Project Design for KarateGo

Boyuan Zhang
Andrew Id: boyuanz

Modules/Technology: pygame, sklearn, Microsoft Kinect, Machine Learning

Project Name: KarateGo

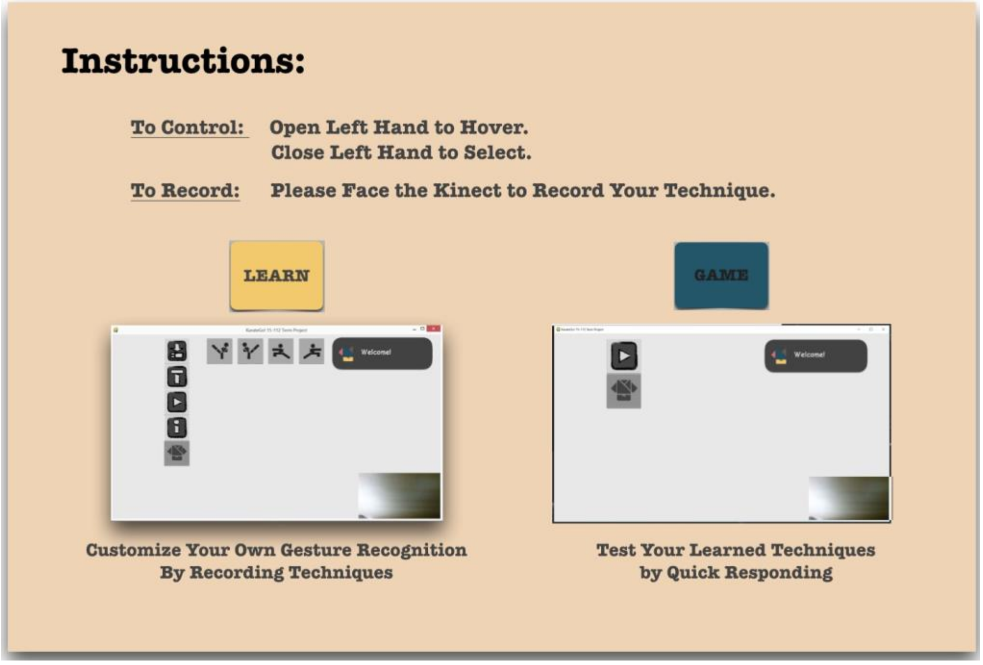
Project Description:

KarateGo is inspired by the following program: NeuMimic (https://www.youtube.com/watch?v=UPJkFVIdjWk), which is a program using Kinect to help patients and doctors to monitor progress of physical therapy. The NeuMimic program can customize the patients' personal physical ability by utilizing information collected by Kinect and report their information to the doctor. KarateGo is inspired by this program and was expected to be able to score users based on their personalized movement inputs.

KarateGo is a educational and entertaining program that can teach the user to do basic karate techniques, such as middle punches, round kicks etc.

The program will take advantage of the the body frame and color frame generated through kinect sensor to get the user's skeleton information. The skeleton information can be further processed by the program. Machine learning is used here to do gesture recognition, that is to determine the technique the user is doing. To control the system, the program will utilize built-in hand-gesture recognizing function of pykinect2 module. Pygame will be used here to support the game frame and game logic.

The project basically has two modes to play with, learning mode and gaming mode. Demo of the two modes UI are as Figure 1.



Instructions:

To Control: Open Left Hand to Hover.
Close Left Hand to Select.

To Record: Please Face the Kinect to Record Your Technique.

Customize Your Own Gesture Recognition
By Recording Techniques

Test Your Learned Techniques
by Quick Responding

Figure 1 . Demo of the Instructional Window

1. Learning mode: Customized training of KNN to do gesture recognition.

Currently, the user can do four basic karate technique recognition, I.e. right/left round kick and right/left middle punch.

In the learning section, the user can performing his/her own techniques to train the model. To do the customized training, the user should input his/her own technique movements by controlling corresponding 'button' on the screen, which can trigger a recording function that take a sequence of 'snapshots' of joints coordinates (x, y, z) captured by Kinect sensor. Multiple sequences of those 'snapshots' then will be stored as numpy array into a '.plk' file, which can be loaded later as training dataset for model training process.

A 'training button' can be triggered to do the training process based on the user input technique data. Since the gesture recognition is based on time serial joints coordinates information, the Hidden Markov Model was considered here at first. However, the performance of the HMM model didn't reach my expectation. So I turned to use KNN, which works very well at the end. The training process will return a model which then can be used in testing user's gesture.

In the testing process, the program can randomly generate a technique command, say 'do a round kick'. The user will have 1 second to complete the technique. The program will record his/her performance and then test it by the trained model mentioned above. If the user do the same technique as the command say, the program will recognize it and give him/her a point, otherwise, his/her score will be deducted by one point.

2. Gaming mode: Test user's learning proficiency on the above technique.

To test how well the user has learned the above technique, the game mode will do a test by means of quick responding. The program will randomly choose 10 techniques from the available types pre-trained by the user in the learning mode. User must follow the instructions on the information board to earn points. Total points will be displayed on the information board after the test.

3. Switching between Modes and instructions are provided at each mode.