

Vivek Sridhar
Sean Reidy

High Detail Kinect Motion Tracking for Sign Language Interpretation and Instruction

Abstract

Our project aims to develop a high-resolution motion tracking and gesture recognition system for the Microsoft Kinect sensor, aimed at teaching American Sign Language, which, unlike most verbal languages, is very difficult to learn and to teach. We will develop this tracking and recognition system for the purpose of teaching ASL, but with the hope that it will be applied to many other areas in the future. The language is based on motion and gestures that can be hard for one to learn without hiring an instructor. Sign language interpreters serve as a good way for the hearing impaired to communicate, but they are not always available to the individual. With over 10 million Xbox One units shipped, a solution to the issue could come from the Xbox's included 3D camera: The Kinect. With an advanced 3D camera available in millions of homes, one can have a digital sign language instructor and interpreter in their home. Building upon the current Kinect API, we plan to capture more detailed positional data about hand and finger positions in order to create a sign language instructional tool. We will build an API that allows developers to access more detailed positional data from the Kinect, and an application that can be used as an instructional tool for American Sign Language. The application will be available to Xbox One users through the Xbox Marketplace, and the API will be available as an open-source library.

Research Question and Significance

The Microsoft Kinect sensor, in particular the Version 2.0, is an extremely powerful piece of technology, complete with full HD cameras, a high-resolution depth sensor, and an infrared sensor. These sensors allow the Kinect to capture full body tracking data for multiple skeletons at the same time, while also being used as a webcam. Currently, the majority of applications for the Kinect motion tracking system are in video games. However, we believe that another important application exists that the Kinect would excel in solving - the interpretation of sign language, particularly to be used as a teaching tool.

Currently, the Kinect sensor's API is very good at detecting general movements and gestures, as well as the positions of the hand and other major joints of the body. This is only part of the challenge, however; the Kinect lacks the finer detail necessary to successfully interpret sign language - the motion of individual fingers. The depth camera on the Kinect is high-resolution enough to give us the data we need to extract the positions of individual fingers. This would be the first step of this research project - developing the technology, and a programming interface, that would allow us and others to track the positions of individual fingers using the Kinect.

The second part of our project would be to design and develop a piece of software that teaches people sign language using the Kinect sensor. This would be an application deployed on the Xbox One platform, because it is by far the largest install base of the Kinect sensor, and deploying to Windows would require users to buy a \$50 adapter for a sensor they already own.

The first part of this project is significant to the programming and Kinect development world because it provides an interface for developers to access more detailed and precise motion

tracking data than the Kinect API currently does. Besides sign language, there are countless applications for finer finger tracking using the Kinect sensor. The first that comes to mind is a greater level of control in video games – being able to implement features such as heads-up displays and virtual keyboards in games using fingers rather than sweeping gestures. Another application is in human-computer interaction. Many robotics applications and interactive installations both in research and in places like theme parks use Kinect sensors to interface with their users, and this improved finger recognition would go a long way to making the user experience smoother and more intuitive. The second part of this project is significant because it fills a need - sign language interpretation and instruction - and also is a step towards branching the Kinect into applications outside of video games.

Project Design and Feasibility

The largest section of this project will be focused on developing the API for getting finger tracking data from the Kinect. To do this, we will primarily use the depth sensor information, in conjunction with the skeleton tracking. Essentially, we will use the skeleton tracking information to narrow down the frame to the camera, and then use depth sensor information to distinguish between the fingers. The next step will be to identify the different fingers as thumb, index, etc. An important step after this will be to detect which position each individual finger is in - most likely, we will do this by determining which joint(s) on the fingers are bent, to tell whether the finger is angled forward, or completely curled up, or in other positions. Once we can detect the positions and orientations of each individual finger, we can put them together into a system that detects the positions and orientations of all the fingers in a hand. This will let us complete the final step of this part of the project, which is associating the motion tracking data of each hand with a database of sign language signs, to identify what sign the person is making.

If it is necessary, we will also use OpenCV image processing on the RGB image to complement the depth data.

The database of signs will be broken down into two parts - large-scale gesture recognition, and small-scale finger position recognition. The macro gestures can already be recognized by the current Kinect libraries, and the finger positions will be recognized by the library that we are writing for this project. To generate the database containing the correct signs, which we will use as our primary teaching tool, we plan to find and contact people who know sign language, and have them essentially record the gestures into our program once we have finished writing the finger tracking libraries. Sean Reidy is also planning on taking a sign language course this summer.

The final step of this project will be actually writing the program to teach people sign language. The end goal of this program is deployment on the Xbox One platform, because it is the largest install base of the Kinect sensor, and it doesn't require people to buy a \$50 Kinect PC adapter to use. We are going to develop our app for the Universal App Platform on Windows 8, making sure to only use hardware that will be available on the Xbox One platform. Then, because of the flexibility of the Universal App Platform, we will port it to the Xbox One platform.

Background

Sean Reidy: I have experience with research with quadcopters; currently one of the instructors of 98-270 (StuCo Quadrotors: Robots in da Skies). I have worked on quadrotor

development using Autodesk Inventor 2015 and 3D printers to design, build, and test various drones for educational purposes. I have been a team leader for two high school robotics teams throughout most of grade school. Within the teams I learned valuable project management skills, and technical skills including but not limited to Java and 3D modeling. I have earned my Eagle Scout award after completing my project to create and manage a Junior FIRST Lego League team for autistic children of wounded veterans. After participating in HackIllinois 2015 I have experience with the tools (Kinect API, and OpenCV) that will be used during most of the project.

Vivek Sridhar: I have taught myself programming in C#. I was also a part of my high school robotics team for four years, where I learned machining, CAD, and design. I was the president of the robotics team in my senior year, and worked on managing all aspects of the team, from designing our robot to dealing with the finances of the team. I am doing undergraduate research with the CMU Biorobotics Lab, where I've gained experience working in C, C++, and C# using Visual Studio. In addition, after participating in HackIllinois 2015, I gained experience working with Visual Studio and the Kinect and OpenCV APIs.

Feedback and Evaluation

David Kosbie, a professor of computer science, reviewed and approved our proposal. The student members of the team will meet at least once a week to discuss progress and goals for the next week. We will check in with our faculty advisor at each milestone of the project, particularly at each step of developing the finger tracking API, and determine if the tracking is precise and successful enough to move forward.

Dissemination of Knowledge

We will present our project at the Meeting of the Minds. We also plan make the application available to download on the Xbox App Store. Finally, our current plan is to make the API for Kinect finger tracking available as an open-source Kinect C# API through a platform like GitHub, so other developer can take advantage of the finer level of tracking detail that the Kinect sensor is capable of providing.

Budget

Item	Cost	Source
Kinect 2.0	\$175	Amazon/Microsoft Store
Kinect for Windows Adaptor	\$60	Amazon/Microsoft Store
Microsoft Xbox One	\$400	Amazon/Microsoft Store
Poster Costs	\$100	Kinkos
Total	\$735	