutional pose machines predict pose estimation map of each body part. b) For each frame, pose estimation maps are aggregated to form a heatmap and a pose. c) Spatial rank pooling is proposed to describe the evolution of heatmaps as a body shape evolution image, which contains one channel. d) Body guided sampling is proposed to describe the evolution of poses as a body pose evolution image, which contains two channels. e) Deep features are extracted from both types of images and the late fusion result predicts action label. Note that both images are normalized to fix-sized color images to facilitate transfer learning.

pose estimation tasks have been integrated to extract pose guided features for recognition. Wang *et al.* [41] improved an existing pose estimation method, and then designed pose features to capture both spatial and temporal configurations of body parts. Xiaohan *et al.* [48] proposed a framework to integrate training and testing of action recognition and pose estimation. They decomposed actions into poses which are further divided to mid-level ST-parts and then parts. Most recently, Du *et al.* [8] proposed an end-to-end recurrent network which can exploit important spatial-temporal evolutions of human pose to assist action recognition in a unified framework. Different from pose features [41] or pose-guided color features [48, 8], this paper recognizes human actions from only pose estimation maps, which have not been explored for action recognition task before.

3. Generation of Pose Estimation Maps

This section predicts pose estimation maps from each frame of a video (Fig. 3 (a)), and then generates a heatmap (Fig. 3 (b)) and a pose (Fig. 3 (c)) to denote each frame.

Pose Estimation Maps: The task of human pose estimation from a single image can be modeled as a structure prediction problem. In [34], a pose machine is proposed to sequentially predict pose estimation maps for body joints, where previous predicted pose estimation maps iteratively improve the estimates in following stages. Let $\mathcal{Y}_k \in \{x, y\}$ denote the set of coordinates from body joint k. The structural output can be formulated as $\mathcal{Y} = \{\mathcal{Y}_1, ..., \mathcal{Y}_k, ..., \mathcal{Y}_K\}$, where K is the total number of body joints. Multi-class classifier g_t^k is trained to predict the k_{th} body joint in the t_{th} stage. For an image position $\mathbf{z}$, the pose estimation map for assigning it to the k_{th} body joint is formulated as:

$$\mathbf{B}_t^k(\mathcal{Y}_k = \mathbf{z}) = g_t^k\left(\mathbf{f}_\mathbf{z}; \bigcup_{i=1,...,K} \psi(\mathbf{z}, \mathbf{B}_{t-1}^i)\right), \quad (1)$$

where $\mathbf{f}_\mathbf{z}$ is the color feature at position $\mathbf{z}$, $\mathbf{B}_{t-1}^i$ is the pose estimation map predicted by g_{t-1}^i, $\cup$ is the operator for vector concatenation, ψ is the feature function for computing contextual features from previous pose estimation maps. After T stages, the generated pose estimation maps are used to predict locations of body joints. The pose machine [34] uses boosted classifier with random forests for the weak learners. Instead, this paper applies the convolutional pose machine [46, 4] to combine pose machine with convolutional architectures, which does not need graphical-model style inference and boosts the performances of pose machine.

Heatmaps & Poses: For the n_{th} frame of a video, K types of pose estimation maps, namely $\{\mathbf{B}_T^{1,n}, ..., \mathbf{B}_T^{K,n}\}$, are generated. To reduce the redundancy of pose estimation maps, we describe them as a heatmap $\mathbf{G}_n$ and a pose $\mathcal{L}_n$. The heatmap $\mathbf{G}_n$ can be expressed as:

$$\mathbf{G}_n = \frac{1}{K}\sum_{k=1}^{K} \mathbf{B}_T^{k,n}, \quad (2)$$

which reflects the global body shape. The pose $\mathcal{L}_n$ can be expressed as $\{\mathbf{z}^{k,n}\}_{k=1}^K$, where $\mathbf{z}^{k,n}$ is often estimated via Maximum A Posterior (MAP) criterion [4]:

$$\mathbf{z}^{k,n} = \underset{\mathbf{z}\in\mathcal{Z}}{arg\,max}\left\{\mathbf{B}_T^{k,n}(\mathcal{Y}_k = \mathbf{z})\right\}, \quad (3)$$

where $\mathcal{Z} \in \mathbb{R}^2$ denote all positions on the image. Till now, each frame of a video is described as a heatmap and a pose. In other words, the video is converted to the evolution of heatmaps and the evolution of poses.

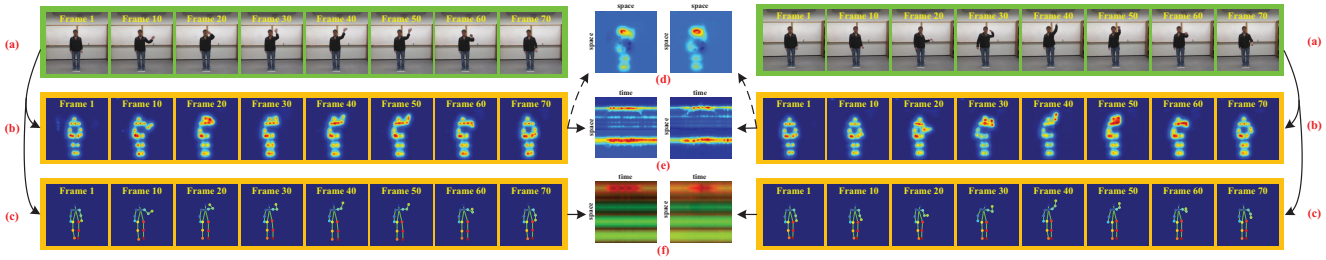

Figure 3: The comparison between features extracted from two videos. (a) Left video denotes action "wave" and right video denotes action "throw". (b) The evolution of heatmaps. Each heatmap is a gray scale image. To facilitate observation, we use colormap to highlight heatmap according to gray scale values. (c) The evolution of poses. Each joint is shown using specific color. To facilitate the observation, we also show the limbs which are colored in green. (d) Body shape evolution image implemented by temporal rank pooling (t-rk). (e) body shape evolution image implemented by spatial rank pooling (s-rk). (f) Body pose evolution image implemented by body guided sampling.

4. Evolution of Pose Estimation Maps

This section describes the evolution of heatmaps as a body shape evolution image using temporal rank pooling (Fig. 3 (d)), based on which spatial rank pooling (Fig. 3 (e)) is developed. Further, body guided sampling is developed to describe the evolution of poses as a body pose evolution image (Fig. 3 (f)). The complementary properties between two images are learned by CNNs.

Temporal Rank Pooling: As a robust and compact video representation method, temporal rank pooling [12, 1] has the ability to aggregate the temporal relevant information throughout a video via a learning to rank approach. The encoded temporal information denotes the temporal order among frames, which is a robust feature showing less sensitive to different types of input data. As heatmaps are distinct from natural images, we treat heatmaps as a new type of data and apply temporal rank pooling to encode the evolution of heatmaps. Suppose a sequence $\mathcal{V}_{\mathbf{G}} = \{\mathbf{G}_1, ..., \mathbf{G}_n, ..., \mathbf{G}_N\}$ contains N heatmaps, and $\mathbf{G}_n \in \mathbb{R}^{P\times Q}$ denotes the n_{th} frame with P rows and Q columns. $\mathbf{G}_{1:n}$ can be mapped to a vector defined as:

$$\mathbf{v}_n = V\big(\frac{1}{n}\sum_{i=1}^{n}\mathbf{G}_i\big), \tag{4}$$

where the function V reshapes a matrix into a vector and $\mathbf{v}_n \in \mathbb{R}^{(P\cdot Q)\times 1}$. Let $\mathbf{v}_{n+1} \succ \mathbf{v}_n$ denote the temporal ordering relationship between $\mathbf{v}_{n+1}$ and $\mathbf{v}_n$. A natural constraint among frames is $\mathbf{v}_N \succ ... \succ \mathbf{v}_{n+1} \succ \mathbf{v}_n \succ ... \succ \mathbf{v}_1$. Temporal rank pooling (t-rk) [12] optimizes parameters $\mathbf{u} \in \mathbb{R}^{(P\cdot Q)\times 1}$ of a linear function $\psi(\mathbf{v};\mathbf{u})$ to ensure that $\forall n_i, n_j, \mathbf{v}_{n_i} \succ \mathbf{v}_{n_j} \Leftrightarrow \mathbf{u}^{\mathrm{T}}\cdot\mathbf{v}_{n_i} > \mathbf{u}^{\mathrm{T}}\cdot\mathbf{v}_{n_j}$. The parameter $\mathbf{u}$ is used as the representation of temporal rank pooling method, as it implicitly encodes the appearance evolution information of the sequence.

Spatial Rank Pooling: We reshape $\mathbf{u}$ as the same size of input frame to facilitate the observation. As shown in Fig. 3 (d), the temporal rank pooling method mainly preserves spatial information while ignores most of the temporal information. To improve this, we propose a novel spatial rank pooling method (s-rk) which takes both spatial and temporal information into account. The observation is that there exists huge spatial redundancy in each heatmap. Therefore, we take advantage of the learning to rank method to reduce each heatmap to a compact feature, which has the ability of preserving spatial order. Concatenating all feature vectors according to the temporal order will generate a body shape evolution image, which can preserve both spatial and temporal information of heatmaps in a compact way. The pipeline of generating body shape evolution image is shown in Fig. 4. Specifically, we partition the n_{th} frame of the sequence $\mathcal{V}$ into P rows, i.e., $\mathbf{G}_n = [(\mathbf{p}_1)^{\mathrm{T}}, ..., (\mathbf{p}_s)^{\mathrm{T}}, ..., (\mathbf{p}_P)^{\mathrm{T}}]^{\mathrm{T}}$, or Q columns, i.e., $\mathbf{G}_n = [\mathbf{q}_1, ..., \mathbf{q}_s, ..., \mathbf{q}_Q]$. Similar to Eq. 4, function V is applied to map $(\mathbf{p}_{1:s})^{\mathrm{T}}$ and $\mathbf{q}_{1:s}$ to $\mathbf{v}_s^{\mathbf{p}}$ and $\mathbf{v}_s^{\mathbf{q}}$, respectively. Using the structural risk minimization and max-margin framework, the objective is defined as:

$$\begin{aligned} &\underset{\mathbf{u}^\eta}{arg\,min}\ \frac{1}{2}\left\|\mathbf{u}^\eta\right\|^2 + W\sum_{\forall i,j\ \mathbf{v}_{s_i}^\eta \succ \mathbf{v}_{s_j}^\eta}\epsilon_{ij}, \\ &s.t.\ (\mathbf{u}^\eta)^{\mathrm{T}}\cdot(\mathbf{v}_{s_i}^\eta - \mathbf{v}_{s_j}^\eta) \geq 1-\epsilon_{ij} \\ &\epsilon_{ij} \geq 0 \end{aligned} \tag{5}$$

where $\eta \in \{\mathbf{p}, \mathbf{q}\}$, $\mathbf{u}^{\mathbf{p}} \in \mathbb{R}^Q$ and $\mathbf{u}^{\mathbf{q}} \in \mathbb{R}^P$. For all N frames, we catenate vectors according to the temporal order, and obtain $\mathbf{U}^{\mathbf{p}} \in \mathbb{R}^{Q\times N}$ and $\mathbf{U}^{\mathbf{q}} \in \mathbb{R}^{P\times N}$. The final matrix $\mathbf{U}$ via spatial rank pooling is defined as $\left[(\mathbf{U}^{\mathbf{p}})^{\mathrm{T}}, (\mathbf{U}^{\mathbf{q}})^{\mathrm{T}}\right]^{\mathrm{T}}$, where $\mathbf{U}$ is called body shape evolution image.

Body Guided Sampling: Since body shape evolution image only considers the shape of accumulated pose estimation map while does not differentiate different joints, we need body pose evolution image to consider the information from each specific joint. To this end, this section builds body pose evolution image from joints, which are densely sampled by body guided sampling to denote the specific pose of human body. Suppose a sequence $\mathcal{V}_{\mathcal{L}} = \{\mathcal{L}_1, ..., \mathcal{L}_n, ..., \mathcal{L}_N\}$ contains N poses, where $\mathcal{L}_n = \{\mathbf{z}^{k,n}\}_{k=1}^{K}$ and $\mathbf{z}^{k,n} = (x^{k,n}, y^{k,n})$, which denote the horizontal and vertical coordinates of the k_{th} joint. When

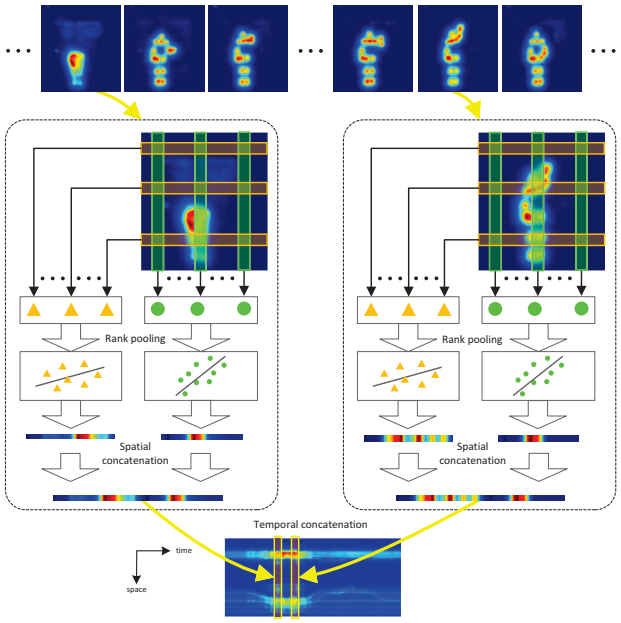

Figure 4: Generation of body shape evolution image via proposed spatial rank pooling

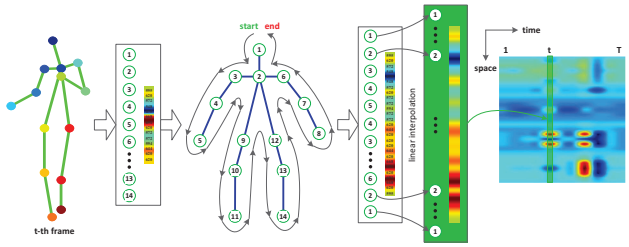

Figure 5: Building one channel of body pose evolution image from horizontal coordinates via body guided sampling

recording pose sequences, the distances between human bodies and the depth video are not strictly the same. In other words, different pose sequences have their own scales, which bring intra-varieties to same types of actions. To this end, $\{\{x^{k,n}\}_{k=1}^{K}\}_{n=1}^{N}$ and $\{\{y^{k,n}\}_{k=1}^{K}\}_{n=1}^{N}$ are normalized to change from 0 to 1, respectively.

To encode the spatial and temporal evolution of pose joints in a compact way, we represent the pre-processed pose sequence $\mathcal{V}_{\mathcal{L}}$ as a body pose evolution image. Since original estimated pose joints are too sparse to represent the human body, we sort the order of joint labels according to the body structure, and use linear interpolation to sample abundant points from pose limbs. The pipeline of building one channel of body pose evolution image from horizontal coordinates is shown in Fig. 5. Another channel of body pose evolution image from vertical coordinates can be similarly built. Mathematically, let $\mathbf{x}^n = [x^{1,n}, ..., x^{K',n}]^{\mathrm{T}}$ and $\mathbf{y}^n = [y^{1,n}, ..., y^{K',n}]^{\mathrm{T}}$ denote the coordinate vector of the generated joints, where K' is the total number of sampled joints. In our experiments, the pose limbs can be roughly denoted by sampling five joints on each limb. Suppose

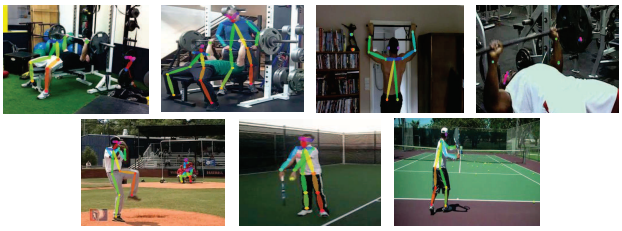

Figure 6: Snaps selected from PennAction dataset

$K = 14$ and the number of limb is $K - 1$, then the K' can be calculated as $K + (K - 1) \times 5 = 79$. Two channels of body pose evolution image are formed as $[\mathbf{x}^1, ..., \mathbf{x}^N]$ and $[\mathbf{y}^1, ..., \mathbf{y}^N]$, denoting horizontal and vertical coordinates, respectively. Both channels reflect the temporal evolution of joints, which are densely sampled to denote the pose of human body.

Late Fusion: A video $\mathcal{I}^c$ has been denoted as a body shape evolution image and a body pose evolution image, where c means the c_{th} sample from a batch that is used for training. As CNN has achieved success in image classification task, we use CNN model that is pre-trained on Imagenet [7] for transfer learning. Since these two images contain significantly different spatial structure, we use separate CNN to explore deep features from them. To accommodate existing CNN models, the single channel of body shape evolution image is repeated three times to form a 3-channel image, and two channels of body pose evolution image are combined with a zero-valued channel to form a 3-channel image. Let $\{\mathbf{I}_m^c\}_{m=1}^2$ denote these two images. Mean removal is adopted for input images to improve the convergence speed. Then, each color image is processed by a CNN. For the image $\mathbf{I}_m^c$, the output Υ_m of the last fully-connected (fc) layer is normalized by the softmax function to obtain the posterior probability: $prob(r \mid \mathbf{I}_m^c) = e^{\Upsilon_m^r} / \sum_{j=1}^{R} e^{\Upsilon_m^j}$, which indicates the probability of image $\mathbf{I}_m^c$ belonging to the r-th action class. R is the number of total action classes. The objective function of our model is to minimize the maximum-likelihood loss function $\mathcal{L}(\mathbf{I}_m) = -\sum_{c=1}^{2} ln \sum_{r=1}^{R} \delta(r - s_c)\, prob(r \mid \mathbf{I}_m^c)$, where function δ equals to one if $r = s_c$ and equals to zero otherwise, s_c is the ground truth label of $\mathbf{I}_m^c$, and C is the batch size. For a sequence $\mathcal{I}$, its final class score is the average of the two posteriors: $score(r \mid \mathcal{I}) = \frac{1}{2} \sum_{m=1}^{2} prob(r \mid \mathbf{I}_m)$, where $prob(r \mid \mathbf{I}_m)$ is the probability of $\mathbf{I}_m$ belonging to the r_{th} action class.

5. Experiments

5.1. Datasets and Settings

PennAction dataset [54] contains 15 action categories and 2326 sequences in total. Since all sequences are collected from the internet, complex body occlusions, large appearance and motion variations make it challenging for

Table 1: The evaluation of body shape evolution image and body pose evolution image on NTU RGB+D, UTD-MHAD and PennAction datasets. BPI is short for body pose evolution image. BSI is short for body shape evolution image. BSI (t-rk) is short for implementing BSI with temporal rank pooling. BSI (s-rk) is short for implementing BSI with spatial rank pooling. To accelerate the computations, approximate rank pooling [1] is used to implement rank pooling method.

Method	Sensor	Data	Feature	NTU RGB+D		UTD-MHAD	PennAction
				CS	CV	CS	half / half
S1	Kinect	2D Pose	BPI	80.52%	85.75%	85.53%	-
S2	Kinect	3D Pose	BPI	82.38%	86.65%	89.44%	-
H1	RGB	2D Pose	BPI	72.96%	77.21%	85.63%	84.08%
H2	RGB	Heatmap	BSI (t-rk)	53.91%	54.10%	58.88%	84.61%
H3	RGB	Heatmap	BSI (s-rk)	72.75%	78.35%	74.88%	87.02%
H1 + H3	RGB	2D Pose + Heatmap	BPI + BSI (s-rk)	78.80%	84.21%	92.51%	91.39%
S1 + H3	Kinect	2D Pose + Heatmap	BPI + BSI (s-rk)	90.90%	94.54%	92.84%	-
S2 + H3	Kinect	3D Pose + Heatmap	BPI + BSI (s-rk)	91.71%	95.26%	94.51%	-

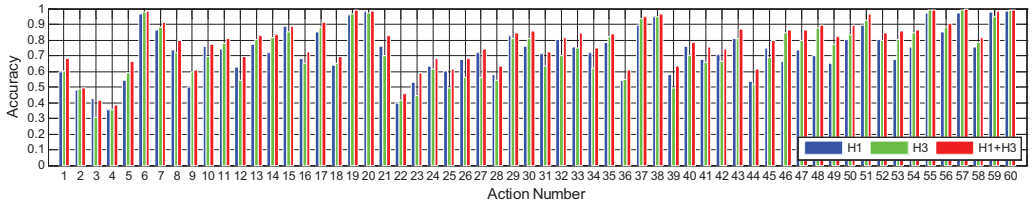

Figure 7: The complementary between body shape evolution image and body pose evolution image

pose-related action recognition [48, 8]. We follow [48] to split the data into half and half for training and testing. Snaps with estimated poses [1] are shown in Fig. 6.

NTU RGB+D dataset [35] contains 60 actions performed by 40 subjects from various views, generating 56880 sequences. Following the cross subject protocol in [35], we split the 40 subjects into training and testing groups. Each group contains samples captured from different views performed by 20 subjects. For this evaluation, the training and testing sets have 40320 and 16560 samples, respectively. Following the cross view protocol in [35], we use all the samples of camera 1 for testing and samples of cameras 2, 3 for training. The training and testing sets have 37920 and 18960 samples, respectively.

UTD-MHAD dataset [6] was collected using a Microsoft Kinect sensor and a wearable inertial sensor in an indoor environment. It contains 27 actions performed by 8 subjects. Each subject repeated each action 4 times, generating 861 sequences. We use this dataset to compare the performances of methods using different data modalities. Cross subject protocol [6] is used for the evaluation.

Implementing details: In our model, each CNN contains five convolutional layers and three fc layers. The first and second fc layers contain 4096 neurons, and the number of neurons in the third one is equal to the total number of action classes. Filter sizes are set to 11×11, 5×5, 3×3, 3×3 and 3×3, respectively. Local Response Normalization (LRN), max pooling and ReLU neuron are adopted and the dropout regularization ratio is set to 0.5. The network weights are learned using the mini-batch stochastic gradient descent with the momentum value set to 0.9 and weight decay set to 0.00005. Learning rate is set to 0.001 and the maximum training cycle is set to 60. When the CNN model achieves 99% accuracy on the training set, the training procedure is stopped beforehand. In each cycle, a mini-batch of C samples is constructed by randomly sampling images from training set. For NTU RGB+D dataset, UTD-MHAD dataset and PenAction dataset, C is set to 64, 16 and 16, considering the size of training set. To reduce the effect of random parameter initialization and random sampling, we repeat the training of CNN model for five times and report the average results. The implementation is based on PyTorch with one TITAN X card and 16G RAM.

[1] These poses are generated by pose estimation method [4], which can be found from https://github.com/tensorboy/pytorch_Realtime_Multi-Person_Pose_Estimation. Four pose joints on the head are not used since they are redundant for denoting actions. For scenes with multi-person, the coordinates of joints are averaged for feature extraction.

5.2. Discussions

2D pose & 3D pose from depth video: As poses estimated from videos lose depth cues, we begin with evaluating the significance of depth information for pose-based human action recognition. In Table 1, a 3D pose sequence extracted from a depth video is described as a three channel body pose evolution image, which is further encoded by CNN to predict action label. This method is called "S2". By setting all values of depth channel to zero, 2D pose sequences from depth videos are used instead. This method is called "S1". Without using depth channel, the accuracy drops from 89.44% to 85.53% on UTD-MHAD dataset. While, accuracies drop by only 1.86% from 82.38% to 80.52% for cross subject setting and 0.90% from 86.65%

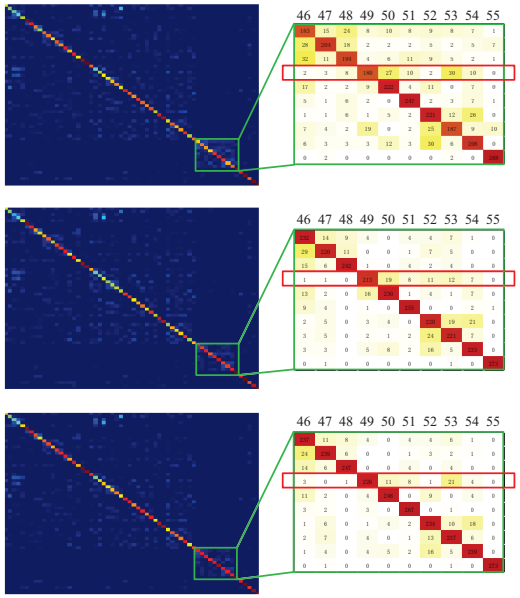

Figure 8: Confusion matrices of body pose evolution image-based method (first row), body shape evolution image-based method (second row), body pose and body shape evolution images-based method (third row) on NTU RGB+D dataset using cross subject protocol. Confusion matrices of ten actions are enlarged. These ten actions are: touch back (backache) (46), touch neck (neck-ache) (47), nausea or vomiting condition (48), use a fan (with hand or paper)/feeling warm (49), punching/slapping other person (50), kicking other person (51), pushing other person (52), pat on back of other person (53), point finger at the other person (54), hugging other person (55). Red boxes verify that the body pose and body shape evolution images compensate for each other and improve the recognition of action (49).

to 85.75% for cross view setting on NTU RGB+D dataset. These results show that depth information can improve the recognition, but the influence of depth channel drops when large scale training data is used, as well-trained CNN model may infer depth cues from 2D pose. These results support the potential of estimating 2D poses for action recognition from videos, where depth cues are not directly available.

2D pose from video: With accurate pose estimation method and additional depth cues, 3D poses from depth videos are more reliable than 2D poses from videos. This part evaluates the performance of noisy 2D poses from videos by comparing with 2D poses and 3D poses from depth videos. "H1" denotes our proposed method using body pose evolution image. On NTU RGB+D dataset, "H1" performs worse than "S1". The reason is that this dataset contains multi-view samples, which brings more ambiguities to 2D pose from video than 3D pose from depth video. The performance of "H1" is comparable with "S1" on UTD-MHAD dataset. The reason is that this dataset contains samples observed from single view, which helps the pose estimation from both RGB and depth videos. Generally,

Table 2: Comparisons between our proposed method and state-of-the-art methods on NTU RGB+D dataset

Method	Year	CS	CV
HON4D [32]	2013	30.56%	7.26%
Super Normal Vector [50]	2014	31.82%	13.61%
Skeletal Quads [10]	2014	38.60%	41.40%
Lie Group [40]	2014	50.10%	52.80%
HBRNN-L [9]	2015	59.07%	63.97%
FTP Dynamic Skeletons [16]	2015	60.23%	65.22%
Deep RNN [35]	2016	59.29%	64.09%
Deep LSTM [35]	2016	60.69%	67.29%
2 Layer P-LSTM [35]	2016	62.93%	70.27%
ST-LSTM + Trust Gate [25]	2016	69.20%	77.70%
Unsupervised Learning [29]	2017	56.00%	-
LieNet-3Blocks [17]	2017	61.37%	66.95%
GCA-LSTM network [26]	2017	74.40%	82.80%
Body-part appearance + skeleton [33]	2017	75.20%	83.10%
Clips + CNN + MTLN [20]	2017	79.57%	84.83%
View-invariant [28]	2017	80.03%	87.21%
Proposed Method: H1 + H3	-	**78.80%**	**84.21%**
Proposed Method: S2 + H3	-	**91.71%**	**95.26%**

Table 3: Comparisons between our proposed method and state-of-the-art methods on UTD-MHAD dataset

Sensor	Method	Year	CS
Kinect	Cov3DJ [18]	2013	85.58%
Kinect	Kinect [6]	2015	66.10%
Inertial	Inertial [6]	2015	67.20%
Kinect + Inertial	Kinect&Inertial [6]	2015	79.10%
Kinect	JTM [44]	2016	85.81%
Kinect	Optical Spectra [15]	2016	86.97%
Kinect	3DHOT-MBC [53]	2017	84.40%
Kinect	JDM [23]	2017	88.10%
RGB	*Proposed Method*: H1 + H3	-	**92.84%**
Kinect	*Proposed Method*: S2 + H3	-	**94.51%**

2D pose from video can only compete with that from depth video in simple scenes; 2D pose from video can barely achieve the performance of 3D pose from depth video.

Heatmap from video: Instead of using sole 2D pose, we evaluate the performance of combining heatmap with 2D pose for recognition from video. The method called "H3" describes heatmap as body shape evolution image using spatial rank pooling. For comparisons, the method called "H2" is implemented by temporal rank pooling. "H3" outperforms "H2" by more than 15% on both NTU RGB+D and UTD-MHAD datasets, which verifies the advantage of spatial rank pooling method in preserving both spatial and temporal cues. The method called "H1 + H3" denotes the combination of both 2D pose and heatmap. "H1 + H3" outperforms at least 5% than "H1". Detailed improvements on NTU RGB+D dataset using cross subject protocol are shown in Fig. 7. These results indicate the complementary property between 2D pose and heatmap. In Fig. 8, we analyze the confusion matrices among 10 types of actions. The red boxes highlight the improvement on action "use a fan (with hand or paper)/feeling warm (49)", by combining 2D pose and heatmap. As shown in Fig. 9, body pose evolution image only captures 2D pose joints, which are noisy due to occlusions. However, body shape evolution image, which captures global shape of heatmap, can provide cues for inferring accurate locations of 2D pose joints.

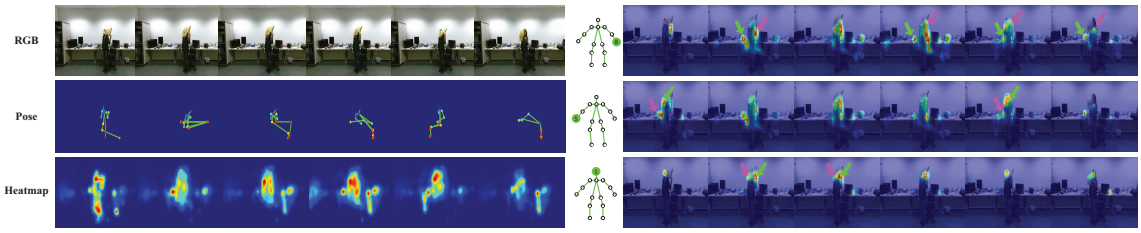

Figure 9: The visualization of 2D poses and heatmaps estimated from the action "use a fan (with hand or paper)/feeling warm (49)". We show the pose estimation maps of the 8-th, 5-th, 1-st joint. On these pose estimation maps, green arrow points out the estimated position of joint, which is inaccurate. Meanwhile, pink arrow points to the region of heatmap which covers the ground truth of the joint position. We claim that heatmaps contain richer cues for inferring locations of joints when estimated 2D poses are inaccurate.

Table 4: Comparisons between our proposed method and state-of-the-art methods on PennAction dataset

Sensor	Method	Year	half / half
RGB	Action Bank [54]	2013	83.90%
RGB	AOG [48]	2015	85.50%
RGB	C3D [3]	2016	86.00%
RGB	JDD [3]	2016	87.40%
RGB	Pose + IDT-FV [19]	2017	92.00%
RGB	RPAN [8]	2017	97.40%
RGB	*Proposed Method*: H1 + H3	-	**91.39%**
RGB	*Proposed Method*: (H1 + H3)*	-	**98.22%**

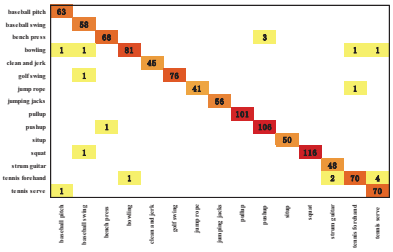

Figure 10: Confusion matrix of our method on PennAction dataset

5.3. Comparisons with State-of-the-arts

Ours versus 3D Pose-based methods: Even with inaccurate 2D pose estimation method, our method outperforms 3D pose, which is extracted by 3D pose estimation method from depth sensors. In Table 2 and Table 3, "H1 + H3" outperforms most state-of-the-art methods using 3D poses. Specifically, "H1 + H3" achieves 78.80% and 84.21% on the currently largest NTU RGB+D dataset. "H1 + H3" also outperforms LSTM-based method, i.e., GCA-LSTM [26]. Although slightly worse, "H1 + H3" approaches to similar performance of the most recent CNN-based method, i.e., View-invariant [28]. On UTD-MHAD dataset, "H1 + H3" outperforms all 3D pose-based methods, e.g., 3DHOT-MBC [53] and JDM [23]. Instead of using 2D poses, it is interesting to combine heatmap with more accurate poses, namely 2D poses from depth video. Table 1 shows that "S1+H3" outperforms "H1+H3". With additional depth information, "S2+H3" achieves the best performances. These results verify that our proposed heatmaps benefit both 2D poses from videos and 3D poses from depth videos.

Ours versus video-based methods: Among approaches using videos, our method is compared with most related 2D pose-based action recognition methods. In Table 4, poselet detected by Action Bank [54] achieves accuracy of 83.90% on PennAction dataset. AOG [48] and Pose + IDT-FV [19] benefit from treating pose estimation and action recognition as a uniform framework, and achieve accuracy of 85.50% and 92.00%. RPAN [8] goes beyond previous studies by training an end-to-end RNN network, and achieves accuracy of 97.40%. Our method jointly learns 2D poses and heatmaps, and the complementary property between them alleviate the effect of noisy 2D poses. We further select one frame for each video and use CNN to extract deep features. Also, we encode annotated poses, which are provided by original dataset. Fused with these additional information, our method "(H1 + H3)*" achieves accuracy of 98.22% and outperforms all recent methods. The confusion matrix of our method is shown in Fig. 10, where most of ambiguities among actions are suppressed.

6. Conclusions

This paper recognizes actions from videos as the evolution of pose estimation maps. Compared with unreliable estimated 2D poses, pose estimation maps provide richer cues for inferring body parts and their movements. By describing the evolution of pose estimation maps as compact body shape evolution image and body pose evolution image, our method can effectively capture movements of both body shape and body parts, thereby outperforming all 2D pose or 3D pose-based methods on benchmark datasets. It is noted that our features only rely on the estimated pose estimation maps rather than original videos, from which the pose estimation maps are estimated. This property indicates the generalization ability of our method by estimating pose estimation maps from various types of input video, e.g., depth or infrared video, for action recognition task.

7. Acknowledgement

This work is supported in part by Singapore Ministry of Education Academic Research Fund Tier 2 MOE2015-T2-2-114 and start-up funds of University at Buffalo.

References

[1] H. Bilen, B. Fernando, E. Gavves, A. Vedaldi, and S. Gould. Dynamic image networks for action recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3034–3042, 2016.

[2] A. Bulat and G. Tzimiropoulos. Human pose estimation via convolutional part heatmap regression. In *European Conference on Computer Vision (ECCV)*, pages 717–732, 2016.

[3] C. Cao, Y. Zhang, C. Zhang, and H. Lu. Action recognition with joints-pooled 3D deep convolutional descriptors. In *International Joint Conferences on Artificial Intelligence (IJCAI)*, pages 3324–3330, 2016.

[4] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh. Realtime multi-person 2D pose estimation using part affinity fields. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[5] J. Carreira, P. Agrawal, K. Fragkiadaki, and J. Malik. Human pose estimation with iterative error feedback. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4733–4742, 2016.

[6] C. Chen, R. Jafari, and N. Kehtarnavaz. UTD-MHAD: A multimodal dataset for human action recognition utilizing a depth camera and a wearable inertial sensor. In *IEEE International Conference on Image Processing (ICIP)*, pages 168–172, 2015.

[7] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 248–255, 2009.

[8] W. Du, Y. Wang, and Y. Qiao. RPAN: An end-to-end recurrent pose-attention network for action recognition in videos. In *IEEE International Conference on Computer Vision (ICCV)*, 2017.

[9] Y. Du, W. Wang, and L. Wang. Hierarchical recurrent neural network for skeleton based action recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1110–1118, 2015.

[10] G. Evangelidis, G. Singh, and R. Horaud. Skeletal quads: Human action recognition using joint quadruples. In *IAPR International Conference on Pattern Recognition (ICPR)*, pages 4513–4518, 2014.

[11] C. Feichtenhofer, A. Pinz, and R. P. Wildes. Spatiotemporal multiplier networks for video action recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4768–4777, 2017.

[12] B. Fernando, E. Gavves, J. M. Oramas, A. Ghodrati, and T. Tuytelaars. Modeling video evolution for action recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5378–5387, 2015.

[13] G. Garcia-Hernando and T.-K. Kim. Transition forests: Learning discriminative temporal transitions for action recognition and detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 432–440, 2017.

[14] G. Gkioxari and J. Malik. Finding action tubes. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 759–768, 2015.

[15] Y. Hou, Z. Li, P. Wang, and W. Li. Skeleton optical spectra based action recognition using convolutional neural networks. *IEEE Transactions on Circuits and Systems for Video Technology*, 2016.

[16] J.-F. Hu, W.-S. Zheng, J. Lai, and J. Zhang. Jointly learning heterogeneous features for RGB-D activity recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5344–5352, 2015.

[17] Z. Huang, C. Wan, T. Probst, and L. Van Gool. Deep learning on lie groups for skeleton-based action recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6099–6108, 2017.

[18] M. E. Hussein, M. Torki, M. A. Gowayyed, and M. El-Saban. Human action recognition using a temporal hierarchy of covariance descriptors on 3D joint locations. In *International Joint Conferences on Artificial Intelligence (IJCAI)*, volume 13, pages 2466–2472, 2013.

[19] U. Iqbal, M. Garbade, and J. Gall. Pose for action-action for pose. In *IEEE International Conference on Automatic Face & Gesture Recognition (FG)*, pages 438–445, 2017.

[20] Q. Ke, M. Bennamoun, S. An, F. Sohel, and F. Boussaid. A new representation of skeleton sequences for 3D action recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[21] Z. Lan, M. Lin, X. Li, A. G. Hauptmann, and B. Raj. Beyond gaussian pyramid: Multi-skip feature stacking for action recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 204–212, 2015.

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

[23] C. Li, Y. Hou, P. Wang, and W. Li. Joint distance maps based action recognition with convolutional neural networks. *IEEE Signal Processing Letters*, 24(5):624–628, 2017.

[24] H. Liu, M. Liu, and Q. Sun. Learning directional co-occurrence for human action classification. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1235–1239, 2014.

[25] J. Liu, A. Shahroudy, D. Xu, and G. Wang. Spatio-temporal LSTM with trust gates for 3D human action recognition. In *European Conference on Computer Vision (ECCV)*, pages 816–833, 2016.

[26] J. Liu, G. Wang, P. Hu, L.-Y. Duan, and A. C. Kot. Global context-aware attention LSTM networks for 3D action recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[27] M. Liu and H. Liu. Depth Context: A new descriptor for human activity recognition by using sole depth sequences. *Neurocomputing*, 175:747–758, 2016.

[28] M. Liu, H. Liu, and C. Chen. Enhanced skeleton visualization for view invariant human action recognition. *Pattern Recognition*, 68:346–362, 2017.

[29] Z. Luo, B. Peng, D.-A. Huang, A. Alahi, and L. Fei-Fei. Unsupervised learning of long-term motion dynamics for videos. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[30] M. Ma, H. Fan, and K. M. Kitani. Going deeper into first-person activity recognition. In *IEEE Conference on Com-*

puter Vision and Pattern Recognition (CVPR), pages 1894–1903, 2016.

[31] S. Ma, L. Sigal, and S. Sclaroff. Space-time tree ensemble for action recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5024–5032, 2015.

[32] O. Oreifej and Z. Liu. Hon4D: Histogram of oriented 4D normals for activity recognition from depth sequences. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 716–723, 2013.

[33] H. Rahmani and M. Bennamoun. Learning action recognition model from depth and skeleton videos. In *IEEE International Conference on Computer Vision (ICCV)*, 2017.

[34] V. Ramakrishna, D. Munoz, M. Hebert, J. A. Bagnell, and Y. Sheikh. Pose machines: Articulated pose estimation via inference machines. In *European Conference on Computer Vision (ECCV)*, pages 33–47, 2014.

[35] A. Shahroudy, J. Liu, T.-T. Ng, and G. Wang. NTU RGB+D: A large scale dataset for 3D human activity analysis. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1010–1019, 2016.

[36] J. Shotton, T. Sharp, A. Kipman, A. Fitzgibbon, M. Finocchio, A. Blake, M. Cook, and R. Moore. Real-time human pose recognition in parts from single depth images. *Communications of the ACM*, 56(1):116–124, 2013.

[37] K. Simonyan and A. Zisserman. Two-stream convolutional networks for action recognition in videos. In *Advances in Neural Information Processing Systems (NIPS)*, pages 568–576, 2014.

[38] S. Singh, C. Arora, and C. Jawahar. First person action recognition using deep learned descriptors. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2620–2628, 2016.

[39] Z. Tu, W. Xie, Q. Qin, R. Poppe, R. C. Veltkamp, B. Li, and J. Yuan. Multi-stream CNN: Learning representations based on human-related regions for action recognition. *Pattern Recognition*, 79:32–43, 2018.

[40] 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 (CVPR)*, pages 588–595, 2014.

[41] C. Wang, Y. Wang, and A. L. Yuille. An approach to pose-based action recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 915–922, 2013.

[42] C. Wang, Y. Wang, and A. L. Yuille. Mining 3D key-pose-motifs for action recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2639–2647, 2016.

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

[44] P. Wang, Z. Li, Y. Hou, and W. Li. Action recognition based on joint trajectory maps using convolutional neural networks. In *ACM on Multimedia Conference (ACM MM)*, pages 102–106, 2016.

[45] Y. Wang, M. Long, J. Wang, and P. S. Yu. Spatiotemporal pyramid network for video action recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1529–1538, 2017.

[46] S.-E. Wei, V. Ramakrishna, T. Kanade, and Y. Sheikh. Convolutional pose machines. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4724–4732, 2016.

[47] J. Weng, C. Weng, and J. Yuan. Spatio-Temporal Naive-Bayes Nearest-Neighbor (ST-NBNN) for skeleton-based action recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4171–4180, 2017.

[48] B. Xiaohan Nie, C. Xiong, and S.-C. Zhu. Joint action recognition and pose estimation from video. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1293–1301, 2015.

[49] S. Yang, C. Yuan, B. Wu, W. Hu, and F. Wang. Multi-feature max-margin hierarchical bayesian model for action recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1610–1618, 2015.

[50] X. Yang and Y. Tian. Super normal vector for activity recognition using depth sequences. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 804–811, 2014.

[51] Y. Yang and D. Ramanan. Articulated human detection with flexible mixtures of parts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(12):2878–2890, 2013.

[52] G. Yu, Z. Liu, and J. Yuan. Discriminative orderlet mining for real-time recognition of human-object interaction. In *Asian Conference on Computer Vision (ACCV)*, pages 50–65, 2014.

[53] B. Zhang, Y. Yang, C. Chen, L. Yang, J. Han, and L. Shao. Action recognition using 3D histograms of texture and a multi-class boosting classifier. *IEEE Transactions on Image Processing*, 26:4648–4660, 2017.

[54] W. Zhang, M. Zhu, and K. G. Derpanis. From actemes to action: A strongly-supervised representation for detailed action understanding. In *IEEE International Conference on Computer Vision (ICCV)*, pages 2248–2255, 2013.

[55] Z. Zhang. Microsoft kinect sensor and its effect. *IEEE multimedia*, 19(2):4–10, 2012.

[56] W. Zhu, J. Hu, G. Sun, X. Cao, and Y. Qiao. A key volume mining deep framework for action recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1991–1999, 2016.