Gesture recognition supporting the interaction of humans with socially assistive robots

Damien Michel, Konstantinos Papoutsakis, and Antonis A. Argyros

Institute of Computer Science, FORTH
and
Computer Science Department, University of Crete

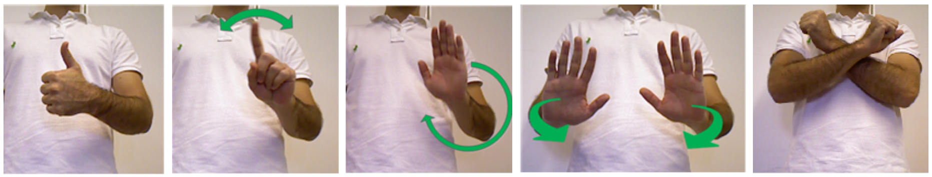
{michel|papoutsa|argyros}@ics.forth.gr – http://www.ics.forth.gr/cvrl/

Abstract. We propose a new approach for vision-based gesture recognition to support robust and efficient human robot interaction towards developing socially assistive robots. The considered gestural vocabulary consists of five, user specified hand gestures that convey messages of fundamental importance in the context of human-robot dialogue. Despite their small number, the recognition of these gestures exhibits considerable challenges. Aiming at natural, easy-to-memorize means of interaction, users have identified gestures consisting of both static and dynamic hand configurations that involve different scales of observation (from arms to fingers) and exhibit intrinsic ambiguities. Moreover, the gestures need to be recognized regardless of the multifaceted variability of the human subjects performing them. Recognition needs to be performed online, in continuous video streams containing other irrelevant/unmodeled motions. All the above need to be achieved by analyzing information acquired by a possibly moving RGBD camera, in cluttered environments with considerable light variations. We present a gesture recognition method that addresses the above challenges, as well as promising experimental results obtained from relevant user trials.

1 Introduction

Vision-based gesture recognition is aiming at recognizing meaningful physical movements that are performed by humans, through the processing and analysis of visual information acquired by a camera system. In recent years, this has been a highly active research area which, in many cases, has been of multidisciplinary nature. The significant research efforts devoted to the problem have been motivated by wide-ranging applications in many commercial/business domains, that can benefit from a robust solution.

Besides being interesting, the problem exhibits significant difficulties. Gestures can be of varying complexity and their recognition is also affected by the scene context, actions that are performed in the fore- or the back-ground at the same time, as well as by preceding and/or following actions. Moreover, gestures are often language- and culture-specific, providing additional evidence to substantiate the interesting as well as challenging nature of the problem.



(a) Yes (b) No (c) Reward (d) Stop/cancel (e) Help

Fig. 1. Illustration of the supported gestures. The correspondence between gestures and physical actions of hands/arms are as follows: (a) “Yes”: A “thumb up” hand posture. (b) “No”: A sideways waiving hand with extended index finger. (c) “Reward”: A circular motion of an open palm at a plane parallel to the image plane. (d) “Stop/cancel”: A two-handed push forward gesture. (e) “Help”: two arms in a cross configuration.

In this work, we focus on vision-based recognition of hand gestures [1, 2]. Our goal is to support robust and natural interaction of a human and an autonomous socially assistive robot, enhancing multi-modal human-robot interaction. The target user group consists of people of all ages and variable familiarity with technology. Therefore, a set of intuitive gestures has been defined, conveying messages of fundamental importance in a dialogue, such as “Yes”, “No”, “Reward”, “Stop/Cancel” and “Help”. These are realized by a subject with a variety of finger, palm, and arm movements (see Fig. 1). A novel method for gesture recognition is proposed consisting of a complete system that detects and tracks arms, hands and fingers, and performs spatio-temporal segmentation and recognition of the set of predefined gestures, based on data acquired by an RGBD sensor. A model of the hand is employed to detect hand and finger candidates. At a higher level, hand posture models are defined and serve as building blocks to recognize gestures based on the temporal evolution of the detected postures. We note that our methodology performs automatic spatial and temporal segmentation of gestures and can simultaneously handle a possibly varying number of hands that may occlude each other in the field of view of the camera. Moreover, it is capable of meeting a broad range of challenging user requirements that are discussed in detail in Sec. 3.

To assess quantitatively and qualitatively the performance of the proposed solution, twelve persons were engaged in user tests. These persons were divided in two different age groups and also varied with respect to their gender and familiarity and expertise in technological systems. The training of all persons were kept minimal. All subjects performed the supported set of gestures several times and in random order in the context of various other gestures, which closely resemble the supported ones. The obtained experimental results confirm the effectiveness and the robustness of the proposed approach.

2 Related Work

According to the review by Aggarwal et al [3] on the broader area of human activity recognition, human activities can be conceptually categorized into four different levels depending on their complexity: gestures, actions, interactions, and group activities. Gestures are defined as elementary movements of a persons body part, most commonly involving the hands, arms, face, head, defining the expressive atomic components that describe the meaningful motion of a person. Gestures can be static or dynamic, while some gestures also have both static and dynamic elements, as in sign languages [4]. Subsequently, actions are single person activities that may be composed of multiple gestures organized temporally, such as "walking", "waving" and "punching". More resources on vision-based action recognition [5], human motion analysis [6] and hand pose estimation [7] are available in the literature.

We focus on the problem of hand-based gesture recognition [2, 8]. Despite the considerable research efforts devoted to it, this remains a challenging task. Early techniques employed HMMs [4], Neural Networks [9], Kalman and particle filtering [10] to efficiently model the hand, estimate its pose and recognize static and dynamic gestures. A real-time implementation of gesture recognition for robot control was developed in [11] combining skin color-based, shape-based recognition and Kalman-filtering for hand detection and tracking, while HMMs are used for temporal segmentation of hand gestures. The work in [12] uses similar tools introducing the notion of HMM codebook towards gesture recognition. A noteworthy technique regards the Finite State Machine (FSM) which model the gesture as a sequence of states in temporal order capturing also the semantics of the movements. It is used in many methodologies [13] to model and analyse simple, complex or manipulative gestures in the spatio-temporal configuration space. A recent method by Baraldi et al [14] in the context of the emerging field of ego-centric vision, combines gesture recognition and hand segmentation, modelling both static and dynamic gestures as a collection of dense trajectories extracted around the detected hand regions.

Other methods rely on extracted 3D trajectories or angles of body joints and/or segmented gestures, concentrating in the classification task. In the work of Raptis et al [15], a classification scheme is designed and trained, based on a novel angular skeletal representation of body motion acquired using the Kinect^{TM} platform, in order to recognize from among 28 gesture classes in real-time, in the context of dancing.

Most recent advancements introduce the concept of personalized gesture recognition as a means to resolve some of the difficulties in the domain. They focus on the interpretation and assignment of the gestures to meaningful user-level system commands, which is a crucial task in order to achieve efficient natural-based interactivity between humans and computers. However, they require an additional process in order to collect personalization data and optimize the performance of the system. In many cases a training procedure [16] is required to be performed by each user in order for a gestural interactive system to collect data and train a learning-based methodology to finally adapt its performance to the individual. In the same context, a learning-based methodology for personalization was recently proposed by Yao et al [17]. Unlike other personalization methods [18] which learn a single classifier that later gets adapted, their approach learns a set (portfolio) of classifiers during training, one of which is selected for each test subject based on the personalization data.

3 Application Requirements and Gestural Vocabulary

In order to define a compact gestural vocabulary, a survey has been conducted based on a group of potential target users that have been interviewed on their preferences on the physical movements to perform to trigger specific robot tasks. An initial set of gestures was provided to the users to vote for their preferences. The selection of those gestures was driven by experts in the fields of human-computer and human-robot interaction. Special attention was put so that the selected gestures are physically easy to be performed by users of all ages and intuitive enough to be remembered in daily routine. Alternatively, the users could propose a new gesture, not belonging to the originally proposed set. This iterative process converged to the definition of five gestures that are illustrated and defined in detail in Fig.1. Despite their small number, their recognition is challenging for a number of reasons.

Mix of static and dynamic components: Gestures are defined as static postures ("Yes", "Help"), but also as temporally evolving, dynamic gestures ("No", "Reward", "Stop/ Cancel").

Broad scale of observations: The recognition of gestures depend on modelling and recognizing human body parts at different scales, e.g., from single-handed hand postures involving fingers ("No"), to bi-manual postures involving two arms ("Help").

Intrinsic ambiguities: In the quest for natural interaction, users defined gestures with intrinsic ambiguities. For example, the hand shape in "Yes" and "No" or in "Reward" and "Stop/cancel" are quite similar.

Broad variability of users: The gestures should be recognized for a broad range of parameters related to the biometric characteristics of the subjects, their age, their mobility capabilities, the specific way they perform gestures, etc.

Recognition in the context of unknown actions: The gestures need to be recognized online, in continuous video streams. Therefore, they should be segmented and identified robustly in the context of other, arbitrary and unmodeled hand motions.

Robustness to illumination changes, camera motion and scene clutter: The defined gestures need to be recognized by an assistive robot operating at a user's home. Therefore, gestures should be recognized by a potentially moving camera, in varying illumination conditions and with robustness to scene clutter.

In the following sections, we provide a detailed description of the proposed framework as well as on the experiments perform to validate it with respect to the above challenging characteristics an requirements.

4 The Proposed Approach

The proposed framework encompasses a collection of techniques that enable robust, real-time and efficient gesture recognition based on visual information acquired by an RGBD camera. To achieve the recognition of the aforementioned gestures, detection and tracking of multiple hands and fingers is initially performed based on an effective layered representation of a hand model consisting of the wrist, palm and fingers. Temporal association of the computed hand candidates across time is also performed. By analysing the available 3D trajectory and geometric properties of the fitted model for each hand candidate in the scene, segmentation and recognition of the gestural actions is finally performed.

4.1 Depth-based edge detection and skeletonization

At each time instant t, an RGBD frame is acquired (see Fig. 2(a),(b)). We denote the depth frame by I_t. We assume that intrinsic calibration data of the camera is available, enabling the conversion of the acquired depth pixels to a 3D point cloud representation.

As a first processing step, the depth-based edge map G_t is calculated. Let $p = I_t(i, j)$ be an image point and let $N(p)$ denote the set of its eight immediate neighbours in its 3×3 neighbourhood. A point $G_t(p)$ is set as a depth edge point, if its 3D Euclidean distance to any of its neighbours is higher than a threshold value T_D. In notation,

$$G_t(p) = \begin{cases} 1, if \ ||I_t(p) - I_t(p')||_2 > T_D, \ \forall p' \epsilon N(p) \\ 0, \text{otherwise.} \end{cases} \tag{1}$$

Additionally, a contour map C_t is computed in order to refine G_t. More specifically,

$$C_t(p) = \begin{cases} 1, \ if \ \sum_{p' \epsilon N(p)} G(p') > 0 \\ 0, \ \text{otherwise.} \end{cases} \tag{2}$$

Practically, a point p is considered as a contour point if at least one of its neighbours is a depth edge point. Subsequently, we produce a binary image map M_t as follows:

$$M_t(p) = \begin{cases} 1, \ if \ I_t(p) \leq T_V, \ C_t(p) = 0 \\ 0, \ \text{otherwise.} \end{cases} \tag{3}$$

The intuition behind M_t is the following. A point in M_t is considered as background (0) if its depth value is greater than a threshold F_V, or if it is a depth discontinuity ($C(p) = 1$). Essentially, this means that all distant scene points are considered irrelevant and further processing is restricted in a depth range defined by T_V. Additionally, depth discontinuities appear as background pixels. In our experiments, the distance thresholds T_D and T_V are set to 30 mm and 1500 mm, respectively. An example map M_t is shown in Fig. 2(c).

As a final preprocessing step, we compute the skeleton of M_t using morphological filtering [19]. Let S_t denote a binary image where only skeletal points appear as foreground. S_t appears in Fig. 2(c) (red pixels superimposed on M_t). A different skeleton is identified for each of the connected components of M_t.

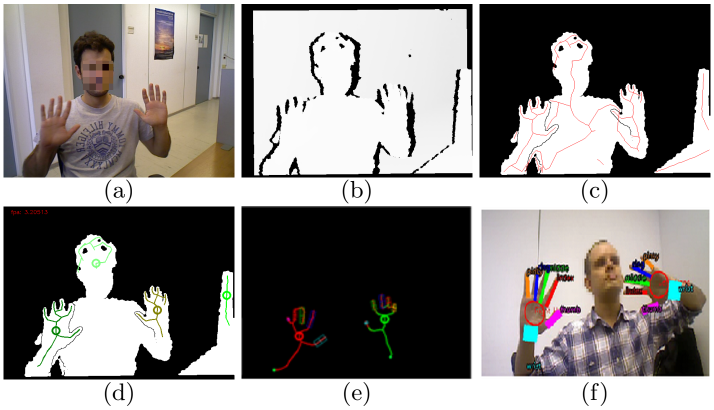

(a) (b) (c)
(d) (e) (f)

Fig. 2. Illustration of intermediate results for hand detection. (a) Input RGB frame. (b) Input depth frame I_t. (c) The binary mask M_t where far-away structures have been suppressed and depth discontinuities of I_t appear as background pixels. Skeleton points S_t are shown superimposed (red pixels). (d) A forest of minimum spanning trees is computed based on (c), identifying initial hand hypotheses. Circles represent the palm centers. (e) Checking hypotheses against a simple hand model facilitates the detection of the actual hands, filtering out wrong hypotheses. (f) Another example showing the detection results (wrist, palm, fingers) in a scene with two hands.

4.2 Forming hand hypotheses

Once S_t is computed, we set out to compute a forest of minimum weight spanning trees [20] (one spanning tree T for each skeleton). This is based on a graph representation of points of a skeleton. More specifically, two points of a specific skeleton are considered connected if their 3D distance is lower than a threshold that is set equal to 100 mm in our experiments. Otherwise, their distance is set equal to infinity.

Based on the fact that candidate hands are more likely to be localized on the extremities (leafs) of each minimum spanning tree T, our goal is to perform segmentation of each of the spanning trees by calculating optimal cut points and tree branches that correspond to hand structures. Searching for an optimal cut, we traverse a minimum spanning tree T starting from any of its leaf nodes towards any other leaf node, as long as the spanning tree nodes and the corresponding structure do not exceed the size of an average human hand (180 mm). Several cuts will satisfy the described constraints, resulting in different overlapping trees. From each set of overlapping trees, we keep the largest one. The remaining trees constitute the initial hand hypotheses h. Figure 2(d) shows four such identified hypotheses. As it can be verified, although all actual hands have been identified, false positives do exist.

4.3 Hand detection

To filter-out wrong hand hypotheses, we deploy a simple 2D hand model that is compared against each of the computed hand hypotheses h. The employed

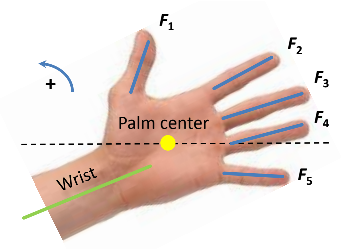


Fig. 3. The 2D hand model utilized to detect hand candidates. Fingers F_n, their orientation δ_n and the wrist orientation ϕ are considered with respect to the detected palm center and the horizontal line of the image plane.

hand model consists of (a) a wrist region and its orientation, (b) a palm center and, (c) up to five fingers (see Fig. 3). The orientations $\delta_i \in [-\pi..\pi)$ of each finger and the orientation $\phi \in [-\pi..\pi)$ of the wrist are computed with respect to the x-axis of the image coordinate system and are considered positive in the counterclockwise direction. These hand components are sequentially fit in each hand hypothesis.

Palm detection: The palm center of the hand candidate is estimated by finding the local maximum of the distance transformed M_t, in the region spanned by the hand hypothesis h. Intuitively, such a point is the center of a relatively large and compact area that matches closely the shape of a palm.

Finger detection: Given the estimated palm center of each hand candidate, we set out to further compute the positions and orientations of the finger candidates. To do so, we compute a *skeletal shape descriptor* K on the skeletal points of S_t. Each such descriptor consists of two components. The first is the local slope of the skeleton. Assuming that the descriptor is computer at a point p, this local slope is estimated by fitting a straight line to the skeleton points located within a radius of 5 pixels from p. The second component of the descriptor is the 3D Euclidean distance of p to the closest background point in M_t in a direction perpendicular to its local slope.

A finger candidate is localized by sequentially grouping skeletal descriptors K, starting from the skeletal point of hand hypothesis h that is closest to the palm center, towards its leaf nodes. This is achieved by applying a set of geometric constraints that reflect the structural properties of the position and orientation of a finger candidate with respect to the palm center.

Based on the computed finger candidates, a set of additional features are also calculated with respect to the skeletal descriptors assigned to each finger. Those regard the center, direction, tip, root, and width of each finger. The center and direction are estimated by averaging the corresponding values of all descriptors, while the tip and the root are defined as the furthest and closest descriptors from the palm center, respectively. Subsequently, a finger candidate must respect a set of constrains based on these feature values in order to be attached to the hand model. More specifically, the orientation of each finger is expected to be

Posture	Wrist	Number of fingers	Finger orientation
Thumb up	$\|\phi\ mod\ \pi\| < \frac{\pi}{4}$	1	$\frac{\pi}{4} < \delta_0 < \frac{3\pi}{4}$
Index up	$\|\phi + \frac{\pi}{2}\| < \frac{\pi}{4}$	1-2	$\frac{\pi}{4} < \delta_0 < \frac{3\pi}{4}$ & $\delta_1 < \frac{\pi}{4}$ (optional)
Other	$\|\phi + \frac{\pi}{2}\| < \frac{3\pi}{4}$	0-5	No ‘thumb up’ or ‘Index-up’

Table 1. The rules used to assign hand hypotheses to posture classes based on the values of the hand model parameters (see Fig.3).

roughly towards the palm center, thereby the projection of the palm center pixel to the line defined by the finger center and the finger direction is considered. The finger candidate is considered as valid if the 3D distance between the center and its projection is less that the expected size of the palm.

Wrist detection: If two or more fingers have been detected for a hand hypothesis, the position and orientation of its wrist is computed by fitting a 3D line to the skeletal points starting from the palm center, in a direction opposite to the fingers and up to a distance of 20 cm.

Figure 2(e) shows that by employing the aforementioned techniques, the false hand hypotheses of Fig. 2(d) have been removed. Figure 2(f) provides a similar example where sample, low level hand detection detection results are shown.

4.4 Hand tracking

In order to track hands in time, we adopt a simple tracking-by-detection approach. More specifically, the hands that are detected at time t are associated to the closest hands detected at time $t-1$. In this context, proximity is defined based on the 3D space covered by each hand hypothesis at each time t. Rules similar to the ones employed in the blob tracker presented in [21] are used to handle the introduction of a new hand and the disappearance of a tracked one.

4.5 Hand posture recognition

Three different hand postures are defined and recognized, “Thumb up”, “Index up” and “Other”. At each time t each detected hand model (see Fig.3) is classified against one of the posture classes by matching the feature values to the posture models shown in Table 1 following a simple best-fit classification scheme.

4.6 Hand gesture recognition

The employed gestural vocabulary is composed of 5 different classes of physical hand-arm actions, as shown in Fig.1. Below, we report on how each of these five gestures are recognized.

“Yes”: The case of the “Yes” gesture is recognized if the posture performed by a single tracked hand is classified as a “Thumb up” posture for a number F_y of consecutive frames. We set $F_y = 10$ in our experiments.

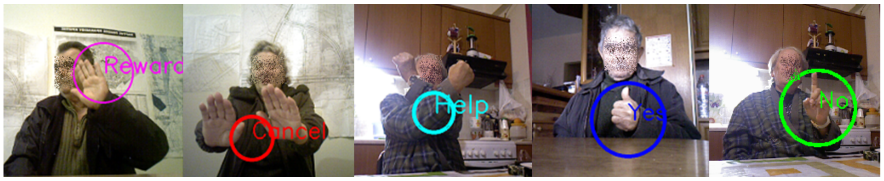


Fig. 4. Sample snapshots from tests with elderly users.

"No": For the "No" gesture it is necessary to detect a waving motion of the tip of the index finger. Therefore, an "Index up" posture should be recognized for a number of consecutive frames. Additional constraints are then applied to the 3D trajectory of the finger tip for those frames. More specifically, we compute its projection onto the x-axis of the image coordinate system. We apply smoothing and assess the consecutive extremity values of the signal which indicate the start and end points of the movement. The "No" gesture is recognized if there is a minimum $R_n = 4$ repetitions of a sideways motion of the hand. The width of each sideways motion has to be at least 30 mm. By setting this minimum quite low, the recognition of the "No" gesture is possible even if only the index finger moves, while the palm remains practically static.

"Stop/Cancel": The "Stop/Cancel" gesture regards the physical movement of both hands moving simultaneously towards the camera with open palms, as in Fig.1(c). The 3D coordinates of the palm centers of both hands are considered. They have to be at a similar distance from the camera at the beginning of the motion, and their trajectories should be mainly towards the camera plane, i.e., their distances to the camera plane must be strictly decreasing over more than 100 mm. During the whole gesture, the depth difference of the two palm centers has to be less than 100 mm.

"Reward": The "Reward" gesture is realized using each of the hands to perform a circular motion with open palm facing the camera. The 3D coordinates of the center of the performing hand are orthogonally projected onto the 2D camera plane. Subsequently, an ellipse is fitted based on the induced 2D coordinates and their angle is assessed with respect to the center of the ellipse. If the angle is continuously increasing (or decreasing) over more than 360 degrees, a "Reward" gesture is triggered.

"Help": The "Help" gesture is triggered upon successful detection of a pair of hands. The absolute value of the angle formed by the two wrist directions has to be in the interval $[\pi/2 \pm \pi/4]$. The line joining the two hand centers must be roughly parallel to the horizon (orientation less than $\pi/4$) and the intersection point of the wrist lines has to be below the centers. If these conditions are satisfied for more than $F_h = 10$ consecutive frames, the gesture is validated.

Gestures	Yes	No	Reward	Stop/Cancel	Help	Unknown
Yes	32/27	-	-	-	-	2/5
No	-	24/29	-	-	-	2/2
Reward	-	-	20/26	-	-	3/1
Stop/Cancel	-	-	-	30/25	-	2/-
Help	-	-	-	-	20/19	-/5
Unknown	1/3	-	3/-	1/-	3/-	46/10

Table 2. Confusion matrix of the conducted tests for both groups (experts/elderly). Actual class or ground truth appears in rows and the predicted class in columns.

5 Experimental Evaluation

To evaluate the proposed gesture recognition methodology two sets of user groups were identified that differ with respect to their age and to the their familiarity with technology. Intentionally, no member of the test group belongs to the group of subjects that participated in the definition of the gestural vocabulary.

Five persons from an academic/research environment participate in the first, experts group. A single example demonstration per gesture was performed to each subject explaining how to perform the gesture. A total of 189 gestures were performed by all subjects in the group, while 54 of them were unknown random gestures or intentionally performed unsupported gestures.

The second group of subjects consists of eight persons between 60-85 years old with practically no previous experience in technology (see Fig.4). Each gesture was demonstrated a few times. Each of the five gestures were performed at least 3 times by each subject. In total, 156 gestures were performed by the second group and 13 out of them were irrelevant, random movements.

Regardless of the group, each subject was recorded in a single video where he performed the gestures at random order and without interruption. The 54+13=67 random movements or unsupported gestures that were intentionally or unintentionally performed by the test subjects were assigned to the "unknown" class. The lack of response of our system to any of these un-modeled gestures was considered as a successful classification in the "unknown" class. Thus, the performance of our method is assessed in the presence of noise and irrelevant actions.

Table 2 shows the confusion matrices for the classification experiments for the two sets of users. Moreover, Table 3 reports the standard measures of statistical analysis for gesture classification. For the group of experts, and excluding the "unknown" class, the precision, recall and F-measure metrics were never below 0.87. For the group of elderly, the minimum scores were 0.90, 0.792 and 0.844, respectively. The group of experts scored the lowest F-measure value for the "Reward" gesture. The qualitative analysis of the recordings showed that this happened because of the high speed of execution of the related circular hand motion. For the group of elderly, the lowest F-measure score appears at the "Help" gesture. This is because, for the elderly people, this appears still to be a hard/complex gesture, given the mobility constraints of some of the sub-

	Group 1 - Experts			*Group 2 - Elderly*		
Gestures	Precision	Recall	F1 score	Precision	Recall	F1 score
Yes	0.970	0.941	0.955	0.900	0.844	0.871
No	**1.000**	0.923	**0.960**	**1.000**	0.935	**0.967**
Reward	0.870	0.870	0.870	**1.000**	0.839	0.912
Stop/Cancel	0.968	**0.938**	0.952	0.926	**1.000**	0.962
Help	0.870	**1.000**	0.930	0.905	**0.792**	0.844
Unknown	**0.836**	0.852	**0.844**	**0.435**	0.796	0.556
TOTAL	**0.919**	0.921	**0.919**	**0.861**	0.863	**0.852**

Table 3. Quantitative results of the proposed method. Precision, Recall and F- measure are reported for both test groups with respect to the set of the supported gestures.

jects. The qualitative analysis of the recordings showed that most of the elderly subjects performed this gesture by touching their arms on their torso, so depth discontinuities were not adequately estimated. The highest number of false positives and negatives was obtained for the "thumb up" gesture and for the group of the elderly. The requirement for a visible wrist-elbow during the execution of the gesture deteriorates the classification results for this group, as several subjects did not recall the relevant instructions while performing the gesture.

Overall, as suggested by the results shown in Table 3, the relatively small difference in the performance of the two groups suggests that the proposed approach is intuitive and can cover successfully the needs of a wide range of users.

Sample experimental results are illustrated in http://youtu.be/eIIzgjG2V7A.

6 Summary

A method for gesture recognition is proposed based on the effective detection of arms, hands and fingers, their tracking and the interpretation of their activities. The proposed method aims at supporting HRI and handles multiple hands that may dynamically enter and exit the field of view of an RGBD sensor. A set of 5 gestures have been identified from potential users. Despite being few, their recognition exhibits several challenges related to the mix of static and dynamic components, the broad scale of observations, their intrinsic ambiguities, the variability of the test subjects, the need to recognize gestures in the context of unknown actions and the requirement for robust performance under illumination changes, camera motion and scene clutter. The quantitative and qualitative assessment of the proposed methodology led to promising results that substantiate the effectiveness of the proposed approach.

Acknowledgements

This research has been partially supported from the EU FP7 project HOBBIT under grant agreement No. 288146.

References

1. Zabulis, X., Baltzakis, H., Argyros, A.: Vision-based hand gesture recognition for human-computer interaction. The Universal Access Handbook. LEA (2009)
2. Mitra, S., Acharya, T.: Gesture recognition: A survey. IEEE Trans. on Systems, Man, and Cybernetics **37** (2007) 311–324
3. Aggarwal, J., Ryoo, M.: Human activity analysis: A review. ACM Comput. Surv. **43** (2011) 16:1–16:43
4. Bowden, R., Zisserman, A., Kadir, T., Brady, M.: Vision based interpretation of natural sign languages. In: ICVS, ACM Press (2003)
5. Poppe, R.: A survey on vision-based human action recognition. Image and Vision Computing **28** (2010) 976 – 990
6. Moeslund, T., Hilton, A., Krüger, V., Sigal, L.: Visual Analysis of Humans: Looking at People. SpringerLink : Bücher. Springer (2011)
7. Erol, A., Bebis, G., Nicolescu, M., Boyle, R., X.Twombly: Vision-based hand pose estimation: A review. Computer Vision and Image Understanding, Special Issue on Vision for HCI **108** (2007) 52 – 73
8. Wu, Y., Huang, T.S.: Vision-based gesture recognition: A review. In: Gesture-based communication in human-computer interaction. Springer (1999)
9. Yang, M.H., Ahuja, N.: Extraction and classification of visual motion patterns for hand gesture recognition. In: IEEE CVPR. (1998)
10. Bretzner, L., Laptev, I., Lindeberg, T.: Hand gesture recognition using multi-scale colour features, hierarchical models and particle filtering. In: IEEE Automatic Face and Gesture Recognition. (2002)
11. Ramamoorthy, A., Vaswani, N., Chaudhury, S., Banerjee, S.: Recognition of dynamic hand gestures. Pattern Recognition (2003)
12. Yoon, H.S., Soh, J., Bae, Y.J., Yang, H.S.: Hand gesture recognition using combined features of location, angle and velocity. Pattern Recognition (2001)
13. Jo, K.H., Kuno, Y., Shirai, Y.: Manipulative hand gesture recognition using task knowledge for human computer interaction. In: IEEE International Conference on Automatic Face and Gesture Recognition. (1998)
14. Baraldi, L., Paci, F., Serra, G., Benini, L., Cucchiara, R.: Gesture recognition in ego-centric videos using dense trajectories and hand segmentation. In: IEEE CVPR Workshops. (2014)
15. Raptis, M., Kirovski, D., Hoppe, H.: Real-time classification of dance gestures from skeleton animation. In: Proceedings of the 2011 ACM SIGGRAPH/Eurographics. SCA '11 (2011)
16. Fothergill, S., Mentis, H., Kohli, P., Nowozin, S.: Instructing people for training gestural interactive systems. In: SIGCHI Conference on Human Factors in Computing Systems. CHI '12 (2012)
17. Yao, A., Van Gool, L., Kohli, P.: Gesture recognition portfolios for personalization. IEEE CVPR (2014)
18. Zhang, C., Hamid, R., Zhang, Z.: Taylor expansion based classifier adaptation: Application to person detection. In: IEEE CVPR. (2008)
19. Gonzalez, R., Woods, R.: Digital Image Processing. 2nd edn. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA (2001)
20. Eppstein, D.: Spanning trees and spanners. In Sack, J.R., Urrutia, J., eds.: Handbook of Computational Geometry. Elsevier (2000) 425–461
21. Argyros, A., Lourakis, M.: Real time tracking of multiple skin-colored objects with a possibly moving camera. In: IEEE ECCV. (2004) 368–379