Integration of a Low-Cost RGB-D Sensor in a Social Robot for Gesture Recognition

Arnaud Ramey
RobotiscLab, Univ. Carlos III of Madrid
Leganés, Spain
arnaud.ramey@m4x.org

Victor Gonzalez-Pacheco
RobotiscLab, Univ. Carlos III of Madrid
Leganés, Spain
vgonzale@ing.uc3m.es

Miguel A. Salichs
RobotiscLab, Univ. Carlos III of Madrid
Leganés, Spain
salichs@ing.uc3m.es

ABSTRACT

An objective of natural Human-Robot Interaction (HRI) is to enable humans to communicate with robots in the same manner humans do between themselves. This includes the use of natural gestures to support and expand the information that is exchanged in the spoken language. To achieve that, robots need robust gesture recognition systems to detect the non-verbal information that is sent to them by the human gestures. Traditional gesture recognition systems highly depend on the light conditions and often require a training process before they can be used. We have integrated a low-cost commercial RGB-D (Red Green Blue - Depth) sensor in a social robot to allow it to recognise dynamic gestures by tracking a skeleton model of the subject and coding the temporal signature of the gestures in a FSM (Finite State Machine). The vision system is independent of low light conditions and does not require a training process.

Categories and Subject Descriptors

I.2.9 [**Robotics**]: Sensors; I.4.8 [**Scene analysis**]: Tracking

General Terms

Algorithms, Design, Experimentation

Keywords

Gesture Recognition, Finite State Machine (FSM), RGB-D Sensor

1. INTRODUCTION

Natural Human-Robot Interaction (HRI) includes non-verbal communication. One of its fields is gesture communication. In order to detect gestures performed by humans, robots use gesture recognition systems. Several works have been carried out to allow robots "seeing" their environment [1]. One of the most important factors of robot-embarked gesture recognition systems is the need of low-cost, robust sensors to acquire visual data of the environment. At the same time, the data provided by the sensor must be analysed and interpreted in real time by the robot. The classical approach to gesture recognition is based on 2D images analysis and requires high computation time or long training phases.

HRI'11, March 6–9, 2011, Lausanne, Switzerland.
ACM 978-1-4503-0561-7/11/03.

Using image data implies the application of complex statistical models for recognition, which are difficult to use in practical applications [3].

We use a low-cost RGB-D camera to track a human performing gestures by extracting and tracking a model of his skeleton. It enables the identification of dynamic hand gestures regardless the relative orientation of the subject to the camera. The gestures are modelled using a Finite State Machine (FSM) and interpreted by the social robot Maggie [2] as commands sent by the subject. Our FSM is similar to [4], but additionally, and thanks to the 3D skeleton model, we added a third dimension to the gesture space. Other works used a 3D skeleton model to track hand gestures with promising results [3] to control video games. Our approach mixed both works seizing the capabilities of the RGB-D sensor and integrated them in the robotic platform.

2. DESCRIPTION OF THE SYSTEM

The gesture recognition system is composed of several modules.The vision system of the robot is the Microsoft's Kinect time-of-flight camera, an RGB-D sensor that mixes standard RGB images with depth information provided by an Infra-Red (IR) sensor. Its price is between 5 and 10 times lower than usual time-of-flight cameras while its precision meets gesture recognition requirements. The sensor data can be accessed and controlled by the open-source openNI framework (http://openni.org/) via the NITE middleware (http://www.primesense.com/). The latter provides real-time tracking of the 3D skeleton model of the subject using the depth information of the sensor. We use this 3D skeleton model to extract the temporal signature of hand gestures using an FSM which codes the direction of the hand in different states. During the execution of a gesture, and while the hand is changing its trajectory, a stream of state changes is sent to a template-based classifier which decides which gesture is being performed by the human.

3. 3D GESTURE IDENTIFICATION

3.1 Extraction of the Body Features

The RGB-D sensor calculates the distance at which the body is located. With this information, the API is capable of detecting human shapes and build a skeleton model of the human who is being detected. Moreover, the API tracks the position and orientation parameters of all the joints of the skeleton model in real time. This information enables us to track down the hands of the subject in real time. The API provides the orientation of the body from the camera

reference coordinates. To monitor the gestures it is easier, however, to transform it to the subject coordinates system. This is done through computing the transformation matrix to the user coordinates using the position of both his shoulders, neck and head. This allows us to capture his gestures even if the human is not directly orientated to the sensor.

3.2 Movement Extraction and Representation

Once we have found the human reference system, we track the hand motion in the following way. We consider a vector V_e defined as $V_e = (x_e - x_s, y_e - y_s, z_e - z_s)$ where (x_e, y_e, z_e) and (x_s, y_s, z_s) are the coordinates of the end effector and the shoulder of the human respectively. While the gesture is being executed, V_e varies. We extract the velocity components $(\dot{x}, \dot{y}, \dot{z})$ of V_e by $M_{V_e} = (\frac{\delta \overline{x}}{\delta t}, \frac{\delta \overline{y}}{\delta t}, \frac{\delta \overline{z}}{\delta t})$. By comparing the components of M_{V_e}, we determine the instantaneous direction of the hand. If its velocity in this direction is above a given value, we consider a motion has just been performed in this direction.

We codify this motion as a state of an FSM. We have chosen the following states to codify the motion properties of the hand: Left (L) and Right (R) for the $\dot{x}$ component; Up (U) and Down (D) for $\dot{y}$; Back (B) and Forward (F) for $\dot{z}$; and Still (S) which means no motion in any of the components. Hence, a gesture is a finite specific sequence of states. Fig. 1 depicts an example of this FSM.

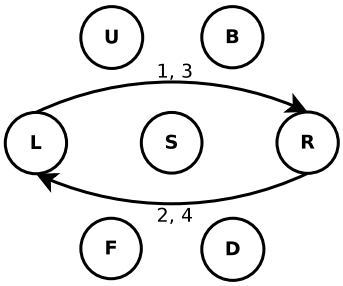

Figure 1: Example of the used FSM. The state represent *"hello"* gesture that involves waving the hand. The numbers indicate the state changing sequence.

3.3 Gesture Classification

Gesture classification is carried out by template matching. We define a pool of gestures which can be identified by the system using a search tree. Each node of the tree corresponds to one possible state of the FSM. While the gesture tracking system is feeding the classifier with a stream of states, the classifier navigates through the branches of tree. If the tracker sends a sequence of states that corresponds with an entire branch of the tree, the classifier will arrive to the leaf of this branch. When this happens, it triggers an *event* stored in this leaf and specific to the gesture which has been detected. *Events* are one of the key communication mechanisms of the software architecture of the robot [2]. One of their possible uses is to make the robot perform certain tasks. For example, if the vision system of the robot detects a gesture indicating it to come closer, the classifier will trigger an event which activates the robot movements in the direction of the user.

4. RESULTS AND CONCLUSIONS

We presented the integration of an RGB-D sensor in a social robotic platform allowing it to recognise dynamic hand gestures in 3D. Our system consists in the exploitation of the sensor capabilities of building 3D human skeleton models in real time. With the skeleton model we track the temporal variations of hand gestures and we model them in an FSM. The states of the FSM correspond to the 3D hand directions of the gesture. While the gesture is being performed by the human, the vision system, feeds a template-based classifier with a stream of the FSM's states. When the classifier detects the gesture, an event is sent notifying the robot. Our system does not require a training phase.

Our robot relies on a dual core 1,6 GHz micro-processor laptop. This binds us to use low CPU consumption algorithms, which led us to discard classical approaches to gesture recognition. The NITE middlewarewe provides an actual framerate of 25 FPS. Additionally, our recognition system is light enough to maintain this framerate and requires a reasonable CPU workload. Hence, it proves to be a real-time solution for gesture recognition in a modest platform.

Because of the IR depth sensor, the system is independent of low light conditions. Furthermore, by building the 3D skeleton of the subject, gestures can be executed regardless the orientation of the subject. The main condition is that the hand must be visible by the vision device during most of the gesture. Future work involves expanding the pool of states of the FSM in order to be able to recognise more gestures and mix them with other natural communication methods such as natural spoken language. Additionally, our gesture recognition system tracks the hand gestures, but since we use a 3D model of the skeleton, it is possible to virtually track any other part of the body. Future revisions of our system will point to that direction.

5. ACKNOWLEDGMENTS

The authors gratefully acknowledge the funds provided by the Spanish MICINN (Ministry of Science and Innovation) through the project "*A new Approach to Social Robotics (AROS)*".

6. REFERENCES

[1] S. Mitra and T. Acharya. Gesture Recognition: A Survey. *IEEE Transactions on Systems, Man and Cybernetics, Part C (Applications and Reviews)*, 37(3):311–324, May 2007.

[2] M. Salichs, R. Barber, A. Khamis, M. Malfaz, J. Gorostiza, R. Pacheco, R. Rivas, A. Corrales, E. Delgado, and D. Garcia. Maggie: A robotic platform for human-robot social interaction. In *2006 IEEE Conference on Robotics, Automation and Mechatronics*, pages 1–7, 2006.

[3] J. Varona, A. Jaume-i Capó, J. Gonzàlez, and F. J. Perales. Toward natural interaction through visual recognition of body gestures in real-time. *Interacting with Computers*, 21(1-2):3–10, Jan. 2009.

[4] M. Yeasin. Visual understanding of dynamic hand gestures. *Pattern Recognition*, 33(11):1805–1817, Nov. 2000.