A REVIEW OF GESTURE RECOGNITION APPROACHES FOR HRI

C. Attolico[1], *V. Renò*[2], *C. Guaragnella*[1], *T. D'Orazio*[2], *G. Cicirelli*[2]

[1] Department of Electric Engineering - Politecnico di Bari, ITALY
[2] Institute of Intelligent Systems for Automation - C.N.R. ITALY

dorazio@ba.issia.cnr.it

ABSTRACT

In the last years Gesture Recognition through visual information has been recognized as one of the main active research topics in the computer vision community. Initially, images acquired from conventional cameras were used to achieve this task. More recently, the wide spread of depth sensors such as the low-cost Microsoft Kinect, allowed the use of a new type of data at hand. The availability of synchronized depth and color (RGB) images permitted the analysis of gestures to be approached by a new perspective. This paper tries to summarize the recent state-of-the-art of gesture recognition approaches that exploit both RGB and depth data.

I. INTRODUCTION

A gesture is defined as a form of non-verbal communication in which visible bodily actions communicate particular messages, either in place of, or in conjunction with speech. A gesture can include movements of hands, face, or other parts of the body. Gestures are the oldest means of human communication. Nowadays gestures are still important as people use them also in an unconscious way in everyday life, but they can be essential in many situations which involve communications in hazardous contexts. From the scientific point of view, gestures are used and then analyzed in several domains such as sign language recognition, vision-based augmented reality, smart surveillance, virtual environments, human-robot interaction, and so on. In this paper we will focus our attention on gesture recognition approaches used in the human-robot interaction context.

Different definitions of the term *gesture* have been provided in literatures and sometimes it has been interchangeably used as a synonym of the term *action*. In this paper we have used the definition provided in [1]: a gesture is a physical movement of hands, arms, face or body, made with the intent of conveying meaningful information. Thus gesture recognition involves not only the plain tracking of movement, but also its interpretation as a semantically meaningful command. We distinguish between gestures, which are intentional movement of the body, and actions which are unconscious elementary movements of the body and can be used to understand human daily activities.

Gestures can be static, when the user assumes a certain pose or configuration, or dynamic with a pre-stroke, stroke and post-stroke phase, as pointed out in [2]. Some gestures also have both static and dynamic elements, as in sign language applications. The automatic recognition requires in the first case the characterization of the spatial disposition of the body parts performing the gesture, whereas in the second case it requires the observation of the sequence of movements generated by the human body.

In this paper, we will consider all the challenging problems related to the development of an automatic gesture recognition system:

- *Feature Extraction:* This is the first step in gesture recognition process. It involves the definition of the features that better and distinctively characterize a specific movement setting it apart from similar items.
- *Gesture Classification:* Gesture recognition can be seen as a classification problem in which examples of gestures are used into a supervised learning scheme (such as SVM or NN) to model the gestures and solve the recognition problem as a class association problem.
- *Spatiotemporal Segmentation:* This is the task of determining, in a video sequence, where the gesturing elements are located and when the gesture starts and ends.

Many good reviews on action recognition approaches have resumed the researches carried out for the recognition of human movements such as walking, jumping, running, and so on. Gesture recognition surveys have also been published [1], [2], with particular emphasis on hand gestures and facial expression by the analysis of images of conventional RGB cameras. In this review, we will focus our attention on the literature which uses depth information to improve gesture recognition performances with a particular attention to the more recent literature which sees a number of publications on approaches based on depth data extracted by low cost RBG-D sensors.

In figure 1, a layer diagram about the classification of the approaches considered in this review is shown. The following sections present the considered literature according to the different aspects reported in this scheme. Finally, after a discussion about the general topic, the last section provides insights into open problems and future research directions.

II. FEATURE EXTRACTION

Selecting features is crucial to gesture recognition, since body gestures are very rich in shape and motion variations. According to [3], in the context of gesture recognition,

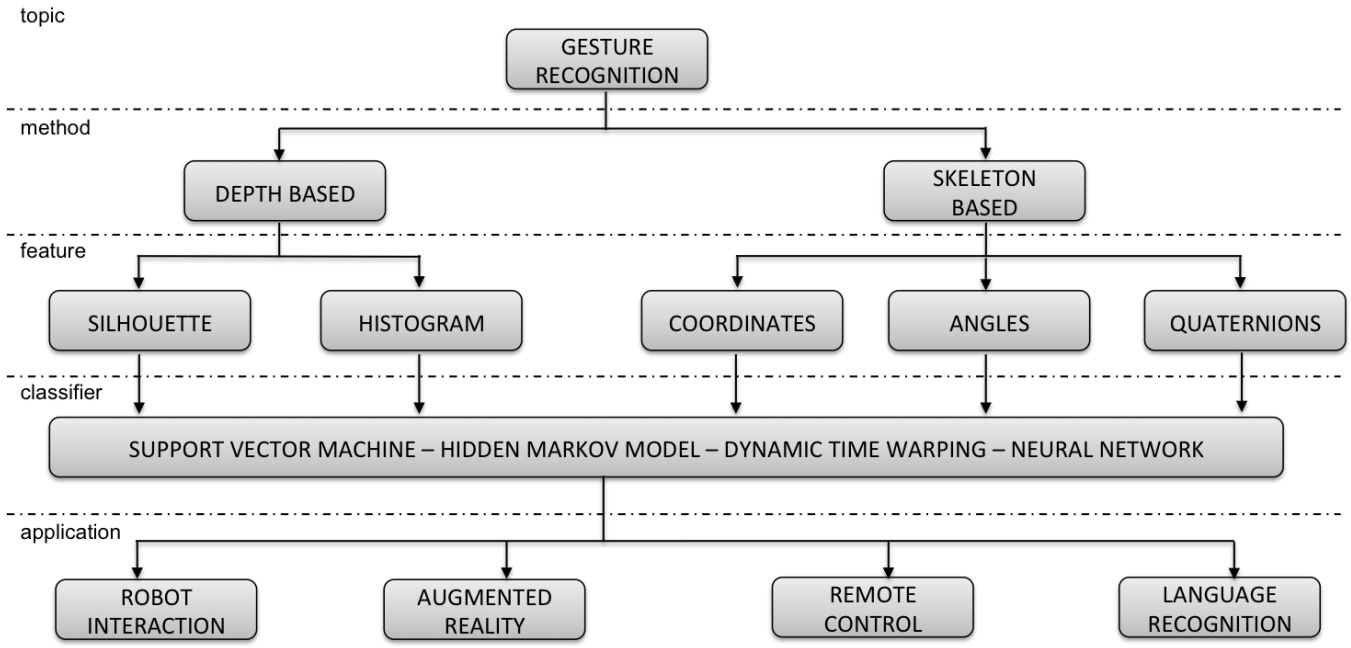

Fig. 1. The picture represents a layer diagram of a gesture recognition system, exploiting the methods that can be used to recognize a gesture, the features that can be extracted, the classifiers that can be used and some relevant real-time applications.

the methods can be classified into two main categories: *Depth-Map Based methods* and *Skeleton-based Methods*. The former ones are primarily based on features extracted directly from the space volume. The latter ones, instead, use features such as angles, coordinates, and other combinations, extracted by skeleton which represents in a synthetic way the human silhouette. Skeletons have been always used by the computer vision community [4], even if their extraction was computationally expensive and often not effective due to the noise of conventional cameras. In the last years, skeleton-based approaches have become again popular with the advent of low cost RGB-D sensors with annexed software frameworks that provide directly the 3D joint positions of the human skeleton.

II-A. Depth-Map Based Features

Depth data processing (DDP) provides a rich informative description of the scene, especially if compared to the analysis of a bi-dimensional image because it adds the knowledge of the third coordinate z to the well known (x, y) frame representation. There are many applications of this method that exploit these information in order to:

- extract three-dimensional shapes based on pose evaluation;
- recognize the gesture without processing the full pose, but applying machine learning techniques to some features extracted from the depth information.

Stereo cameras are used in [5] to generate *depth silhouettes* that improve the binary silhouette model filling pixels with range information. In this way, the authors demonstrate that depth silhouettes are able to register complex poses without tracking feature points. Range cameras in [6] fuse depth and intensity information to produce robust 3D point clouds. The point clouds are therefore represented efficiently using shape context descriptors based on a spherical histogram. View invariant gesture recognition is assured by representing gestures as sequences of 3D motion primitives. In [7] a real-time hand gesture recognition system uses the depth data to extract the hand as the area of the body closer to the camera. A normalization step performs scale and rotation transformations to have view invariance. Also in [8] a three-dimensional shape descriptor is used to represent the hand in a 3D view and to guarantee an hand poses recognition invariant to scale and rotation variations. Depth and RGB images are used in [9], [10] to recognize hands. In [9] the hand's distance and curvature features are used to divide the hand region into three parts: fingers, palms and wrists. After an initial hand segmentation step, SIFT features are extracted in [10] and are hierarchically quantized in a vocabulary tree to recognize the hand gestures representing the numbers from "one" to "ten". A different setup, presented in [11], consists of a multi camera system able to solve occlusion problems which reduce the recognition performances with single camera approaches; salient points on the fingers allow the estimation of the hand pose in challenging situations containing hands and objects in action. A synthetic 3D hand model is used in [12] to perform pixel classification and assign each pixel to a hand part, such that each skeleton joint is at the center of one of the labelled part. The skeleton parameters are manually extracted to train several randomized decision trees which are used to estimate the full hand poses in real time.

II-B. Skeleton-based Features

The study of skeleton-based gesture dates back to the early work by Johansson [13], which demonstrates that a large set of gestures and actions can be recognized solely from the joint positions [3]. This concept has been extensively explored since then. In contrast to the depth maps-based methods, the majority of the skeleton-based methods model temporal dynamics explicitly. One main reason is the natural correspondence of skeleton joints across time, while this is difficult to establish for general visual and depth data. For this reason in the past, many researchers tried to create a 3D Human Model in order to detect some information about joints. In [14], the authors build a 3D articulated human body model which consists of 17 body segments. A linear combination of 2D depth image and 3D human model prototypes is used to reconstruct the 3D human body. The features used to recognize the gestures, are the three angle values of selected body points located at joints of left wrist, left elbow, left shoulder, etc. However, the extraction of human skeleton from 2D or 3D images is considered a complex task, requires many computational resources, and it is strictly dependent on the noise in the images. With the recent diffusion of Microsoft Kinect sensors on the market, skeleton based approaches have become much more popular. In fact, together with the sensors, some software platforms are available to detect and track one or more people in the scene and extract their corresponding human skeleton in real-time. In particular, several platforms have been largely utilized such as the official SDK for Kinect, and others open source such as OpenNI, Libfreenect or ROS. The direct availability of real-time information about joint coordinates and orientations has provided a great impetus to research and many papers have been published in the last couple of years. In this review we have divided the considered approaches according to the features they have used: coordinates, angles and quaternions of skeleton joints. In the next subsections these methods will be introduced with the relative examples.

Coordinates Features Many works use the joints coordinates, as they are immediately provided by the Kinect sensor and for many kind of gestures they are discriminant enough to guarantee their recognition. However, coordinates may depend on the person's height, arm length, and distance from the camera. Each of these variations may impact the gesture model in a different way, therefore, all feature vectors have to be further normalized.

In [15], 7 upper body joints out of the total 20 provided by the Kinect are kept to recognize 8 aircraft gestures used in the military air force. The coordinates are normalized as $C_{scaled} = (C_{original} - min)/(max - min)$ where (max) and (min) are the maximum and minimum values of that particular feature. In [16], all the 20 joints are considered to recognize three human gestures (stand, sit down and lie down) and a Z-score normalization is applied to deal with parameters of different units and scales of body-joint positions. A sequence of *(x, y, z)* coordinates of the person's right hand is stored in [17], to recognize a set of six gestures, that appear in interaction with waiters (such as ask for a bill, cancelling an order, and so on). A simple geometric transformation is applied in [18] to the hand coordinates to set the reference system centered on the human torso, instead of the default sensor-centered reference frame. This transformation provides invariance to starting point of a physical gesture. In other words, the user can perform the three considered hand gestures (push, grasp and tap) at any distance or angle from the robot sensors, and these gestures are always measured with regards to his torso coordinate. Also in [19], [20], only few joints (hand and elbow) are considered significant for the hand gesture recognition. In each frame, the 3-D distances of these joints from the spine joint, which serves as a reference point, are computed to be invariant to the user's height and camera distances. A mixture of joint coordinates and angles are used in [21], to recognize nine different gestures which are characterized by either static pose or dynamic variations of joint positions. In [22] the neck joint of the skeletal model is employed as the origin of coordinates (OC). The remaining joints coordinates are computed with respect to the OC. This transformation allows to relate pose models that are at different depths, being invariant to translation, scale, and tolerant to corporal differences of subjects. In a similar way, in [23], the feature vectors are firstly normalized with respect to the distance between the left and the right shoulders to account for the variations due to people different sizes. A second normalization follows by subtracting the shoulder center from all the elements to account for cases where the user is not in the center of the depth image.

Angles Features Many papers selects angles between joint vectors as significant features to both maximize the invariance of the skeletal representation with respect to camera position and reduce the dimensionality of search space while retaining the character of the motion.

In [24], the angle between specific pairs of direction vectors, (dv_1, dv_2) is computed to obtain the corresponding joint angles with this formula:

$$cos\gamma = \frac{\vec{dv_1} \circ \vec{dv_2}}{|\vec{dv_1}| \bullet |\vec{dv_2}|}$$

For instance, the direction vector of the lower arm is calculated between Elbow and Wrist position, while the direction vector of the higher arm is calculated between Shoulder and Elbow position. In [25], a novel angular skeleton representation is used to map the skeleton motion data to a smaller set of features. The aim is to reduce the overall entropy of the signal, remove dependence on camera position, and avoid unstable parameter configurations. They use the first and second-degree limb joints relative to the torso one. The first-degree joints are all the joints adjacent to the torso, while the second-degree ones are the tips of the wire frame extremities; these include the hands and the feet. The approach is to fit the full torso with a single frame of reference, and to use this frame to parametrize the estimated orientation of both the first- and second-degree limb joints. In some papers the angles provided by the Kinect software platforms such as yaw, pitch and roll angles of some joints are directly used. For example in [26], left elbow yaw and roll, and left shoulder yaw and pitch are used to recognize

five different gestures executed with the left arm.

Quaternions Features Some Kinect software frameworks provide other joints orientation information in addition to the coordinates, this is the quaternion. It is a set of numbers that comprises a four-dimensional vector space and is denoted by $q = a + bi + cj + dk$, where a, b, c, d are real numbers and i, j, k are imaginary units. The quaternion q represents an easy way to code any 3D rotation expressed as a combination of a rotation angle and a rotation axis. Compared to 3-by-3 rotational matrices, quaternions are also more compact, requiring only 4 storage units, instead of 9. The properties of quaternions make their use favorable for representing rotational representations.

In [27] all the quaternion joints are used to recognize 6 different actions which reasonably cover the various movement of arms, legs, torso, and their combination. When the gestures movement involves only some part of the body, different selections can be done. In [28] the quaternions of the shoulder and elbow right nodes are employed as they are enough to represent the direction these bones are pointing to. The human position with respect to the sensor can greatly affect gesture recognition. In order to resolve this dependency, the systems have to be view-invariant. One possible method is that proposed in [29]. The torso joint is taken as the origin and the X-axis is defined as the line from left hip to right hip, the Y-axis as the average of the left and right leg lines and the Z-axis perpendicular to the plane determined by the X and Y axes. The relative orientation of each joint is calculated by abstracting the torso orientation in the starting frame using: $Q_{new} = Q_{old} * Q^*_{torso}$ where Q^* is the conjugate of Q. After this step, the quaternions are all reported in the new coordinate system.

III. CLASSIFICATION METHODS

After the feature extraction step during which features relevant for the considered gesture typology are selected, another important step has to be carried out: the model generation for the gesture recognition task. Some methods consider this problem as a supervised classification process, supposing that a human expert determines the number of classes and provides the sets of samples that belong to the known classes. During a training phase, the sets of samples are used to generate the gesture models which can thereafter be applied during the actual tests for the recognition phase. Different methods can be used to generate gesture models.

Hidden Markov Model (HMM) This is a statistical Markov model in which the system being modeled is assumed to be a Markov process with unobserved (hidden) states. It is characterized by the following components: the number of state in the model, the number of distinct observation symbols per state, the state transition probability distribution, the observation symbol probability distribution and the initial state distribution. In [26] a K-means clustering is used to convert the feature vectors (joint angles) into the observable symbols for HMMs. The orientation of the hand centroid point projection in the image plane is transformed in [30] in a chain code from 1 to 8 by dividing the coordinate system into eight equal portions. The observations provided to the HMM are the orientation chain codes of each observed gestures. A similar approach is used in [31] with a uniform quantization of the orientations in 12 angle sectors (every $\Pi/6$ in order to quantify the different directions). In [32] a cascade of two HMMs is used to calculate a robust estimation of the pointing direction. The first stage HMM takes the estimated hand position and maps it to a more accurate position by modelling the kinematic characteristics of the finger that is pointing. The resulting 3D coordinates are used as input for the second stage HMM that discriminates pointing gestures from other types. Finally, the pointing direction is estimated for the pointing state. In order to assign the state to the HMM, the surface of the hemisphere in which the pointing gesture is performed, is divided into patches at 20° from left-to-right and top-to-bottom.

Support Vector Machine (SVM) This is a discriminative classifier formally defined by a separating hyperplane. In other words, given labelled training data (as the result of a supervised learning), the algorithm outputs an optimal hyperplane which categorizes new examples. SVM approaches with linear and RBF kernels are compared in [15] and demonstrate better performances than decision tree classification. A SVM variant, called multi-class SVM is used in [33] to recognize simultaneously 8 gestures. A multiple kernel learning algorithm is proposed in [34] in conjunction with a multiclass SVM for feature fusion and to enhance the discrimination power of the classifier. 3D shape features and 3D motion features are combined to recognize complex activities.

Neural Network (NN) Multilayer feed forward Neural networks are often used to solve classification problems. In [16], a multi class NN classifier, compared with SVM and DT classifiers, demonstrates the best accuracy for the classification of input frames in three class gestures (stand, sit down, and lie down). A different approach is used in [28], where a cascade of 10 different binary NN classifiers are trained to recognize ten different gestures. The class association is performed by selecting the NN which provides the maximum output value over a fixed threshold.

Dynamic time warping (DTW) This is a well-known technique that has long been used to find the optimal alignment of two signals. The DTW algorithm calculates the distance between each possible pair of points out of two signals in terms of their associated feature values. The algorithm is implemented as described below.

1) The average length of a training samples L_{avg} is calculated.
2) An equally spaced vector is calculated using the length of the signal to be warped.
3) The signal values corresponding to the equally spaced points are calculated by linear interpolation of the signal to be warped. The warped signal has a length equal to L_{avg}.

In [35] DTW is utilized for pairwise coupling, a multi-class classification method that combines all pairwise comparisons between each pair of classes. The maximum value of the probability estimates (of different classes) is calculated to classify a gesture. A variation of this method is used in [23] where the authors propose a weighted DTW method that weights joints by optimizing a discriminant ratio that

depends on body joint activity. By doing so, some joints will be weighted up and some joints will be weighted down to maximize between-class variance and minimize within-class variance.

IV. GESTURE SEGMENTATION

Another important issue for the development of HRI systems concerns the gesture segmentation task. When gestures are executed in a continuous way it is important to identify the starting and ending frames of the gesture. During the transition between one gesture to another, there occur intermediate movements as well. These transition motions need to be eliminated to avoid false matching with reference patterns. Moreover, the same gesture may dynamically vary in shape and duration even for the same gesturer. All the classification methods (except for those based on DTW algorithms [17], [36], [37] which manage different gesture lengths) suffer for these problems and have to solve both the segmentation and the spatial/temporal normalization of the gesture sequences.

In many cases during the training phase of gesture model generation the data streams are marked manually by human observers and some constraints on the gesture lengths are imposed. In [19] and [28] the authors impose that a gesture must be recorded/observed in a fixed time interval (1 or 2 seconds), so the incoming video sequences are quickly exploitable because every gesture has the same temporal length. In [30] the start/end point detection of a hand gesture trajectory is performed measuring the hand centroid point velocity, and setting an initial rectangular region of the image as reference for the system to start capturing the hand point coordinates. A sliding window in [15] is passed over the unedited data stream to estimate the time length of gesture. This process generates one probability estimate per gesture for each position of the sliding window. Then, the cumulative sum of these probability values is computed, and the predicted gesture is the one with the maximum cumulative sum.

V. CONCLUSIONS AND DISCUSSION

Gesture recognition approaches have found application in many fields especially in the last years with the large distribution of Kinect sensors. In table I we summarize the papers we have considered in this review. Different features and classifiers have been used as reported in the table. In the last two columns of the table we have reported two flags to highlight if the approaches have considered the segmentation problem of the starting/ending frame, and if on line experiments have been carried out. Many of them do not consider the segmentation problem at all, as the tests are off line on data sets extracted manually. These works aim to verify the effectiveness of feature extraction and classification methods. According to the gesture complexity different selections of features have been considered. If the selected gestures are very distant in terms of 3D trajectories, the variations of simple coordinates, such as hand elbow and shoulder joint coordinates, characterize the movement. When gestures are similar and differ only for angle variations, features have to be more discriminant, then, rotation angles or quaternions have been considered. Classifiers such as HMMs, SVMs, or NNs have been applied producing comparable performances when the gestures are well segmented and have quite the same length. Some works consider the problem of gesture segmentation or gesture length normalization in order to increase the classifier performances, but do not demonstrate the effectiveness of the proposed approach during on-line tests. Approaches based on DTW are able to manage sequences of different lengths but they still have the problem of selecting the starting frame of each gesture. Especially when gestures are executed in a continuous way there are some preparatory movements that have to be eliminated to avoid false detections. Sliding window approaches represent a good solution to allow the application of gesture recognition on continuous sequences during online experiments. The main drawback is represented by the high computational costs that small shifts of sliding windows can produce. Some papers introduce idle movements between consecutive gestures that can be easily recognized and used to separate different sequences. The last point that has to be considered is the invariance to the position and orientation with respect to the cameras and the execution of each gesture. Some papers introduce different kinds of normalization to be independent of the distances involved. A few works demonstrate that the gesture model learnt on some people is robust also when tested with different subjects performing the same actions.

All these problems remain open and worthy of investigation in the next future. The main purpose is to develop robust HRI that can work on-line in challenging and real situations.

VI. REFERENCES

[1] G. R. S. Murthy and R. S. Jadon, "A review of vision based hand gesture recognition," *International Journal of Information Technology and Knowledge Management*, vol. 2, no. 2, pp. 405–419, 2009.

[2] S. Mitra and T. Acharya, "Gesture recognition: A survey," *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, vol. 37, no. 3, pp. 311–324, 2007.

[3] M. Ye, Q. Zhang, L. Wang, J. Zhu, R. Yang, and J. Gall, "A survey on human motion analysis from depth data," *Time-of-Flight and Depth Imaging. Sensors, Algorithms, and Applications*, pp. 149–187, 2013.

[4] C. Castiello, T. D'Orazio, A.M. Fanelli, P. Spagnolo, and M.A Torsello, "A model free approach for posture classificatin," *IEEE Conf. on Advances Video and Signal Based Surveillance, AVSS*, 2005.

[5] R. Munoz-Salinas, R. Medina-Carnicer, F.J. Madrid-Cuevas, and A. Carmona-Poyato, "Depth silhouettes for gesture recognition," *Pattern Recognition Letters*, vol. 29, no. 3, pp. 319 – 329, 2008.

[6] M.B. Holte, T.B. Moeslund, and P. Fihl, "Fusion of range and intensity information for view invariant gesture recognition," in *Computer Vision and Pattern Recognition Workshops, 2008. CVPRW '08. IEEE Computer Society Conference on*, June 2008, pp. 1–7.

Reference	Body movement	Feature	Classifier	Start/End Seg.	On-line Exp.
[5]	Both Arms	Depth silhouette	HMM, SVM	N	N
[6]	Both Arms	3D point cloud, Spherical histogram	Correlation probability distance	Y	N
[7]	Hand	Cell occupancy silhouette	Action graph	-	Y
[10]	Hand	SIFT	k-Means Clustering	-	Y
[14]	Full Body	Joint Angles	GMM + HMM	Y	-
[15]	Both Arms	Coordinates	SVM	Y	Y
[16]	Full Body	Coordinates	Optimal classifier among NN, SVM, DT and Bayesian	-	N
[17]	Right Hand	Coordinates	DTW	Y	Y
[18]	Single Arm	Coordinates	HMM result of GMM and HMM merge	Y	Y
[19]	Both Arms	Coordinates	Nearest Neighbour	N	Y
[20]	Single Arm	Coordinates	Decision forest SVM	Y	-
[21]	Both Arms	Coordinates	-	Y	Y
[22]	Full Body	Coordinates	Weighted DTW	Y	Y
[23]	Both Arms	Coordinates	Weighted DTW	N	-
[24]	Full Body	Joint Angles	-	-	Y
[26]	Left Arm	Joint Angles	HMM	Y	Y
[27]	Both Arms	Quaternion	DTW	N	N
[28]	Right Arm	Quaternions Joint angles	NN	Y	Y
[35]	Right Arm	Quaternion	DTW + Multi-class probability	Y	-
[30]	Both Hands	Hand centroid pt.	HMM	Y	Y
[31]	Single Hand	Depth sequence	3D random occupancy pattern	-	N
[33]	Both Arms	Depth image sub.	SVM	-	N
[34]	Full Body	Motion and Shape Features fusing with MKL	MultiClass SVM	-	N
[36]	Full Body	Coordinates	DTW	Y	Y
[37]	Both Arms	BoVDW	Probability-based DTW	Y	-

Table I. An overview of the works cited in this paper. Comparison of different gesture recognition systems parameters.

[7] A. Kurakin, Z. Zhang, and Z. Liu, "A real time system for dynamic hand gesture recognition with a depth sensor," in *Signal Processing Conference (EUSIPCO), 2012 Proceedings of the 20th European*, Aug 2012, pp. 1975–1979.

[8] P. Suryanarayan, A. Subramanian, and D. Mandalapu, "Dynamic hand pose recognition using depth data," in *Pattern Recognition (ICPR), 2010 20th International Conference on*, Aug 2010, pp. 3105–3108.

[9] F. Dominio, M. Donadeo, G. Marin, P. Zanuttigh, and G. M. Cortelazzo, "Hand gesture recognition with depth data," in *Proceedings of the 4th ACM/IEEE International Workshop on Analysis and Retrieval of Tracked Events and Motion in Imagery Stream*, New York, NY, USA, 2013, ARTEMIS '13, pp. 9–16, ACM.

[10] M. Hamissi and K. Faez, "Realtime hand gesture recognition based on the depth map for human robot interaction," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 3, no. 6, pp. 770–778, 2013.

[11] L. Ballan, A. Taneja, J. Gall, L. Gool, and M. Pollefeys, "Motion capture of hands in action using discriminative salient points," in *Computer Vision – ECCV 2012*, A. Fitzgibbon, S. Lazebnik, P. Perona, Y. Sato, and C. Schmid, Eds., vol. 7577 of *Lecture Notes in Computer Science*, pp. 640–653. Springer Berlin Heidelberg, 2012.

[12] C. Keskin, F. Kirac, Y.E. Kara, and L. Akarun, "Real time hand pose estimation using depth sensors," in *Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on*, Nov 2011, pp. 1228–1234.

[13] G. Johansson, "Visual motion perception.," *Scientific American*, 1975.

[14] S.-W. Lee, "Automatic gesture recognition for intelligent human-robot interaction," in *Automatic Face and Gesture Recognition, 2006. FGR 2006. 7th International Conference on*. IEEE, 2006, pp. 645–650.

[15] S. Bhattacharya, B. Czejdo, and N. Perez, "Gesture classification with machine learning using kinect sensor data," in *Emerging Applications of Information Technology (EAIT), 2012 Third International Conference on*. IEEE, 2012, pp. 348–351.

[16] O. Patsadu, C. Nukoolkit, and B. Watanapa, "Human gesture recognition using kinect camera," in *Computer Science and Software Engineering (JCSSE), 2012 International Joint Conference on*. IEEE, 2012, pp. 28–32.

[17] S. Bodiroža, G. Doisy, and V.V. Hafner, "Position-invariant, real-time gesture recognition based on dynamic time warping," in *Proceedings of the 8th ACM/IEEE international conference on Human-robot interaction*. IEEE Press, 2013, pp. 87–88.

[18] G. Saponaro, G. Salvi, and A. Bernardino, "Robot anticipation of human intentions through continuous gesture recognition," in *Collaboration Technologies and Systems (CTS), 2013 International Conference on*. IEEE, 2013, pp. 218–225.

[19] K. Lai, J. Konrad, and P. Ishwar, "A gesture-driven computer interface using kinect," in *Image Analysis and Interpretation (SSIAI), 2012 IEEE Southwest Symposium on*. IEEE, 2012, pp. 185–188.

[20] J. Oh, T. Kim, and H. Hong, "Using binary decision tree and multiclass svm for human gesture recognition," in *Information Science and Applications (ICISA), 2013*

International Conference on. IEEE, 2013, pp. 1–4.

[21] T. Hachaj and M.R. Ogiela, "Rule-based approach to recognizing human body poses and gestures in real time," *Multimedia Systems*, vol. 20, no. 1, pp. 81–99, 2014.

[22] M. Reyes, G. Dominguez, and S. Escalera, "Featureweighting in dynamic timewarping for gesture recognition in depth data," in *Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on*. IEEE, 2011, pp. 1182–1188.

[23] S. Celebi, A. S. Aydin, T. T. Temiz, and T. Arici, "Gesture recognition using skeleton data with weighted dynamic time warping," *Computer Vision Theory and Applications. Visapp*, 2013.

[24] I. Almetwally and M. Mallem, "Real-time tele-operation and tele-walking of humanoid robot nao using kinect depth camera," in *Networking, Sensing and Control (ICNSC), 2013 10th IEEE International Conference on*. IEEE, 2013, pp. 463–466.

[25] M. Raptis, D. Kirovski, and H. Hoppe, "Real-time classification of dance gestures from skeleton animation," in *Proceedings of the 2011 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, New York, NY, USA, 2011, SCA '11, pp. 147–156, ACM.

[26] Y. Gu, H. Do, Y. Ou, and W. Sheng, "Human gesture recognition through a kinect sensor," in *Robotics and Biomimetics (ROBIO), 2012 IEEE International Conference on*. IEEE, 2012, pp. 1379–1384.

[27] S. Sempena, N. U. Maulidevi, and P. R. Aryan, "Human action recognition using dynamic time warping," in *Electrical Engineering and Informatics (ICEEI), 2011 International Conference on*. IEEE, 2011, pp. 1–5.

[28] T. D'Orazio, C. Attolico, G. Cicirielli, and C. Guaragnella, "A neural network approach for human gesture recognition with a kinect sensor," in *International Conference on Pattern Recognition Applications and Methods (ICPRAM 2014)*. INSTICC, 2014, pp. 741 – 746.

[29] L. Sun and K. Aizawa, "Action recognition using invariant features under unexampled viewing conditions," in *Proceedings of the 21st ACM international conference on Multimedia*. ACM, 2013, pp. 389–392.

[30] D. Xu, Y.L. Chen, C. Lin, X. Kong, and X. Wu, "Real-time dynamic gesture recognition system based on depth perception for robot navigation," in *Robotics and Biomimetics (ROBIO), 2012 IEEE International Conference on*. IEEE, 2012, pp. 689–694.

[31] J. Wang, Z. Liu, J. Chorowski, Z. Chen, and Y. Wu, "Robust 3d action recognition with random occupancy patterns," in *Proceedings of the 12th European Conference on Computer Vision - Volume Part II*, Berlin, Heidelberg, 2012, ECCV'12, pp. 872–885, Springer-Verlag.

[32] C.B. Park and S.W. Lee, "Real-time 3d pointing gesture recognition for mobile robots with cascade hmm and particle filter," *Image and Vision Computing*, vol. 29, no. 1, pp. 51–63, 2011.

[33] KK Biswas and S.K. Basu, "Gesture recognition using microsoft kinect®," in *Automation, Robotics and Applications (ICARA), 2011 5th International Conference on*. IEEE, 2011, pp. 100–103.

[34] S. Althloothi, M.H. Mahoor, X. Zhang, and R.M. Voyles, "Human activity recognition using multi-features and multiple kernel learning," *Pattern Recognition*, 2013.

[35] P.K. Pisharady and M. Saerbeck, "Robust gesture detection and recognition using dynamic time warping and multi-class probability estimates," in *Computational Intelligence for Multimedia, Signal and Vision Processing (CIMSIVP), 2013 IEEE Symposium on*. IEEE, 2013, pp. 30–36.

[36] A. A. Chaaraoui, J. R. Padilla-Lopez, P. Climent-Perez, and F. Florez-Revuelta, "Evolutionary joint selection to improve human action recognition with rgb-d devices," *Expert Systems with Applications*, vol. 41, no. 3, pp. 786 – 794, 2014, Methods and Applications of Artificial and Computational Intelligence.

[37] A. Hernandez-Vela, M. Ángel Bautista, X. Perez-Sala, V. Ponce-Lopez, S. Escalera, X. Bara, O. Pujol, and C. Angulo, "Probability-based dynamic time warping and bag-of-visual-and-depth-words for human gesture recognition in rgb-d," *Pattern Recognition Letters*, , no. 0, pp. –, 2013.