Virtual Flying Experience Contents Using Upper-Body Gesture Recognition

Jae-wan Park, Chi-min Oh, and Chil-woo Lee

Dept. of Electronics and Computer Engineering,
Chonnam National University, Gwang-ju, Korea
cyanlip@naver.com, sapeyes@gmail.com, leecw@jnu.ac.kr

Abstract. In this paper, we describe an algorithm and an interactive content using the idea to experience feeling of bird's flying by using gesture recognition of a user's upper body. In the algorithm we assume that gesture is composed of several key poses. So, in order to recognize the user's gesture, we firstly classify the user pose into the several predefined key poses and then analyze the sequence of the poses. In the key pose recognition procedure, the information of upper-body configuration is estimated by using joint locations of depth image from a Kinect camera. If the user performs a consecutive motion, the content recognizes the key poses and then synthesizes a gesture according to the order of the key poses. The stage of the content is consisted with three parts in order to enjoy the various flight experiences.

Keywords: Virtual Flight Experience, Gesture Interface, pose recognition.

1 Introduction

Body Gesture is one of the most important non-verbal communication methods for human being. Since it does not need direct contact and any physical channel, it has been used as a key technology for constructing multi-modal information system. However, it is very difficult to use gesture for real-time interaction system because human body is a very complicated object with many limbs and the gesture is composed of articulated motion with very high degree of freedom. In this paper, we present an algorithm and interactive content using that idea for animating bird's flying motion synchronized by user's gesture. Using the predefined gestures on our system, the users can vividly enjoy various feeling of bird's flying in real time.

Recently, KINECT camera is very popular in constructing interactive system such as game. It can easily segment moving region of human body by using depth information and produce sequential information of the joint locations of the user. However, even though the camera gives good information about a motion or human pose, it is another thing to recognize sequential static poses as a gesture. That means that gesture recognition needs some specific knowledge and procedures.

C. Stephanidis (Ed.): Posters, Part I, HCII 2013, CCIS 373, pp. 367–371, 2013.

In this research we assume that a gesture is composed of sequential specific poses, and we call it key poses. Therefore if we sequentially classify the poses captured with camera in correct order, we can recognize the meaning of the motion as the gesture. That implies if we are able to set a certain scenario which ruling the user's pose as we can meet in simple situation of interactive game, and then gesture recognition becomes more convenient and effective one to be used in various applications. In this paper, we suppose such simple situation of animating flying motion and gesture of upper-body would be applied to make a dynamic motion of a flying object.

2 The Plot of the Flying Animation

In the beginning of the animation, a user can see an avatar having a shape of human bird in virtual world, and the avatar waits for admission from the user. As the user comes on the playing stage, it gets a signal to start and animation is executed according to user's gesture. While the user is watching the scene of the virtual world with two bare eyes makes specific gestures to control the speed, direction, and height of avatar. Flapping of the both arms of the user can produce beat of flying motion; speed of the flight, and the height difference of the arms decides the direction of the flight. All the behavior of the both arm and upper-body is mapped onto the motion of the avatar.

When user makes slow swing in steady speed, the avatar flies upward from standing position and it continues the flying animation according to user's gesture. The flapping speed of user's arm determines the acceleration or deceleration condition. Namely that speed is calculated from the number of the flapping arms per a second. During the simple flying animation, we cannot have immersive feeling of real flight, so we insert some missions of avoiding obstacles and taking coins. When the avatar arrives the fixed landing position, a specific routine for landing motion is executed. Otherwise, the animation ended with failure.

Avatar used in the contents utilizes only predefined particular gestures and as the player's response is requested, it imitates the motion of the player in the virtual world. Using this animation repeating the player's motion, we can have the feeling of free flight with the virtual object which can overcome the space limitation.

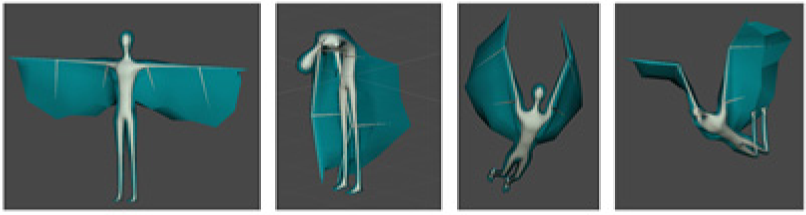

Fig. 1. The 3D Avatar used in the content and some examples of specific pose

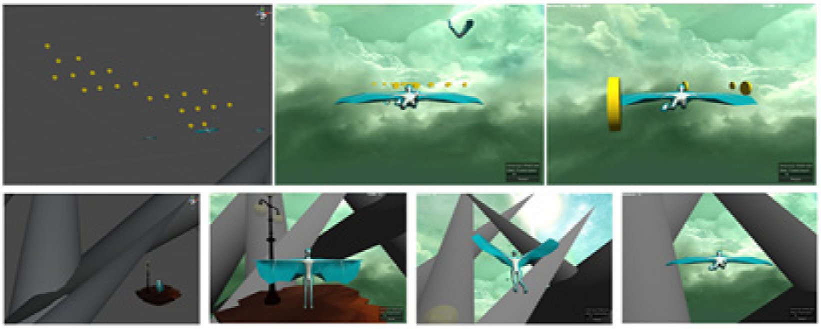

Fig. 2. The objects which are used for special missions: obstacle avoidance, coin collection, and etc

The animation routines which are used in the content are shown in figure 3. It is a sequential motion of taking-off, steady flying(flying without flapping), flying, turning, and landing motion as we can suppose. These animation routines are invoked by particular gestures which are defined as key poses, and consequently user can execute animations of flight by using only simple upper-body motion.

The whole experience of the content is divided into three stages. In the first stage, it makes the user to learn flapping motion which is needed for the continuous flight by acting the taking-off gesture. In the second stage, user can enjoy the experience of specific mission; avoiding obstacles by using the “steady flying” and “turning” gesture as already mentioned.

In the third stage, we can collect coins scattered in the virtual space to experience feeling of immersive interaction. When the user arrives the finish point, user should stop flapping to make landing at the finishing point.

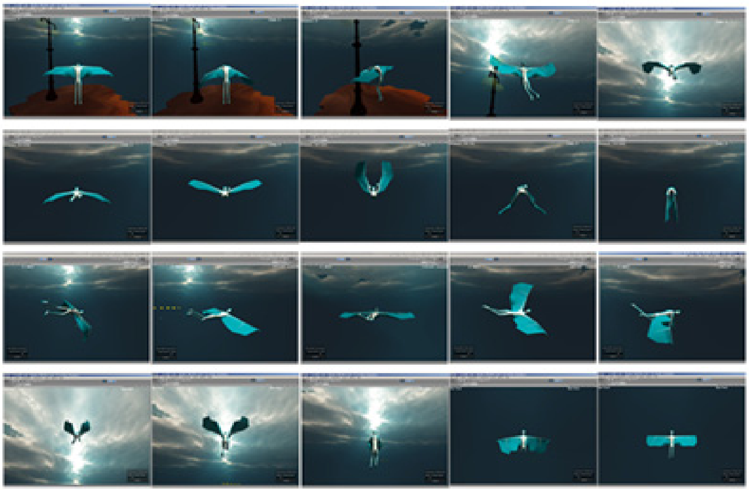

Fig. 3. Some examples of animated scene using gesture recognition

3 Key-Pose Definition and Gesture Recognition

In figure 4, we show the key-poses which are used for the "taking-off" gesture, "right or left movement", and "flight maintaining" gestures. All gesture of upper-body is judged by these key-poses. In order to recognize the gestures we compare the information of body joints obtained with Kinect to the model joints which are previously constructed as a gesture template for the individual key-poses. The advantage of our approach is that joints configuration is very simple, and so that pose estimation is very fast because joint locations are estimated by sequential transformation for a base joint. In order to compare the configuration of the joints, it is needed to estimate the angle between the joints. In this step, we use only major joints; red joints in figure 5, which determine important appearance of the body. In the right image of figure 5, it shows one example of the angle estimation of the left elbow joint.

Fig. 4. Key-Poses used in Contents

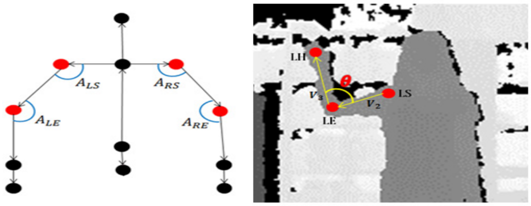

Fig. 5. Some examples of animated scene using gesture recognition

In Equation (1), V1 is the left shoulder joint vector and V2 is the left elbow joint vector. The angle of the left elbow joint can be extracted by using these two vectors. The angle of the two vectors can be extracted by projecting the two vectors onto the x-y, y-z, and z-x plane respectively.

$$|V'_1| = \sqrt{(x'_1 - x'_2)^2 + (z'_1 - z'_2)^2} \quad (1)$$

$$|V'_2| = \sqrt{(x'_3 - x'_2)^2 + (z'_3 - z'_2)^2} \quad (2)$$

$$V'_1 \cdot V'_2 = (x'_{v1} \times x'_{v2}) + (z'_{v1} \times z'_{v2}) \quad (3)$$

$$\theta_{xz} = \cos^{-1}(V'_1 \cdot V'_2) \tag{4}$$

In above equations, x, y, and z are used for expressing 3D location of joints in camera space. After obtaining the inner product of two vectors with equation (3), we determine the angle between two vectors by using the equation (4). We can obtain three angles per one joint if we repeat these processes to each projection plane. These angle values are used for evaluating matched level of the joints. Each pose template is composed of the joint vectors of 11 joints. It means one pose template has 10 vectors as feature information. When user makes some pose, it recognizes gesture to compare to closest pose template.

4 Conclusion

In this paper, we describe an algorithm and an interactive content using the idea to experience feeling of bird's flying by using gesture recognition of a user's upper body. In the algorithm, a gesture is previously defined as a set of some sequential poses. And a pose is estimated with difference of joint angles between models and input joints coming from Kinect camera. Combining the consequential pose estimation, we can recognize the meaning of user's motion as a gesture. Using this result we can animate the player's motion with the avatar, and then we can experience the feeling of free flight of virtual object.

Acknowledgment. "This research was supported by the MKE(The Ministry of Knowledge Economy), Korea, under the ITRC(Information Technology Research Center) support program (NIPA-2013-H0301-13-3005) supervised by the NIPA(National IT Industry Promotion Agency)".

References

1. Haritaoglu, I., Harwood, D., Davis, L.S.: Ghost: A human body part labeling system using silhouettes. In: Proc. of the 14th International Conference on Pattern Recognition, pp. 77–82 (1998)
2. Iwasawa, S., Ebihara, K., Ohya, J., Morichima, S.: Real-Time Estimation of Human Body Posture from Monocular Thermal Images. In: Proc. of 1997 IEEE Computer Society Conference on computer Vision and Pattern Recognition, pp. 15–20 (1997)
3. Goldberg, D.E.: Genetic Algorithm in search, optimization, and machine learning. Addison-Wesley (1989)
4. Micilotta, A.S., Ong, E.-J., Bowden, R.: Real-Time Upper Body Detection and 3D Pose Estimation in Monoscopic Images. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3953, pp. 139–150. Springer, Heidelberg (2006)
5. Gavrila, D., Davis, L.: 3-D model-based tracking of humans in action: a multi-view approach. In: Proc. of IEEE Conference on Computer Vision and Pattern Recognition, san Francisco, pp. 73–80 (1996)