Skeleton-based Dynamic hand gesture recognition

Quentin De Smedt, Hazem Wannous, Jean-Philippe Vandeborre
Télécom Lille, Univ. Lille, CNRS, UMR 9189 - CRIStAL, F-59000 Lille, France
{desmedt,wannous,vandeborre}@telecom-lille.fr

Abstract

In this paper, a new skeleton-based approach is proposed for 3D hand gesture recognition. Specifically, we exploit the geometric shape of the hand to extract an effective descriptor from hand skeleton connected joints returned by the Intel RealSense depth camera. Each descriptor is then encoded by a Fisher Vector representation obtained using a Gaussian Mixture Model. A multi-level representation of Fisher Vectors and other skeleton-based geometric features is guaranteed by a temporal pyramid to obtain the final feature vector, used later to achieve the classification by a linear SVM classifier.

The proposed approach is evaluated on a challenging hand gesture dataset containing 14 gestures, performed by 20 participants performing the same gesture with two different numbers of fingers. Experimental results show that our skeleton-based approach consistently achieves superior performance over a depth-based approach.

1. Introduction

Among other human body parts, the hand is the most effective interaction tool in mostly Human-Computer Interaction (HCI) applications. To date, the most reliable tools used to capture the hand gesture are motion capture magnetic devices, which employ sensors attached to a glove able to determine precisely the hand gesture, delivering real-time measurements of the hand. However, they present several drawbacks in terms of the naturalness of hand gesture, price, in addition to their complex calibration setup process.

Recently, thanks to the advence in information technologies, effective and inexpensive depth sensors, like Microsoft Kinect or Intel RealSense, are increasingly used in the domain of computer vision. The development of theses sensors has brought new opportunities for the hand gesture recognition area. Compared to 2D cameras, these sensors are more robust concerning common low-level issues in RGB imagery like background subtraction and light variation.

Hand gesture recognition is becoming a central key for different types of application such as virtual game control, sign language recognition, human computer interaction, robot control, etc. Consequently, the improvements in hand gesture interpretation can benefit a wide area of research domains. In this paper, we present a novel hand gesture recognition solution, where the main advantage of our approach is the use of 3D skeleton-based features. We also contribute to the community with a new depth and skeleton-based dynamic hand gesture dataset. The rest of this paper is structured as follows. Related work on hand gesture in terms of datasets and recognition approaches are briefly reviewed in Section 2. In Section 3, we provide details on our dynamic hand gesture dataset. Our recognition approach is described in Section 4. The experimental results are presented in Section 5 before concluding.

2. Related work

Hand gesture recognition has been an active research field for the past 20 years, where various different approaches have been proposed. Over the past six years, advances in commercial 3D depth sensors have substantially promoted the search of hand gesture detection and recognition. The most of recent works in human motion analysis pay more attention to the full-body human poses and actions [3, 23]. Some other works have focused on the movements of certain body parts like hands [19]. The approaches reviewed mainly focus on 3D hand gesture recognition, which can be gathered into two main categories so far: **static** and **dynamic** hand gesture recognition.

In most of the **static** approaches, 3D depth information can be used to extract hand silhouettes or simply hand areas and the focus will be on the feature extraction from segmented hand region. Features are usually based on a global information as proposed by Kuznetsova et al. [9], where an ensemble of histograms is computed on random points in the hand point cloud. Other local descriptors are expressed as the distribution of points in the divided hand region into cells [26]. Instead of using the distribution of points in the region of the hand, Ren et al. [19] represented the hand shape as time-series curve and used distance metric called Finger-Earth Mover Distance to distinguish hand

gestures from collected dataset of 10 different gestures. The time-series curve representation is also used by Cheng et al. [2], to generate a fingerlet ensemble representing the hand gesture. Sign language recognition with hand gestures has been widely investigated. Pugeault and Bowden [18] proposed a method using Gabor filter for hand shape representation and a Random Forest for gesture classification. They applied their method on a collected *ASL Finger Spelling* dataset, containing 48000 samples of RGB-D images labelled following 24 static gestures of the American Sign Language. Recently, Dong et al. [4] outperformed the previous results on this database by going more deeply into the hand representation. They proposed a hierarchical mode-seeking method to localize hand joint positions under kinematic constraints, segmenting the hand region into 11 natural parts (one for the palm and two for each finger). A Random Forest classifier is then built to recognize ASL signs using a feature vector of joint angles. Finally, Marin et al. [11] released a publicly database of 10 static hand gestures giving the depth image from a *Kinect* but also information about the hand using the hand pose recognition device *LeapMotion*. They also proposed a classification algorithm using fingertips distances, angles and elevations and also curvature and correlation features on the depth map.

Unlike the static approaches based on hand description on a single image, **dynamic** methods exploit the temporal character of hand motion, by considering the gesture as a sequence of hand shape. Kurakin et al. [8] presented the MSR-3D hand gesture database containing 12 dynamic *American Sign Language*. They recorded 360 sequences of depth images from a *Kinect*. Their recognition algorithm is based on a hand depth cell occupancy and a silhouette descriptor. They used an action graph to represent the dynamic part of the gesture. Recently, using a histogram of 3D facets to encode 3D hand shape information from depth maps, Zhang et al. [28] outperformed last results on the MSR 3D gesture dataset using a dynamic programming-based temporal segmentation. One of the track of the *Chalearn 2014* [5] consists in using a multimodal database of 4,000 gestures drawn from a vocabulary of 20 dynamic Italian sign gesture categories. They provided sequences of depth images of the whole human body and body skeletons. On this database, Monnier et al. [13] employ both body skeleton-based and Histogram of Oriented Gradients (HOG) features on the depth around the hand to perform a gesture classification using a boosted cascade classifier. Recently, the use of deep learning has changed the paradigm of many research fields in computer vision. Recognition algorithms using specific neural network — like Convolutional Neural Network (ConvNet) – obtain previously unattainable performance in many research field. Still on the *Chalearn 2014* [5], Neverova et al. [14] used stacked ConvNets on raw intensity and depth sequences around the hand and neural network on body skeletons. In order to study real-time hand gesture recognition for automotive interfaces, Ohn-Bar and Trivedi [15] made a publicly available database of 19 gestures performed in a car using the *Kinect*. The initial resolution obtained by such a sensor is 640x480 and the final region of interest is 115x250. Moreover, at some distance from the camera, with the illumination varying in the car, the resulting depth is very noisy, making the challenge of gesture recognition thougher. They compare the accuracy of gestures recognition using several known features (HOG, HOG3D, HOG^2). Using stacked 3D ConvNets combining multiple spatial scales, Molchanov et al. [12] recently outperformed their results.

In contrast to activity and action recognition, we can notice from this brief review a lack of publicly available dynamic hand gesture datasets for benchmarking and comparing methods for hand gesture recognition. Even for existing ones, there is no available dataset that provides both depth and 3D joint hand with ground-truth. In term of recognition approaches, there would still appear to be room for improvement, especially using recent approaches of hand pose estimation [25].

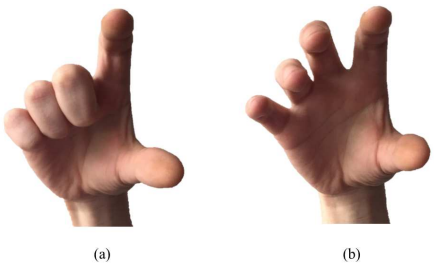
3. Dynamic Hand Gesture dataset (DHG-14/28)

Skeleton-based action recognition approaches have become popular as Shotton et al. [22] proposed a real-time method to accurately predict the 3-D positions of body joints from depth images. Hence, several descriptors in the literature proved how the position, motion, and orientation of joints could be excellent descriptors for human actions. Collected datasets for action recognition purpose like [27, 10] provide usually the depth data in addition to the 3D body skeleton of the person performing the action. However, in the context of hand gesture recognition, there are no publicly released dataset of dynamic hand gestures providing sequences of labelled hand gestures with the depth *and* hand skeleton. We present below a *Dynamic Hand Gesture 14-28* (DHG) dataset, which provides sequences of hand skeleton in addition to the depth image. Such a dataset will facilitate the analysis of hand gestures and open new scientific axes to consider[1].

3.1. Overview and protocol

The DHG-14/28 dataset contains 14 gestures performed in two ways: using one finger and the whole hand (an example is shown in Figure 1). Each gesture is performed 5 times by 20 participants in 2 ways, resulting in 2800 sequences. Sequences are labelled following their gesture, the number of fingers used, the performer and the trial. Each frame contains a depth image, the coordinates of 22 joints both in the

[1]Downloadable at: http://www-rech.telecom-lille.fr/DHGdataset



(a) (b)

Figure 1. Two images of a hand illustrating *Grab* gesture performed (a) with one finger and (b) with the whole hand.

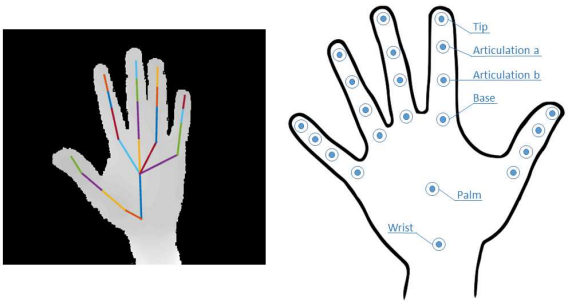


Figure 2. Depth and hand skeleton of the DHG-14/28 dataset. The 22 joints of the hand skeleton returned by the *Intel RealSense* camera. The joints include: 1 for the center of the palm, 1 for the position of the wrist and 4 joints for each finger represent the tip, the 2 articulations and the base. All joints are represented in $\mathbb{R}^3$.

2D depth image space and in the 3D world space forming a full hand skeleton. The *Intel RealSense* short range depth camera is used to collect our dataset. The depth images and hand skeletons were captured at 30 frames per second, with a 640x480 resolution of the depth image. The length of sample gestures ranges goes from 20 to 50 frames.

Fothergill et al. [7] investigated the problem of the most appropriate semiotic modalities of instructions for conveying to performers the movements the system developer needs to perform. They found out that a gesture recognition algorithm not only must need examples of desired gestures but also in order to cope with a wide array of users, the dataset must include common desired variants of the gestures. To achieve a good correctness in our dataset, we use 2 semiotic modalities to explain what we waited from our performers. First, the register explains in an abstractive way the gesture (example for a swipe gesture with one finger: "You're going to mime a swipe in the air with only one finger"), then we were showing them a video of someone performing the gesture.

In terms of hand pose estimation, much attention has been received over the last two years in the computer vision community [25, 21]. The Software Development Kit (SDK) released for *Intel RealSense* F200 provides a full 3D skeleton of the hand corresponding to 22 joints labelled as shown in Figure 2. However, the sensor still has trouble to properly recognize the skeleton when the hand is closed, perpendicular to the camera, without a well initialization or when the user performs a quick gesture. Our participants are asked to start each sequence by one or two seconds of the hand well opened in front of the camera. This may be necessary for some state-of-the-art hand pose estimation algorithms requiring an initialization, which can be tested on our depth sequences. For those who do not need initialisation, we manually labelled the effective beginning and end of each gesture sequence.

3.2. DHG-14/28 challenges

The list of our gestures proposed can be found in Table 1. Most of them have been chosen to be close to the state-of-the-art, like the VIVA challenges dataset [15]. Nevertheless, we removed the differentiation between normal and scroll swipe as you can find it in our number-of-fingers approach. The same thing appears with the pair of gesture *Pinch/Expand* and *Open/Close*. In addition, we supplement this base with the gesture *Grab* because of its usefulness in the augmented reality applications, but also for its scientific challenges related to the high potentially variation among performers. We also add the gesture *Shake*, as it can be interesting for recognition algorithm to be able to differentiate gesture composed of other gestures (a shake gesture can be seen as a repetition of opposed swipe gestures).

We emphasized our main challenges compared to existing hand gesture datasets: (1) Study the dynamic hand gesture recognition using depth *and* full hand skeleton; (2) Evaluate the effectiveness of recognition process in terms of coverage of the hand shape depending on the number of fingers used. The same movement is performed with one or more fingers, and the sequence are labelled according to 28 label classes, depending on the gesture represented and the number of fingers used; (3) Make distinctions between both fine-grained and coarse-grained gestures. Indeed, dividing the gesture sequences in two categories: coarse and fine gesture sequences contribute to increasing difficulty facing the recognition algorithm. Gesture categories are given in Table 1.

4. Feature extraction from 3D skeleton

In order to represent a hand gesture entirely, we propose to mainly capture the hand shape variation based on skeleton joints, but also the movement and the rotation of the hand in space are also computed. The temporal nature of gestures is encoded using a temporal pyramid and the classification process is performed by a linear Support Vector Machines (SVM) classifier. Figure 3 shows a general overview of the proposed approach.

Gesture	Labelization	Tag name
Grab	Fine	G
Expand	Fine	E
Pinch	Fine	P
Rotation CW	Fine	R-CW
Rotation CCW	Fine	R-CCW
Tap	Coarse	T
Swipe Right	Coarse	S-R
Swipe Left	Coarse	S-L
Swipe Up	Coarse	S-U
Swipe Down	Coarse	S-D
Swipe X	Coarse	S-X
Swipe V	Coarse	S-V
Swipe +	Coarse	S-+
Shake	Coarse	Sh

Table 1. List of the gestures included in the DHG-14/28 dataset.

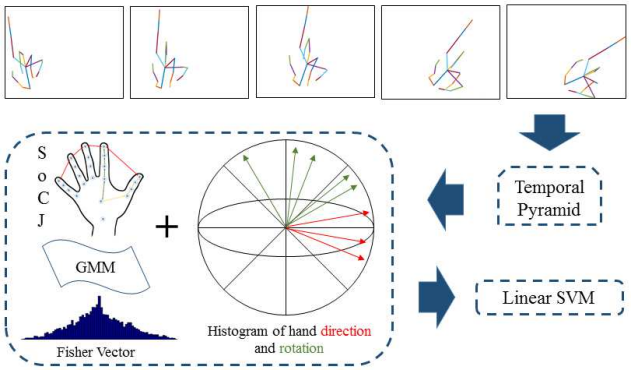


Figure 3. Pipeline of our gesture recognition system. A fisher vector representation is computed from our SoCJ descriptor. The later is concatenate with histograms of the hand direction and the wrist orientation. A temporal pyramid is used to take into account the temporal information and a linear SVM is used for classification.

4.1. Shape of Connected Joints (SoCJ)

To represent the hand shape using a full skeleton, we propose a new descriptor based on several relevant sets of joints, denoted as *Shape Of Connected Joints* (SoCJ).

Hand skeleton returned from sensor consists of 3D coordinates of hand joints, represented in the camera coordinate system. Therefore, they vary with the rotation and translation of the hand with respect to the camera. To make our hand shape descriptor relatively invariant to hand geometric transformations, we normalize it following 2 steps. Firstly, we removed the difference of hand size between performers, by estimating the average size of each bone of the hand skeleton. Then, carefully keeping the angles between bones, we change their sizes by their mean found previously. Secondly, we create a fake hand H_f which is open and in front of the camera with its palm node at $[0\ 0\ 0]$. Let

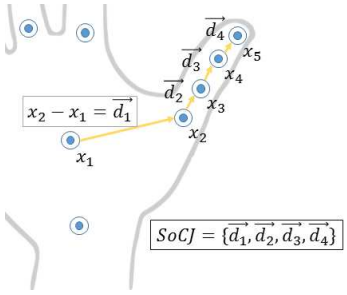


Figure 4. An example of the SoCJ descriptor. A 5-tuples is constructed using the thumb joints, $T = \{x_1, x_2, x_3, x_4, x_5\}$ where $x_i \in \mathbb{R}^3$. We compute the displacements from points to their respective right neighbor resulting in the SoCJ vector $\left\{\vec{d_1}, \vec{d_2}, \vec{d_3}, \vec{d_4}\right\}$.

B_f be a set of 2 vectors $\in \mathbb{R}^3$ defined by the coordinates in H_f of the vectors going from the palm node and respectively to the wrist node and to the base of the thumb. For each hand in a sequence, we create the same set of vectors B_c and we compute the optimal translation and rotation using a *Singular Value Decomposition* from B_c to B_f. Once the optimal translation and rotation are found, we apply this transformation to all joints of the hand skeleton resulting of a skeleton centered around $[0\ 0\ 0]$ and its palm facing the camera.

To describe the hand shape, we use nine 5-tuples of joints according to the hand physical structure on which we will perform our SoCJ descriptor. Five of these 5-tuples are constructed with the 4 joints of each finger plus the palm one. The 4 remaining concern the 5 tips, the 5 first articulations, the 5 second articulations and the 5 bases. Notice that the points of each tuple follow the same order.

Let $T_j = \{x_1, x_2, x_3, x_4, x_5\}$ be a 5-tuple and x_i a point in $\mathbb{R}^3$ representing one particular joint coordinate. To represent the shape of the joint connections, we compute the displacement from one point to its right neighbor:

$$SoCJ(T_j) = \{x_{i+1} - x_i\}_{[i=1...4]} \quad (1)$$

This results in a descriptor in $\mathbb{R}^{12}$. We compute our 9 SoCJs on each frame and regroup them along the sequence resulting in a set $T_{seq} = \{T_j\}_{[1 \leq j \leq 9N]}$ where N is the number of frame in the sequence. Figure 4 shows an example of a particular SoCJ around a thumb.

4.2. Fisher Vector representation

Fisher Vector (FV) coding method was firstly proposed for large-scale image classification. Il can be considered as an extension of the Bag-Of-Word (BOW) method by going beyond count analysis. It encodes additional information about the distribution of the descriptors. Its superiority against BOW has been analysed in the image classification

[20]. It also has been used over the past years in action recognition [6, 16].

The FV consists of fitting K parametric models to the descriptor and then encoding the derivative of each log-likelihood of the models with respect to their parameters. The common way to obtain such models is to train a K-component Gaussian Mixture Model (GMM). We denote the parameters of a GMM by $\lambda = \{\pi_k, \mu_k, \sigma_k\}_{[1 \leq k \leq K]}$ where π_k, μ_k, σ_k are respectively the prior weight, mean and covariance of the Gaussian k. After the training process, we are able to model any new sequence represented by its set of SoCJ, T_{seq}, as follow:

$$p(T_{seq}|\lambda) = \prod_{j=1}^{9N} \sum_{k=1}^{K} \pi_k p(T_j|\lambda_k) \tag{2}$$

Once we have the set of Gaussian Models, we can compute our FV, which is given by the gradient of the formula of Eq. (2):

$$\mathcal{G}_{\lambda}^{T_{seq}} = \frac{1}{9N} \nabla_{\lambda} \log p(T_{seq}|\lambda) \tag{3}$$

The normalization term $\frac{1}{9N}$ avoids dependency related to the size of T_{seq}. The derivatives in Eq. (3) are computed separately with respect to mean and standard deviation parameters, leading to the final Fisher Vector :

$$\Phi(T_{seq}) = \{\mathcal{G}_{\mu_k}^{T_{seq}}, \mathcal{G}_{\sigma_k}^{T_{seq}}\}_{[1 \leq k \leq K]} \tag{4}$$

Where $\mathcal{G}_{\mu_k}^{T_{seq}}$ and $\mathcal{G}_{\sigma_k}^{T_{seq}}$ have the same size as the descriptor used to train the GMM. We also normalize the final vector with a $l2$ and power normalization to eliminates the sparseness of the FV and increase its discriminability. We refer the reader Sanchez et al. [17] for more details.

We noticed that the final size of a Fisher Vector is $2dK$ where d is the size of the descriptor and K the number of cluster in the classification process. It can be a strong disadvantage against BOW, which has a size of K, when applying on a long descriptor.

4.3. Other relevant features

We chose to characterize the different aspects of the hand movement independently. To this end, before normalizing the hand in order to extract its shape information, we computed two other descriptors:

Histogram of hand directions (HoHD): Some gestures are defined practically only by the way the hand moves into space (e.g. *swipes*). To take this information into account, we first computed a direction vector using the position of the palm node noted x_{palm} along the sequence.

$$d(S) = \{x_{palm}^{t} - x_{palm} t - L\}_{[L+1 \leq i \leq N]}$$

where N is the size of the sequence and L a constant chosen by experiment. As the amplitude of the movement may vary from performer to another, we remove the magnitude of each vector using spherical coordinate representation (ρ, θ, φ), letting aside the radial distance ρ. The range of the features are $0 \leq \theta \leq \pi$ and $0 \leq \varphi \leq 2\pi$. Inspired by [24], which previously worked in 2D, the space of θ and φ are respectively divided in n_θ and m_φ resulting in a global partitioning of the 3D space into $n_\theta m_\varphi$ bins. Each direction are then localized at a unique bin and used to construct a histogram of size $n_\theta m_\varphi$. Figure 5 shows the construction of this descriptor.

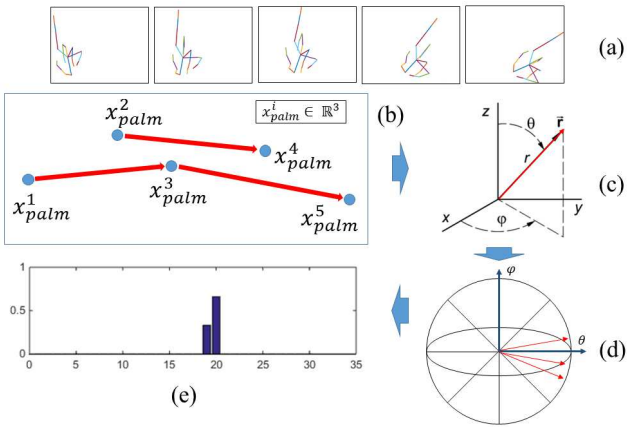


Figure 5. Computing of histogram of hand direction: (a) an example of *Swipe Right* gesture of 5 frames, (b) only the palm joint of each frame is kept and the direction vectors is computed with an offset L = 2, (c) each direction vector is then represented in the spherical coordinate, (d) the 3D space is divided into N bins allowing to localize each direction vector (e) the resulting N-dimensional histogram.

Histogram of wrist rotations (HoWR): The rotation of the wrist can also be important for some gestures (e.g. *R-CW, R-CCW*). For each frame, we use the direction vector from the wrist node to the palm node to get the rotational information of the hand skeleton. As for the HoHD, we transpose our vector into the spherical coordinates, divide the space into $n_\theta m_\varphi$ bins, localize our vector into a unique one and construct a histogram.

4.4. Temporal modelling and classification

Our three descriptors SoCJ, HoHD and HoWR allow us to describe the hand shape and geometric variation inside the sequence without taking into consideration the temporal nature of the gesture. Some inversed gestures like *Pinch / Expand* may be confused in this case. To add the temporal information, we use the simple representation called *Temporal Pyramid* (TP) which is widely used in action and hand gestures recognition [6, 28]. The principle of the TP is to divide the sequence into j sub-sequences at each j^{th} level of the pyramid (Figure 6). We compute our three descriptors en each sub-sequence and concatenate them. Adding more

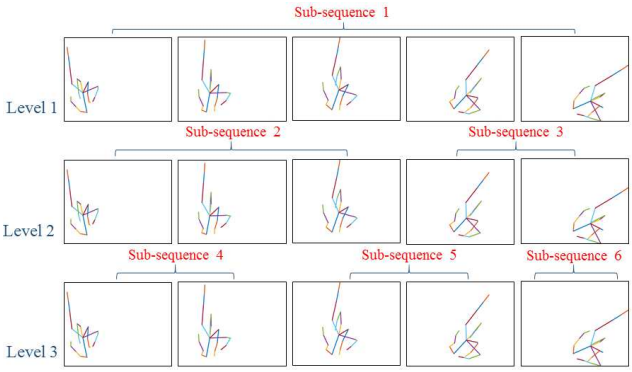


Figure 6. An example of a temporal pyramid of size 3. We compute each descriptor on each sub-sequence. Notice that a TP of size 3 multiply the final size of our descriptor by 6.

level to the pyramid allows the results to be more precise, but increase the size of the final descriptor and the computing time substantially.

The final size of our descriptor is then $(\sum_{i=1}^{L_{pyr}} i) \times (2kD + 2n_\theta m_\varphi)$, where L_{pyr} is the level of the TP, k is the number of cluster in the GMM, D is the size of the SoCJ descriptor and $n_\theta m_\varphi$ is the number of bins in the rotation and direction histograms. For gesture classification, we used the supervised learning classifier SVM. We choose the linear kernel as it easily deal with our high-dimensional representation. We employed a *one-vs-rest* strategy resulting in G binary classifier, where G is the number of different gestures in the experiment. We make use of the implementation contained in the LIBSVM package [1].

5. Experiments

First, we evaluate our approach in two cases by considering 14 and 28 classes of gestures thus taking account of the number of fingers used. Then, a comparison analysis on depth-vs-skeleton based descriptors is presented. Finally, we discuss the impact of taking into account the number of fingers in the gesture recognition accuracy. For each gesture, 9 SoCJs of size $D = 12$ per frame are computed. We also compute a GMM of 128 clusters using SoCJs from the training data, thus leading to 3072-element FVs. For the HoHD and the HOWR, the values, n_θ and m_φ, used for partitioning the 3D space into bins are respectively 8 and 6. Finally, we consider 4 levels ($L_{pyr} = 4$), leading to 10 FVs (1 + 2 + 3 + 4) , resulting in 31680-dimensional vector descriptor by sequence.

For all following experiments, we use a *leave-one-subject-out cross-validation* strategy. The dataset contains 2800 sequences of hand gestures. The depth images and hand skeletons are provided with some others information (e.g. timestamp, region of interest of the hand in the depth images,...). Each sequence is labelled following the gesture

Features	fine	coarse	both
HoHD	39.90%	83.06%	67.64%
HoWR	42.70%	31.67%	35.61%
SoCJ	67.40%	61.00%	63.29%
SoCJ + HoHD	70.70%	88.72%	82.29%
SoCJ + HoHD + HoWR	73.60%	88.33%	**83.07**%

Table 2. Results of our method for 14 gestures on the DHG dataset using **skeleton** data. Fine and Coarse columns are respectively the mean accuracies of fine and coarse gestures, obtained from the confusion matrix of Figure 7.

represented, a performer id and the number of fingers used while performing the gesture. Notice that the sequences are previously cropped using the effective beginning and end of the gestures manually labelled by the dataset makers.

5.1. 14-gestures classification

To assess the effectiveness of our algorithm to classify the gestures of the DHG dataset into 14 classes, we compare the results for each descriptor separately. The Table 2 presents the results of our skeleton-based approach obtained using each of our descriptors independently and by combining them. The results introduced in this table represent mean accuracies calculated for each descriptor. For clarity, we divide the result by coarse and fine gestures according to the labels from Table 1, allowing us to analyse the impact of each descriptor on each category.

Using all skeleton-based descriptors presented in Section 4, the final accuracy of our algorithm on the DHG-14 is 83%. It can reach 88% of recognition for the coarse gestures, but for the fine ones the accuracy is below the 75%. However, a large difference can be observed between accuracies obtained for the fine and the coarse gestures, respectively 40% and 83% when using only HoHD. These results attest the interest of the subdivision of our dataset into 2 meaningful sets of gestures where the coarse one can be more described by the movement of the hand through space.

The analysis of the results obtained using only our SoCJ descriptor encoded by its FVs, shows that the hand shape is the most effective feature for the fine gestures with an accuracy of 67%. On the other hand, this result shows that the hand shape is also a way to describe the coarse gestures with a not-so-low accuracy of 61%. If the HoWR descriptor shows a low mean accuracy of 36% , it's a valuable feature for pair of alike gestures as *R-CW* and *R-CCW*, and exclude it decreases the accuracy of 3% concerning the fine gestures.

To better understand the behaviour of our approach according to the recognition per class, the confusion matrix is illustrated in Figure 7.

The first observation is that using our approach, 10 gestures out of 14 are more than 85% correctly classified. The

Figure 7. The confusion matrix of the proposed approach for DHG-14

second observation concern a wide confusion between the gestures *Grab* and *Pinch*. By analyzing the sequences, we observe that they are very similar and hard to distinguish even for human eyes. The main difference between them is the amplitude of the hand movement in the space. As our method doesn't take this information into account, we let it for future work and as an open challenge for the community.

As shown in Table 2, combining the descriptors leads to a significant gain in performance (+15%). With a final accuracy of 83% obtained on DHG-14 dataset, we notice that the recognition of dynamic hand gestures is still challenging whether in terms of handling the wide difference between gestures performed by different persons, resulting in a challenging coverage of the gestures but also by improving the hand pose estimation or finding more pertinent features and their temporal representations.

5.2. 28-gestures classification

In order to meet the challenge about gesture recognition performed with 2 different numbers of fingers, proposed in Section 3, we consider the sequences of the DHG-14/28 dataset as belonging to 28 classes related to the gesture but also the way it have been performed (with one finger or the whole hand). The resulting confusion matrix is shown in Figure 8. Using our 3 skeleton-based descriptors, we obtain an accuracy of 80%. So, by multiplying the number of classes by 2, we only loose 3% of accuracy.

5.3. Depth-vs-Skeleton based descriptors

To evaluate the contribution of the skeleton relative to the depth information, we computed three depth-based descriptors similar to those used in our approach. For the depth-based HoHD, we computed the center of mass of the region of interest of the hand in the depth to estimate the palm center position. For the depth-based HoWR, we used a *Principal Component Analysis* (PCA) on the hand depth data to find an approximation of the rotation of the hand in space.

Features	fine	coarse	both
HoHD	46.50%	78.72%	67.21%
HoWR	25.10%	38.44%	33.68%
Shape descr.	53.00%	54.28%	53.82%
HoHD + SDV	65.14%	86.51%	77.70%
HoHD+HoWR+SDV	66.90%	85.94%	**79.14%**

Table 3. Results of our method for 14 gestures on the DHG dataset using **depth**-based descriptors.

In order to represent the hand shape using the depth images, we implemented the 2 descriptors proposed by Kurakin et al. [8]. The first consists of dividing the hand image into several uniform grid (4×4, 8×8 or 16× 16). For each cell of the grid, we calculate its occupancy (area of the cell occupied by hand mesh) and the average depth after normalization. The second one divides the whole image into a number of fan-like sectors. For each one, we computed the average distance from the hand contour in the sector to the center of mass of the hand depth. We finaly concatecated the 2 descriptors in a shape descriptor vector. Results obtained using the depth-based descriptors are shown in Table 3. As noticed, depth-based HoHD and HoWR give more or less the same results as skeleton-based ones. However, for the hand shape description, the SoCJ gives better result (63%) compared to the depth-based descriptor (54%). Using the descriptors computed on the depth images leads in an overall decrease of the accuracy of 4%, mostly coming from the misclassification of fine gestures. We also observe a decrease of accuracy of 5% when going from 14 to 28 classes using depth-based descriptors. Moreover, we point out that finer depth feature could yield better results.

5.4. Discussion

In order to study the confusion of recognition rates between same gestures performed with different number of fingers, we propose to compute a metric, denoted as *Loss of Accuracy when Removing the Finger Differentiation* (LARFD). The LARFD metric assesses if the loss of accuracy when passing from 14 to 28 gestures is coming from the different number of fingers (**intra-gesture** confusion) or from the confusion with other gestures (**inter-gesture** confusion). Below, we use the notation $\mathcal{M}_g$ and $\mathcal{M}_{gf}$ respectively to denote the confusion matrix using 14 and 28 gesture classes (Figures 7 and 8).

$$\mathcal{E}^{+}_{larfd}(G_i) = (\mathcal{M}_g(G_i, G_i) - \frac{\sum_{j=1}^{N_{HF}} \sum_{k=1}^{N_{HF}} \mathcal{M}_{gf}(G_i^j, G_i^k)}{N_{HF}})$$

where G_i^j is a class gesture i performed with j fingers

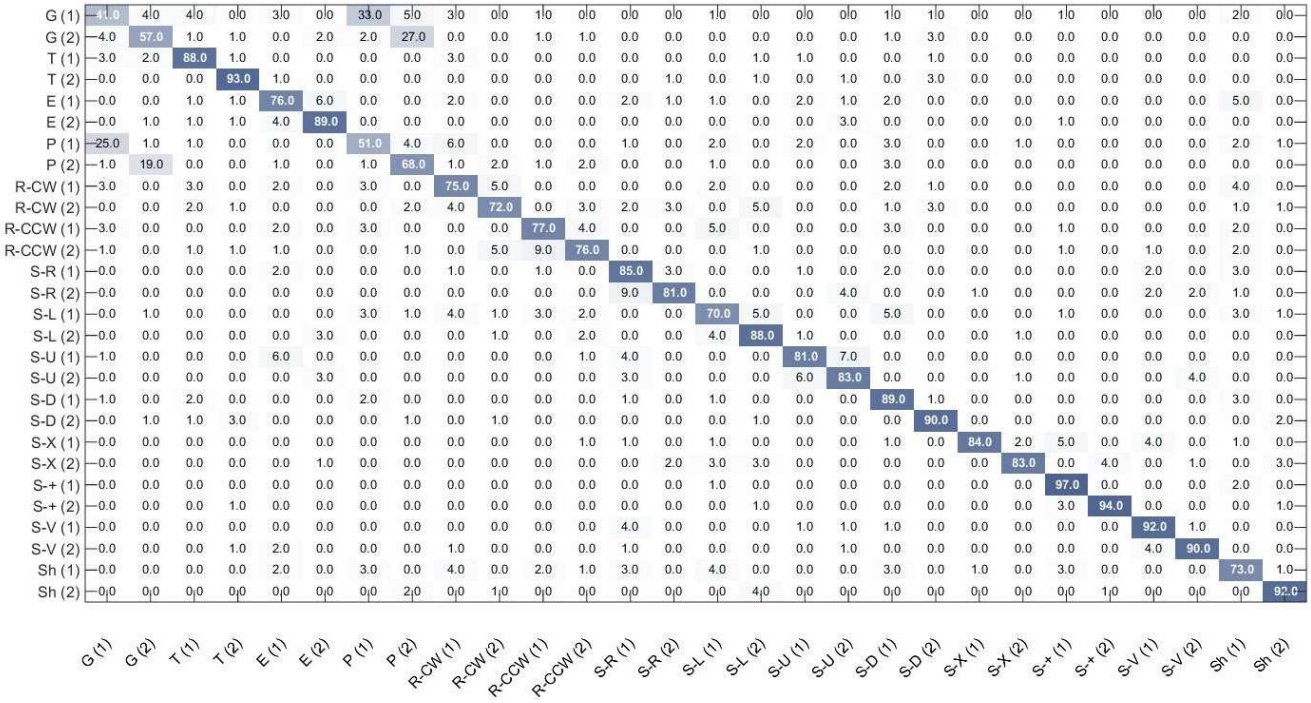


Figure 8. The confusion matrix of the proposed approach for DHG-28. The gestures annotated (1) and (2) were respectively performed using one finger and with the whole hand.

in the confusion matrices and N_{HF} the amount of different number of fingers used in the dataset.

For example, in the case of the *R-CCW* gesture, an accuracy of 85.5% is obtained when considering 14 classes. For 28 classes, we see that its score using one finger is 77% but also that 4% of them are seeing as performed with the whole hand. So their sum means that 81% of the *R-CCW* gesture performed with one finger are well recognized as *R-CCW* gesture. For the case of the same gesture performed with the whole hand, we obtained (76+9) 85%. The average of both is 83%. So, the loss of recognition accuracy considering 28 classes on the *R-CCW* gesture (without taking into account the number of fingers) is equal to (85.5-83) $2,5\%$. If we take into account the differentiation between the numbers of fingers (meaning 77% and 76%), we get a loss of accuracy of 9%. It means that in the general loss of accuracy of 9% of the *R-CCW* gesture, 2.5% of them are from **intra-gestures** confusion and the rest are from **inter-gestures** confusion.

The average of the LARFD over all gestures, when using skeleton-based descriptors, is equal to 0.0114. This score shows that the loss of accuracy when passing from 14 to 28 classes is due more to intra-gestures confusion that inter-gestures one (because on the 3% of general loss of accuracy, only 1% is due to inter-gestures confusion). Finally, when using depth-based descriptors, the obtained LARFD metric is equal to 0.0157. This result shows that intra-gesture confusion is greater when using the depth information than the skeleton. This can be explained by the fact that the hand skeleton provides more informative descriptions of the hand shape that the depth information.

6. Conclusion

This work suggests the advantage of using 3D hand skeleton data to describe hand gestures, and points out a promising direction of performing gesture recognition tasks using skeleton-like information. We present an approach to recognize dynamic hand gesture as time series of 3D hand skeleton returned by the Intel RealSense depth camera. We take as input a several set of relevant joints inferred from 3D hand skeleton. We propose a compact representation using Fisher Vector kernel and on multi-level encoding the temporal nature of gestures. Experimental results, conducted on enrolled dynamic hand gesture dataset, show the performance of our proposed method. Moreover, our approach achieves a performance accuracy of 83% on a challenging dataset, which is encouraging.

As future work, skeleton-based features can be combined with the depth-based features to provide more informative description and produce algorithms with better recognition robustness.

References

[1] C.-C. Chang and C.-J. Lin. Libsvm: a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2(3):27, 2011.

[2] H. Cheng, Z. Dai, and Z. Liu. Image-to-class dynamic time warping for 3d hand gesture recognition. In *2013 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–6, July 2013.

[3] M. Devanne, H. Wannous, S. Berretti, P. Pala, M. Daoudi, and A. Del Bimbo. 3-d human action recognition by shape analysis of motion trajectories on riemannian manifold. *IEEE Transactions on Cybernetics*, 45(7):1340–1352, 2015.

[4] C. Dong, M. C. Leu, and Z. Yin. American sign language alphabet recognition using microsoft kinect. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 44–52, June 2015.

[5] S. Escalera, X. Baró, J. Gonzalez, M. A. Bautista, M. Madadi, M. Reyes, V. Ponce-López, H. J. Escalante, J. Shotton, and I. Guyon. Chalearn looking at people challenge 2014: Dataset and results. In *Computer Vision-ECCV 2014 Workshops*, pages 459–473. Springer, 2014.

[6] G. Evangelidis, G. Singh, and R. Horaud. Skeletal quads: Human action recognition using joint quadruples. In *International Conference on Pattern Recognition*, 2014.

[7] S. Fothergill, H. M. Mentis, P. Kohli, and S. Nowozin. Instructing people for training gestural interactive systems. In J. A. Konstan, E. H. Chi, and K. Höök, editors, *CHI*, pages 1737–1746. ACM, 2012.

[8] A. Kurakin, Z. Zhang, and Z. Liu. A real time system for dynamic hand gesture recognition with a depth sensor. In *20th European Signal Processing Conference (EUSIPCO)*, pages 1975–1979, Aug 2012.

[9] A. Kuznetsova, L. Leal-Taix, and B. Rosenhahn. Real-time sign language recognition using a consumer depth camera. In *IEEE International Conference on Computer Vision Workshops (ICCVW)*, pages 83–90, Dec 2013.

[10] W. Li, Z. Zhang, and Z. Liu. Action recognition based on a bag of 3d points. In *IEEE Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 9–14, June 2010.

[11] G. Marin, F. Dominio, and P. Zanuttigh. Hand gesture recognition with leap motion and kinect devices. In *IEEE International Conference on Image Processing (ICIP)*, pages 1565–1569, 2014.

[12] P. Molchanov, S. Gupta, K. Kim, and J. Kautz. Hand gesture recognition with 3d convolutional neural networks. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 1–7, 2015.

[13] C. Monnier, S. German, and A. Ost. A multi-scale boosted detector for efficient and robust gesture recognition. In *Computer Vision-ECCV 2014 Workshops*, pages 491–502. Springer, 2014.

[14] N. Neverova, C. Wolf, G. W. Taylor, and F. Nebout. ModDrop: adaptive multi-modal gesture recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Apr. 2016.

[15] E. Ohn-Bar and M. M. Trivedi. Hand gesture recognition in real time for automotive interfaces: A multimodal vision-based approach and evaluations. *IEEE Trans. on Intelligent Transportation Systems*, 15(6):2368–2377, 2014.

[16] X. Peng, C. Zou, Y. Qiao, and Q. Peng. Action recognition with stacked fisher vectors. In *Computer Vision–ECCV 2014*, pages 581–595. Springer, 2014.

[17] F. Perronnin, J. Sánchez, and T. Mensink. Improving the fisher kernel for large-scale image classification. In *Computer Vision–ECCV*, pages 143–156. Springer, 2010.

[18] N. Pugeault and R. Bowden. Spelling it out: Real-time asl fingerspelling recognition. In *IEEE computer Vision Workshops (ICCV Workshops)*, pages 1114–1119, Nov 2011.

[19] Z. Ren, J. Yuan, and Z. Zhang. Robust hand gesture recognition based on finger-earth mover's distance with a commodity depth camera. In *ACM International Conference on Multimedia*, MM '11, pages 1093–1096, New York, NY, USA, 2011. ACM.

[20] J. Sánchez, F. Perronnin, T. Mensink, and J. Verbeek. Image classification with the fisher vector: Theory and practice. *International journal of computer vision*, 105(3):222–245, 2013.

[21] T. Sharp, C. Keskin, D. Robertson, J. Taylor, J. Shotton, D. Kim, C. Rhemann, I. Leichter, A. Vinnikov, Y. Wei, D. Freedman, P. Kohli, E. Krupka, A. Fitzgibbon, and S. Izadi. Accurate, robust, and flexible real-time hand tracking. CHI, April 2015.

[22] 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 a single depth image. In *Computer Vision on Pattern Recognition (CVPR)*, June 2011.

[23] R. Slama, H. Wannous, M. Daoudi, and A. Srivastava. Accurate 3d action recognition using learning on the grassmann manifold. *Pattern Recognition*, 48(2):556 – 567, 2015.

[24] H.-I. Suk, B.-K. Sin, and S.-W. Lee. Hand gesture recognition based on dynamic bayesian network framework. *Pattern Recognition*, 43(9):3059 – 3072, 2010.

[25] D. Tang, H. J. Chang, A. Tejani, and T. K. Kim. Latent regression forest: Structured estimation of 3d articulated hand posture. In *Computer Vision and Pattern Recognition (CVPR)*, pages 3786–3793, June 2014.

[26] H. Wang, Q. Wang, and X. Chen. Hand posture recognition from disparity cost map. In *ACCV (2)*, volume 7725 of *Lecture Notes in Computer Science*, pages 722–733. Springer, 2012.

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

[28] C. Zhang, X. Yang, and Y. Tian. Histogram of 3d facets: A characteristic descriptor for hand gesture recognition. In *IEEE Int. Conference and Workshops on Automatic Face and Gesture Recognition (FG),*, pages 1–8, April 2013.