Studies of AR Drone on Gesture Control

MA Lu, CHENG Lee Lung

Department of Electronic Engineering City University of Hong Kong

Hong Kong, China

luma26-c@my.cityu.edu.hk; itacheng@cityu.edu.hk

Keywords: gesture control; Kinect; AR Drone; quaternion; human-computer interaction; machine vision

Abstract. This paper focuses on the recent drone and gesture recognition technology, and we try to apply the gesture recognition technology to a drone platform. Thus, a drone system capable of gesture-controlled navigation is implemented. The major components comprising the system are Parrot AR. Drone [1], Kinect [2] and PC, while LabVIEW [3, 4] is chosen as the programming language and Kinect is utilized as the gesture recognition device. Besides, specific methods are discussed about how to properly design gestures easy for manipulating the AR. Drone, and how to generate the control signals from these gestures. As a result, the response of gesture-based flight control system is accurate and prompt, according to its decent performance in the video [5].

I. Introduction

Nowadays, Unmanned Aerial Vehicle (UAV), also referred to as Drone, has become a notable tool in lots of civilian applications, such as 3D building digitalization [6], area surveillance, and delivery service. Among all kinds of UAVs, quadcopter is currently one of the most popular UAVs, due to its low cost, and excellent stability.

At the same time, many researches on how to elevate the efficiency of interaction between human and machine [7, 8] have been made. In contrast to the conventional Human-Computer Interaction (HCI) method, such as mouse and keyboard, there is a trend that gesture-based interaction methods [9, 10, 11] are replacing the conventional methods in certain fields, like Virtual Reality (VR), Robotic, mobile devices.

Thus, it can be envisioned that gesture recognition technology could be incorporated into aerial vehicles to enable the user to control them more naturally and perform even more complicated operations than before.

In this paper the idea is realized by implementing the system in Fig. 1 which consists of Kinect, AR. Drone, and PC.

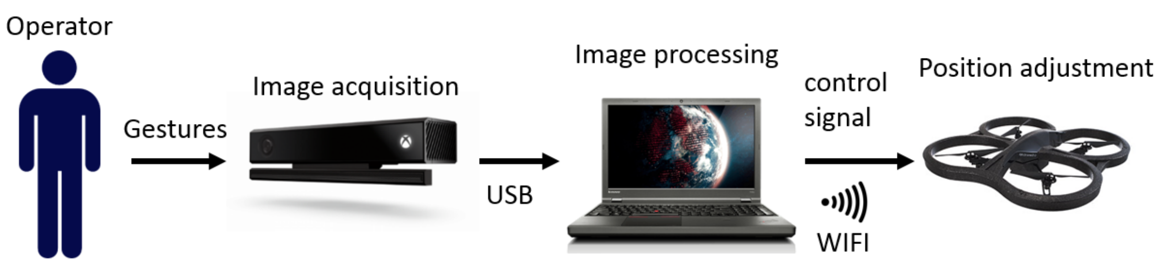
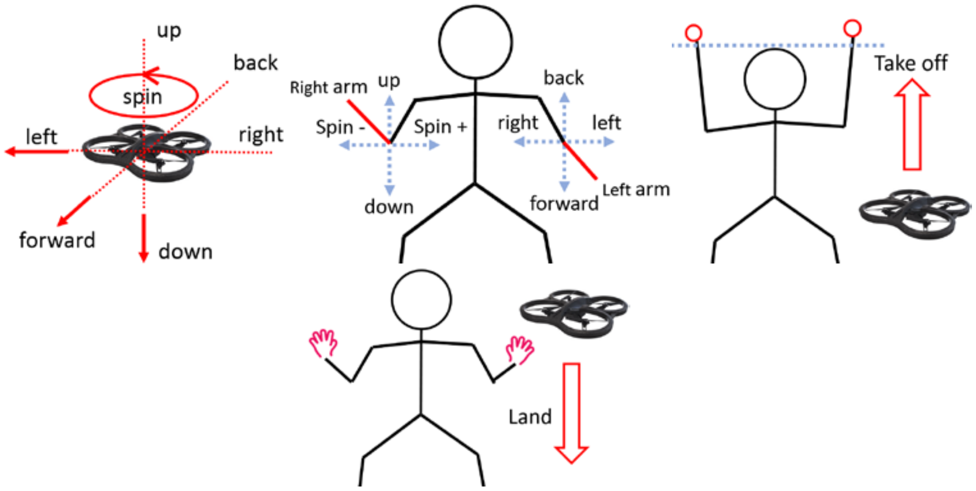


Fig. 1 the whole gesture control system

II. Method of Gesture Control

Gesture Design: The AR. Drone has four directions of movement (cf. Fig. 2 (a)). The gestures captured by Kinect are represented as skeletons as in Fig. 2 (b, c, d), and the building blocks of the skeletons are joints, such as head joint and wrist joint, which has its own coordinate position and orientation. Inspired by the joystick, we design the flight control gestures as in Fig. 2 (b). Pointing the arms to different directions will result in various projections on the blue dashed axes which can be used as signals for the control of each of the directions of drone. For example, you can order the drone to move up by bending your right arm upward.

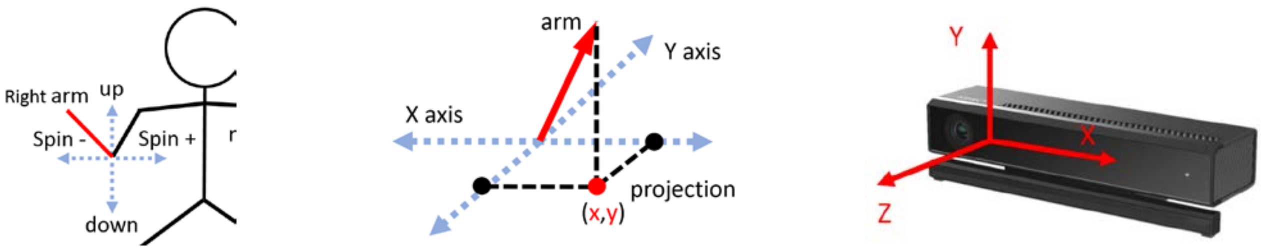


(a) directions (b) gestures of flight (c) gesture of taking off (d) gesture of landing

Fig. 2 the direction of drone and gestures

In addition, the gesture for taking off and landing are designed separately as in Fig. 2 (c) and (d). If both joints of hands are higher than the joint of head, the drone will take off. If the status of both hands are 'open', the drone will land.

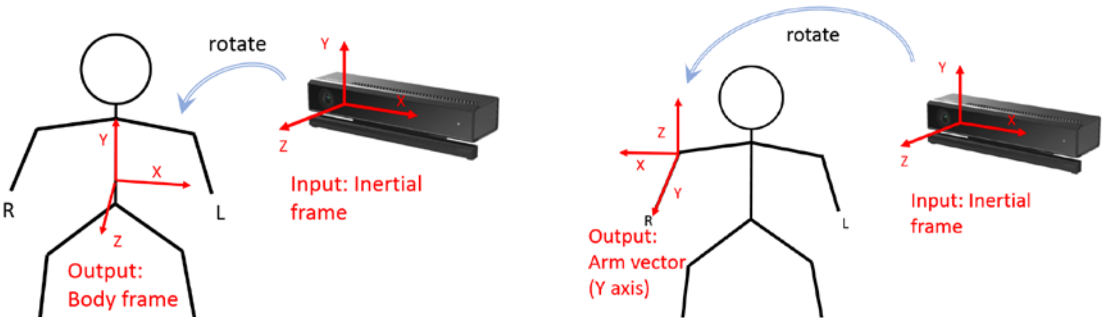
Generating Control Signal: As is mentioned above, the projection of arm vector on the body frame (blue dashed axes in Fig. 3 (b)) is used as the control signal. As opposed to the *inertial frame* (Fig. 3 (c)) fixed on Kinect itself, the *body frame* is fixed on the operator himself as in Fig. 3 (a).



(a) body frame (b) projection on body Frame (c) inertial frame

Fig. 3 projections of the arm vector

Both the skeletons and *body frame* are represented under the *inertial frame*. The data of the skeletons is provided by Kinect whereas the body frame is not initially given. Thus, the quaternion [12] is introduced to perform rotations of *inertial frame* in order to obtain the *body frame* as in Fig. 4 (a).



(a) body frame (b) arm vector

Fig. 4 rotation from inertial frame to body frame and arm vector

Because the body frame is fixed on the operator's body, the Z axis of the body frame always points to the front of operator's body with X axis always pointing left and Y axis pointing right, no matter which direction the operator is facing. Similarly, arm vector can also be obtained by rotation as in Fig. 4 (b).

According to [13], each quaternion represented as Eq. (1) corresponds to a particular rotation matrix as in Eq. (2) with which the rotation of vectors can be performed as Eq. (3) where v_I represents the vector being rotated and v_B represents the vector obtained after rotation.

Quaternion:

$$q_i^b = [a \quad b \quad c \quad d]^T \quad \text{Eq. (1)}$$

Rotation matrix:

$$R_i^b(q_i^b) = \begin{bmatrix} a^2+b^2-c^2-d^2 & 2bc-2ad & 2bd+2ac \\ 2bd+2ad & a^2-b^2+c^2-d^2 & 2cd-2ab \\ 2bd-2ac & 2cd+2ab & a^2-b^2-c^2+d^2 \end{bmatrix} \quad \text{Eq. (2)}$$

Rotated vector:

$$v_B = R_i^b(q_i^b)v_I \quad \text{Eq. (3)}$$

Now with the body frame obtained, we can calculate the projection by multiplying unit vectors of arm and *body frame* as Fig. 3 (b). With each arm contributing two variables, the control signal consists of 4 variables in total with each of them ∈ (-1, 1), and the negative value denotes the opposite direction. Finally, the whole process of producing the control signal can be summarized as Fig. 5.

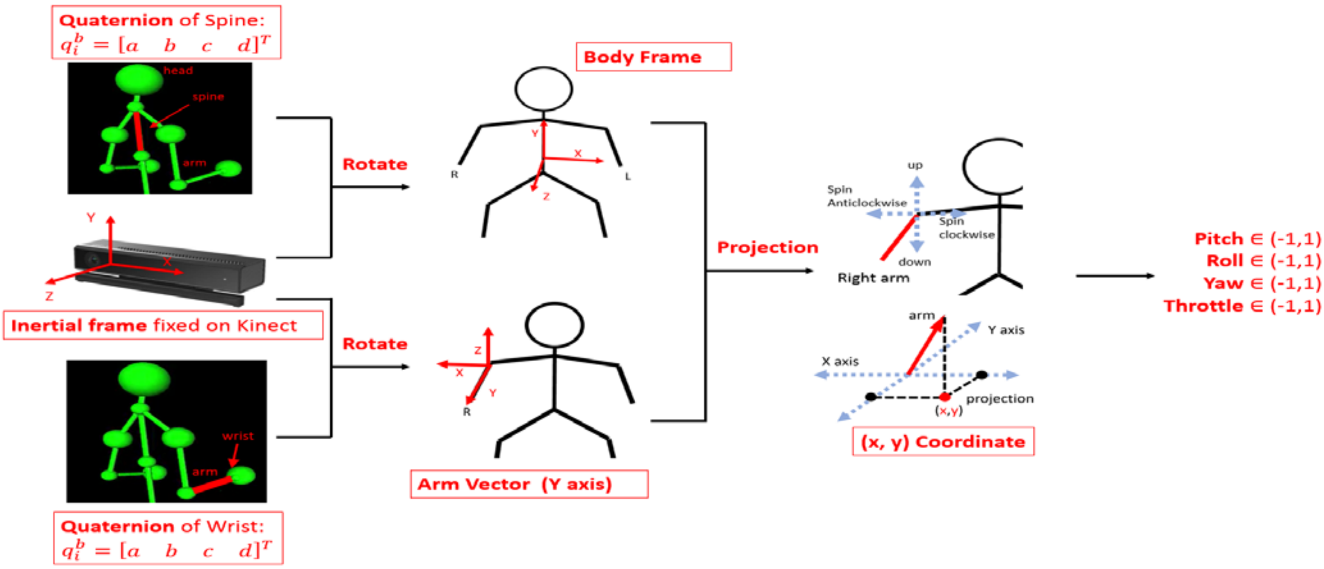


Fig. 5 the process of producing the control signal

III. Summary

In this paper, we introduce a method to apply the gesture recognition technology to the flight control system of AR. Drone.

To implement the gesture control system, proper gestures (Fig. 2) are designed for the user to easily control the four directions of drone at same time. Kinect is used as the gesture recognition device and LabVIEW is chosen as the programming language.

As a result, the response of the gesture-based flight control system is accurate, and prompt, according to its decent performance in the video [5].

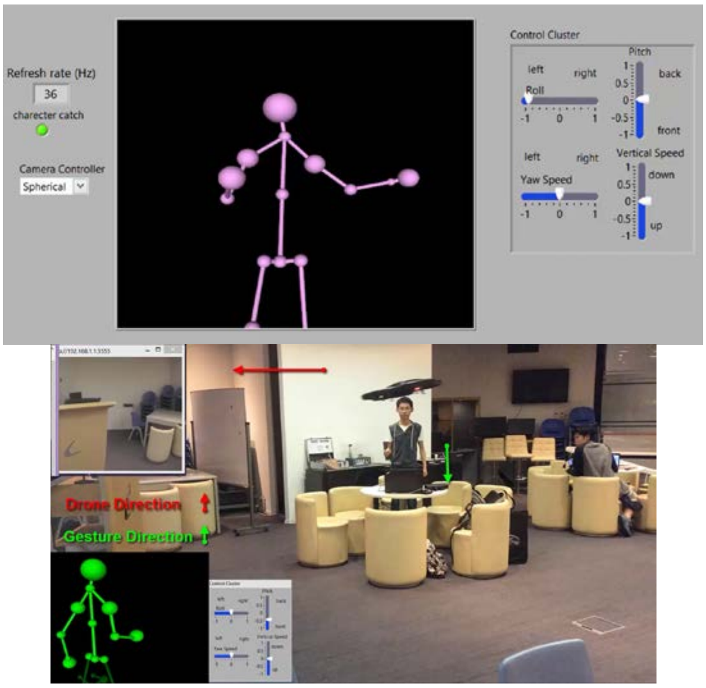


Fig. 6 screenshots of the demonstration video

IV. References

[1] WIKIPEDIA, "Design and development of AR.Drone," [Online]. Available: https://en.wikipedia.org/wiki/Parrot_AR.Drone.

[2] "About Kinect," [Online]. Available: https://en.wikipedia.org/wiki/Kinect.

[3] "LabVIEW," National Instrumentation, [Online]. Available: http://www.ni.com/labview/.

[4] H. LLC, "Haro User guide version 1.0.0 , 3D Visualization Library for LabVIEW™," 2014.

[5] M. Lu, "Video of Flight Demonstration," [Online]. Available: https://youtu.be/IO0FPjVEGpw.

[6] R.-M. Tauer, "3D building digitalization by DJI Phantom Vision+," [Online]. Available: https://www.youtube.com/watch?v=VaEKm4nV9TQ.

[7] A. Dix, Human Computer Interaction, Pearson Education, 2004.

[8] Rautaray, S. S. , Agrawal, A., "Vision based hand gesture recognition for human computer interaction: a survey," *Artificial Intelligence Review,* vol. 43, pp. 1-54, 01 2015.

[9] Sanna, A., Lamberti, F., Paravati, G.,, "A Kinect-Based Natural Interface for Quadrotor Control," in *Intelligent Technologies for Interactive Entertainment*, Springer Berlin Heidelberg, 2011, pp. 48-56.

[10] J. N. P. a. A. P. M. Pedro Neto, "Accelerometer-Based Control of an Industrial Robotic Arm".

[11] Gowing, M., Ahmadi, A., Destelle, F., "Kinect vs. Low-cost Inertial Sensing for Gesture Recognition," in *MultiMedia Modeling*, Springer International Publishing, 2014, pp. 484-495.

[12] J. Vince, "Quaternions in Space," in *Quaternions for Computer Graphics*, Springer London,

2011, pp. 89-129.
[13] "Understanding Quaternions," [Online]. Available: http://www.chrobotics.com/library/understanding-quaternions.