HON4D: Histogram of Oriented 4D Normals for Activity Recognition from Depth Sequences

Omar Oreifej
University of Central Florida
Orlando, FL
oreifej@eecs.ucf.edu

Zicheng Liu
Microsoft Research
Redmond, WA
zliu@microsoft.edu

Abstract

We present a new descriptor for activity recognition from videos acquired by a depth sensor. Previous descriptors mostly compute shape and motion features independently; thus, they often fail to capture the complex joint shape-motion cues at pixel-level. In contrast, we describe the depth sequence using a histogram capturing the distribution of the surface normal orientation in the 4D space of time, depth, and spatial coordinates. To build the histogram, we create 4D projectors, which quantize the 4D space and represent the possible directions for the 4D normal. We initialize the projectors using the vertices of a regular polychoron. Consequently, we refine the projectors using a discriminative density measure, such that additional projectors are induced in the directions where the 4D normals are more dense and discriminative. Through extensive experiments, we demonstrate that our descriptor better captures the joint shape-motion cues in the depth sequence, and thus outperforms the state-of-the-art on all relevant benchmarks.

1. Introduction

Depth sensors have been available for many decades. Though, their applications have been limited due to the high cost and complexity of operation. However, the recent emergence of low-cost depth sensors such as Kinect [18], triggered significant attention to revisit problems such as object detection and activity recognition using depth images as input instead of color.

Compared with conventional color images, depth maps provide several advantages. For example, depth maps reflect pure geometry and shape cues, which can often be more discriminative than color and texture in many problems including segmentation, object detection, and activity recognition. Moreover, depth images are insensitive to changes in lighting conditions. In this context, it seems natural to employ depth data in many computer vision problems. However, it is also intuitive to wonder whether conventional RGB-based methods would also perform well in depth sequences?

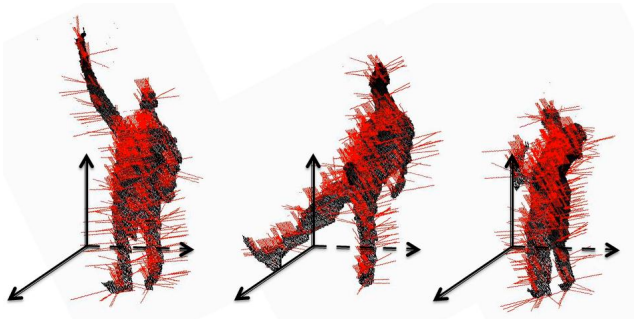

Figure 1. Surface normals overlayed on three examples from MSR Actions 3D dataset [12]. The surface normals capture the shape cues at a specific time instance, while the change in the surface normal over time captures the motion cues. In this paper, we use 4D normals computed in the space of depth, time, and spatial coordinates in order to obtain rich descriptors of activities. Note that in the figure we illustrate 3D surface normals since it is difficult to visualize the 4D normals used in the paper.

In activity recognition, which is the topic of this paper, two significant aspects arise when adopting conventional color-based methods for depth sequences. First, the captured depth images are often contaminated with undefined depth points, which appear in the sequence as spatially and temporally discontinues black regions [1]. This hinders the application of local interest points detectors such as Dollar [5] and STIP [10], which falsely fire on these regions instead of the regions where important events are occurring. To verify that, we conducted an experiment using MSR-Daily Activity Dataset [24], and found that 60% of the detected Dollar interest points were fired on locations irrelevant to the action of interest. Therefore, the corresponding classification accuracy is very low (52%). To handle that, recent approaches resorted to selecting the informative points using the joints from a skeleton detector [18], as in [24], or using a discriminative sampling scheme as in [23].

Second, and more importantly, the depth images provide natural surfaces which can be exploited to capture the geometrical features of the observed scene in a rich descriptor. For instance, it was recently shown in [20] that for the purpose of object detection, the shape of the object can be better described using the normal vectors in depth images, instead of the gradients in color images.

Our work in this paper proceeds along this direction. We propose a novel activity descriptor for depth sequences, which is analogous to the histogram of gradients in color sequences [4, 9], and extends the histogram of normals in static images [20]. As the depth sequence represents a depth function of space and time, we propose to capture the observed changing structure using a histogram of oriented 4D surface normals (HON4D). In order to construct HON4D, the 4D space is initially quantized using a regular 4D extension of a 2D polygon, namely, a 600-cell Polychoron [3]. Consequently, the quantization is refined using a novel discriminative density measure, which we compute along the quantized directions in the 4D space. Figure 2 summarizes the steps involved in computing HON4D.

Our proposed histogram operates in the 4D space, thus, jointly capturing the distribution of the changing shape and motion cues along with their correlations, instead of an adhoc concatenation of features as in [24]. Additionally, our descriptor is a holistic representation for the entire sequence; therefore, it is robust against noise and occlusion, from which local methods often suffer [23]. Moreover, it does not require a skeleton tracker as in [24] and [19]. Compared to the other holistic methods, we model the distribution of the normal vectors for each cell in the 4D space, which is richer and more discriminative than the average 4D occupancy used in [21]. Furthermore, unlike [26], the temporal order of the events in the sequence is encoded and not ignored. More importantly, as we will demonstrate, HON4D is superior in performance to all previous methods.

The main contributions of this paper are: First, we propose a novel descriptor for activity recognition from depth sequences, in which we encode the distribution of the surface normal orientation in the 4D space of depth, time, and spatial coordinates. Second, we demonstrate how to quantize the 4D space using the vertices of a polychoron, and then refine the quantization to become more discriminative. The rest of the paper is organized as follows: Section 2 reviews the related work. Section 3 describes the process of computing the 4D surface normals. In Section 4, we describe the quantization of the 4D space, and show how to build the HON4D. Section 5 describes the 4D quantization refinement approach. The experimental results are presented in Section 6. Finally, Section 7 concludes the paper.

2. Related Work

Early methods for activity recognition from depth sequences attempted to adopt techniques originally developed for color sequences. For instance, in [12], Li et al. proposed to obtain a bag of 3D points (analogous to a bag of words) by sampling points from the silhouette of the depth images, then clustering the points in order to obtain salient postures (vocabulary). Consequently, a GMM is used to globally model the postures, and an action graph [11] is used for inference. On the other hand, parallel to the approaches developed for temporal modelling of human actions in color videos such as [15, 2], Lv and Nevatia in [14] employ a hidden Markov model (HMM) to represent the transition probability for pre-defined 3D joint positions. Similarly, in [8], the 3D joint position is described using another generative model, which is a conditional random field (CRF).

Adopting local interest point-based methods to operate in depth sequences is difficult because, as discussed earlier, detectors such as STIP [10] and Dollar [5] are not reliable in depth sequences. Additionally, standard methods for automatically acquiring motion trajectories in color images as in [25, 22] are also not reliable in depth sequences. Therefore, recent methods for activity recognition in depth sequences resorted to alternative approaches in order to obtain reliable interest points and tracks. For instance, in [24], Jiang et al. extract the skeleton of the human using the skeleton tracking algorithm in [18]. Consequently, the joints of the skeleton are used as interest points. In that, the shape of the area surrounding the joint along with the joint location information are captured using a local occupancy pattern feature and a pairwise distance feature, respectively. These features are extracted for each joint at each frame, then the fourier transform coefficients are employed to describe the temporal variation of the features. On the other hand, in [23], random subvolumes are selected from the space of all possible subvolumes in the depth sequences. The subvolume selection is based on LDA [16], where the most discriminative subvolumes are retained.

Furthermore, holistic approaches for activity recognition from depth sequences are recently becoming popular. In that, instead of using local points, a global feature is obtained for the entire sequence. For example, in [26], the depth video is summarized in one image (a motion map), which is the average difference between the depth frames. Consequently, a single HOG descriptor is extracted from the motion map. This method collapses the temporal variations, and thus suffers when the temporal order is of significance. On the other hand, in [21], the sequence is divided into a spatiotemporal grid, then a simple feature called the global occupancy pattern is used, where the number of occupied pixels is recorded for each grid cell.

Our proposed method in this paper falls in the holistic methods category. Evidently, holistic methods such as

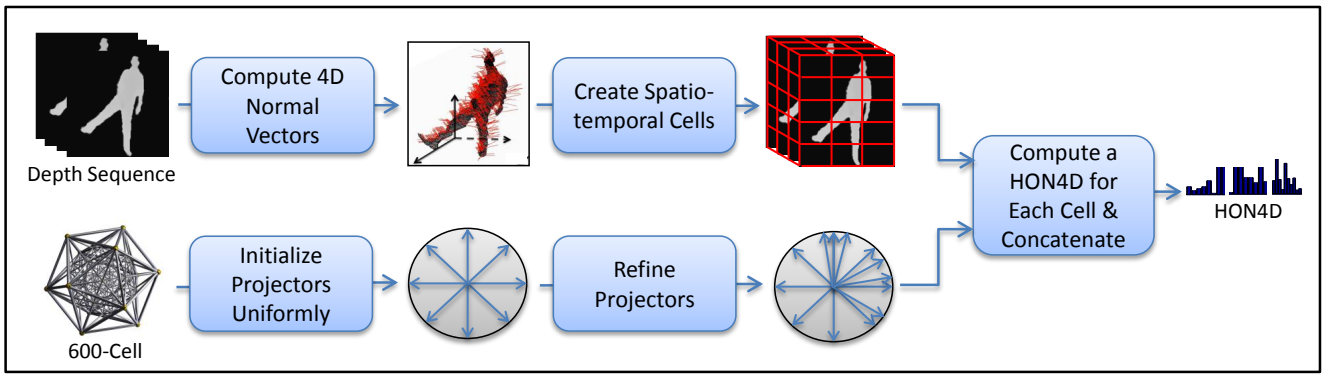

Figure 2. The various steps for computing HON4D descriptor.

[26, 21] are generally simpler, computationally efficient, and often outperform local approaches. We demonstrate that our method captures the complex and articulated structure and motion within the sequence using a richer and more discriminative descriptor than [21] and [26]. We additionally bypass the use of a skeleton tracker, which can often be unreliable. Though, we still outperform the methods which rely on the skeleton detector such as [24]. Moreover, since global descriptors generally assume coarse spatiotemporal alignment, we show that a local version of our descriptor can be derived and employed for significantly unaligned datasets.

3. The 4D Surface Normal

Given a sequence of depth images $\{I_1, I_2 \dots I_N\}$ containing a person performing an activity, our goal is to compute a global descriptor which is able to discriminate the class of action being performed. 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, t, z) satisfying $S(x, y, t, z) = f(x, y, t) - z = 0$. The normal to the surface S is computed as

$$\mathbf{n} = \nabla S = (\frac{\partial z}{\partial x}, \frac{\partial z}{\partial y}, \frac{\partial z}{\partial t}, -1)^T. \quad (1)$$

Only the orientation of the normal is relevant to the shape of the 4D surface S; therefore, we normalize the computed normal to a unit length normal $\hat{\mathbf{n}}$. Note that the components of the surface normal are the gradients in space and time, along with a scalar (-1). Therefore, the normal orientation might falsely appear as equivalent to the gradient orientation, and thus one might expect a histogram of 4D normal orientation (HON4D) to coincide with a histogram of 3D gradient orientation (HOG3D). In fact, there is an inherent difference, which allows the HON4D to capture richer information. The normal orientation has one extra dimension; therefore, the corresponding distribution over the bins is significantly different. Note that in a unit normal, the fourth dimension encodes the magnitude of the gradient $-1/||(f_x, f_y, f_t, 1)^T||_2$. This allows the HON4D to select different bins based on the gradient magnitude (i.e. two normals with different corresponding gradient magnitudes may fall into different bins). In contrast, in the histogram of gradient orientation, the magnitude is either ignored or only used as a weight for the bins.

To better illustrate that, consider for example the shapes in figure 3, which shows two space-time surfaces, where surface 1 has a higher inclination than surface 2. The gradient orientation is similar for both surfaces because the component of the gradient along the shape dimension is negligible. In contrast, the orientation of the normal is significantly different. Therefore, a histogram of normal orientation can differentiate between these surfaces, while a histogram of gradient orientation cannot. We argue, and verify by experiments, that the depth sequences provide natural surface functions, from which we can compute rich geometric properties such as the distribution of the normal orientation in 4D, without having to resort to less informative representations such as the gradient orientation. In the coming section we demonstrate how we compute the histogram of oriented normals in the 4D space.

4. Histogram of 4D Normals

Given the surface normals computed as in equation 1 using finite gray-value difference over all voxels in the depth sequence, we compute the corresponding distribution of 4D surface normal orientation. This requires quantizing the corresponding space into specific bins. In HOG, the gradient is two-dimensional; therefore, it is trivial to quantize a circle in order to obtain the bins of the histogram. Most recent implementations such as in [6] use predefined orientation vectors, and project the gradient to these vectors in order to obtain the corresponding response per direction. Consequently, either the maximum response is selected as the corresponding bin (hard-decision), or all the responses are accumulated (soft-decision). Similarly, HOG3D [9] employs an analogous process. In contrast, in the depth se-

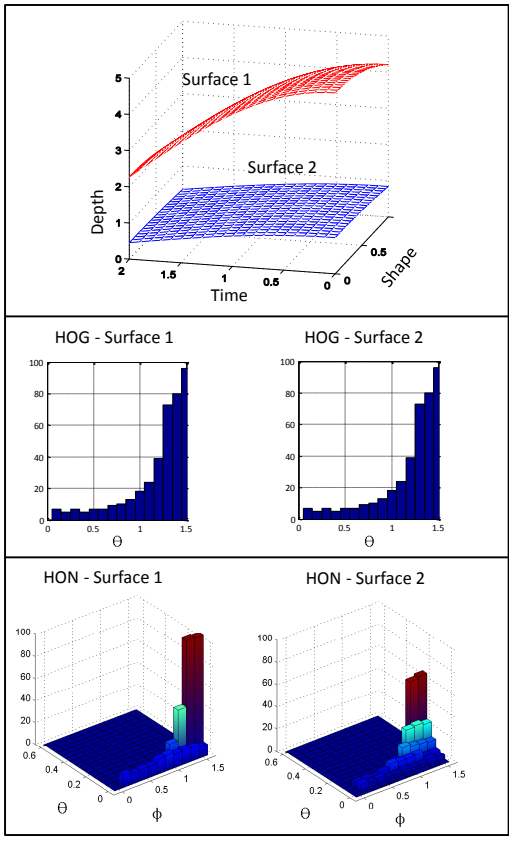

Figure 3. Example showing the difference between the gradient orientation histogram and the normal orientation histogram. For better visualization, in this example, we assume we have only one spatial dimension; therefore, we have 2D gradients and 3D normals instead of the actual 3D gradients and 4D normals of a depth sequence. The orientation of the gradient is determined by an angle Θ, while the orientation of the normal is determined by two angles, Θ and Φ. Top: Two surfaces produced as a result of a shape (line) moving in time, where surface 1 has a higher inclination than surface 2. Middle: The histogram of gradient orientation for surface 1 (left), and surface 2 (right). Bottom: The histogram of normal orientation for surface 1 (left), and surface 2 (right). The gradient direction for the two surfaces is similar (because the derivative along the shape dimension is negligible). Therefore, it cannot differentiate between the two surfaces. On the other hand, the direction of the normal for surface 1 is significantly different than surface 2, and the corresponding histogram of normal orientation evidently captures this difference.

quences, the obtained normal vectors live in a 4D space. In order to quantize the 4D space, we consider 4D regular geometric objects called polychorons [3, 7]. These objects are 4D extensions of a 2D polygon. A regular polychoron divides the 4D space uniformly with its vertices; therefore, it is a proper quantization of the 4D space. In particular, from the set of regular polychorons, we consider the 600-cell for two reasons: First, it has 120 vertices, which is a proper dimensionality size (compared for example to 600 vertices in the 120-cell). Second, empirically, the performance using the 600-cell is superior to the others.

In [3], it is shown that in the 4D space, the vertices of a 600-cell centered at the origin are given as:

- 8 vertices obtained by permutations of $(0, 0, 0, \pm 1)$.
- 16 vertices obtained by permutations of $(\pm 1/2, \pm 1/2, \pm 1/2, \pm 1/2)$.
- 96 vertices obtained by even permutations of $1/2(\pm\varphi, \pm 1, \pm 1/\varphi, 0)$, where $1/\varphi$ is the edge length of the 600-cell, and is set to a constant called the golden ratio $2/(1+\sqrt{5})$ [3].

We quantize the 4D space using these vertices, and refer to each vertex vector as a "projector". Therefore, given the set of 120 projectors $\mathcal{P} = \{\mathbf{p}_i\}$, and the set of unit normals $\mathcal{N} = \{\hat{\mathbf{n}}_j\}$ computed over all the spatiotemporal locations of the depth sequence, we compute the component of each normal in each direction by an inner product with the corresponding projector

$$c(\hat{\mathbf{n}}_j, \mathbf{p}_i) = max(0, \hat{\mathbf{n}}_j^T \mathbf{p}_i). \tag{2}$$

Therefore, the distribution of the 4D normal orientation for a depth sequence is estimated by accumulating the contributions from the computed normals, followed by a normalization using the sum across all projectors, such that the final distribution sums to one:

$$\Pr(\mathbf{p}_i|\mathcal{N}) = \frac{\sum_{j\in\mathcal{N}} c(\hat{\mathbf{n}}_j, \mathbf{p}_i)}{\sum_{\mathbf{p}_v\in\mathcal{P}} \sum_{j\in\mathcal{N}} c(\hat{\mathbf{n}}_j, \mathbf{p}_v)}. \tag{3}$$

Hence, we obtain a 120 dimensional HON4D descriptor for the video. In order to further introduce cues from the spatiotemporal context, we divide the sequence into $w \times h \times t$ spatiotemporal cells, and obtain a separate HON4D for each. The final descriptor is a concatenation of the HON4Ds obtained from all the cells.

5. Non-Uniform Quantization

Histogram-based descriptors (for example SIFT [13], SIFT 3D [17], HOG [4], HOG3D [9], and our proposed HON4D), mostly employ uniform space quantization in order to build their histograms. It is, however, not difficult to find examples where such quantization is not optimal. For instance, consider the case where two different classes of activities are quite close in the feature space such that their samples mostly fall in similar bins. This results in a significant confusion between the two classes, which could evidently be avoided through a finer quantization at the regions of confusion. As the dimension of the space to be quantized becomes larger (4D in our case), different possible quantizations could potentially be employed, and therefore this observation becomes more prominent.

Finding the optimal projectors (bins of the histogram) is unarguably a highly non-convex optimization process, since in principle, it should involve learning both the classifier and the projectors jointly. It is also likely that the resulting classifier will suffer from overfitting. Therefore, finding the optimal projectors is still an open-ended problem, which we leave for future work, and instead, we resort to relaxing the problem into refining the projectors to better capture the distribution of the normals in a discriminative manner. In particular, given a dataset with training HON4D descriptors $\mathcal{X} = \{\mathbf{x}_k\}$, note that each descriptor $\mathbf{x}_k$ is obtained for a video k by projecting the corresponding set of surface normals $\mathcal{N}_k = \{\hat{\mathbf{n}}_j\}$ on the projectors $\mathcal{P} = \{\mathbf{p}_i\}$ as in equation 2. Therefore, we can compute the density of the projectors by estimating how many unit normals fall into each of them

$$D(\mathbf{p}_i) = \sum_{k \in \mathcal{X}} \sum_{j \in \mathcal{N}_j} c(\hat{\mathbf{n}}_{k,j}, \mathbf{p}_i), \tag{4}$$

where $\hat{\mathbf{n}}_{k,j}$ is the unit normal number j from depth sequence k. It is obvious that the density in equation 4 is not discriminative, meaning that a bin with higher density does not necessarily contribute more in deciding to which class the sample $\mathbf{x}_k$ belongs. Now, consider a SVM classifier which scores a sample $\mathbf{x}_k$ using:

$$\text{score}(\mathbf{x}_k) = \sum_s \alpha_s \mathbf{w}_s^T \mathbf{x}_k, \tag{5}$$

where $\mathbf{w}$ is a support vector and α is the weight corresponding to the support vector, which are learned by minimizing a loss function such as the hinge loss. The final class decision is made by thresholding the score (typically using 0 threshold if a bias is also learned). Note that the set of support vectors $\mathcal{W} = \{\mathbf{w}_i\}$ correspond to videos selected from the training data and weighted in order to best discriminate between classes. Therefore, these specific samples directly contribute in the decision value. Based on that, a discriminative version of equation 4 can be formulated as

$$D_{disc}(\mathbf{p}_i) = \sum_{j \in \mathcal{W}} \sum_{k \in \mathcal{N}_j} \alpha_j c(\hat{\mathbf{n}}_{k,j}, \mathbf{p}_i). \tag{6}$$

Note that the density now is computed using only the weighted set of support vectors, which makes it more robust and discriminative. In other words, not only the projector with higher discriminative density D_{disc} has higher accumulation of normal vectors, but also it has a higher contribution in the final classification score. Therefore, it is intuitive to place more emphasis on that direction. To that end, we sort the projectors according to their discriminative density, and induce m random perturbations of each of the highest l projectors according to their density, where m is computed for a projector $\mathbf{p}_i$ as:

$$m(\mathbf{p}_i) = \begin{cases} \lambda \frac{D(\mathbf{p}_i)}{\sum_{\mathbf{p}_v \in \mathcal{P}} D(\mathbf{p}_v)} & \text{if } i \leq l \\ 0 & \text{if } i > l, \end{cases}$$

and λ is a parameters reflecting the total number of projectors to be induced. The random perturbations for projector $\mathbf{p}_i$ constitute a new set of projectors $\{\mathbf{p}_{i,q} | q = 1 \dots \lfloor m(\mathbf{p}_i) \rfloor\}$, which we compute as

$$\mathbf{p}_{i,q} = \frac{\mathbf{p}_i + \beta \mathbf{r}_q}{||\mathbf{p}_i + \beta \mathbf{r}_q||_2}, \tag{7}$$

where $\mathbf{r} \in \mathbb{R}^4$ is a unit random vector, and $(0 < \beta \ll 1)$ is the perturbation amplitude.

We augment the density-learned projectors to the original 120 projectors, and obtain the final set of projectors. Using that, we compute the final HON4D descriptors and train a new SVM. It is rather important to note the following: First, the initial SVM from which we learn the discriminative density is different from the final SVM we use for the classification. The final SVM is trained on newly induced projectors which have never been seen in the initial SVM. Second, only the training set is involved in learning the density. Therefore, the process of refining the projectors, as we verify in the experiments, is far from overfitting. We use a polynomial kernel in all experiments, though the proposed method of refining the projectors using the discriminative density is general enough to apply to any kernel.

6. Experiments

We extensively experimented on the proposed ideas using three standard 3D activity datasets including MSR Actions 3D [12], MSR Gesture 3D [23], and MSR Daily Activity 3D [24]. We additionally collected a new type of 3D dataset, which we refer to as "3D Action Pairs" dataset. The actions in the new dataset are selected in pairs such that the two actions of each pair are similar in motion (have similar trajectories) and shape (have similar objects); however, the motion-shape relation is different. This emphasizes the importance of capturing the shape and the motion cues jointly in the activity sequence as in HON4D, in contrast to capturing these features independently as in most previous methods. We discuss this in more detail in subsection 6.3. Both the code and the datasets are available on our website http://www.cs.ucf.edu/~oreifej/.

In all experiments, we initialize the projectors using the 120 vertices of the 600-cell polychoron, and compute the initial HON4D descriptors. Consequently, we learn the discriminative density, refine the projectors, and compute the final descriptors set. We finally end up with a number of projectors typically ~ 300, which becomes the dimensionality of the HON4D descriptor obtained per cell. All the parameters are learned using cross-validation. The frame size in all datasets is 320×240, and each video sequence is divided into spatiotemporal cells, which are typically $4 \times 3 \times 3$

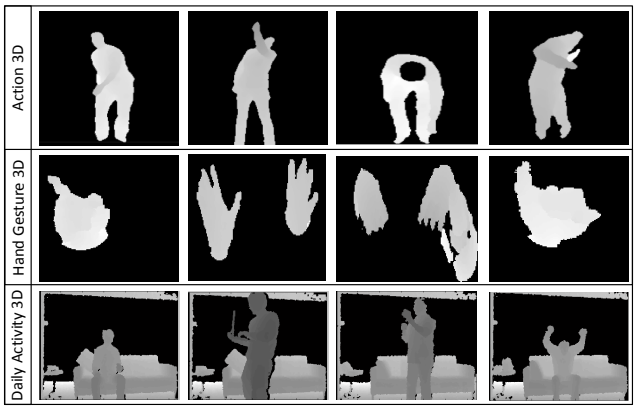

Figure 4. Example frames from different actions obtained from MSR Action 3D dataset [12], MSR Hand Gesture dataset [23], and MSR Daily Activity 3D [24].

in width, height, and number of frames, respectively. We compare with several recent methods including: (1) Yang et al. [26], where motion maps are obtained by accumulating the differences in the depth frames, and then HOG is used to describe the motion maps. (2) Klaser et al. [9], which employs a histogram of gradients in space-time (HOG3D). (3) Jiang et al. [24], where the local occupancy pattern features (LOP) are used over the skeleton joints. (4) Jiang et al. [23], where the depth sequence is randomly sampled then the most discriminative samples are selected and described using LOP descriptor. (5) Interest point detection within a bag of words framework, where the interest points are detected using Dollar detector [5] and STIP [10], then the descriptors are computed (spatiotemporal derivatives [5] and HOG/HOF [10]). Consequently, the descriptors are quantized and represented using a histogram of video words' frequency.

6.1. MSR Action 3D Dataset

MSR Action 3D dataset [24] is an action dataset of depth sequences captured by a depth camera. It contains twenty actions: "high arm wave", "horizontal arm wave", "hammer", "hand catch", "forward punch", "high throw", "draw x", "draw tick", "draw circle", "hand clap", "two hand wave", "sideboxing", "bend", "forward kick", "side kick", "jogging", "tennis swing", "tennis serve", "golf swing", "pick up & throw". Each action was performed by ten subjects for three times. Example depth sequences from this dataset are shown in figure 4.

In this dataset, the background is pre-processed to clear the discontinuities created from undefined depth regions. Nevertheless, this dataset is still challenging as many activities appear very similar. Using HON4D we obtain the state-of-the-art accuracy of 88.89% with the same experiment setup as in [24] (first five actors for training, and the rest for testing). Before refining the projectors, the obtained accuracy is 85.85%, which proves the advantage of our discriminative density method. We compare with several recent methods and summarize the results in table 1. It is important to note that in our method we do not use a skeleton tracker, and yet we outperform the skeleton-based method [24]. Additionally, note that the accuracy of [26] in table 1 is different than the number reported in their paper, the reason is that their experiment setup is different; therefore, we obtained their code and ran it within our experiment setup.

Table 1. The performance of our method on MSR Action 3D dataset, compared to previous approaches.

Method	Accuracy %
HON4D + D_{disc}	**88.89**
HON4D	85.85
Jiang et al. [24]	88.20
Jiang et al. [23]	86.50
Yang et al. [26]	85.52
Dollar [5] + BOW	72.40
STIP [10] + BOW	69.57
Vieira et al. [21]	78.20
Klaser et al. [9]	81.43

We further conduct a cross validation experiment to verify that the process of refining the projectors does not depend on specific training data. We consider all the possible combinations of choosing half of the actors for training, which are 252 folds for choosing 5 actors out of 10 in this dataset. At each fold, we train using all the videos from a certain combination of 5 actors, and test on the rest. We conduct this experiment first using the uniformly distributed projectors, and obtain an average accuracy of $79.38 \pm 4.40\%$ (mean $\pm$ std). Consequently, we conduct the experiment again, however, with refining the projectors at each fold, and obtain an average accuracy of $82.15 \pm 4.18\%$. This provides a clear evidence that the refined projectors do not depend on specific training data, and the corresponding trained models are not overfit.

6.2. MSR Hand Gesture Dataset

The Gesture3D dataset [23] is a hand gesture dataset of depth sequences captured by a depth camera. It contains a set of dynamic gestures defined by American Sign Language (ASL). There are 12 gestures in the dataset: "bathroom", "blue", "finish", "green", "hungry", "milk", "past", "pig", "store", "where", "j", "z". In total, the dataset contains 333 depth sequences, and is considered challenging mainly because of self-occlusion issues. Example frames from different gestures are shown in figure 4. We follow the experiment setup in [23] and obtain the accuracies described in table 2, where our descriptor outperforms all previous approaches.

Table 2. The performance of our method on MSR Hand Gesture 3D dataset, compared to previous approaches.

Method	Accuracy %
HON4D + D_{disc}	**92.45**
HON4D	87.29
Jiang et al. [23]	88.50
Yang et al. [26]	89.20
Klaser et al. [9]	85.23

6.3. 3D Action Pairs Dataset

This is a new type of activity dataset, which we collected in order to emphasize two points: First, though skeleton trajectories seem reliable when the person is in upright position, many actions share similar motion cues; therefore, relying on motion solely is insufficient. This was also pointed out in [24]. Second, the motion and the shape cues are correlated in the depth sequences, and it is rather insufficient to capture them independently. Therefore, in this dataset, we select pairs of activities, such that within each pair the motion and the shape cues are similar, but their correlations vary. For example, "Pick up" and "Put down" actions have similar motion and shape; however, the co-occurrence of the object shape and the hand motion is in different spatiotemporal order (refer to figure 6). This dataset is useful to evaluate how well the descriptors capture the prominent cues jointly in the sequence. We collected six pairs of actions: "Pick up a box/Put down a box", "Lift a box/Place a box", "Push a chair/Pull a chair", "Wear a hat/Take off a hat", "Put on a backpack/Take off a backpack", "Stick a poster/Remove a poster". Each action is performed three times using ten different actors, where the first five actors are used for testing, and the rest for training. We compare our performance in this dataset with three methods. First, we compute skeleton-based pair-wise features and LOP features as described in [24] and train a SVM on that. Second, we enhance the previous features by applying a temporal pyramid as described in [24]. Finally, we also compare with the motion map method from [26]. We summarize the results in table 3, and demonstrate the confusion tables in figure 5. It is clear that our method significantly outperforms the other approaches, which suffer from within-pairs confusion. In [24] (Skeleton + LOP), though both motion and shape features are obtained, they are simply concatenated; therefore, their relations are not encoded. Adding the temporal pyramid captures the temporal order and improves the accuracy, though still inferior to our method. Additionally, in [26], the whole sequence is collapsed into one image, which eliminates the temporal order of shape/motion cues, and thus this method suffers in this dataset. Our HON4D operates in the 4D space of shape and motion; therefore, it captures both features jointly, and outperforms the other methods significantly.

Table 3. The performance of our method on 3D action pairs dataset, compared to previous approaches.

Method	Accuracy %
HON4D + D_{disc}	**96.67**
HON4D	93.33
[24] (Skeleton + LOP)	63.33
[24] (Skeleton + LOP + Pyramid)	82.22
Yang et al. [26]	66.11

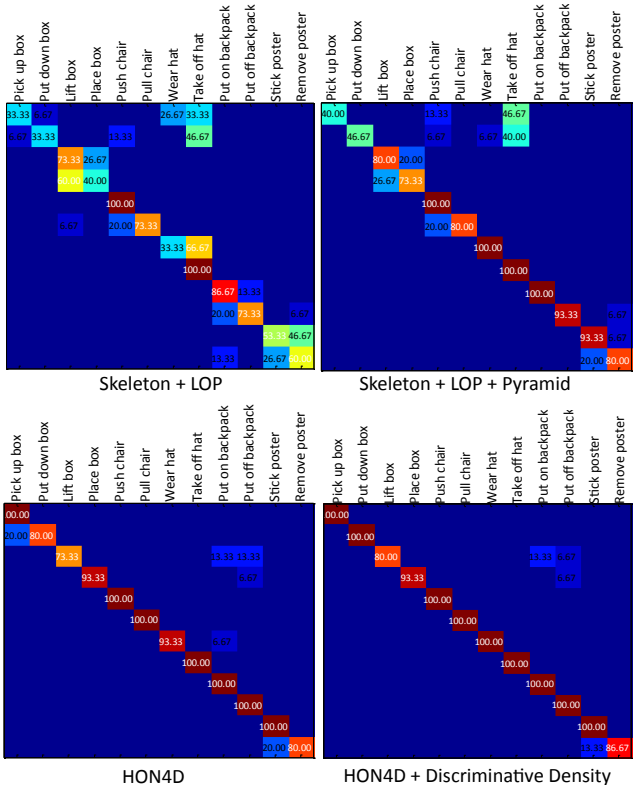

Figure 5. The confusion tables for 3D Action Pairs dataset. Top: Pair-wise skeleton features and LOP features from [24] without temporal pyramid (left), and with pyramid (right). Bottom: HON4D features as is (left), and after refining the projectors using the discriminative density (right).

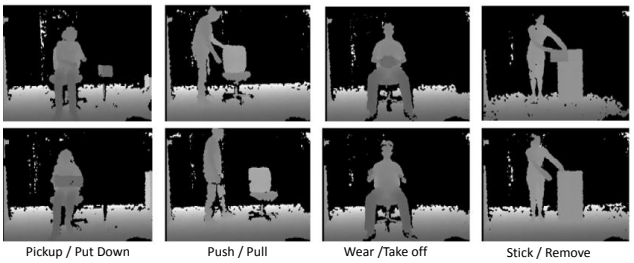

Figure 6. Example frames for four pairs from 3D Action Pairs dataset. Each column shows two images from a pair of actions. Note that, for example in the first column, both "Pick up a box" and "Put down a box" have similar motion and shape; however, they occur in different spatiotemporal order.

6.4. Local HON4D

The HON4D descriptor discussed earlier is a holistic feature, similar in that to [26] and [21]. This intrinsically assumes coarse spatial and temporal correspondence between the spatiotemporal cells across the sequences. This is a valid assumption for many practical scenarios, such as in the datasets discussed above, and generally in videos captured for Kinect applications and games. This assumption is also required (but often not explicitly mentioned) in some non-holistic methods as in [23].

On the other hand, in the scenarios where the actors significantly change their spatial locations, and the temporal extent of the activities significantly vary, we use a local HON4D descriptor, which is computed exactly as the global HON4D, except over spatiotemporal patches centered at skeleton joints obtained using [18]. We use a patch size of $12 \times 12 \times 6$, and divide it into a $3 \times 3 \times 1$ grid, where the numbers are selected using cross validation. To capture the temporal variation in the features, we follow a process similar to [24], however, replacing their LOP feature with the local HON4D. In particular, we compute the local HON4D for each joint, and for each frame, then the fourier transform is applied, and a SVM is trained on the fourier transform coefficients. For evaluation , we use the Daily Activity 3D Dataset [24], which contains 16 actions of common daily behaviors such as talking on the phone or reading a book ... etc. We achieve an average accuracy of 80.00%, compared to 67.50% when the original LOP feature is used, which proves that HON4D is also superior for significantly non-aligned sequences. It is important to note that [24] proposes additional steps to improve the accuracy, which generally apply to any descriptor. In our implementation, we do not include these steps, as our aim is to directly compare our descriptor with theirs.

7. Conclusion

We presented a novel, simple, and easily implementable descriptor for activity recognition from depth sequences. Our descriptor captures motion and geometry cues jointly using a histogram of normal orientation in the 4D space of depth, time, and spatial coordinates. We initially quantize the 4D space using the vertices of a 600-cell polychoron, and use that to compute the distribution of the 4D normal orientation for each depth sequence. Consequently, we estimate the discriminative density at each vertex of the polychoron, and induce further vertices accordingly, thus placing more emphasis on the discriminative bins of the histogram. We showed by experiments that the proposed method outperforms all previous approaches on all relevant benchmark datasets.

References

[1] M. Camplani, L. Salgado, and G. Imagenes. Efficient spatio-temporal hole filling strategy for kinect depth maps. *SPIE*, 2012. 1

[2] H. S. Chen, H. T. Chen, Y. W. Chen, , and S. Y. Lee. Human action recognition using star skeleton. *4th ACM international workshop on Video surveillance and sensor networks*, 2006. 2

[3] H. S. M. Coxeter. Regular polytopes. In *3rd. ed., Dover Publications. ISBN 0-486-61480-8*, 1973. 2, 4

[4] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. *CVPR*, 2005. 2, 4

[5] P. Dollar, V. Rabaud, G. Cottrell, and S. Belongie. Behavior recognition via sparse spatio-temporal features. *ICCV*, 2005. 1, 2, 6

[6] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, D., and Ramanan. Object detection with discriminatively trained part based models. *PAMI*, 2010. 3

[7] B. Grnbaum, V. Kaibel, V. Klee, and G. M. Ziegler. Convex polytopes (2nd ed.). In *New York and London: Springer-Verlag, ISBN 0-387-00424-6*, 2003. 4

[8] L. Han, X. Wu, W. Liang, G. Hou, and Y. Jia. Discriminative human action recognition in the learned hierarchical manifold space. *Image and Vision Computing*, 2010. 2

[9] A. Klaser, M. Marszalek, and C. Schmid. A spatio-temporal descriptor based on 3d-gradients. In *BMVC*, 2008. 2, 3, 4, 6, 7

[10] I. Laptev. On space-time interst points. In *IJCV*, 2005. 1, 2, 6

[11] W. Li, Z. Zhang, and Z. Liu. Expandable data-driven graphical modeling of human actions based on salient postures. *IEEE Transactions on Circuits and Systems for Video Technology*, 2008. 2

[12] W. Li, Z. Zhang, and Z. Liu. Action recognition based on a bag of 3d points. In *CVPR*, 2010. 1, 2, 5, 6

[13] D. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 2004. 4

[14] F. Lv and R. Nevatia. Recognition and segmentation of 3-d human action using hmm and multi-class adaboost. *ECCV*, 2006. 2

[15] H. Ning, W. Xu, Y. Gong, and T. Huang. Latent pose estimator for continuous action. *ECCV*, 2008. 2

[16] G. Ratsch, J. Weston, B. Scholkopf, and K. Mullers. Fisher discriminant analysis with kernels. *IEEE Signal Processing*, 1999. 2

[17] P. Scovanner, S. Ali, and M. Shah. A 3-dimensional sift descriptor and its application to action recognition. *ACM MM*, 2007. 4

[18] 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. *CVPR*, 2011. 1, 2, 8

[19] J. Sung, C. Ponce, B. Selman, and A. Saxena. Human activity detection from rgbd images. *In AAAI workshop on PAIR*, 2011. 2

[20] S. Tang, X. Wang, T. Han, J. Keller, M. Skubic, S. Lao, and Z. He. Histogram of oriented normal vectors for object recognition with a depth sensor. *ACCV*, 2012. 2

[21] A. W. Vieira, E. R. Nascimento, G. L. Oliveira, Z. Liu, , and M. F. M. Campos. Stop: Space-time occupancy patterns for 3d action recognition from depth map sequences. In *17th Iberoamerican Congress on Pattern Recognition (CIARP)*, 2012. 2, 3, 6, 8

[22] H. Wang, A. Kläser, C. Schmid, and C. Liu. Action recognition by dense trajectories. *CVPR*, 2011. 2

[23] J. Wang, Z. Liu, J. Chorowski, Z. Chen, , and Y. Wu. Robust 3d action recognition with random occupancy patterns. In *ECCV*, 2012. 1, 2, 5, 6, 7, 8

[24] J. Wang, Z. Liu, Y. Wu, , and J. Yuan. Mining actionlet ensemble for action recognition with depth cameras. In *CVPR*, 2012. 1, 2, 3, 5, 6, 7, 8

[25] S. Wu, O. Oreifej, and M. Shah. Action recognition in videos acquired by a moving camera using motion decomposition of lagrangian particle trajectories. *ICCV*, 2011. 2

[26] X. Yang, C. Zhang, , and Y. Tian. Recognizing actions using depth motion maps-based histograms of oriented gradients. In *ACM Multimedia*, 2012. 2, 3, 6, 7, 8