Effective Surface Normals Based Action Recognition in Depth Images

Xuan Son Nguyen, Thanh Phuong Nguyen, François Charpillet

Effective Surface Normals Based Action Recognition in Depth Images

Xuan Son Nguyen
Université de Lorraine
54600 Villers-Lès-Nancy, France
email: xuan-son.nguyen@inria.fr

Thanh Phuong Nguyen
Université de Toulon
83957 La Garde, France
email: thanh-phuong.nguyen@univ-tln.fr

François Charpillet
INRIA Nancy Grand-Est
54600 Villers-Lès-Nancy, France
email: francois.charpillet@inria.fr

***Abstract*—In this paper, we propose a new local descriptor for action recognition in depth images. The proposed descriptor relies on surface normals in 4D space of depth, time, spatial coordinates and higher-order partial derivatives of depth values along spatial coordinates. In order to classify actions, we follow the traditional Bag-of-words (BoW) approach, and propose two encoding methods termed Multi-Scale Fisher Vector (MSFV) and Temporal Sparse Coding based Fisher Vector Coding (TSCFVC) to form global representations of depth sequences. The high-dimensional action descriptors resulted from the two encoding methods are fed to a linear SVM for efficient action classification. Our proposed methods are evaluated on two public benchmark datasets, MSRAction3D and MSRGesture3D. The experimental result shows the effectiveness of the proposed methods on both the datasets.**

I. Introduction

Approaches for human action recognition in depth images have received a large attention in recent years thanks to the rich information provided by depth sensors. These approaches usually exploit depth information to build highly discriminative low-level descriptors, or use skeletal data which can be more easily obtained using depth images to build high-level descriptors. Although many approaches have achieved impressive results, they still face a number of challenges, e.g. rate variations, temporal misalignment, composite actions, noise, human-object interaction. Moreover, most of them require high computation time to extract features and recognize actions.

In this paper, we propose a new local descriptor that relies on surface normals in 4D space of depth, time and spatial coordinates which have been shown to carry important shape and motion cues for action recognition [1], [2]. In our proposed descriptor, the shape cue is enhanced by augmenting surface normals with higher-order partial derivatives of depth values along spatial coordinates, while the motion cue is characterized by combining a set of local descriptors extracted at consecutive 3D points along the temporal dimension. In order to effectively encode local descriptors into a global representation of depth sequences, we propose two encoding methods MSFV and TSCFVC, which are based on Fisher Vector (FV) [3] and Sparse Coding based Fisher Vector Coding (SCFVC) [4].

This paper is organized as follows. Section 2 introduces the related work on action recognition in depth images. Section 3 explains our proposed local descriptor, MSFV and TSCFVC for feature encoding. Section 4 presents the experimental evaluation of the proposed methods. Finally, Section 5 offers some conclusions and ideas for future work.

II. Related work

Existing approaches for action recognition in depth images can be broadly grouped into three main categories: skeleton-based, depth map-based and hybrid approaches. Xia et al. [5] partitioned the 3D space using a spherical coordinate defined at the hip joint position. Each 3D joint casted a vote into a bin to generate a histogram of 3D joint. These histograms were then used to construct visual words whose temporal evolutions were modeled using discrete Hidden Markov Models [6]. Yang and Tian [7] learned EigenJoints from differences of joint positions and used Naïve-Bayes-Nearest-Neighbor [8] for action classification. Zanfir et al. [9] relied on the configuration, speed, and acceleration of joints to construct the action descriptor. A modified kNN classifier was then used for action classification. Vemulapalli et al. [10] used rotations and translations to represent 3D geometric relationships of body parts in a Lie group [11], and then employed Dynamic Time Warping [12] and Fourier Temporal Pyramid [13] to model the temporal dynamics. Eweiwi et al. [14] learned a compact representation of depth sequences from joint locations, joint velocities and joint movement normals. Evangelidis et al. [15] proposed skeletal quad to describe the positions of nearby joints in the human skeleton and used FV for feature encoding.

Depth map-based approaches usually rely on low-level features from the space-time volume of depth sequences to compute action descriptors. Li et al. [16] proposed a bag of 3D points to capture the shape of the human body and used an action graph [17] to model the dynamics of actions. Representative 3D points used to describe a posture were sampled from a very small set of points in depth maps. Kurakin et al. [18] proposed cell occupancy-based and silhouette-based features which were then used with action graphs for gesture recognition. Wang et al. [19] introduced random occupancy patterns which were computed from subvolumes of the space-time volume of depth sequences with different sizes and at different locations. Since the number of subvolumes can be extremely large, a weighted random sampling scheme was proposed to effectively explore the space-time volume. Yang et al. [20] projected depth maps onto three orthogonal Cartesian planes to obtain Depth Motion Maps (DMMs) which were

used to extract HOG descriptors [21] for action recognition. Xia and Aggarwal [22] proposed a filtering method to extract local spatio-temporal interest points of depth sequences. The histograms of depth pixels in 3D cuboids centered around the extracted interest points were calculated and used in a BoW approach for action recognition. Wang et al. [23] represented actions by histograms of spatial-part-sets and temporal-part-sets, where spatial-part-sets are sets of frequently co-occurring spatial configurations of body parts in a single frame, and temporal-part-sets are co-occurring sequences of evolving body parts. Oreifej and Liu [1] and Yang and Tian [2] relied on surface normals in 4D space of depth, time, and spatial coordinates to capture the shape and motion cues in depth sequences. However, they used different methods to construct action descriptors. The method of [1] used polychorons to quantize possible directions of 4D normals, while the method of [2] used SC to compute visual words and spatial average pooling and temporal max pooling to aggregate local descriptors into a global representation of depth sequences.

Hybrid approaches combine skeletal data and depth maps to create action descriptors. Wang et al. [13] introduced local occupancy patterns computed in spatio-temporal cells around 3D joints which were treated as the depth appearance of these joints. They proposed Actionlet Ensemble Model where each actionlet is a particular conjunction of the features for a subset of 3D joints. An action was then represented as a linear combination of a set of discriminative actionlets which were learned using data mining. Du et al. [24] divided the human skeleton into five parts, and fed them to five subnets of a recurrent neural network [25]. As the number of layers increases, the representations extracted by the subnets are hierarchically fused to be the inputs of higher layers. Once the final representations of the skeleton sequences have been obtained, actions are classified using a fully connected layer and a softmax layer.

III. Our method

We follow the traditional BoW approach. For feature extraction, we rely on surface normals in 4D space of depth, time, and spatial coordinates and higher-order partial derivatives of depth values along spatial coordinates. In order to encode local descriptors into a global representations of depth sequences, we propose two encoding methods termed MSFV and TSCFVC. Action recognition is performed using a linear SVM classifier. In what follows we explain in more detail the different steps of our method.

A. Local Feature Extraction

Our local descriptor is extracted using surface normals in 4D space (SN4D) of depth, time, and spatial coordinates, which are the extensions of surface normals in 3D space of depth and spatial coordinates [26]. The depth sequence can be considered as a function $\mathbb{R}^3 \rightarrow \mathbb{R}^1 : z = f(x, y, t)$, which constitutes a surface in the 4D space represented as the set of points (x, y, z, t) satisfying $S(x, y, t, z) = f(x, y, t) - z = 0$. The normal to the surface S is computed as:

$$\mathbf{n} = \nabla S = [\frac{\partial z}{\partial x}; \frac{\partial z}{\partial y}; \frac{\partial z}{\partial t}; -1].$$

The first three components of $\mathbf{n}$ are the partial derivatives of depth values along the spatial and temporal dimensions, which have been shown [1], [2] to carry important shape and motion cues. For each pixel $\mathbf{p}_t = (x, y, z)$ at frame t of the depth sequence, we combine the first three components of $\mathbf{n}$ with the third-order partial derivatives of depth values along x and y axes to form a 5-dimensional feature descriptor $\mathbf{l}(\mathbf{p}_t)$ as follows:

$$\mathbf{l}(\mathbf{p}_t) = [\frac{\partial z}{\partial x}; \frac{\partial z}{\partial y}; \frac{\partial z}{\partial t}; \frac{\partial^3 z}{\partial x^3}; \frac{\partial^3 z}{\partial y^3}].$$

To better characterize the local motion, we concatenate $\mathbf{l}$ for a set of n neighboring pixels $\{\mathbf{p}_{t-\frac{n-1}{2}}, \mathbf{p}_{t-\frac{n-1}{2}+1}, \ldots, \mathbf{p}_{t+\frac{n-1}{2}}\}$ of $\mathbf{p}_t$ in the temporal dimension. The final feature descriptor $\mathbf{v}$ extracted at pixel $\mathbf{p}_t$ is thus given as:

$$\mathbf{v} = [\mathbf{l}(\mathbf{p}_{t-\frac{n-1}{2}}); \mathbf{l}(\mathbf{p}_{t-\frac{n-1}{2}+1}); \ldots \mathbf{l}(\mathbf{p}_{t+\frac{n-1}{2}})].$$

In our approach, the shape and motion cues are effectively encoded into a compact representation, where the shape cue is described by the first and third order partial derivatives of depth values along the spatial coordinates, and the motion cue is characterized by the first-order partial derivatives of depth values along the temporal dimension and by combining a set of local descriptors computed at consecutive 3D points along the temporal dimension.

B. Feature Encoding

Given a depth sequence which can be written as a set of whitened vectors $\mathbf{X} = \{\mathbf{v}_t, t = 1, \ldots, T\}$, where $\mathbf{v}_t \in \mathbb{R}^M$, M is the dimension of a local descriptor, T is the number of local descriptors extracted in the sequence, we propose two encoding methods to construct a global representation of the sequence. In the following, we explain MSFV and TSCFVC for this purpose.

1) MSFV: FV was first introduced in [27] which assumes that the generation process of local descriptors $\mathbf{v}_t$ can be modeled by a probability density function $p(.; \theta)$ with parameters θ. In order to describe the contribution of individual parameters to the generative process, one can compute the gradient of the log-likelihood of the data on the model:

$$\mathcal{G}_\theta^{\mathbf{X}} = \frac{1}{T} \nabla_\theta \log p(\mathbf{X}; \theta).$$

The probability density function is usually modeled by Gaussian Mixture Model (GMM), and $\theta = \{w_1, \boldsymbol{\mu}_1, \boldsymbol{\sigma}_1, \ldots, w_K, \boldsymbol{\mu}_K, \boldsymbol{\sigma}_K\}$ are the model parameters of the K-component GMM, where w_k, $\boldsymbol{\mu}_k$ and $\boldsymbol{\sigma}_k$ are respectively the mixture weight, mean vector, and diagonal

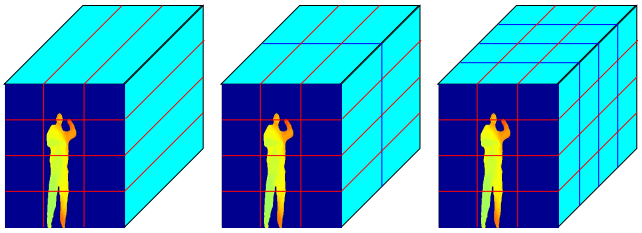

Fig. 1. The spatio-temporal grids used in our experiments. From left to right: the first, second and third temporal pyramids.

covariance matrix of Gaussian k. In our work, we use the following formulas [28] to calculate the FV components:

$$\mathcal{G}^{\mathbf{X}}_{\boldsymbol{\mu}_k} = \frac{1}{T\sqrt{w_k}} \sum_{t=1}^{T} \gamma_t(k) \left(\frac{\mathbf{v}_t - \boldsymbol{\mu}_k}{\boldsymbol{\sigma}_k} \right),$$

$$\mathcal{G}^{\mathbf{X}}_{\boldsymbol{\sigma}_k} = \frac{1}{T\sqrt{2w_k}} \sum_{t=1}^{T} \gamma_t(k) \left(\frac{(\mathbf{v}_t - \boldsymbol{\mu}_k)^2}{\boldsymbol{\sigma}_k^2} - 1 \right),$$

where $\gamma_t(k)$ is the soft assignment of $\mathbf{v}_t$ to Gaussian k. The FV representation of the sequence is the concatenation of $\mathcal{G}^{\mathbf{X}}_{\boldsymbol{\mu}_k}$ and $\mathcal{G}^{\mathbf{X}}_{\boldsymbol{\sigma}_k}$. Since $\mathcal{G}^{\mathbf{X}}_{\boldsymbol{\mu}_k}$ and $\mathcal{G}^{\mathbf{X}}_{\boldsymbol{\sigma}_k}$ are M-dimensional vectors, our FVs are $2MK$-dimensional vectors.

Following [2], we partition the space-time volume of the depth sequence into the spatio-temporal grids illustrated in Fig. 1, where the largest spatio-temporal grid corresponds to the bounding box of the action, and adaptive temporal pyramid is used to take into account the variations in motion speed and frequency when different people perform the same action. We calculate the FV for each grid, and apply two normalization steps: l_2-normalization and power normalization [3]. We then concatenate the FVs of all the grids to form the final representation of the depth sequence.

In order to improve the recognition accuracy, we rely on multiple scales of the depth sequence. The diagram of our proposed method is described in Fig.2, where three different scales of the sequence are used. First, local descriptors are extracted at different scales of the depth sequence. Then, a fixed proportion of local descriptors is randomly selected from each scale and the selected local descriptors are combined to train a GMM. In the testing phase, the FV of each scaled sequence is computed using the trained GMM and the local descriptors of that scale (for each scaled sequence we use the spatio-temporal grids given in Fig. 1). These FVs are then concatenated to form the final FV of the sequence.

2) TSCFVC: FV relies on the assumption that $p(.;\theta)$ is a Gaussian mixture with a fixed number of components K. As the dimensionality of the feature space increases, K must also increase to model the feature space accurately. This results in the explosion of the FV representation dimensionality. In order to deal with this problem, Liu et al. [4] proposed SCFVC which assumes that local descriptors are drawn from a Gaussian distribution $\mathcal{N}(\mathbf{Bu}, \sigma I)$, where $\mathbf{B} = [\mathbf{b}_1, \ldots, \mathbf{b}_K]$ is a matrix of bases (visual words) and $\mathbf{u}$ is a latent coding vector randomly generated from a zero mean Laplacian distribution.

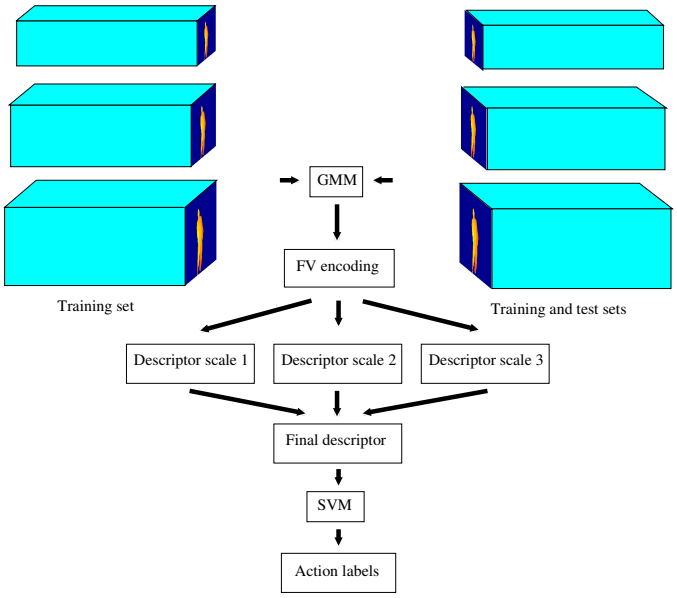

Fig. 2. The diagram of MSFV.

This corresponds to modeling $p(.;\theta)$ using a Gaussian mixture with an infinite number of components. The generative model can be written as:

$$p(\mathbf{v}, \mathbf{u}|\mathbf{B}) = p(\mathbf{v}|\mathbf{u}, \mathbf{B})p(\mathbf{u}).$$

Thus:

$$p(\mathbf{v}) = \int_{\mathbf{u}} p(\mathbf{v}, \mathbf{u}|\mathbf{B})d\mathbf{u} = \int_{\mathbf{u}} p(\mathbf{v}|\mathbf{u}, \mathbf{B})p(\mathbf{u})d\mathbf{u}.$$

Denote $\mathbf{u}^* = \arg\max_{\mathbf{u}} p(\mathbf{v}|\mathbf{u}, \mathbf{B})p(\mathbf{u})$, then $p(\mathbf{v})$ can be approximated by:

$$p(\mathbf{v}) \approx p(\mathbf{v}|\mathbf{u}^*, \mathbf{B})p(\mathbf{u}^*).$$

Taking the logarithm of $p(\mathbf{v})$, one obtains:

$$\log(p(\mathbf{v}|\mathbf{B})) = \min_{\mathbf{u}} \frac{1}{\sigma^2}\|\mathbf{v} - \mathbf{Bu}\|_2^2 + \lambda\|\mathbf{u}\|_1. \quad (1)$$

Eq. 1 reveals the relationship between the generative model and a sparse coding model. Liu et al. [4] showed that the FV representation of $\mathbf{X}$ is given by:

$$\mathbf{Q} = [\mathbf{q}_1; \ldots; \mathbf{q}_K],$$

$$\mathbf{q}_k = \sum_{t=1}^{T} u_t^*(k)(\mathbf{v}_t - \mathbf{Bu}_t^*), \quad (2)$$

where $\mathbf{u}_t^*$ is the solution of the sparse coding problem, $u_t^*(k)$ is the k^{th} dimension of $\mathbf{u}_t^*$.

In order to take into account the temporal order of local descriptors in $\mathbf{X}$, we follow the approach of [2]. For the k^{th} visual word, we compute the FV of the set of local descriptors extracted at the same frame of the depth sequence:

Method	Accuracy
Li et al., 2010 [16]	74.70
Xia et al., 2012 [5]	79.00
Yang and Tian, 2012 [7]	82.30
Wang et al., 2012 [19]	86.50
Wang et al., 2012 [13]	88.20
Yang et al., 2012 [20]	88.73
Oreifej and Liu, 2013 [1]	88.89
Xia and Aggarwal, 2013 [22]	89.30
Wang et al., 2013 [23]	90.00
Zanfir et al., 2013 [9]	91.70
Vemulapalli et al., 2014 [10]	92.46
Yang and Tian, 2014 [2]	93.09
SN4D-MSFV	**93.27**
SN4D-TSCFVC	**95.42**

TABLE I
RECOGNITION ACCURACY COMPARISON OF OUR METHODS AND PREVIOUS APPROACHES ON MSRACTION3D.

$$\mathbf{q}_k(f) = \frac{1}{|N_f|} \sum_{i \in N_f} u_i^*(k)(\mathbf{v}_i - \mathbf{B}\mathbf{u}_i^*), \tag{3}$$

where N_f is the set of 3D points at frame f of the depth sequence. The FV is normalized to avoid the dependence on the image resolution.

If $\mathbf{q}_k(f) = [q_k^1(f); \ldots; q_k^M(f)]$, then the j^{th} dimension q_k^j of $\mathbf{q}_k$ is computed as:

$$q_k^j = \max_{f=1,\ldots,F} q_k^j(f), \text{ for } j = 1, \ldots, M, \tag{4}$$

where the frame indices of the depth sequence are supposed to be $1, \ldots, F$.

Now, $\mathbf{q}_k$ is calculated using Eqs. 3 and 4 instead of Eq. 2. As for MSFV, we use the spatio-temporal grids described in Fig.1 to capture the spatial geometry and temporal order of the depth sequence. The final representation of the depth sequence is the concatenation of the FVs from all the grids.

C. Action Recognition

The FV has been shown [28], [15] to give good performance on image classification and action recognition when it is combined with linear classifiers. Thus for both the encoding methods, we rely on linear SVMs trained in an one-versus-all fashion to build a multi-class classifier for action recognition. A significant benefit of linear classifiers is that they are efficient to train and test. In our experiments, we use the LIBLINEAR library [29] that provides the implementation of linear SVMs.

IV. EXPERIMENTS

In this section, we evaluate the proposed methods on two benchmark datasets: MSRAction3D [16] and MSRGesture3D [19]. The two encoding methods presented in Section III-B result in two algorithms termed SN4D-MSFV and SN4D-TSCFVC, which use MSFV (see Section III-B1) and TSCFVC (see Section III-B2) for feature encoding, respectively. These two methods are compared against several state-of-the-art methods. We use the recognition accuracy reported in the original papers for comparison. The number of neighboring pixels n was set to $n = 5$, which has been experimentally found to give good results. For SN4D-TSCFVC, the number of visual words was set to $K = 100$. Since each local descriptor has 25 dimensions, the vector representation of a spatio-temporal grid has 2500 dimensions. The final representation of a depth sequence is the concatenation of $3 \times 4 \times 7$ 2500-dimensional vectors, which is a 210000-dimensional vector. For SN4D-MSFV, the number of GMM components K was set to $K = 50$. The vector representation of a sequence for each scale is the concatenation of $3 \times 4 \times 7$ FVs which is 210000-dimensional. We used 3 spatial scales spaced by a factor of $1/\sqrt{2}$, which has been experimentally found satisfactory. The image width and height at the first scale are those of the original depth images. The final representation of a sequence is the concatenation of its vector representations in 3 scales and is thus a 630000-dimensional vector.

Method	AS1	AS2	AS3	Ave.
Chen et al., 2013 [30]	**96.2**	83.2	92.0	90.47
Gowayyed et al., 2013 [31]	92.39	90.18	91.43	91.26
Vemulapalli et al., 2014 [10]	95.29	83.87	**98.22**	92.46
Du et al., 2015 [24]	93.33	94.64	95.50	94.49
SN4D-MSFV	93.58	90.88	95.35	93.27
SN4D-TSCFVC	92.92	**95.57**	97.76	**95.42**

TABLE II
RECOGNITION ACCURACY COMPARISON OF OUR METHODS AND PREVIOUS APPROACHES ON AS1, AS2 AND AS3 OF MSRACTION3D.

A. MSRAction3D Dataset

The MSRAction3D is an action dataset captured using a depth sensor similar to Kinect. It contains 20 actions performed by 10 different subjects. Each subject performs every action two or three times.

For a fair comparison, we used the experimental setting described in [16]. We divided the 20 actions into three subsets AS1, AS2 and AS3, each having 8 actions. The AS1 and AS2 were intended to group actions with similar movement, while AS3 was intended to group complex actions together. Action recognition was performed on each subset separately. We followed the cross-subject test setting, in which subjects 1,3,5,7,9 were used for training and subjects 2,4,6,8,10 were used for testing. Tab. I shows the accuracy of the proposed methods and different state-of-the-art methods. SN4D-TSCFVC achieves an accuracy of 95.42% which is the best method among the competing ones. Note that SN4D-TSCFVC outperforms the methods in [1], [2] which also rely on surface normals in 4D space of depth, time, and spatial coordinates to calculate local descriptors. SN4D-MSFV has lower accuracy than SN4D-TSCFVC but it still outperforms the majority of the methods. In Tab. II, we compare SN4D-TSCFVC and SN4D-MSFV against other state-of-the-art methods that reported the results on AS1, AS2 and AS3 separately. As can be observed, SN4D-TSCFVC achieves the highest accuracy for AS2 and the highest average accuracy. The confusion matrices are shown in Fig.3. The most confusions occur between the actions *hammer*

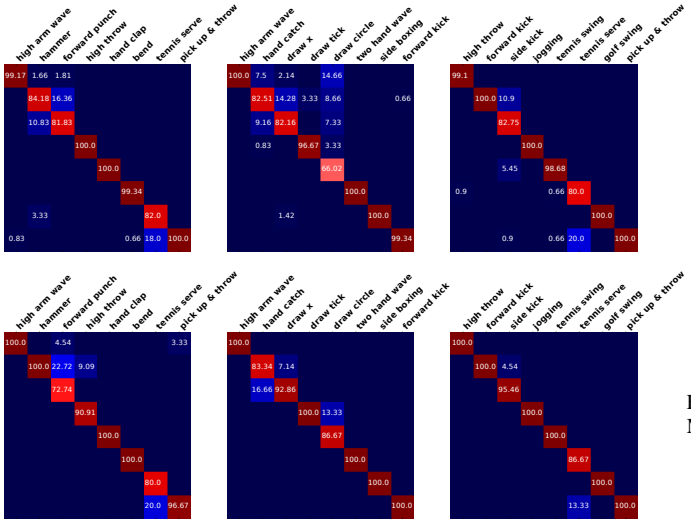

Fig. 3. (Best viewed in color) The confusion matrices of our methods on MSRAction3D (top: SN4D-MSFV, bottom: SN4D-TSCFVC, left: AS1, middle: AS2, right: AS3).

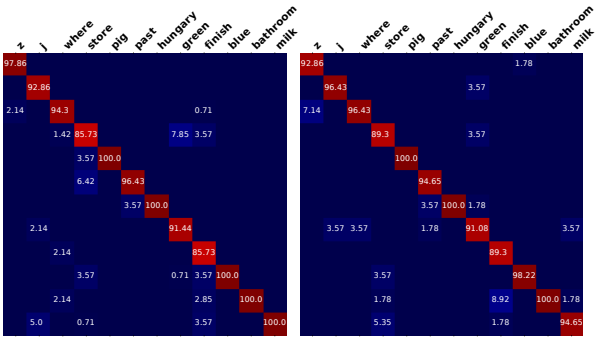

Fig. 4. (Best viewed in color) The confusion matrices of our methods on MSRGesture3D (left: SN4D-MSFV, right: SN4D-TSCFVC).

and *forward punch* (AS1), *tennis serve* and *pick up & throw* (AS1 and AS3), *hand catch* and *draw x* (AS2), *draw circle* and *draw tick* (AS2), *draw circle* and *high arm wave* (AS2), *side kick* and *forward kick* (AS3). These actions tend to be confused because of similarity in arm or leg motion. For example, in the action *tennis serve*, the arm of the subject moves through a circular motion which is similar to the throwing action in the second phase of the action *pick up & throw*. The actions *draw circle* and *draw tick* are highly similar in arm motion, while the actions *side kick* and *forward kick* share similar leg motion.

B. MSRGesture3D Dataset

The MSRGesture3D dataset contains 12 dynamic hand gestures defined by the American sign language. Each gesture is performed two or three times by 10 subjects.

For a fair comparison, we used the leave-one-subject-out cross validation scheme proposed by [19]. The accuracy of the proposed methods and different state-of-the-art methods are given in Tab. III. For this dataset, SN4D-MSFV and SN4D-TSCFVC perform comparably with an accuracy of 95.39% and 95.27%, and are the best methods among the competing ones. The confusion matrices are shown in Fig. 4. Most of the confusions are between the actions *store* and *green* (SN4D-MSFV), *finish* and *bathroom* (SN4D-TSCFVC).

C. Effect of Higher-Order Partial Derivatives in The Local Descriptor

In this section, we compare the performance of different local descriptors obtained by combining the first three components of a surface normal with different higher-order partial derivatives of z along x and y axes. Three descriptors are considered: $\mathbf{d}_1 = [\frac{\partial z}{\partial x}; \frac{\partial z}{\partial y}; \frac{\partial z}{\partial t}]$, $\mathbf{d}_2 = [\frac{\partial z}{\partial x}; \frac{\partial z}{\partial y}; \frac{\partial z}{\partial t}; \frac{\partial^2 z}{\partial x^2}; \frac{\partial^2 z}{\partial y^2}]$, and $\mathbf{d}_3 = [\frac{\partial z}{\partial x}; \frac{\partial z}{\partial y}; \frac{\partial z}{\partial t}; \frac{\partial^2 z}{\partial x^2}; \frac{\partial^2 z}{\partial y^2}; \frac{\partial^3 z}{\partial x^3}; \frac{\partial^3 z}{\partial y^3}]$. We do not consider other higher-order partial derivatives of z along x and y axes as well as higher-order partial derivatives of z along t since in our experiments, we did not observe any improvement in recognition accuracy. Using MSFV and TSCFVC for feature encoding, we obtain 6 methods for comparison. Tab. IV shows the accuracy of these methods. For both the datasets, our descriptor outperforms the other descriptors if the same encoding method is used for all the descriptors.

Method	Accuracy
Kurakin et al., 2012 [18]	87.70
Wang et al., 2012 [19]	88.50
Yang et al., 2012 [20]	89.20
Oreifej and Liu, 2013 [1]	92.45
Yang and Tian, 2014 [2]	94.74
SN4D-MSFV	**95.39**
SN4D-TSCFVC	**95.27**

TABLE III
RECOGNITION ACCURACY COMPARISON OF OUR METHODS AND PREVIOUS APPROACHES ON MSRGESTURE3D.

D. Comparison of MSFV against FV

We compare SN4D-MSFV against two methods: SN4D-FV in which FV is used instead of MSFV in the encoding step, and SN4D-MSFV2 in which one GMM is trained for each scale and the final FV representation of a depth sequence is the concatenation of three FVs from three scales. Hence, the difference between SN4D-MSFV and SN4D-MSFV2 is

		MSRAction3D	MSRGesture3D	Ave.
$\mathbf{d}_1$	**MSFV**	88.23	95.21	91.72
	TSCFVC	94.74	94.28	94.51
$\mathbf{d}_2$	**MSFV**	91.37	95.03	93.2
	TSCFVC	95.15	94.14	94.65
$\mathbf{d}_3$	**MSFV**	92.32	94.55	93.43
	TSCFVC	95.15	94.83	94.99
Ours	**MSFV**	**93.27**	**95.39**	**94.33**
	TSCFVC	**95.42**	**95.27**	**95.35**

TABLE IV
EFFECT OF HIGHER-ORDER PARTIAL DERIVATIVES IN THE LOCAL DESCRIPTOR.

	SN4D-FV	SN4D-MSFV2	SN4D-MSFV
MSRAction3D	91.54	92.58	**93.27**
MSRGesture3D	94.2	94.74	**95.39**

TABLE V
COMPARISON OF SN4D-MSFV AGAINST SN4D-MSFV2 AND SN4D-FV.

	SNV-SC	SN4D-SCFVC	SN4D-TSCFVC
MSRAction3D	94.07	94.87	**95.42**
MSRGesture3D	94.29	94.02	**95.27**

TABLE VI
COMPARISON OF SN4D-TSCFVC AGAINST ITS VARIANTS.

that only one GMM is trained in SN4D-MSFV while three different GMMs are trained in SN4D-MSFV2. Tab. V shows the accuracy of these methods. As can be observed, SN4D-MSFV outperforms both SN4D-MSFV2 and SN4D-FV. The accuracy of the proposed multi-scale encoding method is 1.73% better than that of FV on MSRAction3D, and is 1.19% better than that of FV on MSRGesture3D.

E. Comparison of TSCFVC against SCFVC

We compare SN4D-TSCFVC against its two variants: SN4D-SCFVC in which TSCFVC is replaced by SCFVC, and SNV-SC in which TSCFVC is replaced by the encoding method of [2]. Tab. VI shows the accuracy of these methods. SN4D-TSCFVC outperforms SNV-SC by 1.35% on MSRAction3D and by 0.98% on MSRGesture3D. SN4D-TSCFVC outperforms SN4D-SCFVC by 0.55% on MSRAction3D and by 1.25% on MSRGesture3D.

V. CONCLUSIONS AND FUTURE WORK

We have proposed a new descriptor for human action recognition in depth images. Our proposed descriptor is based on surface normals in 4D space of depth, time, spatial coordinates and higher-order partial derivatives of depth values along spatial coordinates. We have proposed MSFV and TSCFVC to effectively encode local descriptors into a global representation of depth sequences. Action recognition can then be simply performed using a linear SVM. We have presented the experimental evaluation on two benchmark datasets showing the effectiveness of the proposed methods.

For future research, we study the fusion of the proposed descriptor with other descriptors. We also consider improving it by using the joint trajectories as methods which compute local descriptors along the trajectories of feature points have achieved state-of-the-art performance for human action recognition in color images.

REFERENCES

[1] O. Oreifej and Z. Liu, "HON4D: Histogram of Oriented 4D Normals for Activity Recognition from Depth Sequences," in *CVPR*, 2013, pp. 716–723.

[2] X. Yang and Y. Tian, "Super Normal Vector for Activity Recognition Using Depth Sequences," in *CVPR*, 2014, pp. 804–811.

[3] J. Sanchez, F. Perronnin, T. Mensink, and J. Verbeek, "Image Classification with the Fisher Vector: Theory and Practice," *IJCV*, vol. 105, no. 3, pp. 222–245, 2013.

[4] L. Liu, C. Shen, L. Wang, A. van den Hengel, and C. Wang, "Encoding High Dimensional Local Features by Sparse Coding Based Fisher Vectors," in *NIPS*, 2014, pp. 1143–1151.

[5] L. Xia, C. C. Chen, and J. K. Aggarwal, "View Invariant Human Action Recognition Using Histograms of 3D Joints," in *CVPRW*, 2012, pp. 20–27.

[6] L. R. Rabiner, "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–286, 1989.

[7] X. Yang and Y. L. Tian, "EigenJoints-based Action Recognition Using Naive-Bayes-Nearest-Neighbor," in *CVPRW*, 2012, pp. 14–19.

[8] O. Boiman, E. Shechtman, and M. Irani, "In Defense of Nearest-Neighbor Based Image Classification," in *CVPR*, 2008, pp. 1–8.

[9] M. Zanfir, M. Leordeanu, and C. Sminchisescu, "The Moving Pose: An Efficient 3D Kinematics Descriptor for Low-Latency Action Recognition and Detection," in *ICCV*, 2013, pp. 2752–2759.

[10] R. Vemulapalli, F. Arrate, and R. Chellappa, "Human Action Recognition by Representing 3D Skeletons as Points in a Lie Group," in *CVPR*, 2014, pp. 588–595.

[11] R. M. Murray, S. S. Sastry, and L. Zexiang, *A Mathematical Introduction to Robotic Manipulation*. CRC Press, Inc., 1994.

[12] M. Müller, *Information Retrieval for Music and Motion*. Springer-Verlag New York, Inc., 2007.

[13] J. Wang, Z. Liu, Y. Wu, and J. Yuan, "Mining Actionlet Ensemble for Action Recognition with Depth Cameras," in *CVPR*, 2012, pp. 1290–1297.

[14] A. Eweiwi, M. S. Cheema, C. Bauckhage, and J. Gall, "Efficient Pose-Based Action Recognition," in *ACCV*, 2014, pp. 428–443.

[15] G. Evangelidis, G. Singh, and R. Horaud, "Skeletal Quads: Human Action Recognition Using Joint Quadruples," in *ICPR*, 2014, pp. 4513–4518.

[16] W. Li, Z. Zhang, and Z. Liu, "Action Recognition Based on A Bag of 3D Points," in *CVPRW*, 2010, pp. 9–14.

[17] ——, "Expandable Data-Driven Graphical Modeling of Human Actions Based on Salient Postures," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, no. 11, pp. 1499–1510, 2008.

[18] A. Kurakin, Z. Zhang, and Z. Liu, "A Read-Time System for Dynamic Hand Gesture Recognition with A Depth Sensor," in *EUSIPCO*, 2012.

[19] J. Wang, Z. Liu, J. Chorowski, Z. Chen, and Y. Wu, "Robust 3D Action Recognition with Random Occupancy Patterns," in *ECCV*, 2012, pp. 872–885.

[20] X. Yang, C. Zhang, and Y. Tian, "Recognizing Actions Using Depth Motion Maps-based Histograms of Oriented Gradients," in *Proceedings of the 20th ACM International Conference on Multimedia*, 2012, pp. 1057–1060.

[21] K. Mikolajczyk and C. Schmid, "A Performance Evaluation of Local Descriptors," *TPAMI*, vol. 27, no. 10, pp. 1615–1630, 2005.

[22] L. Xia and J. K. Aggarwal, "Spatio-temporal Depth Cuboid Similarity Feature for Activity Recognition Using Depth Camera," in *CVPR*, 2013, pp. 2834–2841.

[23] C. Wang, Y. Wang, and A. L. Yuille, "An Approach to Pose-Based Action Recognition," in *CVPR*, 2013, pp. 915–922.

[24] Y. Du, W. Wang, and L. Wang, "Hierarchical Recurrent Neural Network for Skeleton Based Action Recognition," in *CVPR*, 2015, pp. 1110–1118.

[25] J. Schmidhuber, "Deep Learning in Neural Networks: An Overview," *Neural Networks*, vol. 61, pp. 85–117, 2015.

[26] S. Tang, X. Wang, X. Lv, T. X. Han, J. Keller, Z. He, M. Skubic, and S. Lao, "Histogram of Oriented Normal Vectors for Object Recognition with a Depth Sensor," in *ACCV*, 2013, pp. 525–538.

[27] F. Perronnin and C. Dance, "Fisher Kernels on Visual Vocabularies for Image Categorization," in *CVPR*, 2007, pp. 1–8.

[28] F. Perronnin, J. Sanchez, and T. Mensink, "Improving the Fisher Kernel for Large-scale Image Classification," in *ECCV*, 2010, pp. 143–156.

[29] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin, "LIBLINEAR: A Library for Large Linear Classification," *Journal of Machine Learning Research*, vol. 9, pp. 1871–1874, 2008.

[30] C. Chen, K. Liu, and N. Kehtarnavaz, "Real-time Human Action Recognition Based on Depth Motion Maps," *Journal of Real-Time Image Processing*, pp. 1–9, 2013.

[31] M. A. Gowayyed, M. Torki, M. E. Hussein, and M. El-Saban, "Histogram of Oriented Displacements (HOD): Describing Trajectories of Human Joints for Action Recognition," in *IJCAI*, 2013, pp. 1351–1357.