Real-time Motion-based Hand Gestures Recognition from Time-of-Flight Video

Javier Molina[1] · José Antonio Pajuelo[1] · José M. Martínez[1]

Received: 26 August 2014 / Revised: 20 April 2015 / Accepted: 17 November 2015 / Published online: 25 November 2015

Abstract This paper presents an innovative solution based on Time-Of-Flight (TOF) video technology to motion patterns detection for real-time dynamic hand gesture recognition. The resulting system is able to detect motion-based hand gestures getting as input depth images. The recognizable motion patterns are modeled on the basis of the human arm anatomy and its degrees of freedom, generating a collection of synthetic motion patterns that is compared with the captured input patterns in order to finally classify the input gesture. For the evaluation of our system a significant collection of gestures has been compiled, getting results for 3D pattern classification as well as a comparison with the results using only 2D information.

Keywords Computer vision · Human-computer interaction · Hand gesture recognition

1 Introduction

Human Computer Interaction (HCI) technologies and algorithms are becoming more important in the last years, a time in which users ask for new ways of communication with computers and of interaction with virtual environments. The user experience of high technological services is not always optimal and HCI might help bringing these services to the mass market. As mentioned in [21], in the last years 3D user interfaces (3D UI) are becoming more important in the console gaming scenario.[1,2,3] Besides, in desktop computers interfaces, the usage of the hand as input device provides natural human-computer interaction [24]. Usability constitutes a main issue in the development of HCI systems and some of the aspects are pointed out in [11]; in [3] we find a study devoted to improve user experience.

The ultimate goal of this work is to provide the user with a natural interaction and a good experience when interacting with a computer in contexts of application such as the interaction with maps,[4] allowing intuitive movements of the earth surface. Other contexts of application of this approach can be the control of multimedia menus [31] or the point of view on a virtual environment. Other motion based gestures recognition could allow the interpretation of sign languages [9, 13].

The paper is structured as follows: In Section 2 the State Of Art is exposed and the innovations of our system are pointed out before giving an overview of it in Section 3. In Section 4 the proposed dictionary of gestures and the compilation of users executions is described. In Section 5 the approach followed for gestures detection is explained for later, in Section 6, presenting the significant user-independent evaluation figures and enumerating the achieved conclusions in Section 7.

✉ Javier Molina
javier.molina@uam.es

[1] Video Processing and Understanding Laboratory Escuela Politécnica Superior, Universidad Autónoma de Madrid Avda. Francisco Tomás y Valiente, 11 Ciudad Universitaria de Cantoblanco, Ctra. de Colmenar Viejo, km 15, E-28049 Madrid, Spain

[1] http://wii.com

[2] http://www.xbox.com/kinect/

[3] http://playstation.com/psmove/

[4] Atlas Gloves: A DIY Hand Gesture Interface for Google Earth, http://atlasgloves.org/about

2 Related Work

There are several works focused on hand gesture recognition based on range data, as the use of depth information has been recurrent in the last years. Some examples of the use of depth information can be found in [7, 28] where stereo-vision systems applied to gesture recognition are presented. In [18] they estimate the 3D trajectory of hand by using markers. Another approach consists in the adjustment of 3D models to 2D images [1, 32]. A recent research line is the use of Time-of-Flight (TOF) range cameras that supply real-time depth information per pixel [31] at low cost. An example of the use of this technology can be found in [8] where it is used to improve people tracking in a smart room. TOF technology can also present some problems, such as optical noise existence, unmatched boundaries or temporal inconsistency [16]. The use of depth information results in an enrichment of the communication between user and machine by means of gestural interfaces. In [22] some advantages are remarked: robustness to illumination changes and easy segmentation even when there is camera motion. In [2] a 3D hand model is adjusted to the cloud of points obtained from the captured depth image. In [17, 25–27, 29, 31] experiments, using depth sensors, are performed over static hand gestures collections, pointing out the advantages of using depth information. Another technology for obtaining range data is the one proposed in [23] where the scene is illuminated with a colored pattern, captured by a common RGB camera and later processed to infer depth information.

More concretely, there are several works which focus on the detection of motion pattern based gestures. In [36] a system for the detection of shape and motion based gestures is presented, using 2D images as input. It is evaluated for four different gestures, but only two different trajectories. Yoon et al. [37] recognizes 26 alphabetical gestures on the basis of features of location, angle and velocity. In [5], based on 3D motion captures obtained with an accelerometer, digits 0 to 9 drawn to the air are recognized. Kim et al. [15] presents a solution based on neural networks fed with spatiotemporal information. In [25] two simple motion patterns are taken into account (i.e. *MenuOpen* and *MenuClose*) which correspond to two of the gestures introduced in Section 4 (i.e. N and S). In Section 5 of [27] a whole motion-based gestures dictionary is proposed, it is the one used in this paper. In [19, 35] authours perform experiments using the MSRGesture3D dataset,[5] which includes 12 dynamic American Sign Language gestures. Among these gestures, following the taxonomy proposed in [27], we can find pose-based, pose-motion based and compound gestures, while the approach proposed in this paper is focused in motion-based ones. There are other datasets, such as MSRC-12 Kinect gesture data set [6], which includes a collection of gestures based on human body parts movements, something out of the scope of this work.

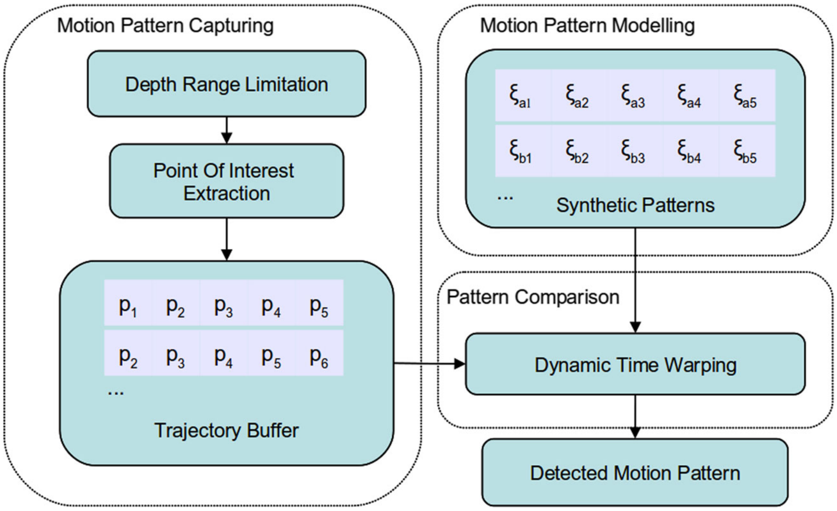
In this paper we present a novel non intrusive (i.e. there is no need of gloves or markers like in [13, 14, 34] or accelerometers like in [5]) real-time approach to the detection of intuitive motion based gestures usable in different application contexts. The learning phase of our approach does not need the capture of ground-truth real data, since the patterns are defined synthetically by using a human arm model (see Section 5.1) making it is user independent (differently to [5, 15, 36, 37]). During evaluation, performed with the collaboration of several users, the system worked properly, as the results presented in this paper confirm (see Section 6). Thanks to the proposed normalization (see Section 5.4) and the representativity of the chosen arm model (see Section 5.1) the system is robust to variations in the distance to the camera, in the height of the user and in the size of arm and hand. The use of TOF technology, apart from providing an accurate segmentation robust to low illumination conditions (not as in color camera based systems [4, 28, 32, 33, 38]), offers a representative point of the hand motion, the closest one to the camera, with no need of application of traditional segmentation techniques.

3 System Overview

In Fig. 1, an overview of the system is presented. First of all, the depth data range is limited to a maximum distance of 3 m, as explained in Section 4. The Point Of Interest (POI) to be tracked is computed, storing its coordinates from frame to frame (i.e. each p_i represents the 3 coordinates of the POI at frame i) which are an estimation of the hand trajectory. More concretely, the proposed POI is the point detected closest to the camera. An alternative POI is also proposed for evaluating purposes, this is the geodesic center of the segmented hand mask (see Section 5.3). Five samples trajectory segments (i.e. four translation segments) are compared with synthetically generated motion patterns (i.e. each ξ_{ai} represents the coordinates of pattern associated to gesture a at sample i) using the Dynamic Time Warping (DTW) distance as explained in Section 5.4. So, each translation segment will be locally labeled with the closest synthetic pattern. This results, along a gesture execution, in a collection of assigned labels to several translation segments. The

[5] http://research.microsoft.com/en-us/um/people/zliu/actionrecorsrc/

Figure 1 Overview of the system.


Motion Pattern Capturing
Depth Range Limitation
Point Of Interest Extraction
p_1 p_2 p_3 p_4 p_5
p_2 p_3 p_4 p_5 p_6
...
Trajectory Buffer
Motion Pattern Modelling
ξ_{a1} ξ_{a2} ξ_{a3} ξ_{a4} ξ_{a5}
ξ_{b1} ξ_{b2} ξ_{b3} ξ_{b4} ξ_{b5}
...
Synthetic Patterns
Pattern Comparison
Dynamic Time Warping
Detected Motion Pattern

final label of the gesture will be the most common assigned label.

4 Data Set

It is very important to have a representative data collection in order to obtain significant evaluation results. For this we use one of the dictionaries described in the dataset proposed in [27]. This is compound of nine gestures (see Fig. 2): slaps in 8 directions (named as the cardinal directions: N, NE, E, SE, S, SW, W and NW) and one slap getting closer and further to the camera (named IO, Inwards-Outwards). For compiling this collection 11 users were asked to execute 5 repetitions of each of the 9 gestures, what makes a total of 495 videos.[6] This collection is entirely used for evaluation purposes, since the knowledge used by the detection system is expressed by the motion patterns defined via the arm model described in Section 5.1. For recording the videos a TOF camera (SR4000 developed by Mesa Imaging)[7] was placed 1.5 m above the floor, with an horizontal orientation orthogonal to the user. This camera captures depth images with QCIF resolution (176 × 144 pixels) and a depth precision of ±1cm. It was configured to capture 30fps and to operate in a 3 m depth range (0.3–3.3m) in order to remove background objects. The recorded users were not asked to keep a certain distance to the camera neither to perform the gestures with any speed restriction. As well, the users had different heights, what makes the collection certainly representative of the potential users of the system. Some captures of this data set can be found in Fig. 3.

This dictionary of gestures was proposed following usability criteria, slaps executed in different directions are an intuitive way of interacting with a virtual environment. Two usability objectives [11] were taken into account in the gestures selection process: learnability and minimization of support requirements. In terms of learnability, it can be said that none of the users showed difficulties in learning the dictionary and that they only required of a brief introduction: they were asked to perform the indicated gestures as if they were interacting with a menu environment. In terms of minimization of support requirements, it can be said that no user presented doubts about how to execute the gestures.

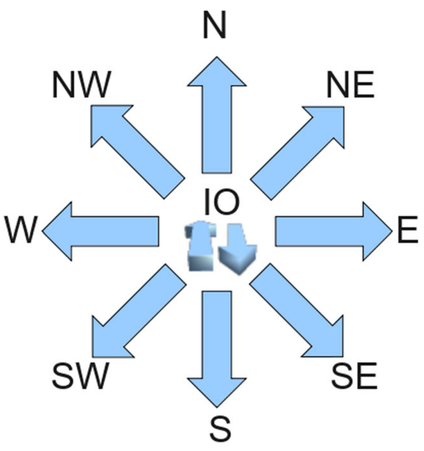


Figure 2 Gestures observed from user's point of view.

[6] http://www-vpu.eps.uam.es/publications/papermotion/indexpaper.html, (user: vision, password: visionpaper)

[7] http://www.mesa-imaging.ch/

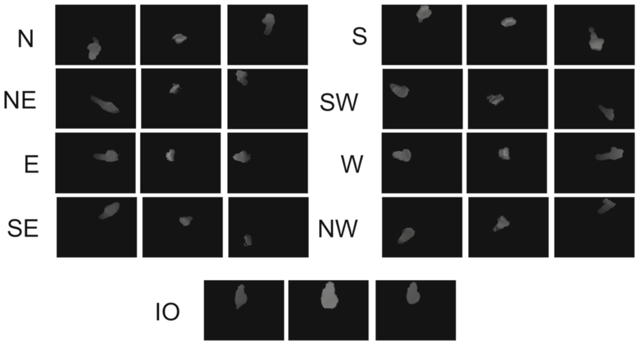


Figure 3 Depth captures of the proposed gestures for user 1. Notice that the temporal coordinate of the captures evolves from left to right.

5 Methodology

Our approach consists of the definition of synthetic motion patterns which will be compared with the hand motion estimations computed from the real data set videos.

5.1 Motion Pattern Modelling

An arm model, responding to human anatomy, has been proposed for the definition of the considered motion patterns. We consider two arm segments (see Fig. 4): the upper arm represented by the vector $\overrightarrow{r_U}$ which goes from the shoulder to the elbow and the lower arm reprensented by $\overrightarrow{r_L}$, from the elbow and to the wrist. The hand is not considered explicitly in this model, since the variation that could introduce is non significant in comparison with the ones shown by the arm movements. The lengths for these upper and lower segments were defined with fixed length: $\left|\overrightarrow{r_U}\right| = \left|\overrightarrow{r_L}\right| = 1$. Finally, the vector that describes the trajectory of the wrist to be analized is $\overrightarrow{r} = \overrightarrow{r_U} + \overrightarrow{r_L}$. In Fig. 4 some set-ups of the arm model are shown. Notice that for a variation of $\triangle\theta$ in angles θ^x and θ^y for the upper segment, the lower segment presents a variation of $2\triangle\theta$, acumulating this way the variation of the upper segment. The expression of the vectors $\overrightarrow{r_U}$ and $\overrightarrow{r_L}$ are the following:

- For gestures N and S (see Fig. 4a):

  $$\overrightarrow{r_U} = \left[0, -sin(\theta^x), cos(\theta^x)\right]$$

  $$\overrightarrow{r_L} = \left[0, -sin(2\theta^x), cos(2\theta^x)\right]$$

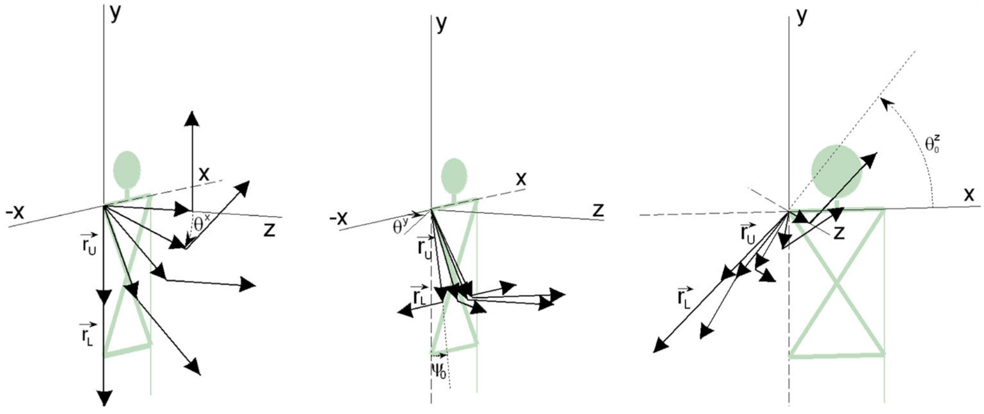
  where $\theta^x \in [0, \pi/2]$. For gesture N θ^x goes from $\pi/2$ to 0, while for gesture S from 0 to $\pi/2$. Notice that these two motion patterns are contained in plane yz.
- For gestures E and W (see Fig. 4b):

  $$\overrightarrow{r_U} = -sin(\psi_0)\left[cos(\theta^y), \frac{cos(\psi_0)}{sin(\psi_0)}, -sin\ (\theta^y)\right)\right]$$

  $$\overrightarrow{r_L} = \left[-cos(2\theta^y - \pi/2), 0, sin(2\theta^y - \pi/2)\right]$$

  where $\theta^y \in [\pi/4, 3\pi/4]$ and $\psi_0 = \frac{25° \times \pi rad}{180°}$. ψ_0 is the angle formed by the upper segment of the arm and $-\hat{y}$. For gesture E θ^y goes from $3\pi/4$ to $\pi/4$, while for gesture W from $\pi/4$ to $3\pi/4$. Notice that these two motion patterns are contained in plane xz.
- For NE, SE, SW and NW : a rotation about the z axis is performed over the gestures N and S (see Fig. 4c). This rotation matrix, R, is:

  $$R = \begin{bmatrix} sin\left(\theta_0^z\right) & cos\left(\theta_0^z\right) & 0 & 0 \\ -cos\left(\theta_0^z\right) & sin\left(\theta_0^z\right) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$



(a) N and S gestures. (b) E and W gestures. (c) NW and SE gestures.

Figure 4 Model set-ups of the arm model. $\overrightarrow{r_U}$ is a vector that goes from the shoulder to the elbow and $\overrightarrow{r_L}$ from the elbow to the wrist. The angles θ^x and θ^y are variables which define the trajectory of the arm in Fig. 4a, b, while ψ_0 and θ_0^z are fixed angles that define the position of the elbow at the beggining of the execution of the movement in Fig. 4b and c respectively. ψ_0 is the angle formed by $\overrightarrow{r_U}$ and $-\hat{y}$ (see Fig. 4b). θ_0^z indicates the rotation angle applied to N and S gestures, which results in the set-up shown in Fig. 4c.

and so, the homogenous coordinates for vectors $\overrightarrow{r_U}$ and $\overrightarrow{r_L}$ are:

$$\overrightarrow{r_U^{hom}} = R \times \left[0, -sin(\theta^x), cos(\theta^x), 0\right]'$$

$$\overrightarrow{r_L^{hom}} = R \times \left[0, -sin(2\theta^x), cos(2\theta^x), 0\right]'$$

where $\theta^x \in [0, \pi/2]$, as for gestures N and S, $\theta_0^z = \pi/4$ for gestures NW and SE and $\theta_0^z = 3\pi/4$ for gestures NE and SW. The application of these rotation matrixes implies that the modelled patterns are contained in the plane xz rotated about the z axis.

5.2 Motion Pattern Definition

The direction in which the defined intervals are covered depends on the direction of execution of the specific gesture, for example, in the case of gesture N θ^x for $\overrightarrow{r_U}$ begins in $\pi/2$ and ends in 0 while for gesture S is the other way around. In order to consider different speeds in the execution of the gestures 6 different patterns per gesture are presented: 1 for the whole arc , 1 for each half and 1 for each third. This makes 6 synthetic patterns per gesture. The selected length for these patterns was 5 samples (i.e. 4 translation segments) what defines the temporal window used for the comparison of synthetic and real patterns (see Fig. 1).

For the definition of the IO synthetic pattern no angles or arm model were considered, just a simpler approach was followed: the pattern was defined as a sequence of movements in the z axis. Three kinds of translations segments (i.e., an homogeneous motion interval) were considered: I, translation getting closer to the camera; O, moving away from the camera; S, staticity between two frames (applying the normalization described in Section 5.4 spurious translations are considered as staticity). Following the line of considering different execution speeds, various motion patterns (composed by 4 translation segments) were defined: IIII, IIIS, IISS, SSOO, SOOO, OOOO, IIIO, IIOO and IOOO. For example, if the execution of the gesture is very fast and only 5 samples are captured during it, the expected segments pattern would be IISS or SSOO. While, if the execution is slower sequences such as IIII or OOOO could be detected.

5.3 Motion Pattern Capturing

In order to capture a representative trajectory of the hand motion it is important to choose an easily traceable point. An unstable point would present noisy translations that could produce wrong estimations of the hand motion. The use of range information provides us with a robust to illumination and easy to detect POI, the closest to the camera. For the detection of this point it is not even necessary to previously segment the image.

With the intention of showing the advantages of using depth information, we also present an approach that makes no use of depth information (except for the depth range limitation): it extracts the tracking point considering the segmentation mask image resulting from the depth range limitation as binary (considering foreground all the pixels of the depth image with value over zero). In this case, the chosen tracking POI is the geodesic center of the binary mask, which is estimated by performing the ultimate erosion [20] up to a point.

5.4 Patterns Comparison

The comparison between two patterns is performed, not over the absolute coordinates of the trajectory, but over the translation of the POI between two frames. For calculating the distance between two patterns a previous normalization is performed, consisting of setting to one the length of each displacement between two sucesives samples frames of the POI. This solution has been used in problems such as hand writing recognition [10] or motion hand based gestures detection, like in [36] where the length of the translations is not used as a feature, something equivalent to fixing their length. In order to filter spurious errors in the detection of the tracked point when it is static (for gesture IO), this normalization is only applied when the magnitude of the translation of the POI between consecutive frames is over the third of the maximum one within the gesture execution. This defines an enough wide range of speeds for the proposed gestures which are intuitively executed in an homogenous way. The presented normalization makes the system independent to variations in the distance to the camera, in the angle of view, in the heigth of the user and in the size of the arm.

Once the synthetic (see Section 4) and captured motion patterns (see Section 5.3) are normalized, they are compared. The Dynamic Time Warping (DTW) distance has shown good performance when comparing temporal patterns executed at different speeds, concretely it has been widely applied to speech recognition problem [30]. An example of its application to hand gesture recognition can be found in [36]. Notice that each new captured motion pattern has four translation vectors, which describe the hand trajectory for five frames. It is then compared with each of the synthetic motion patterns present in the collection described in Section 5.1. This way we obtain a histogram of incidence of the closest synthetic patterns to this new captured motion pattern. The most common one gives us the label to assign to the gesture capture. If there is a tie between labels, the label 'Unknow' is the one assigned.

6 Experiments

6.1 Experimental Set-up

This section presents two different evaluation scenarios, both of them user independent since the learning process is performed using synthetic data and the evaluation is done with 11 different users (see Section 4):

1. 2.5D scenario: the tracked POI is the closest point to the camera and its depth coordinate (apart from x and y coordinates) is used for modelling the trajectory.
2. 2D information scenario: this second scenario was set-up considering the input images as binary masks as explained in Section 5.3. The depth information is implicitly used in the set-up of the camera (see Section 4), resulting in a segmentation mask, but this info is not used in the estimation of the hand trajectory. In this case, the tracked POI is the geodesic center of the binary mask, obtained with an iterative algorithm process [25]. Although the depth information is used for the calculation of this mask the z coordinate is not used in the comparison of the patterns.

The comparison of the results obtained for these two set-ups will permit to obtain conclusions about the utility of using depth information in hand gesture recognition.

6.2 Results

This section compiles the results obtained for the two evaluation scenarios introduced in Section 6.1:

1. 2.5D scenario: the resulting confussion matrix can be found in Table 1. The obtained accuracy rate is 0.951.
2. 2D information scenario: The obtained accuracy rate is 0.780 (see Table 2).

From the results compiled in Table 1 there are several aspects to point out:

- The label IO is the one assigned more times erroneously. It introduces 10 false negatives for executions of other gestures. This is due to the fact that the users tend to introduce the hand in the interaction area (and move it away) with upward and downward trajectories. These patterns are present in the definition of other gestures, apart from IO, producing misclassifications.
- When the assigned labels within an execution results on the same score for 2 or more gestures the assigned label is *Unknown* (U). This situation produces 7 misclassifications.
- Without taking into account the missclassifications produced by the inclusion of the IO gesture (i.e. the only one which translation is fundamentally takes place in the depth coordinate), the obtained accuracy rates are, 0.873 for the 2D scenario and 0.977 for the 2.5D one. So, the use of depth information improves the results even when the gestures are apparently detectable using only 2D information.

Table 2 presents not such good results, mainly due to the instability of the geodesic center. Since no depth information is considered, the representative point to be tracked needs to be estimated on the basis of a segmentation which is noisy due to variation in its shape and size. So, noisy translations are added to the real translations of the hand.

As far as we know, no user-indepent evaluations have been performed for motion based gestures detection, consequently we enumerate the evaluation figures of some works in which the absence of overlap between train and evaluation corpora is not ensured. In [36] a 0.97 accuracy rate is obtained in separating only two motion patterns. [5] presents results for an intrusive approach based on the use of an accelerometer: obtaining 0.93 for 5-fold cross validation and 0.98 for 10-fold cross validation, in the detection

Table 1 Confusion matrix for the 2.5D scenario.

	U	N	S	W	E	SW	NW	SE	NE	IO
N	0	52	0	0	0	0	0	0	0	3
S	1	0	50	0	0	0	0	0	0	4
W	1	0	0	53	0	0	0	0	0	1
E	0	0	0	0	55	0	0	0	0	0
SW	0	0	0	0	0	55	0	0	0	0
NW	2	2	0	0	0	0	51	0	0	0
SE	2	0	0	0	0	0	0	51	0	2
NE	1	0	0	0	1	0	0	0	53	0
IO	0	1	0	2	0	0	0	0	1	51

Gestures described in Section 4 and "U" for Unknown.

Table 2 Confusion matrix for the 2D scenario.

	U	N	S	W	E	SW	NW	SE	NE	IO
N	0	51	0	0	0	0	0	0	0	4
S	1	0	26	0	0	1	0	0	0	27
W	1	0	0	37	0	0	16	0	0	1
E	0	0	0	0	38	0	0	6	9	2
SW	0	0	0	1	0	47	1	0	0	6
NW	2	4	0	2	0	0	44	1	0	2
SE	2	0	0	0	0	0	0	49	0	4
NE	1	2	0	0	0	0	0	0	51	1
IO	0	1	0	1	0	0	7	1	2	43

Gestures described in Section 4 and "U" for Unknown.

of 0 to 9 digits. Kim et al. [15] separates 6 gestures on the basis of the posture and motion of the hand, obtaining an accuracy of 0.975 for the best setup. In [37], the highest accuracy rate in the detection of 26 gestures drawn to the air is 0.932. In [25], two of the considered gestures were N and S, obtaining a mean recall of 0.938 in their detection. So we can say that our approach achieves results comparable to the ones of the State Of Art, even when they do not present user-independent evaluations.

6.3 Computational Cost

We can express the computational cost as a function depending on the number of translation segments for each motion pattern, N , and the number of synthetical patterns, N_{SynPat}, contained in the collection described in Section 5.1. We have consider, as significant, the periods necessary for performing a sum, T_S, a product, T_P , and a square root T_{sqrt}. The different stages considered on this work will present the following computational times per frame:

1. POI sampling: In the case of the 2.5D scenario, this is the time needed to compute the position of the closest pixel, for what is necessary to perform $width \times height - 1$ comparisons, so $T_{A-3D} = (width \times height - 1) \times (N + 1) \times T_S$. In the 2D scenario we have to take into account the time for extracting the geodesic center of the binary mask as described in [25], $T_{A-2D} = 4.311\ msec$.
2. Trajectory computation: This is the time needed for calculating the trajectory vector on the basis of the point coordinates, $T_B = 3 \times N \times T_S$.
3. Trajectory Normalization: as described in Section 5.4, $T_C = N \times (5 \times T_S + 6 \times T_P + T_{sqrt})$.
4. DTW computation: $T_D = N^2 \times N_{SynPat} \times (5 \times T_S + 3 \times T_P + T_{sqrt})$.

Current Float Point Units offer a solution for the computation of arithmetic operations with dedicated hardware, achieving computational times in the same order of magnitude for sum, product and squared root. On the basis of Pentium speed tests[8] we can establish the following relation between T_S, T_P and T_{sqrt}, defining T_0 as the reference computational time: $T_S \backsimeq T_P = T_0$ and $T_{sqrt} = 2 \times T_0$. Doing so, and on the basis of the presented expressions, we obtain a total computational time of $T = T_A + T_B + T_C + T_D = T_A + T_0 \times N \times (16 + 10 \times N \times N_{SynPat})$. With $N = 4$ and $N_{SynPat} = 54$ we obtain $T = T_A + 8704 \times T_0$. A CPU performance test was run on an Intel(R) Core(TM)2 Duo CPU E7500 @ 2.93Ghz with 2.98GB RAM, as in [25], being the obtained T_0 below $1 nsec$. So $T_{3D} = T_{A-3D} + 8704 \times T_0 = 135419 \times T_0$ ($T_{3D} < 0.136\ msecs$) and $T_{2D} = T_{A-2D} + 8704 \times T_0$ ($T_{2D} < 4.321\ msecs$).

Table 3 Computational Costs per frame and Accuracy for the two considered scenarios.

Scenario→	2.5D	2D
Comp. cost($msec/frame$)	< 0.136	< 4.321
Accuracy	0.951	0.780

As shown in Table 3, the described approaches require much less than $1/25 sec$ per frame, enabling real-time HCI.

7 Conclusions

In this paper a non intrusive motion-based hand gesture detection system using range data is presented. It is able to work in real-time allowing the interaction between a user and a virtual environment or computer menu. It is robust to the relative camera position and to the speed of execution of the gestures. It is, as well, user-independent, being able to work with a collection of gestures executed by users of different heights and arm's sizes. A novel definition of the motion patterns, based on human anatomy, is presented: the obtained results bear witness to its remarkable representation capacity. A significant data set of depth videos has been compiled and made available for researching purposes (see Section 4).

From the results we confirm that the use of depth information for the hand trajectory estimation implies a significant increase in gesture detection accuracy rate. Our approach (2.5D scenario) works without the need of applying any segmentation algorithm (apart from limitating the depth range of the capture) or calculating the geodesic center of the hand mask, as in the 2D scenario, which means a lower computation time (see Table 3). The achieved accuracy rate for the proposed dictionary, performing a user-independent evaluation , is 0.951, a very promising value, as already mentioned, comparable to the results of the State Of Art. The experiments performed in this work also show that the 2.5D approach performs better that the 2D, even without considering the only gesture with a clear translation just in the depth coordinate, the IO gesture.

In the light of the results described in Section 6 we consider two main future work lines:

- The use of a Hidden Markov Model in order to manage the temporal sequence of detected labels. This could solve some misclasification situations in which the order of the detections is relevant.

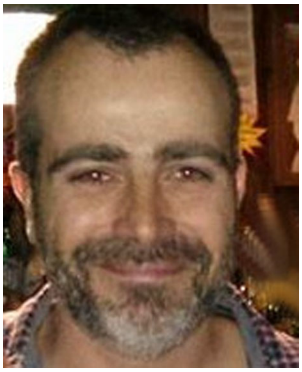
[8] http://www.obliquity.com/computer/speedtest.html

- The use of color-depth registration approaches [12] could improve the quality of the hand motion estimation, and make feasible the detection of more complex gestures.

References

1. Athitsos, V., & Sclaroff, S. (2003). Estimating 3d hand pose from a cluttered image. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, *2*, 432.
2. Breuer, P., Eckes, C., & Muller, S. (2007). Hand gesture recognition with a novel ir time-of-flight range camera: a pilot study. In *Computer vision/computer graphics collaboration techniques 3rd international conference, MIRAGE* (pp. 247–260).
3. Castilla, D., Miralles, I., Jorquera, M., Botella, C., Baños, R., Montesa, J., & Ferran, C. (2009). Analysis and testing of metaphors for the definition of a gestual language based on real users interaction: vision project. In *13th international conference on human-computer interaction*. San Diego.
4. Chen, Y.T., & Tseng, K.T. (2007). Developing a multiple-angle hand gesture recognition system for human machine interactions. In *33rd annual conference of the IEEE industrial electronics society, 2007. IECON 2007* (pp. 489–492).
5. Cheng, J., Xie, C., Bian, W., & Tao, D. (2012). Feature fusion for 3d hand gesture recognition by learning a shared hidden space. *Pattern Recognition Letters*, *33*(4), 476–484. Intelligent Multimedia Interactivity.
6. Fothergill, S., Mentis, H.M., Kohli, P., & Nowozin, S. (2012). Instructing people for training gestural interactive systems, In J. A. Konstan, E. H. Chi, & K. Höök (Eds.), *CHI* (pp. 1737–1746). ACM.
7. Grzeszczuk, R., Bradski, G., Chu, M., & Bouguet, J. (2000). Stereo based gesture recognition invariant to 3d pose and lighting. In *IEEE conference on computer vision and pattern recognition* (pp. I: 826–833).
8. Guomundsson, S., Pardás, M., Larsen, R., Aanaes H., & Casas, J.R. (2010). TOF imaging in smart room environments towards improved people tracking. *Computer Vision and Image Understanding*, *114*(12), 1376–1384.
9. Holden, E.J., Lee, G., & Owens, R. (2005). Australian sign language recognition. *Machine Vision and Applications*, *16*, 312–320.
10. Hu, J., Brown, M.K., & Turin, W. (1996). Hmm based on-line handwriting recognition. *IEEE Trans. Pattern Anal. Mach. Intell.*, *18*, 1039–1045.
11. ISO9241-11. (1998). Ergonomic requirements for office work with visual display terminals (vdts) - part 11: Guidance on usability.
12. Jang, I.Y., & Lee, K. (2010). Depth video based human model reconstruction resolving self-occlusion. *IEEE Transactions on Consumer Electronics*, *56*(3), 1933–1941.
13. Kelly, D., McDonald, J., & Markham, C. (2010). A person independent system for recognition of hand postures used in sign language. *Pattern Recognition Letters*, *31*(11), 1359–1368.
14. Keskin, C., & Akarun, L. (2009). Stars: sign tracking and recognition system using input-output hmms. *Pattern Recognition Letters*, *30*(12), 1086–1095. Image/video-based Pattern Analysis and HCI Applications.
15. Kim, H.J., Lee, J., & Park, J.H. (2008). Dynamic hand gesture recognition using a cnn model with 3d receptive fields. In *2008 international conference on neural networks and signal processing* (pp. 14–19).
16. Kim, S.Y., Cho, J.H., Koschan, A., & Abidi, M.A. (2010). *Spatial and temporal enhancement of depth images captured by a time-of-flight depth sensor*, (pp. 2358–2361).
17. Kollorz, E., Penne, J., Hornegger, J., & Barke, A. (2008). Gesture recognition with a time-of-flight camera. *International Journal of Intelligent Systems Technologies and Applications*, *5*(3/4), 334–343.
18. Kong, W., & Ranganath, S. (2010). Sign language phoneme transcription with rule-based hand trajectory segmentation. *Journal of Signal Processing Systems*, *59*(2), 211–222.
19. Kurakin, A., Zhang, Z., & Liu, Z. (2012). A real time system for dynamic hand gesture recognition with a depth sensor. In *Proceedings of the 20th european signal processing conference, EUSIPCO 2012, Bucharest, Romania* (pp. 1975–1979).
20. Lantuejoul, C., & Maisonneuve, F. (1984). Geodesic methods in quantitative image analysis. *Pattern Recognition*, *17*(2), 177–187.
21. Laviola, J.J. (2008). Bringing vr and spatial 3d interaction to the masses through video games. *IEEE Computer Graphics and Applications*, *28*(5), 10–15.
22. Liu, X., & Fujimura, K. (2004). Hand gesture recognition using depth data. In *Proceedings of the 6th IEEE international conference on automatic face and gesture recognition, 2004* (pp. 529–534).
23. Malassiotis, S., & Strintzis, M. (2008). Real-time hand posture recognition using range data. *Image and Vision Computing*, *26*(7), 1027–1037.
24. Mitra, S., & Acharya, T. (2007). Gesture recognition: a survey. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, *37*(3), 311–324.
25. Molina, J., Escudero-Viñolo, M., Signoriello, A., Pardás, M., Ferrán, C., Bescós, J., Marqués, F., & Martínez, J. (2013). Real-time user independent hand gesture recognition from time-of-flight camera video using static and dynamic models. *Machine Vision and Applications*, *24*(1), 187–204.
26. Molina, J., & Martínez, J.M. (2014). A synthetic training framework for providing gesture scalability to 2.5d pose-based hand gesture recognition systems. *Machine Vision And Applications*, *25*(5), 1309–1315.
27. Molina, J., Pajuelo, J.A., Escudero-Viñolo, M., Bescós, J., & Martínez, J.M. (2014). A natural and synthetic corpus for benchmarking of hand gesture recognition systems. *Machine Vision and Applications*, *25*(4), 943–954.
28. Nickel, K., & Stiefelhagen, R. (2007). Visual recognition of pointing gestures for human-robot interaction. *Image and Vision Computing*, *25*(12), 1875–1884.
29. Qin, S., Zhu, X., Yang, Y., & Jiang, Y. (2014). Real-time hand gesture recognition from depth images using convex shape decomposition method. *Journal of Signal Processing Systems*, *74*(1), 47–58.
30. Sakoe, H., & Chiba, S. (1978). Dynamic programming algorithm optimization for spoken word recognition. IEEE Transactions on Acoustics. *Speech and Signal Processing*, *26*(1), 43–49.
31. Soutschek, S., Penne, J., Hornegger, J., & Kornhuber, J. (2008). 3-d gesture-based scene navigation in medical imaging applications using time-of-flight cameras. In *IEEE computer society conference on computer vision and pattern recognition workshops* (pp. 1–6).

32. Stenger, B., Thayananthan, A., Torr, P.H.S., & Cipolla, R. (2006). Model-based hand tracking using a hierarchical bayesian filter. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *28*(9), 1372–1384.
33. Teng, X., Wu, B., Yu, W., & Liu, C. (2005). A hand gesture recognition system based on local linear embedding. *Journal of Visual Languages and Computing*, *16*, 442–454.
34. Usabiaga, J., Erol, A., Bebis, G., Boyle, R., & Twombly, X. (2009). Global hand pose estimation by multiple camera ellipse tracking. *Machine Vision and Applications*, *21*, 1–15.
35. Wang, J., Liu, Z., Chorowski, J., Chen, Z., & Wu, Y. (2012). Robust 3d action recognition with random occupancy patterns. In *Proceedings of the 12th european conference on computer vision - volume part II, ECCV'12* (pp. 872–885). Berlin: Springer.
36. Wenjun, T., Chengdong, W., Shuying, Z., & Li, J. (2010). Dynamic hand gesture recognition using motion trajectories and key frames. In *2010 2nd international conference on advanced computer control (ICACC)* (vol. 3 pp. 163–167).
37. Yoon, H.S., Soh, J., Bae, Y.J., & Yang, H.S. (2001). Hand gesture recognition using combined features of location, angle and velocity. *Pattern Recognition*, *34*(7), 1491–1501.
38. Zheng, G., Wang, C.J., & Boult, T.E. (2007). Application of projective invariants in hand geometry biometrics. *IEEE Transactions on Information Forensics and Security*, *2*(4), 758–768.



Javier Molina received his title as Ingeniero de Telecomunicación (M.S. degree in electrical engineering) in 2002 at the Universidad Politécnica de Madrid (UPM) and the Doctor Ingeniero en Informática y Telecomunicaciones degree (Ph.D. in Computer Science and Telecommunications) in 2012 at the Universidad Autónoma de Madrid (UAM). He worked as a Research and Development Engineer in Teldat S.L. in 2004. In 2005, he moved to the Video Processing and Understanding Laboratory (VPU-Lab) in UAM and became a Ph.D. candidate. Since then, he has been actively involved in European and national research projects. His professional interests include image/video understanding and human-computer interaction. He has been working as a freelance Computer Vision specialist since January 2013.

José A. Pajuelo received the Ingeniero de Telecomunicación title (M.S. degree in electrical engineering) in 2010 and his M.Phil. degree in multimedia signal processing at 2012, both from the Universidad Autónoma de Madrid. Since 2009 he has been a member of the VPU-Lab, focusing his research in the analysis of semantic information of multimedia content (object recognition) and Human-Computer Interaction (hand gesture recognition).

José M. Martínez received the Ingeniero de Telecomunicación degree (six years engineering program) in 1991 and the Doctor Ingeniero de Telecomunicación degree (PhD in Communications) in 1998, both from the E.T.S. Ingenieros de Telecomunicación of the Universidad Politécnica de Madrid. Since 2002 he is Associate Professor at the Escuela Politécnica Superior of the Universidad Autónoma de Madrid. His professional interests cover different video processing and understanding aspects, in the last year as with a special focus of advanced video surveillance systems and multimedia information systems. Besides his participation in several Spanish national projects (both with public and private funding), he has been actively involved in European projects dealing with multimedia information systems applied to the cultural heritage, education and semantic multimedia networked systems.

He is author and co-author of more than 130 papers in international journals and conferences, and coauthor of the first book about the MPEG-7 Standard published 2002. He was actively involved in the development of the MPEG-7 standard, being co-editor of two parts of it.

He has acted as reviewer for journals and conferences. Since 2014 he is Associate Editor of IEEE Trans. On Circuits and Systems for Video Technology.