

KinectMonitor++

User Guide

Version 1.0



Because this software is still under developing, then only a quick starting is introduced here to help you get a quick overview of this simple demo and see what is going on with current developing. In addition, we will also provide some tips for your further development, which we hope would be helpful to you.

1 General Information

Physical Therapy today is one of the least efficient treatments. Not because there aren't talented clinicians and cutting edge treatments, but clinicians have no way of monitoring the process, ensuring accuracy of movements and patients receive vague instructions on how to perform them and often don't comply.

Hence, there is great potential and urgent demand to build up a bridge between clinicians and patients, such that doctors could obtain the timely feedbacks about the healthy condition of patients and patients would be under guidance of clinicians to recover more fast and better.

Right now, we are trying to develop a client software to obtain the patients' information through some advanced cameras or monitor devices. So this demo, "KinectMonitor++", is aimed at verifying the correctness of Dynamic Human Gesture Recognition based on Kinect Sensor and providing a friendly enough user interface for human-machine interaction.

1.1 The Working Environment

This demo is based on the Kinect sensor v1, one of advanced products from Microsoft, to implement a window-oriented application. Therefore, there are some minimum hardware requirements for correctly running a Kinect sensor:

- 32-bit (x86) or 64-bit (x64) processors
- Dual-core, 2.66-GHz or faster processor
- USB 2.0 bus dedicated to the Kinect
- 2 GB of RAM
- Graphics card that supports DirectX 9.0
- A Microsoft Kinect for windows sensor

And the supported OS

- Windows 7
- Windows 8
- Windows Embedded Standard 7
- Windows Embedded Standard 8

- If your Windows 7 edition is Windows 7 N or Windows 7 KN, you must install the Media Feature Pack, which is required by the Kinect for Windows runtime.

1.2 Installing KinectMonitor++

First, we are using the “Kinect for Windows SDK 1.8” to develop our software, so the corresponding drive of Kinect sensor should be installed before your using.

Then, because it is nothing but a testing demo, we don’t plan to build a release version for installing. So the only thing you need to do after you install the drive correctly is simply to double click the file “KinectDemoApp.exe” to start the demo as shown in fig1 below.

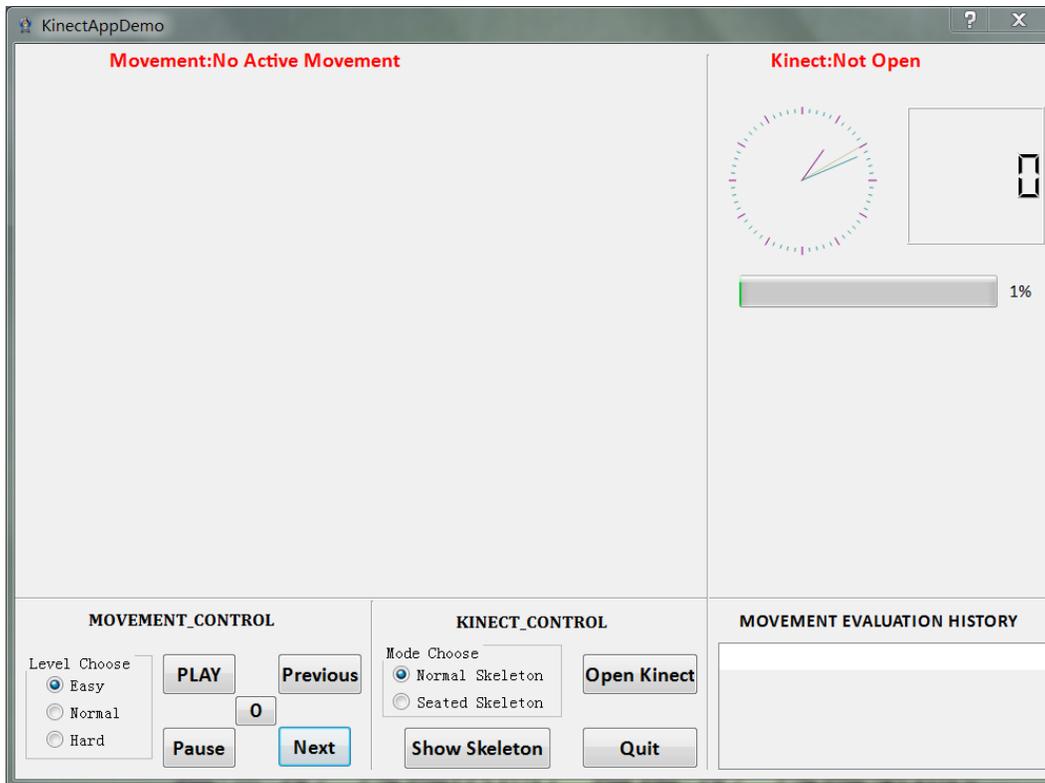


Figure 1 Demo Layout

1.3 Develop Information

UI lib: QT4.8

Kinect SDK: SDK v1.8

Developing language: C++ (C++11)

Developing OS&IDE: Win7 / Visual Studio2010

Image Processing Lib: OpenCV (cross file computer vision lib)

2 UI interaction

In this section, we will illustrate the functions of each components in the UI, and propose a recommended way to use this demoApp.

2.1 Introduction

We are going to introduce each elements in the GUI one by one as shown in the fig 2, and explain corresponding states in detail.

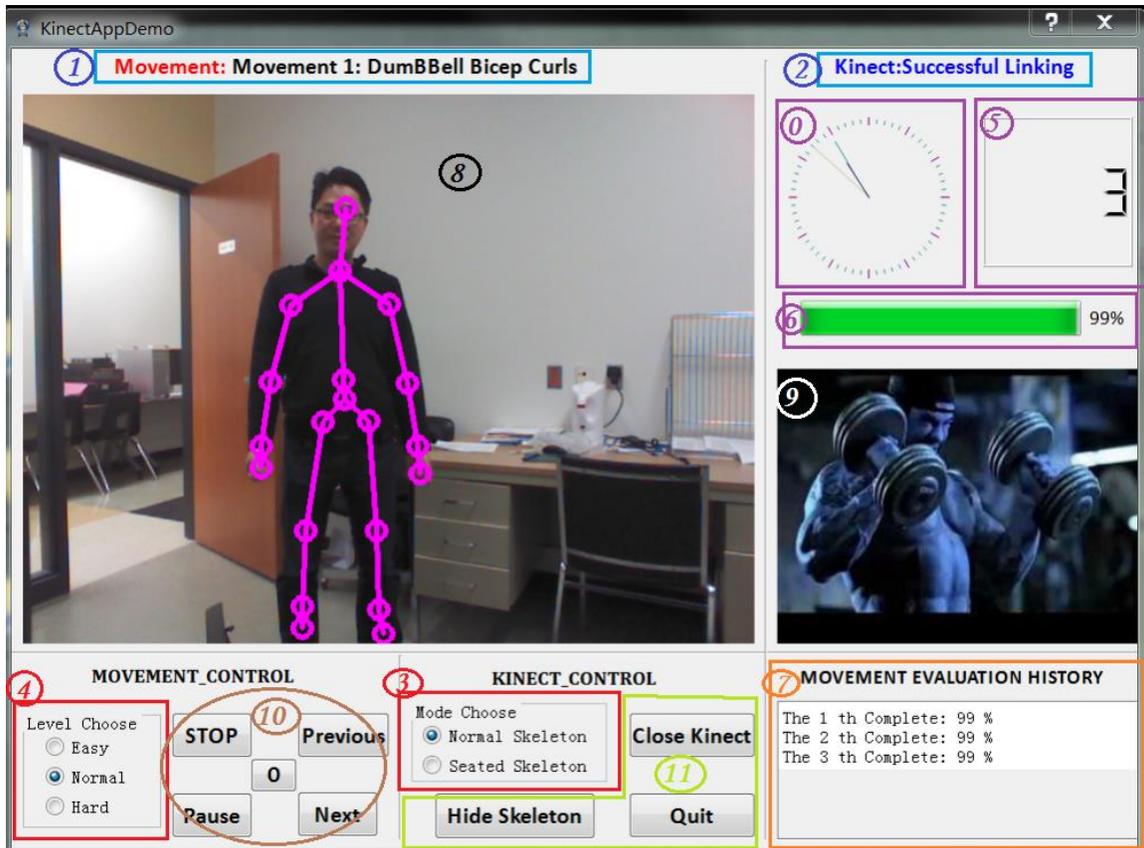


Figure 2 Introduction to GUI

Section 1: State Label

Item 1: shows the current movement information, its ID and name. If nothing has been chose, it will show “No active movement” in red.

Item 2: shows the Kinect state, which has been defined in three different ones: “successful linking”, “Not open” and “linking failed”.

Section 2: Radio Button Group

Item 3: consists of two options: normal skeleton in which the sensor will track the whole body and all the skeleton joints; seated skeleton in which the sensor will just track the two hands and shoulders.

Item 4: includes three different modes. In easy mode, according to current setting, only one feature need to be matched. More difficult level means that more feature judgments will be triggered.

Section 3: Result View

Item 0: simply shows the current system time on the analog clock.

Item 5: shows the complete times of the current movement.

Item 6: shows the rough assessment of current movement.

Section 4: Result History

Item 7: shows the records of current movement, and save them into an output file in “.txt” format.

Section 5: Graphic view

Item 8: shows the color image obtained from the camera, about 25 fps.

Item 9: shows the standard movement image.

Section 6: Push Button Group

Item 10: “Stop & Play”, to open up the movement assessment system; “Previous” and “Next”, to choose different movements; “Pause”, to pause the movement assessment system, and the Color Image; “O” to clear your complete times, meaning you want to replay this movement.

Item 11: “Close & Open Kinect”, to open up and close the Kinect sensor; note that only if Kinect sensor is open, other components are effective, otherwise, you will be reminded to open your Kinect sensor firstly. “Quit”, to shut down the whole application directly; “Hide & Show Skeleton”, to show or hide your skeleton info in the Color Image.

2.2 Using Tips

There is a recommended four-step procedure to test this demoApp before using. After that, you could use it in your own way.

Step1: Push the button “Close Kinect” to check whether your Kinect sensor works and has been initialized properly through the info on *state label*. If everything is correct, you will see the demo running as shown in fig2; otherwise, please make sure all the following problems is Ok:

- Kinect sensor is working elsewhere.
- The Kinect sensor is charged and connected to your computer through USB 2.0;
- The Kinect sensor drive has been correctly installed. Check it in the device manager on Win7 OS.
- Your computer and OS could satisfy the minimum requirements.

Step2: Trigger the movement assessment system by pushing the “PLAY”. Then you can try to do the movement according to the standard image, to see whether your action has been tracked correctly and whether the Result View component works. It is strongly recommended to show the skeleton info while playing, which will potentially correct your actions and make yourself stand in an appropriate position.

Step3: Adjust the assessment system as what you want. In this procedure, if you push the “previous” button while the current movement is the first one, or if you choose the “Next” while the current movement is the last one, you will be warned of error operation. Besides, the “Normal” level is recommend for your common choice. But certainly, you could change it whatever and whenever you want.

Step4: Push “Close Kinect” or “quit” button to check whether the Kinect sensor would be shut down correctly. This is of significant importance, because you can’t use the sensor anymore if your sensor hasn’t been softly shut down properly. Surely, relinking the sensor or cutting off electricity always works, but which is not a good and recommended way.

If all this four steps have been verified to be OK, then just enjoy this demo and play it in your own.

3 Function Testing

The current demo could roughly implement the counting and evaluating functions for two different movements, namely “dumbbell shoulder press” and “dumbbell bicep curls”.

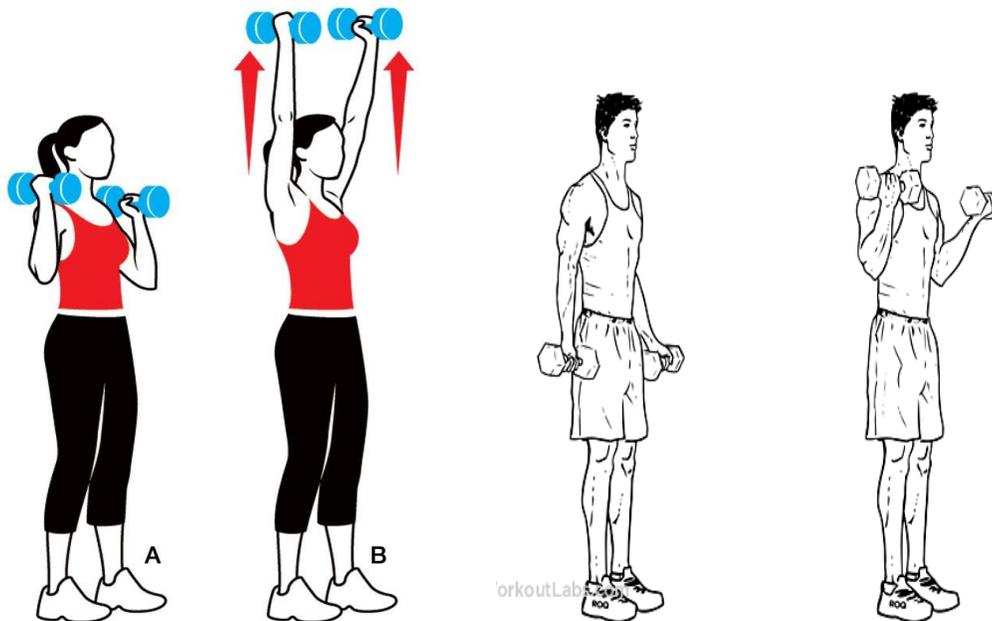


Figure 3 Two complete actions

4 Further Developing

In this section, we will give a brief introduction to the current demo from these 2 aspects:

Designed Class:

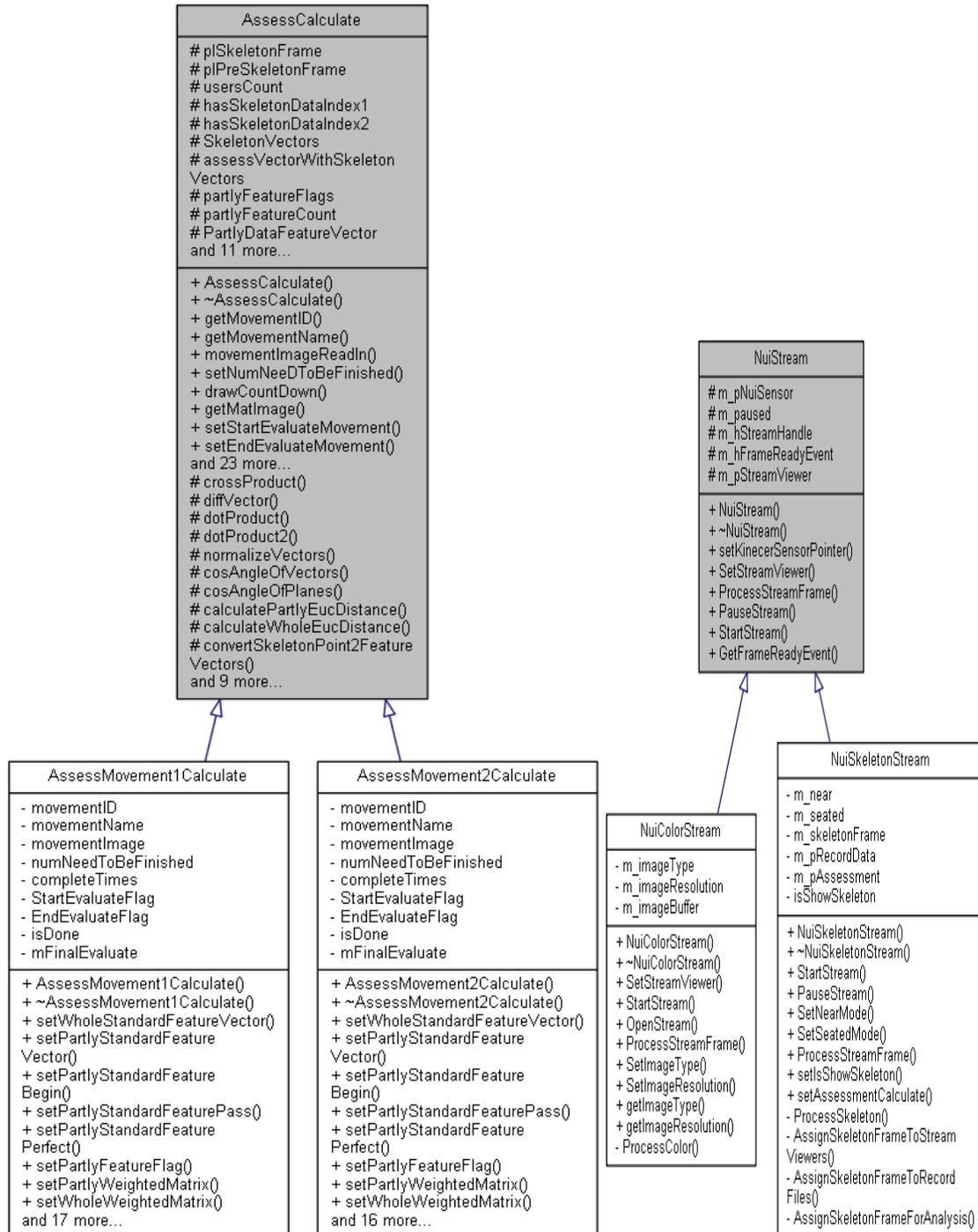


Figure 4 Two main classes

