Human-Robot Interaction Based on Gestures for Service Robots

Patrick de Sousa[1](✉), Tiago Esteves[2], Daniel Campos[2], Fábio Duarte[2], Joana Santos[2], João Leão[2], José Xavier[2], Luís de Matos[2], Manuel Camarneiro[2], Marcelo Penas[2], Maria Miranda[2], Ricardo Silva[2], António J.R. Neves[3], and Luís Teixeira[4]

[1] Faculty of Engineering, University of Porto, Porto, Portugal
ee11183@fe.up.pt
[2] Follow Inspiration, Fundão, Portugal
[3] IEETA/DETI, University of Aveiro, Aveiro, Portugal
an@ua.pt
[4] INESC TEC, Faculty of Engineering, University of Porto, Porto, Portugal
luisft@fe.up.pt

Abstract. Gesture recognition is very important for Human-Robot Interfaces. In this paper, we present a novel depth based method for gesture recognition to improve the interaction of a service robot autonomous shopping cart, mostly used by reduced mobility people. In the proposed solution, the identification of the user is already implemented by the software present on the robot where a bounding box focusing on the user is extracted. Based on the analysis of the depth histogram, the distance from the user to the robot is calculated and the user is segmented using from the background. Then, a region growing algorithm is applied to delete all other objects in the image. We apply again a threshold technique to the original image, to obtain all the objects in front of the user. Intercepting the threshold based segmentation result with the region growing resulting image, we obtain candidate objects to be arms of the user. By applying a labelling algorithm to obtain each object individually, a Principal Component Analysis is computed to each one to obtain its center and orientation. Using that information, we intercept the silhouette of the arm with a line obtaining the upper point of the interception which indicates the hand position. A Kalman filter is then applied to track the hand and based on state machines to describe gestures (`Start`, `Stop`, `Pause`) we perform gesture recognition. We tested the proposed approach in a real case scenario with different users and we obtained an accuracy around 89,7%.

1 Introduction

Nowadays, with robots entering in our daily lives, it is becoming important to provide the users a simple and intuitive way to interact with them. Human-Robot interactions has already proved to be a major field in robotics with an

J.M.R.S. Tavares and R.M. Natal Jorge (eds.), *VipIMAGE 2017*,
Lecture Notes in Computational Vision and Biomechanics 27,
DOI 10.1007/978-3-319-68195-5_76

increasingly investing in more rich and innovative kinds of interaction. Most of those interactions are based on verbal communication, enabling robots to identify voice commands or based on non-verbal interaction composed by gestures, face, eyes and body motion recognition [1].

Gestures can be divided into two types accordingly with their movement along time: static or dynamic. Static gestures does not change with time, they are described by the pose/posture in a single instant. Dynamic gestures changes the posture across time and the gestures are described by its movement [14].

In order to perform the gesture recognition we need to acquire the data from the user and for that there are usually two types of approaches: inertial sensor-based or vision based. Inertial sensor-based approaches are intrusive for the user and does not allow a very natural interaction [9]. Vision-based solutions are user independent and have emerged to give a better experience where RGB cameras were the first used to acquire data but they restrict the information to a 2D plane. Stereo vision, Time of flight (ToF) and Structured Light cameras like the Kinect were able to obtain a 3D space since they also obtain the depth information.

The vision-based approach, on the opposite the inertial sensor-based approach, gives hand features by performing hand/arm segmentation and extracting the desired features from it to recognize the gestures.

In this work, we propose a gesture recognition method to be used in a service robot. This service robot is targeted to help people with reduced mobility on the shopping process. It is used in dynamic and crowded environments, like supermarkets, and it has to be simple and intuitive to the user. We expect that our approach allows to control basic behaviours by simple gestures. The identification of the user is already implemented by the robot's current software where a bounding box focusing on the user (x, y, $width$ and $height$) is calculated. Due to the non-controlled environment and the possible physical limitations of the user, it is not appropriate for the system to have an initialization phase each time a user appears in the field of view and it is not possible to assume that the closest object near the robot are the user's hands.

In Sect. 2 relevant approaches for gesture recognition from another author are presented. In Sect. 3 our approach for gesture recognition and its implementation is explained. The accuracy of our system is showed and discussed in the Sect. 4.

2 Related Work

In order to perform hand/arm segmentation, the most popular method is to do a segmentation based on the skin-color. Argyros et al. proposed a method for detecting skin-colored objects using a Bayesian classifier with a small set of training with an on-line adaptation of skin-color probabilities to cope with illumination changes [3]. These kind of approaches are efficient but they have the problem that the user can not wear any kind of gloves and it should not appear skin colored objects in the background.

A common method used in the cameras with depth information is to do a simple segmentation applying a threshold based on the distance. The distance considered can be regarding to another part of the user. Cerlinca et al. used the head's distance to the sensor as a threshold to obtain the hands, assuming that the hands are always in front of the head and are the closest object to the sensor [6]. Sometimes both skin-color and depth information are used. Chen used a region growing technique with the seed on the estimated center of the hand based on the previous frame, the first position of the hand was obtained by a initialization to detect the hand [7]. Bergh et al. used a ToF and a RGB camera, the face was detected and the distance from it to the camera was measured. Based on this distance, a threshold was applied to the depth image to discard background objects. The remaining pixels, together with skin color detection, were used to detect the hands [5]. Park et al. proposed a different approach were the hands were detected by using motion clusters and predefined wave motion [12]. With the emergence of skeleton tracking algorithms like OpenNI with NITE [11] and Kinect SDK [10], it was possible to obtain the skeleton of the user with the information of the most important joints including arms and hands. Bellmore et al. used NITE to obtain the pose of the observer to interact with an interactive display. This approach requires a calibration pose to initialize body tracking [4].

After performing the segmentation of the hand/arm and obtained its desired features, several methods to track the hand are used. Park used a Kalman filter to continuously track the hand's location [12]. Mean-shift algorithms are also used. Chen used mean-shift to track the hand identifying the center of the palm [7].

In order to identify the gesture over time, some classifiers algorithms are used. Hidden Markov Models (HMM) are good for dynamic gestures who vary across the time. Yang et al. applied an HMM to identify eight gestures to control a music application in to adjust the volume and change the music [15]. For simple and easy model gestures, finite state machines can be applied. Ramey et al. used a finite state machine to classify a simple gesture of waving hand varying the x coordinate to left and right, in order to integrate with a social robot [13].

3 System Overview

Our proposed approach is divided in 4 main phases (Fig. 1). In the first, the acquisition data from the robot is performed. The current robot's software captures the depth and RGB images and computes the user's position. In the segmentation phase, the user is extracted from the background and then the arms are segmented. In the arm pose estimation phase (features extraction), the position and orientation of the hand is obtained. Given these data, in the Tracking

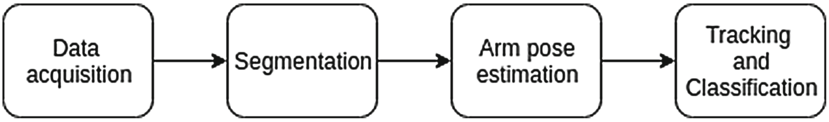


Fig. 1. System overview of the proposed solution.

and Classification phase a Kalman filter is applied to track and filter the hand position so we can identify the gesture in the classification part.

3.1 Gestures Parameterization

In order to start our Human-Robot Interface, it was important to discuss what type of gestures would be best suitable for the target end-user. Since the main target is people with reduced mobility, it is important to minimize the constraints on their use, due to possible physical limitations. Since the user will not receive any training to operate it, the gestures have to be natural and simple so that he can learn them and do not forget it until the next utilization. Given those facts, we reached to the conclusion that the gesture should be performed by one single arm due to people using mobility aids. Besides that, due to possible low sensitivity in the movements of hands it was better to get the arm position for the gesture instead of the hand pose.

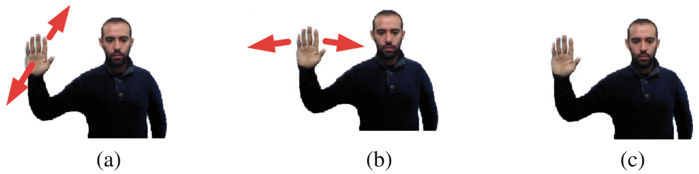


Fig. 2. Gestures parameterization: (a) `Start` gesture where the hand movement is forward and backward towards the robot. (b) `Pause` gesture formed by the lateral movement of the hand. (c) `Stop` Gesture where the hand does not change position.

The chosen gestures, `Start` (where the robot correctly initiate its process), `Stop` (shuts down the robot processes) and `Pause` (puts the robot in pause mode but still working), are represented in Fig. 2. For those gestures the only information necessary are the coordinates of the the hand (x, y, z) and the arm's orientation.

3.2 Data Acquisition

To acquire the data mentioned previously, a depth based solution using a RGB-D camera was developed. The identification of the user is already implemented by the software present on the robot where a bounding box focusing on the user (x, y, $width$ and $height$) is extracted. It allow us to focus on the user and reduce the noise of the scene, removing other people or objects standing next to him. Face detection is also implemented on the robot and since gestures have to be made facing the robot, this is information is used. The face detection result also allows us to restrict the gestures to a certain area. The data acquired from the robot is presented in the Fig. 3.

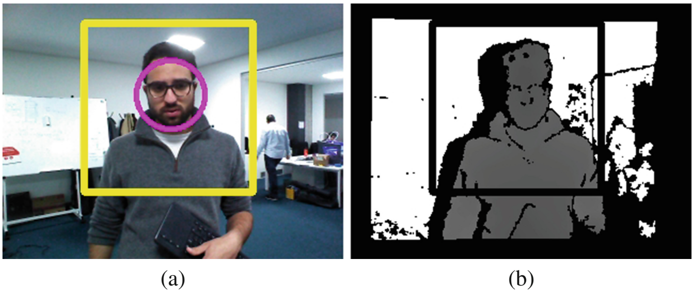


Fig. 3. Data acquisition: (a) RGB image with user detection (yellow rectangle) and face detection (pink circle). (b) Depth image with user detection.

3.3 Segmentation

Since the system will be used in a non-controlled environment it is necessary to ensure that only the user will interact with the robot. As we can observe in Fig. 4 the person next to the user is appearing in the image interfering in the segmentation result.

To separate the user from the background, we apply an histogram approach to find out the distance from the robot to the user's chest. For this, we consider the location information given by the robot and taking into consideration the user area we find the mode value since it will occupy most of that area. After that, we apply a threshold, where the values higher than the threshold value are turned to zero. With the threshold value calculated by the histogram approach we add 15 cm to the calculated threshold to ensure that we segment the user totally (Fig. 5(b)). Finally, we apply a morphological close operation to the segmentation

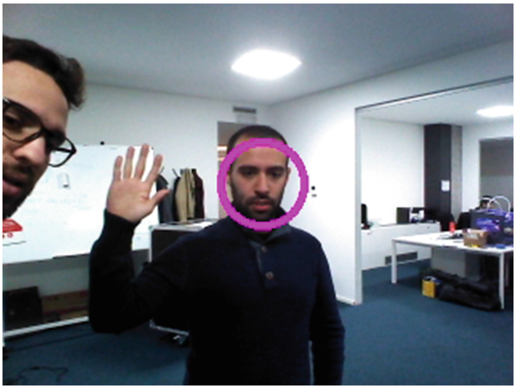


Fig. 4. RGB image received from the robot with the face detection of the user.

result to reduce the noise. In order to focus only on the user, we applied a region growing algorithm [8] with the seed in the center of the user (based on the bounding box sent by the robot). This gives us a binary image (Fig. 5(d)) with the user segmented.

In order to isolate the arms from the body another image was obtained by applying another threshold to the original image. Using the previous value calculated from the histogram it is applied a threshold of this value subtracting 4 cm to obtain an image only with the objects in front of the user (Fig. 5(e)). Then we apply an interception between the image after the close operation (Fig. 5(c)) with the region growing mask (Fig. 5(d)) and the image of the threshold ahead

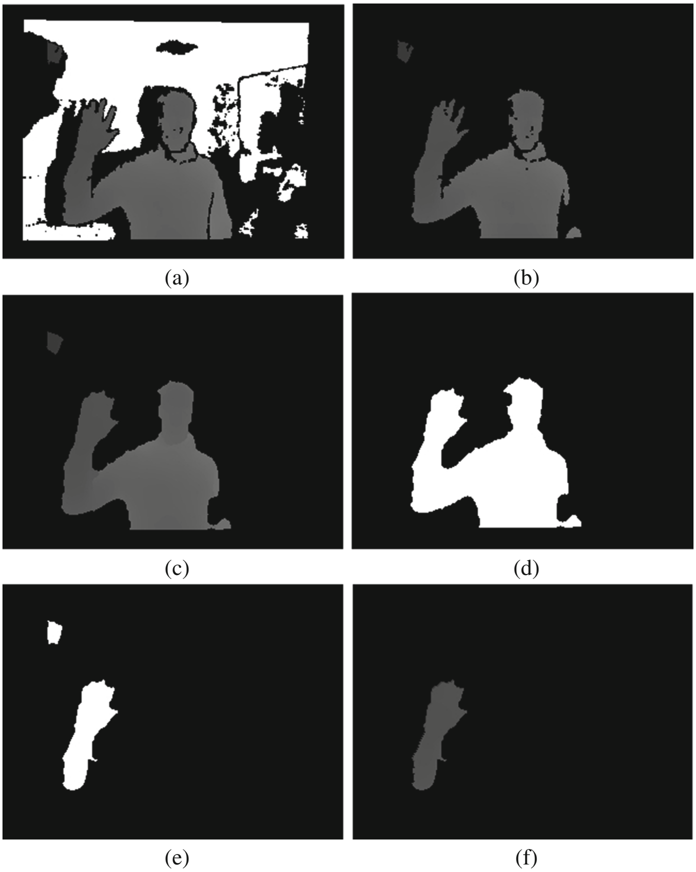


Fig. 5. User image segmentation: (a) Depth image received from the robot. (b) Depth image after a threshold to perform the background segmentation. (c) Morphological close operation on the threshold image. (d) Region growing result with the seed on the center of the user. (e) Original depth image after another threshold to extract what is in front of the user. (f) Interception of the images c, d and e resulting in the parts of the user in front of him.

of the user (Fig. 5(e)) to obtain only the regions of the user near the camera that we assume as a possible arm of the user.

3.4 Identification and Validation

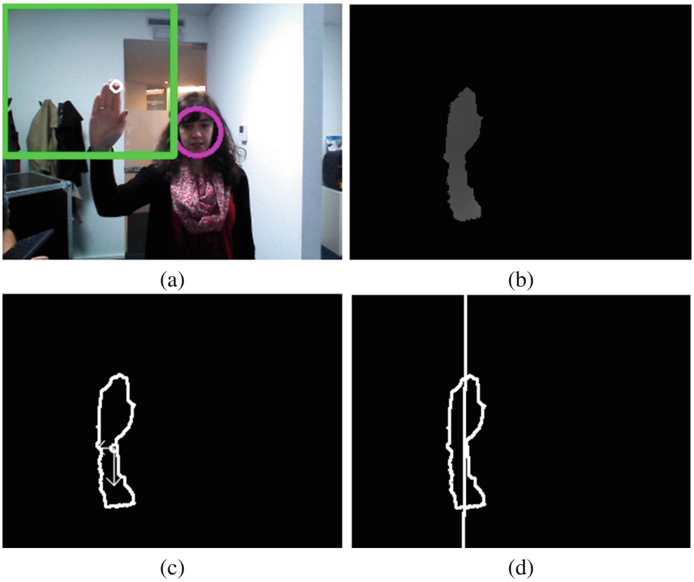
After performing the segmentation which retrieves the most important regions on the image, it is necessary to separate them as distinct objects and find the position of the hands in order to detect gestures. We start by labelling the components and then we perform a Principle Component Analysis (PCA) [2] to each labelled object, which allows the algorithm to understand for each object how its data is distributed across the image, retrieving its center of mass and eigen vectors. The orientation of eigen vectors is considered to draw a line which passes through the center of mass and intersects the silhouette of the object in two different points. We also normalize the orientation's angle to be sure it is pointing to the upper part of the image, assuming that the gesture has to be made with the hand at the top of the arm. Thus we can guarantee that the tip of the hand will be the upper intersection point of the line with the object contour.

Gestures presented in Sect. 3.1 are validated using the face position obtained in Sect. 3.2, considering only those which are made on the lateral parts of the face and above the chain, given by the bottom part of the rectangle which defines user's face.

3.5 Tracking and Classification

Given the hand tip's position, it is necessary to track the hand's position so that we can do the gesture's classification. For the tracking algorithm, a Kalman filter was implemented to track the hand coordinates and the arms orientation. The Kalman filter is an efficient filter that estimates the state of a dynamic system from a series of incomplete and noisy measurements. The main goal of the Kalman filter is to estimate the state of a system from measurements which contain noise and its previous state [12]. We applied the Kalman filter considering a constant position model making the assumption that the next state is defined by the previous state.

In order to identify which gesture was made by the user, a simple state machine for each gesture was implemented. Those state machines were designed according to the movement of the proposed gestures. Since the hand moves along different coordinate axis (along the z axis for the `Start` and along the x axis for the `Pause`) or do not even move (in case it is a `Stop`) the proposed approach was of simple implementation. An example of the gestures are presented on Figs. 7, 8 and 9.



(a) (b) (c) (d)

Fig. 6. Features extraction: (a) RGB image with the identification of the hand's tip inside the region of interest (green rectangle). (b) Segmentation of the hand and arm. (c) Silhouette of the hand and arm with the center of mass and the eigen vectors represented. (d) Line obtained by the center of mass and the longer eigen vector crossing the silhouette.

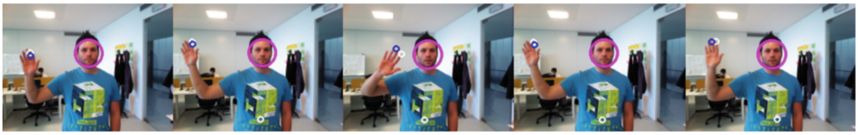


Fig. 7. Time-lapse of the `Pause` gesture performed over time.

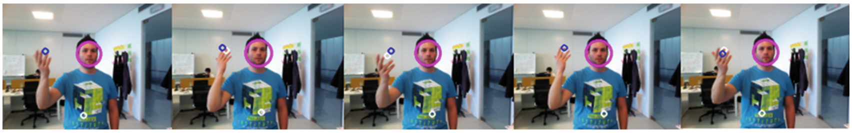


Fig. 8. Time-lapse of the `Start` gesture performed across time.

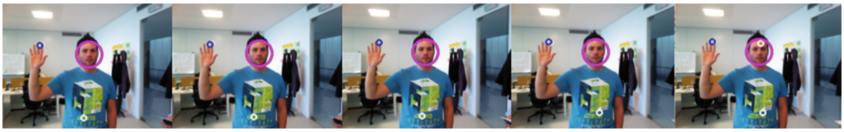


Fig. 9. Time-lapse of the `Stop` gesture performed across time.

4 Experimental Results

To evaluate the efficiency of our method, 13 volunteers were asked to perform the gestures. After explaining how to perform the gestures, the volunteers performed them 3 times alternating between gestures.

Table 1. Correct rate of identification for each gesture.

Gesture	Accuracy(%)
Pause	84,6
Stop	97,4
Start	87,2
Average	89,7

As presented on the Table 1, for each gesture performed it was achieved a correct rate of 84,6% for the "Pause", 97,4% for the "Stop" and 87,2% for the "Start", with a global accuracy of 89,7%. We finished the method evaluation by asking the users to perform random gestures in order to check if we get false positives. We got one of the implemented gestures 17,1% of the times. Nevertheless, 10% of that value corresponds to "Pause" gestures where the users single move the hand to the side several times just like this gesture is made. In this case the random gesture was actually the "Pause" gesture.

5 Conclusions

We proposed a new approach for the recognition of hand gestures to be used in Human-Robot Interfaces. Our approach was designed to recognize three gestures: "Start", "Stop" and "Pause". The gestures are identified based on implemented state machines that recognize specific features in the hand movement of each gesture. The hand movement is obtained using a Kalman filter and considering the hand segmentation from an RGB-D camera. The proposed approach was tested with several persons in a real case scenario, where a robot already existent in the market was used and controlled by the proposed user gestures. A global accuracy of 89,7% was achieved which indicates the robustness of our proposed approach. As part of the ongoing work, it can be interesting to implement a probabilistic classification method like the Hidden Markov models in order to improve the accuracy of our system as it will allow us to add complex gestures to the Human-Robot interface.

Acknowledgements. This work is supported in part by the Faculty of Engineering of University of Porto in collaboration with the company Follow Inspiration S.A.

References

1. Verbal and Non-verbal Communication, pp. 223–235. Springer, Dordrecht (1991)
2. Abdi, H., Williams, L.J.: Principal component analysis. Wiley Interdisc. Rev. Comput. Stat. **2**(4), 433–459 (2010). doi:10.1002/wics.101
3. Argyros, A.A., Lourakis, M.I.A.: Real-time tracking of multiple skin-colored objects with a possibly moving camera, pp. 368–379. Springer, Heidelberg (2004)
4. Bellmore, C., Ptucha, R., Savakis, A.: Interactive display using depth and RGB sensors for face and gesture control. In: 2011 Western New York Image Processing Workshop, pp. 1–4 (2011). doi:10.1109/WNYIPW.2011.6122883
5. den Bergh, M.V., Gool, L.V.: Combining RGB and ToF cameras for real-time 3D hand gesture interaction. In: 2011 IEEE Workshop on Applications of Computer Vision (WACV), pp. 66–72 (2011). doi:10.1109/WACV.2011.5711485
6. Cerlinca, T.I., Pentiuc, S.G.: Robust 3D hand detection for gestures recognition, pp. 259–264. Springer, Heidelberg (2012)
7. Chen, C.P., Chen, Y.T., Lee, P.H., Tsai, Y.P., Lei, S.: Real-time hand tracking on depth images. In: 2011 Visual Communications and Image Processing (VCIP), pp. 1–4 (2011). doi:10.1109/VCIP.2011.6115983
8. Gonzalez, R.C., Woods, R.E.: Digital Image Processing, 3rd edn. Prentice-Hall Inc., Upper Saddle River (2006)
9. Jambhulkar, K.R.: Review on sensor based hand gesture recognition system. Int. J. Res. Eng. Adv. Technol. **5**(1), 33–36 (2017)
10. Microsoft: Kinect. https://developer.microsoft.com/pt-pt/windows/kinect. Accessed April 2017
11. Openni: Nite. http://openni.ru/files/nite/. Accessed April 2017
12. Park, S., Yu, S., Kim, J., Kim, S., Lee, S.: 3D hand tracking using Kalman filter in depth space. EURASIP J. Adv. Signal Process. **2012**(1), 36 (2012)
13. Ramey, A., Gonzalez-Pacheco, V., Salichs, M.A.: Integration of a low-cost RGB-D sensor in a social robot for gesture recognition. In: 2011 6th ACM/IEEE International Conference on Human-Robot Interaction (HRI), pp. 229–230 (2011). doi:10.1145/1957656.1957745
14. Rautaray, S.S., Agrawal, A.: Vision based hand gesture recognition for human computer interaction: a survey. Artif. Intell. Rev. **43**(1), 1–54 (2015)
15. Yang, C., Jang, Y., Beh, J., Han, D., Ko, H.: Gesture recognition using depth-based hand tracking for contactless controller application. In: 2012 IEEE International Conference on Consumer Electronics (ICCE), pp. 297–298 (2012). doi:10.1109/ICCE.2012.6161876