

Studio Applications and Software Development Kits for Microsoft Kinect: A Survey

Muhammad Shoaib Farooq^{1,2}, Bilal Hassan¹, Mudassar Naseer³, Adnan Abid¹, Yaser Daanial khan^{1,2}, Nabeel Sabir Khan¹, Muhammad Usman Akram³, Saif Ullah³

¹Department of Computer Science/SST, University of Management & Technology Lahore, Pakistan

²Department of Computer Science, Abdul Wali Khan University, Mardan, Pakistan

³Department of Computer Science, COMSATS Institute of Information Technology, Lahore, Pakistan

Received: September 1, 2014

Accepted: November 13, 2014

ABSTRACT

The invention of Motion sensing camera “Microsoft Kinect for XBOX” opened new dimension for the research community. First objective of Kinect was to work with XBOX for playing games. However, people started using it for the recognition of human activities. Kinect for XBOX provided two types of information 1) RGB and 2) Depth. People developed tools, drivers and algorithms to work with it. Later, Microsoft itself released the Windows edition of Kinect with a software development kit (SDK). However, several open source and propriety vendors modified the performance of their existing tools and drivers to work with Kinect for Windows. It was now difficult for the newer one to select an appropriate set of studio application and SDK to work with Kinect for Windows. Keeping in view this point, we have presented a study for the Windows edition of Kinect based on several features like platform, file formats etc. Our study will definitely help the research community involved with Kinect for Windows to select an appropriate set of drivers and SDK's for their work. This study will be a valuable contribution not only for the users of Kinect, also for the owners of various tools.

KEYWORDS: Kinect; Microsoft; studio applications; features; comparative study; software development kit.

1 INTRODUCTION

Initially the purpose of Microsoft Kinect was to work with XBOX for playing games [1]. This device broke the Guinness books world record of most selling device in the first 60 days of its launch [1]. The community working in the domain of human activity recognition took a different sight towards this motion sensing camera. They developed a number of tools for capturing visual information from Microsoft Kinect. They developed algorithms for processing the RGB and depth information provided by the Microsoft Kinect [2- 5]. They also modified already developed SDK's and drivers to work with the windows edition of Microsoft Kinect.

The Kinect for Windows edition got lot of attention from the research community as a motion sensing device. People developed set of drivers and development environments for XBOX Kinect. Later they modified them to work with Kinect for Windows edition. Currently a number of tools are available in the market. The problem arises which to use, why to use and how much benefit it will provide on the other one. Although, Microsoft is also present in the race with lot of functionalities. But the other competitors in the market are also not behind.

We picked up this problem and decided to provide a detailed study which will definitely help the new ones working with Kinect for Windows. Our study will also address the questions asked in the above paragraph. To the best of our knowledge, we decided to compare the tools on the basis of few most commonly used and asked questions. These features are the common ones among those who are asked by a novel user first of all.

In our work we have evaluated 05 most commonly used tools / SDK's freely available in the market. We evaluated tools on the basis of various parameters such as license information, supported platform,

* **Corresponding Author:** Muhammad Shoaib Farooq, Department of Computer Science/SST, University of Management & Technology Lahore, Pakistan

supported development environment, information components extracted, exported file formats and the skeleton information extracted etc. As discussed above, these are the features which are normally asked by a user.

We divided our work into two sections. In the first section we present the introduction of each tool with its features in detail and in the second section we present a comparative overview of these tools.

2 SDK's and Drivers

2.1 Microsoft SDK & Drivers

Microsoft launched its Kinect for Windows Software Development Kit (SDK) in 2011 [6]. Microsoft has never thought to support the Kinect for Windows for development. It was very much surprising for Microsoft a great interest by the people in the device for development purpose. Presently the Microsoft is aggressively providing support in terms of drivers and SDK [7]. The Microsoft SDK is available free of cost and can run on windows environment with its development possible in C++ and C# languages [8]. The exported file format depends upon the development environment. It provides RGB, Depth and Skeleton information and can track all 20 joints of human body. The major limitation of Microsoft SDK is that it couldn't store Depth and Skeleton information on the disk for offline use. However, it provides enhanced facilities like Background removal, Realistic capture with Kinect fusion, improved tracking robustness with Kinect fusion, HTML interaction sample, Multiple sensor Kinect sample and adaptive UI sample [9], [10] and [11].

2.2 CL NUI SDK and Drivers

This is a partially free SDK package along with drivers. It has trial version for 30 days and a propriety version available after that [3]. It's for windows platform and supports development environment like C, C++ and C#. It exports file in the format which is dependent upon the development environment. The skeleton information can be extracted by applying different algorithms on the depth map information which decreases the overall performance of the system. It can support only one Kinect sensor at a time.

2.3 Open Kinect

It's an open source product that provides drivers and SDK to work with Kinect for XBOX [2]. It provides support for multiple platforms like Windows, Linux and MAC. It also support for multiple languages such as Python, C, C++, C# and more. Its exported file format depends upon the development environment. It provides only RGB and depth map with no direct skeleton information. However, the SDK provides special depth processing algorithms to extract skeleton information from depth map this extra processing slows down the efficiency of overall system.

2.4 Open NI

Open NI is one of the oldest set of SDK and drivers for Kinect sensor [4]. It supports one Kinect at a time and provides only RGB and Depth information. It can run on Windows and UNIX platforms and supports only C++ development environment. It provides a set of algorithms to process the depth information obtained from Kinect to extract skeleton information. However, the skeleton data provides information for only 17 joints of the human body.

2.5 NUI Capture

NUI Capture Analyze is an application for recording, analyzing and storing Depth, RGB, skeleton and audio data from Microsoft Kinect [4]. It provides the facility for exporting provided information into Matlab file format [12].

The support documents are divided into 04 sections provided by NUI Capture [13]. The first section covers installation information [14]. The capture section discussed the camera settings information in various environments [15]. The export section provides information about procedure to export data along with supported file formats. The last one is play back section which provides reuse information about previously recorded sessions [16].

The installation section provides information about both installation and UN installation of NUI Capture in windows environment. The capture section provides option of the data capturing with different settings along with previewing and permanently storing videos in the disk. We can first adjust frames per second rate, audio levels and file formats during preview and then to record data permanently on disk. The

information recorded by NUI capture can be exported in the form of RGB, Depth and Skeleton frames along with .avi and .mat file formats. You can also play back the previously recorded session using NUI capture [16].

The major information exported by NUI capture lies in the .mat file. The .mat file has plenty of information for each frame of a recorded session. Each frame has its own depth, color, skeleton and face values, depth frame to color frame mapping, skeleton frame to color frame mapping and the face triangle.

However, it's just a data capturing tool and has no particular development environment. It provides information of all the 20 joints of human body. It exports information in the form of .mat file which can then easily be used in any development environment on any platform. It also provides the functionality of 3-D session viewer [17].

Another important aspect is the requirement of high graphics resolution while using these capturing tools. However, this can be achieved by using multiprocessor systems as discussed by Jan, A., & Khan, S. A in [18].

3A Grid Analysis of Kinect Studio Applications

Which Kinect data capturing tool (drivers and SDK etc) would be appropriate for a specific work is a big question before a person starts developing dataset for his/her project. In any project, you may need different functionalities like; offline availability of skeleton information, direct skeleton information, storage of Depth, RGB and Skelton for later use, multiple Kinect sensors and portable files format etc.

Keeping in view, all of the requirements, we have provided a comparative study of the major Kinect studio tools available to the best of our knowledge. It includes both open source and commercialized ones. We have evaluated five tools; Open Kinect [2], CL NUI SDK [3], Open NI [4] and NUI Capture with Microsoft's own drivers and SDK [5 - 6]. To the best of our knowledge, we have provided a grid analysis of the tools discussed above as per the order of discussion above. We have provided our results in distributed form from Table 1 to 3.

Table 1: General System Requirements

License Info	OS	Graphic Proce
Freeware	Windows	Windows 7
Partially Freeware	Windows	Windows 7
Open Source	UNIX, MAC & Windows	Generalized Graphics card available for 32 bit Machine
Open Source	UNIX, MAC & Windows	Win(32, 64), Unix (32, 64, ARM) and MAC (OS X)
Freeware	Windows 7	As Required by Win 7

Table 2: OS Requirements, Supported and Languages and Output File Formats

System Requirements	Language	Exported File Format
Windows 7	C++ and C#	Depends on Development Environment
Wind 7 Machine	C, C++ and C#	Depends on Development Environment
Any 32 bit Machine	Python, C, C++, and C#	Depends on Development Environment
Win(32, 64), Unix (32, 64, ARM) and MAC (OS X)	C++	Depends on Development Environment
As Required by Win 7	Provides .mat file that can be used on any platform	.mat File

Table 3: Information for Future Processing

Information	Direct Skeleton (No of Tracked Joints)	Offline Information
RGB, Depth and Skeleton	20	RGB, Depth Map
RGB and Depth Map	NILL	RGB and Depth Map
RGB and Depth Map	NILL	RGB and Depth Map
RGB and Depth Map	NILL	RGB and Depth Map
RGB, Depth Map and skeleton	20	RGB, Depth, Skeleton and more

4 Conclusions

In this paper, we discussed the origin and use of Microsoft Kinect as motion sensing device. We provide a brief introduction of Kinect for XBOX and Kinect for Windows. To the best of our knowledge, we selected 05 SDK's and data capturing tools available in the market. We compared them on the basis of nine different features. As per our thinking, these could be the basic features required by the individual who is involved with Kinect for Windows. The study presented in Table 1 will definitely help the research community working in the field of human activity recognition using Kinect sensor to select an appropriate tool.

REFERENCES

1. Ramos, E. (2012). Kinect Basics. In *Arduino and Kinect Projects* (pp. 23-34). Apress.
2. Kinect, O. (2014). OpenKinect. Openkinect.org. Retrieved 18 September 2014, from <http://openkinect.org/>
3. Capture, N. (2014). Retrieved 18 September 2014, from <http://code laboratories.com/kb/nui>
4. NI, O. (2014). Retrieved 18 September 2014, from <http://www.openni.org/>
5. Capture, N. (2014). nuiCapture - Record, export, and playback Kinect data for analysis effortlessly. Nuicapture.com. Retrieved 18 September 2014, from <http://www.nuicapture.com/>
6. Webb, J., & Ashley, J. (2012). *Beginning Kinect Programming with the Microsoft Kinect SDK*. Apress.
7. MSDN, B. (2014). Updated SDK, with HTML5, Kinect Fusion improvements, and more - Kinect for Windows Product Blog - Site Home - MSDN Blogs. Blogs.msdn.com. Retrieved 18 September 2014, from <http://blogs.msdn.com/b/kinectforwindows/archive/2013/09/16/updated-sdk-with-html5-kinect-fusion-improvements-and-more.aspx>
8. Jana, A. (2012). *Kinect for Windows SDK Programming Guide*. Packt Publishing Ltd.
9. Kean, S., Hall, J. C., & Perry, P. (2011). Microsoft's Kinect SDK. In *Meet the Kinect* (pp. 151-173). Apress.
10. SDK, K. (2014). Kinect for Windows. Research.microsoft.com. Retrieved 18 September 2014, from <http://research.microsoft.com/kinectsdk/>
11. Zhang, Z. (2012). Microsoft kinect sensor and its effect. *MultiMedia, IEEE*, 19(2), 4-10
12. Export, N. (2014). How to export a nuiCapture Session - nuiCapture. Nuicapture.com. Retrieved 18 September 2014, from <http://nuicapture.com/support/how-to-export/>
13. Support, N. (2014). Support - nuiCapture. Nuicapture.com. Retrieved 18 September 2014, from <http://nuicapture.com/support/>

14. Installation, N. (2014). How to Install/Uninstall nuiCapture - nuiCapture. Nuicapture.com. Retrieved 18 September 2014, from <http://nuicapture.com/support/how-to-installuninstall-nuicapture>
15. Installation, N. (2014). How to Install/Uninstall nuiCapture - nuiCapture. Nuicapture.com. Retrieved 18 September 2014, from <http://nuicapture.com/support/how-to-capture>
16. Installation, N. (2014). How to Install/Uninstall nuiCapture - nuiCapture. Nuicapture.com. Retrieved 18 September 2014, from <http://nuicapture.com/support/how-to-play>
17. Installation, N. (2014). How to Install/Uninstall nuiCapture - nuiCapture. Nuicapture.com. Retrieved 18 September 2014, from <http://nuicapture.com/support/how-to-play/#using-the-3-D-viewer>
18. Jan, A., & Khan, S. A. (2013). Review of different approaches for optimal performance of multi-processors. *VFAST Transactions on Software Engineering*, 1(2), 7-11.