

Leap Motion FirstThoughts

Posted on June 17, 2014 by Wan Chi Lau in Development Environment, iOS Programming, Natural User Interface, RainyDayEngineering, UI Prototyping// 0 Comments

A little more than **a year ago** we wrote a piece on the upcoming release of a device which we thought would make a huge impact on the state of natural user interfaces for the general market. The **Leap Motion** controller went on sale at Best Buy at the end of July 2013. Unfortunately for Leap Motion the controller was not useful for anything other than the most primitive of interactions, and even then, its effectiveness was highly questionable.

We thought about writing an article on the Leap Motion controller at the time, but held off because there were already a slew of reviews (**MIT Tech Review**, **Gizmodo**, **Amazon**, etc) on its disappointing performance. We had nothing new to add and did not feel the need to pile on. So was the Leap Motion gadget DOA? Far from it.



The problem back then with the Leap Motion was not with the controller. The hardware was ready. The issue was the software released with the device, while stable, was clearly in the alpha state in terms of functionality. It showed the device's tantalizing potential, but even if one was willing to put up with its limitations, it was good only for demo purposes.

Our hope was that Leap Motion would stay alive long enough to improve the software. Today, we are happy to report that, judging by the features and responsiveness of V2.0 of the SDK, the Leap Motion controller may soon be actually ready for prime time.



There is not much **in the box**: the controller, **special USB cable**. Unlike **the Kinect**, a 1"x3" Leap Motion stick is **not packed** with electronics. There are two cameras and some circuitry, but the heavy lifting (motion processing, spatial computations, etc) is left to the computer. The software needed to setup the module is on the Leap Motion site. Keeping the device simple and minimizing what came in the box may be why Leap Motion can offer the controller at a retail price of **just \$80**. It is a bit expensive as a secondary UI device, but the price is competitive with accessories such as a high-end gaming mouse or large trackpad.



Setting up the device was surprisingly quick and simple. We downloaded the installer, ran the setup program, and was up and **running** in a few minutes. There is an app store with a growing collection of both free and paid apps. We would recommend trying the free ones to get a feel for what the device can do.



We did exactly that with the V1.0 software and was intrigued by the potential of the controller, then we put it aside for almost a year. A few weeks ago, we got an email notice from Leap Motion that **Beta release of V2.0 software** was now available. We decided to take a look to see how they had been spending the time. We were glad we did.

Skeletal Tracking Beta

Since launch, we've been listening to your feedback and building the next generation of tracking. Now it's here for you to try – but it's still a work in progress. Let us know what you think!

[Please, sign in before downloading the SDK](#)

Download v.2.0.2.16391 Beta for OSX

Also available for Windows and Linux

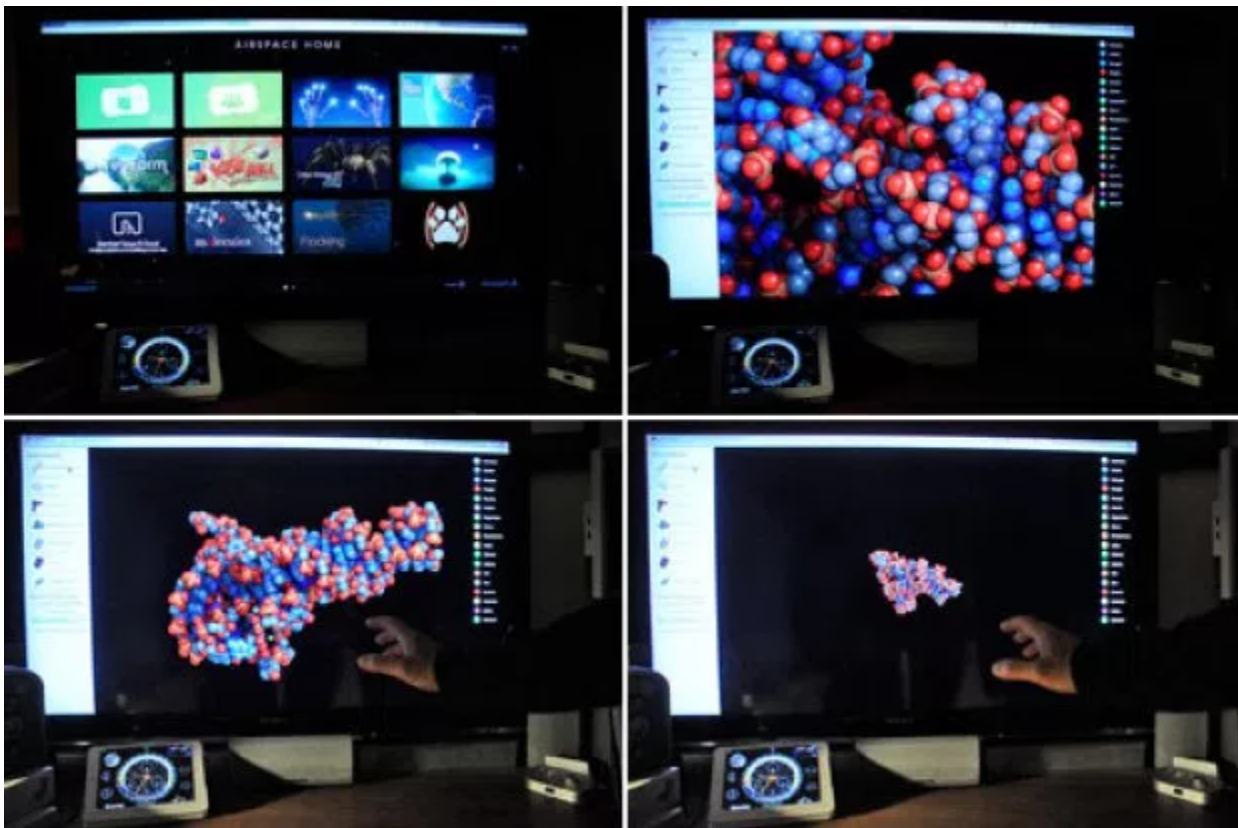
A laptop screen displaying a skeletal tracking application. Two arms and hands are visible on the screen, one in purple and one in green, against a grid background. A large play button icon is centered on the screen. The laptop is shown from a slightly elevated angle, and the background is a solid blue color.

This beta release is what the hype was about a year ago. What is different about V2.0? **Skeletal Tracking**. Instead of just seeing the hands as two flat surfaces moving in three dimensions, the software now can track the **fingers and joints** in 3D space. The difference is rather astounding.



When the Leap Motion controller was first demonstrated, everyone was excited at the prospect of having their own *Minority Report*-style hands-free gestural UI. While not perfect, V2.0 is one big step in that direction. Seeing this version working on a laptop screen was pretty cool, but **using it on a 46" screen** definitely sparked the imagination.



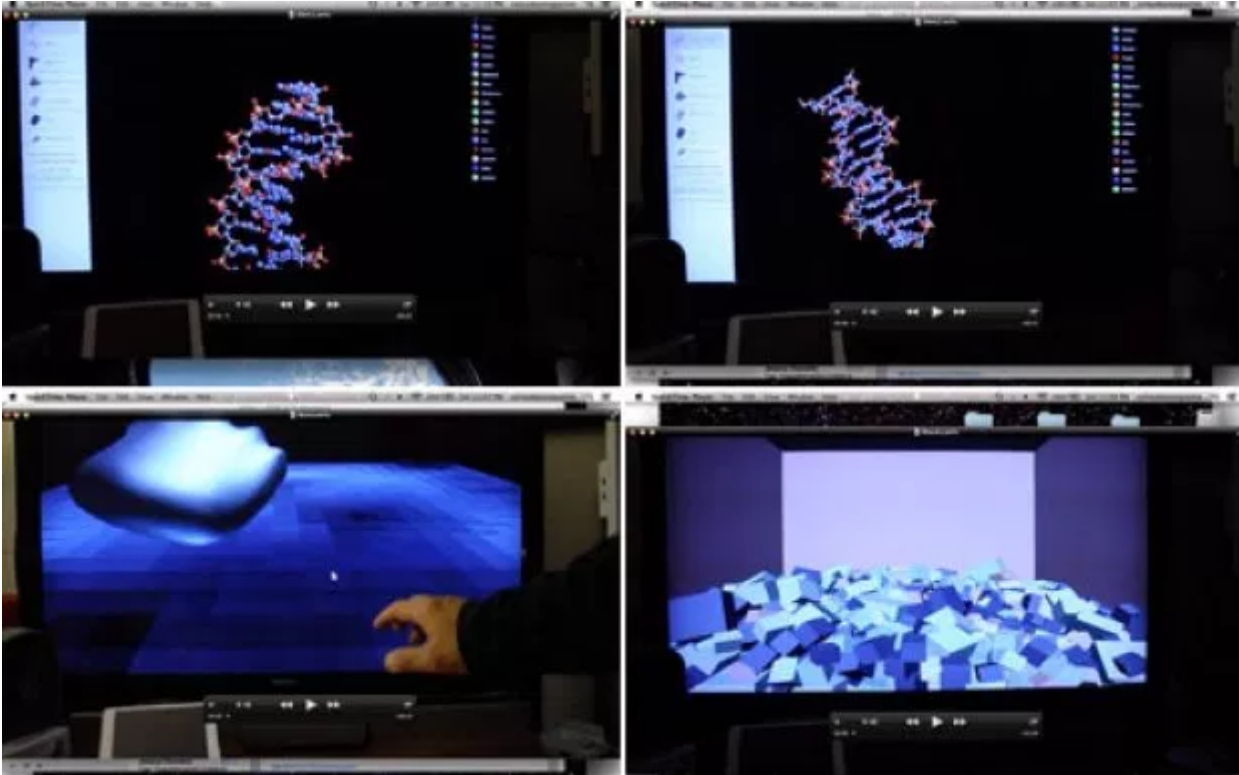


We are not sure if it was because we have a faster computer or that the Leap Motion folks have optimized the code (probably both), but **this version's response time** was definitely much quicker. Smooth hand motions and staying in the “visual” zone of the controller were also key to getting the best responses from the controller. As there are no visual indicators to the controller’s sensor boundaries, we would occasionally wander “outside” if we were too animated. Perhaps there will be the ability to link multiple controllers, and create a larger “visual” zone, for those who are prone to exuberant gesticulations?



Below are four more videos (**DNA1**, **DNA2**, **Wave**, **Blocks**) of us using the Leap Motion controller to interact with various virtual objects. All of the demo programs are available for download from the Leap Motion Apps Store. Some of them are intuitive while others were a bit confusing at first. This is to be expected as there is no “standard set” of gestures for many of the typical actions. Couple that with a slight

lag time and a fuzzy interaction boundary, one may sometimes end up with unexpected actions. Still, the Leap Motion controller has improved by orders of magnitude in a relatively short time. We congratulate them on their progress.



With greater functionality and standardized gestures, the Leap Motion controller may even supplant the mouse/trackpad and become the primary interface for interacting with the computer. There are, of course, still quite a bit of work to be done. The **BRICKS demo** shows the limitation of the current 3D UI. We can't really judge when we have grasped the object when trying to pick one up. While there is depth information, there is no tactile feedback.

Microsoft had a huge head start in this space, but they pretty much squandered their lead. Like the Kinect, we were able to interact with virtual objects in a VR environment. Unlike the Kinect, the Leap Motion device works with any computer, not just the Xbox console. Yes, there are three-party libraries which allow the Kinect to "talk" to a PC/Mac, but they are more proof-of-concepts/demo-ware and definitely not ready for prime time. We imagine that it would not be difficult for Microsoft to make the Kinect do all the things the Leap Motion controller now does. The Kinect still has some advantages over the Leap Motion controller (whole body motion, room view, facial data, etc), but without a compelling and superior use-case for the mass market, why bother?

We are happy that Leap Motion kept at it. Leap Motion's V 1.0 release gave everyone a taste of the possibilities, but the hype ran too far ahead of reality. V2.0, when available to the general public, should bring us all closer to the day when we will be able to control the computer with a wave of the hand. We are looking forward to that day.

Related