The Molecular Control Toolkit: Controlling 3D Molecular Graphics via Gesture and Voice

Kenneth Sabir*
Garvan Institute of Medical Research, Australia

Christian Stolte†
CSIRO Computational Informatics, Australia

Bruce Tabor‡
CSIRO Computational Informatics, Australia

Seán I. O'Donoghue§
CSIRO Computational Informatics, Australia
and
Garvan Institute of Medical Research, Australia

Figure 1: Molecular Control Toolkit using Leap Motion and Aquaria.

ABSTRACT

Three-dimensional (3D) molecular graphic systems are widely used in the life sciences, both for research and communication. These systems need to enable a rich set of 3D operations, including three-axis rotation and translation, selection of parts of macromolecules, and the ability to redefine the center of rotation. As a result, graphical interfaces for these systems typically require users to learn complex keyboard and mouse combinations. This can be a significant barrier for new or occasional users, and even for experts, precise control of 3D molecular structures can be challenging. To help address these challenges, we developed the Molecular Control Toolkit to support multiple consumer gesture and voice recognition devices, and provide an API that allows adaption to multiple molecular graphics systems. The toolkit allows intuitive control, almost as if users are directly manipulating 3D objects in their hands. We applied the toolkit to the Kinect and Leap Motion devices, and to the Aquaria molecular graphics system. We did a pilot user study with 18 life scientists to test the resulting system in different scenarios. Overall, users gave quite favorable ratings to using the Kinect and Leap Motion gesture devices to control molecular graphics, even though these devices initially proved less efficient for common 3D control tasks, compared to the more familiar mouse/keyboard. To our knowledge, this is the first toolkit for macromolecular graphics that supports multiple devices with a set of controls sufficiently rich to be useful in the day-to-day work of a broad range of life scientists. The Molecular Control Toolkit and Aquaria can be accessed at `http://aquaria.ws`.

Keywords: User Interface Toolkits, Gestural Input, Pointing, Voice Control, Molecular Graphics, Macromolecules.

Index Terms: J.3 [Life and Medical Sciences]: Biology and genetics—; I.3.6 [Computer Graphics]: Methodology and Techniques—Interaction Techniques

1 INTRODUCTION

Understanding the function of biological systems often requires use of molecular graphics to examine atomic-resolution, three-dimensional (3D) structures of macromolecules, such as proteins, RNA, or DNA, and their interaction with small molecules. The state of the art method for manipulating these structures is via a combination of mouse and keyboard commands. While this is generally adequate, for many infrequent users of molecular graphics it can be a steep learning curve to remember the required mouse/keyboard combinations needed to achieve full 3D control.

Even for expert users, the state of the art is less than perfect; for example, molecular replacement [29] — one of the key tasks in X-ray crystallography — often requires intricate, manual manipulation of 3D molecular models into electron density maps. For this, the standard mouse/keyboard controls are often not sufficient, and experts use a variety of other control devices, such as dials, in addition to making use of a relatively large number of mouse/keyboard combinations and shortcuts.

*e-mail: k.sabir@garvan.org.au
†e-mail: christian.stolte@csiro.au
‡e-mail: bruce.tabor@csiro.au
§e-mail: sean@odonoghuelab.org

IEEE Symposium on Biological Data Visualization 2013
October 13 - 14, Atlanta, Georgia, USA
978-1-4799-1659-7/13/$31.00 ©2013 IEEE

A distinctive aspect of molecular graphics when compared to other 3D visualizations is the defined hierarchy of molecules, ranging from atoms, residues, secondary structural elements, up to polymer chains. The common interactions used for molecular graphics can be summarized as translation, rotation, zooming, selection of items at each level of the hierarchical structure, and changing display modes [16, 19, 33]. In addition, there are many other challenges for molecular graphics arising for new and increasingly complex datasets [25].

In recognition of these challenges, there is a strong tradition in applying the latest developments in Human Computer Interaction (HCI) to molecular graphics. For example, in 1996 Pavlovic et al. created a hand gesture and speech interface to the existing molecular graphics application VMD [27]. While Pavlovic's solution includes using a finger as a three-dimensional (3D) pointer that had to be developed from scratch using multiple cameras, modern HCI consumer interfaces, such as the Leap Motion camera [17], released in July 2013, support this feature out of the box. There are a wide variety of HCI consumer interfaces becoming available now with the competition and new technology driving the device prices under $US100 [8, 15].

Providing six degrees of freedom (comprising of three axes for rotation plus three for translation) has been inherently difficult when mapped to two dimensional touch interfaces such as a mouse or touchscreen. Mappings have relied on the use of meta-keys to change the axis of focus or to toggle between translation and rotation. Recently, multi-touch displays have offered novel methods for interacting with 3D biological data [19] as they can be demonstrated to operate in six degrees of freedom using finger gestures [18] and further research has compared the advantages of using one hand and two hands for multi-touch operation [23]. However, a 2D surface is still arguably less intuitive than using 3D space to manipulate 3D objects.

Control in 3D can be achieved using so-called perceptual input systems, in which computer vision — possibly combined with audio — enables gesture control, without requiring an intermediate device like a mouse, touch-screen or glove [13]. Perceptual input has been explored using non-spatial computer vision to determine hand gestures in prior research and commercial products, such as Artag [4], Sony's EyeToy and Playstation Eye [2]. Control in 3D can be also be achieved using non-perceptual systems [13], such as Nintendo Wii and Sony Move, which require a physical device to manipulate objects in 3D space. For the analysis and control of molecular structures, there have been many studies on non-perceptual input, particularly the use of 3D printed models [25, 6]. Building on this concept, haptic interfaces and augmented reality present physical objects to manipulate with overlay information and have been used as teaching aids for structural biology [30, 5]. While a physical object may provide an immersive experience by utilizing the sense of touch, it does require extra hardware that is not yet commercially widespread for easy adoption by biologists.

The advent of low cost depth cameras has provided novel ways for intuitive 3D gesture detection [3, 28]. Spatial gesture recognition devices such as Kinect and Leap Motion can be used to manipulate objects in 3D space using six degrees of freedom. To date, there has been several preliminary reports using a Kinect to control translation and rotation with molecular graphics systems such as Jmol and PyMOL [12, 1, 14, 24] — and one study that also supported basic speech recognition to activate commands [22]. However to our knowledge, no studies to date support six degrees of freedom.

We are motivated by the hypothesis that molecular control using current gesture devices could be used to create methods for controlling 3D molecular graphics that are more intuitive to learn, and would be preferred in some user scenarios to the current state of the art, namely mouse/keyboard.

To test this hypothesis, we developed the Molecular Control Toolkit, which supports gesture input using multiple consumer devices and provides an Application Programming Interface (API) to allow adaption for molecular graphics applications. In contrast to previous gesture toolkits [20, 31, 35], to our knowledge ours is the first to support multiple consumer devices while providing callbacks through a single interface for applications. Also, in contrast to previous work using affordable gesture recognition devices with molecular graphics [22, 12, 1, 14, 24], ours is the first to provide a sufficiently rich set of operations to be useful for most life scientists during day-to-day use, or while giving presentations. These operations provide a full six degrees of freedom, the ability to redefine center of rotation, integrated voice commands, and the ability to select, deselect, and select by proximity. We integrated our Molecular Control Toolkit into Aquaria (`http://aquaria.ws`), a new molecular graphics system we are developing, branched from the existing open-source product SRS 3D from BioWisdom Ltd, and originally developed at Lion Bioscience AG [26].

We then investigated whether the resulting system allows normal life sciences to use gesture devices as a viable alternative to mouse/keyboard control of molecular graphics. We conducted a pilot user study on first-time users for Kinect, Leap Motion, and Aquaria, to gauge the barriers for adaption and the scenarios for which it may excel.

2 Methods

2.1 Toolkit Overview

The Molecular Control Toolkit was initially designed to support two gesture devices — the Kinect and the Leap Motion — to control and interact with a Java application. A developer adding support to a particular molecular graphics application for using the Molecular Control Toolkit will not be required to access any specific gesture device API directly.

The toolkit monitors the 3D cartesian coordinates of the hands from the gesture devices over time to identify gestures. These gestures map to common operations identified for a molecular graphics application, such as zooming, panning, rotating and selection. The toolkit also supports voice commands when available from the connected device.

2.2 Toolkit Architecture

A molecular graphics application is required to implement the toolkit listeners to receive events from the gesture devices. The Java application used for this research was a macromolecular graphics application we are developing called Aquaria, which is based on SRS 3D [26]. The Aquaria software uses Java3D to obtain hardware acceleration, providing dynamic lighting and rapid rendering response.

Adding each gesture device to the toolkit required developing a specific Connector. The Connectors support a plugin framework, so as more gesture devices come onto the market, new Connectors can be incorporated into the toolkit. Currently, the Molecular Control Toolkit supports two gesture devices — the Kinect and the Leap Motion — to control and interact with a Java application. The Connector processes the events generated from the gesture devices events are processed by the toolkit's device-specific Connector, which then dispatch a message through the toolkit Dispatcher to the application through the use of registered callbacks (Fig. 2). The callbacks are registered by supplying an application specific listener to the Molecular Control Toolkit dispatcher. The Connector interface has methods for initializing the device and sends gesture and voice events to the registered dispatchers. Both devices support gestures for six degrees of freedom covering three translational movements and three rotational movements. The integration with the Aquaria application also allows for changing the center of rotation depending on what is selected.

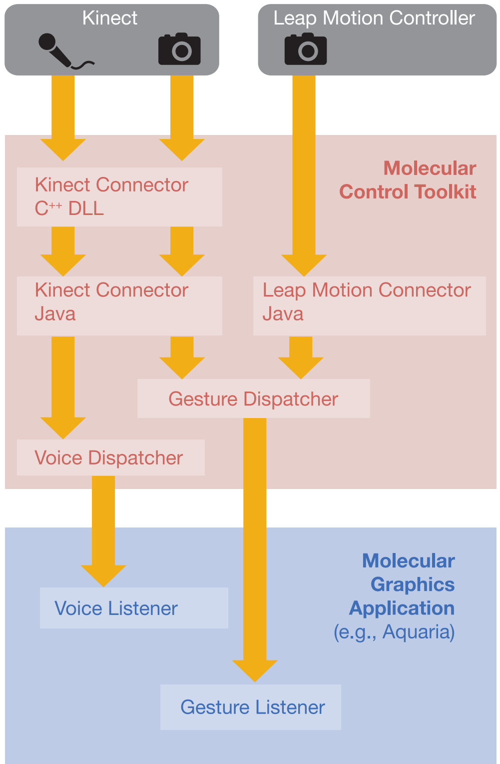

Figure 2: Architecture of the Molecular Control Toolkit

The Kinect supports voice activation, for which a specialized dictionary was created for effectively navigating a macromolecular structure. This dictionary contains domain specific terms such as ligand, chain and residue that will be familiar to a molecular biologist. This allows selection at the various levels of the molecular structure hierarchy, from chains down to atoms. Through the use of voice and gestures, the toolkit supports many of the common interactions required for molecular graphics. Both the Kinect and Leap Motion Connectors support mouse pointer movement by pointing one finger at the screen.

As the Leap Motion Software Development Kit (SDK) supports finger recognition, the toolkit allows the Leap Motion to be usable with one hand as the orientation of the palm is tracked.

2.3 Toolkit Design

Details on the implementation of the toolkit Connectors are in the Supplementary Information, and lists of the listener interface methods to implement are in Supplementary Table 1 & 2.

2.3.1 Connector for Leap Motion gesture recognition

We built a Connector for the Leap Motion device that translated recognized one-handed gestures to 3D controls. The Leap Motion Connector supports moving the mouse cursor directly by controlling it with one finger pointed at the screen and the gesture for poking at the screen is used to simulate the mouse press and release event. This allows the user to navigate the application without using a mouse. Through testing, the action of poking the screen was found to be difficult to master as it was hard to maintain the same X and Y position of the tip of the finger while moving in the Z-axis alone, where the Z-axis refers to the direction towards the computer display. To alleviate this issue, we developed an algorithm that scales down the X and Y translation when the Z velocity is over a given threshold, as given below:

$$S = \begin{cases} T^2/|v_z|^2 & v_z > T \\ 1 & v_z \leq T \end{cases} \tag{1}$$

where S is the scaling factor of the current frame and T is the velocity threshold for the current Z velocity (v_z). If $v_z \leq T$, then the scaling factor is ignored. The scaled x and y positions are calculated as below:

$$\begin{bmatrix} x_{scaled} \\ y_{scaled} \end{bmatrix} = \begin{bmatrix} S \times (x_{new} - x_{old}) + x_{old} \\ S \times (y_{new} - y_{old}) + y_{old} \end{bmatrix} \tag{2}$$

where x_{new} is the current X position of the tip of the finger, and x_{old} is the X position of the tip of the finger for the preceding frame. x_{scaled} is the new value of X after scaling. The same algorithm applies in principle for the y_{scaled} position. An alternative to poking the screen for selection is for the user to hold a wireless mouse or 'presentation clicker' in their non-gesture hand to enable a mouse click for the current mouse cursor. This still allows for the user to position the cursor through the use of gestures. For translation and rotation, the user moves one hand with their fingers outstretched as if rolling an imaginary ball. For example, to rotate in the X-axis the user would roll their hand forward (Fig. 3b). To perform translation the user will move their hand in three-dimensional space to reflect how the object should be translated. For example to zoom in, the user will pull their hand back. The user can disable the tracking at any time by closing their hand into a fist. The Leap Motion Connector will be tracking for translation and rotation events once it detects over three fingers visible on a single hand.

2.3.2 Connector for Kinect gesture recognition

The connector for gestures recognized by the Kinect device was written in C++ and packaged as a dynamic link library (DLL) that communicates with the Microsoft Kinect C++ API directly. The Microsoft Kinect PC API provides skeleton coordinates with three-dimensional Cartesian coordinates for each of the major joints including hand, elbow, shoulders and head. The toolkit's Kinect Connector monitors the hand joints and the torso to judge the distance of the hands away from the body. When the hands pass a threshold of distance away from the body, they are then monitored for rotation and translation movements. To cancel the tracking of the hands, the user can bring their hands back close to their body.

The design of the translation and rotation movements were to imply the user was manipulating a three dimensional elastic object in front of them, hence, the user would spread both hands apart to facilitate translation in the Z-axis, known as 'zooming-in' (Fig. 3n). Translation in the X and Y axis, known as panning, would be moving both hands in the same direction (Figs 3j & 3l) on a vertical plane parallel to the user. For Y-axis rotation, the user would need to push one hand forward and pull the other back (Fig. 3m). Z-axis rotation is moving one hand up and the other down (Fig. 3o). X-axis rotation is similar to Y-axis rotation, however the hands are initially placed one high, the other low, and are then pushed and pulled like Y-axis rotation (Fig. 3k). The Kinect Connector determines whether X-axis or Y-axis is in progress by analyzing whether the orientation of a line connecting the hands is, respectively, vertical or horizontal.

The Kinect API continuously provides frames indicating new updates to the skeleton positioning of the joints. The Kinect Connector compares the current frame to the previous to determine the direction of hand movements to find the appropriate operation, if any. When selecting the operation for the current frame, the Kinect Connector algorithm gives bias preference to the operation that is currently being performed to avoid jitter between operations as the user is moving their arms around. This would make it harder for a

Table 1: Supported voice commands for Molecular Control Toolkit

Voice Command	Action
Reset	Zoom to show whole structure
Hands On	Enable gesture monitoring
Hands Off	Disable gesture monitoring
Wake Up	Enable speech monitoring
Go to Sleep	Disable speech monitoring
Zoom	Zoom to current selection
Select All	Select all residues
Select Up	Select next level in hierarchy
Select Ligand	Select first ligand
Select Proximity	Select atoms 6Å from current selection
Select Residue	Select residue indicated by mouse cursor
Select Chain	Select atoms bonded to current selection
Next Ligand	Select next ligand
Deselect	Deselect all
Rotate	Spins molecule around Y-axis
Copy	Copy current selection to clipboard
Paste	Select subsequences matching clipboard

zoom operation to be accidently regarded halfway as a rotation operation, though not impossible if the rotation movements are strong enough.

The Kinect SDK version 1.6 did not support finger tracking so it required two hands to intuitively handle the three degrees of rotation.

2.3.3 Connector for Kinect voice recognition

The Microsoft Kinect API supports speech recognition through the Microsoft Speech API. This allows for recognition of words from a custom grammar dictionary. The toolkit's Kinect Connector grammar dictionary contains commands for common molecular graphics operations such as 'zoom' and 'select ligand'. The full list is found in Table 1. The grammar dictionary supports multiple words to trigger the same action, for example 'spin' and 'rotate' both trigger the rotate action.

2.4 Pilot User Study

A pilot user study was undertaken to gauge the initial experience and effectiveness of novice users when using Aquaria with the Kinect and Leap Motion Controller, compared with the more familiar keyboard/mouse combination. Each participant was first instructed on the basic operation of each interface and allowed time to get familiar with it.

2.4.1 Subjective Evaluation

Each user was provided a subjective score from 1 to 5 to indicate their background literacy in computing, in molecular graphics and in using gesture devices, such as the Kinect and Wii. Following training, they also provided a score from 1 to 5, indicating how they assessed each interface would suit three different scenarios, with the scenarios being: typical molecular structure research at an office desk, demonstrating a molecular structure in a presentation to an audience, and viewing a molecular in a public interactive installation for a scientific outreach project. A qualitative survey was also taken, in which the participants provided feedback on their experiences and impressions with the different interfaces.

2.4.2 Objective Evaluation

A series of objective quantitative tests were constructed to measure the users ability to use each of the three interfaces, namely the keyboard and mouse, the Leap Motion and the Kinect. There were three tasks to complete for each interface and the time taken to complete each task was recorded. The three tasks were: (1) rotating a given protein to a particular orientation, (2) selecting a particular amino acid residue in the structure, and (3) zooming through the entire structure. Each task started from the same protein structure and same initial orientation.

2.4.3 Testing Environment

The users were instructed vocally with the task requirements and supplied with images of the final rotation and translation required for each test. The first two tests were completed using a 15 inch MacBook Pro Retina, and the Kinect test was completed on the same laptop however for this test the display was shown on two 46 Inch LCD displays stacked vertically. The reason for this was that the Leap Motion did not support a wide radius for moving the hand gestures while the Kinect required the user to stand back from the screen. This could have been alleviated if the testing space supported the leap motion to be further back from the monitors.

2.4.4 User Study Statistics

The non-parametric repeated measures Friedman Test [9, p. 139-146] was used to evaluate differences between control mechanisms for both subjective (self-assessed ability) and objective (time to complete task) measures. A significant result ($p < 0.05$) was followed by post-hoc analysis using the Wilcoxon-Nemenyi-McDonald-Thompson Test [10] to determine where the differences lay. The time measurements were first transformed to log values prior to testing due to the wide variance.

3 Results

3.1 Aquaria Integration

The Aquaria application was modified to create a Leap Motion and/or Kinect Connector based on a command line argument. The Aquaria integration with the Molecular Control Toolkit could be run as a standalone Java application or as a Java Applet running in the browser.

3.2 Pilot User Study

User study participants were obtained by a call for volunteers at the Garvan Institute of Medical Research in Sydney, Australia. Eighteen staff members responded, all currently working full-time in the life sciences. Overall, based on their own self-assessments, participants had high computer literacy, were reasonably experienced using molecular graphics, but had little experience with gesture devices such as Kinect or Wii (Fig. 4).

Each participant took part in the study for a total of 20-30 minutes; during this time they were individually trained then tested on the basic operations for controlling molecular graphics via mouse/keyboard, Leap Motion, and the Kinect.

Figs. 5a, 5b and 5c show how users rated the utility of these three control devices (mouse/keyboard, Leap Motion, and Kinect) in different usage scenarios. For the scenario of normal computer use, the Kinect was judged to be significantly less useful than mouse/keyboard ($p = 0.013$); the Leap Motion was judged to be about equally useful as the mouse/keyboard; however, there was no significant difference between the ratings of the Leap Motion and Kinect. For giving presentations, users considered Kinect, Leap Motion, and mouse/keyboard to be all about equally useful. For interactive installation, the Kinect and Leap Motion was judged to be about equally useful, and both more useful than the mouse/keyboard ($p = 0.000026$ and $p = 0.048$, respectively).

Figs. 5d, 5e and 5f show the time taken for users to complete each of the three tasks (rotate, select, and zoom). For the rotation task, the median time required with the Leap Motion was significantly slower than with the keyboard/mouse ($p = 0.017$); the Kinect

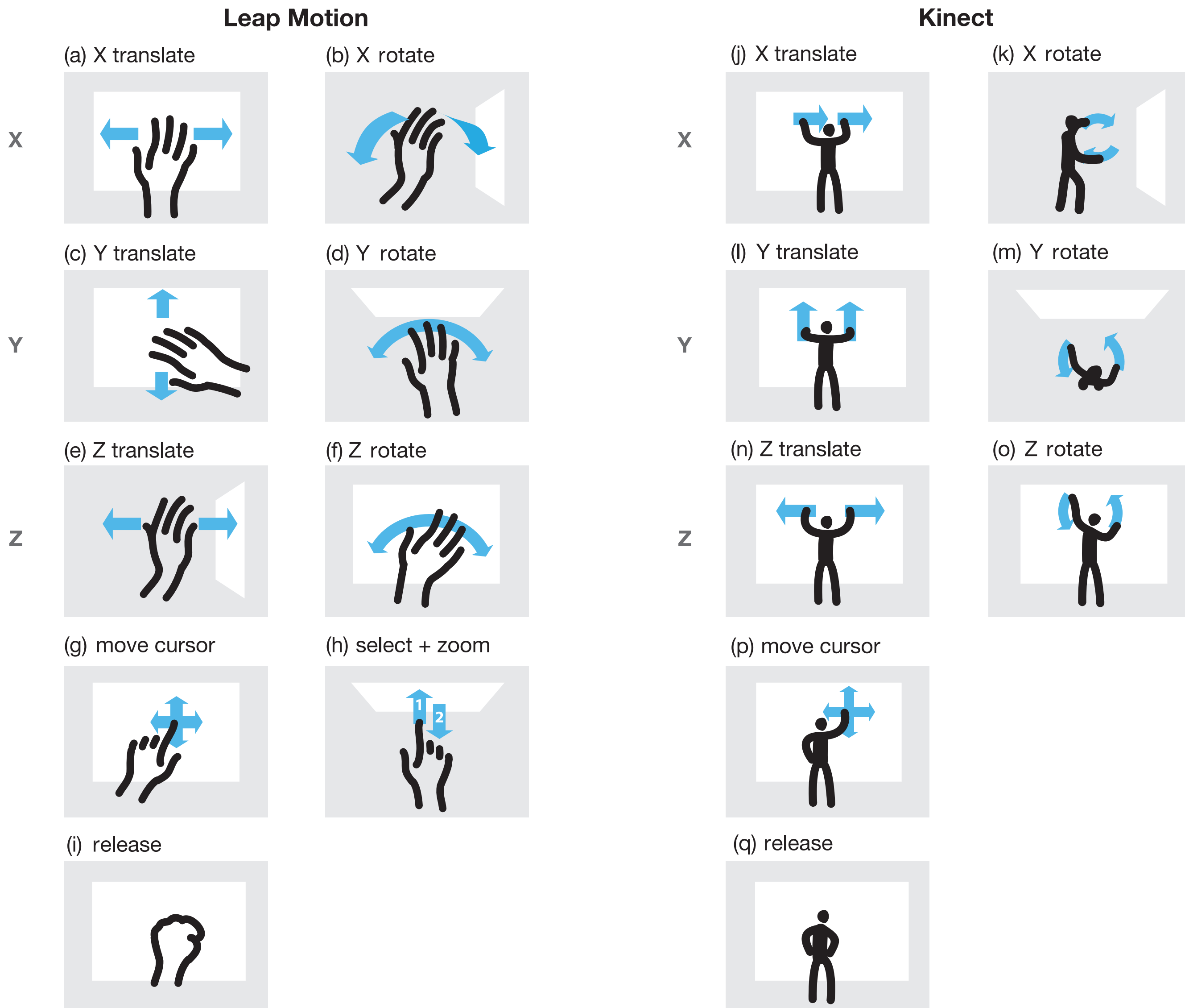

Figure 3: Supported gesture controls for Leap Motion and Kinect. The white rectangle indicates the orientation of the computer display. Making gestures for the toolkit requires one hand for the Leap Motion and two hands for the Kinect. Object selection simulating a mouse click is performed by poking the screen for the Leap Motion and by voice commands for the Kinect.

was also significantly slower ($p = 0.010$). Similarly, for the selection task, compared with the keyboard/mouse, the median times were significantly slower with Leap Motion ($p = 0.0029$) and with Kinect ($p = 0.0000078$). The same pattern was also seen for the zoom task: compared with the keyboard/mouse, the median times were significantly slower with Leap Motion ($p = 0.017$) and with Kinect ($p = 0.0099$).

For each of the three tasks, compared to mouse/keyboard, the Leap Motion was about two to three times slower, and the Kinect was about three to four times slower. The slight difference in performance between the Leap Motion and Kinect results suggest that it may be slightly easier to control molecular graphics with the Leap Motion; however due to rather large variances, these differences were not statistically significant at $p = 0.05$, and a larger sample size would be required to verify this trend.

Full details of the pilot study are provided in the Supplementary Information.

4 Discussion

Overall, the study participants were quite positive about the utility of the Leap Motion and Kinect, even though the time taken to complete tasks was significantly slower than the mouse/keyboard combination, which is currently the method most widely used for controlling molecular graphics. The key factor behind the relatively slow task performance is probably due to the very short training time available in this study, plus the relative lack of experience with similar devices. Indeed, several participants stated that, with more training and familiarity, they would like to adopt, especially, the Leap Motion. This matches with experiences made by one of us (KS), who has now used the devices extensively and can use the Leap Motion faster than the mouse/keyboard for rotation tasks (See Supplementary Video and `http://tinyurl.com/mct-vid`).

As more gesture devices are appearing on the market, application vendors may feel compelled to support each one to not lose market share to competitors. The Molecular Control Toolkit simplifies the

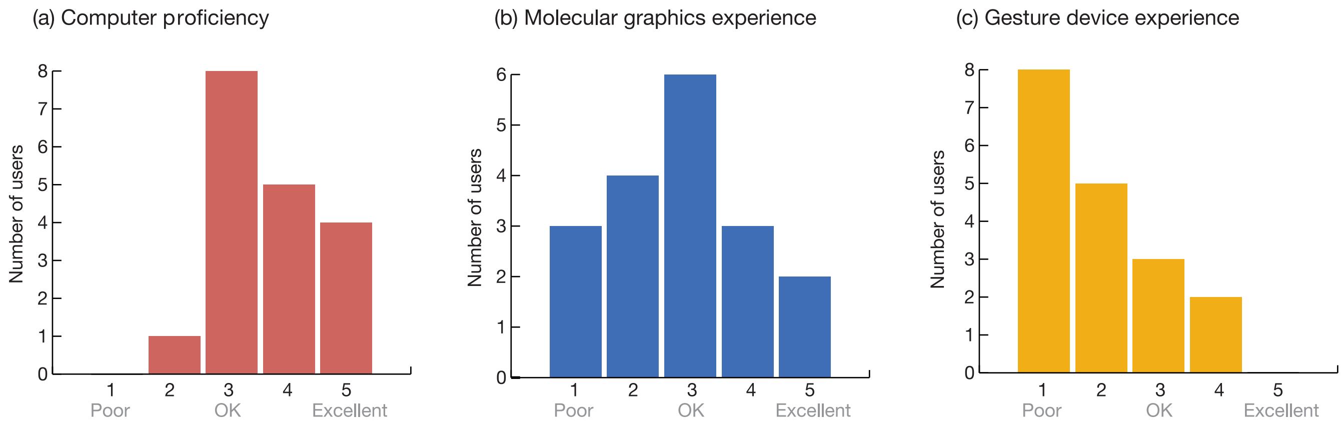

Figure 4: Participants background experience

application vendors task by providing a single gesture interface to support. This gesture interface can provide control for one or two hands, depending on the device capabilities.

Both gesture devices have blind spots where the device would lose track of the hands or fingers due to the devices angle of vision. The Kinect is mounted either below or above the display and scans horizontally for user input. Its blind spot was when the hands were aligned horizontally in front of each other. The Leap Motion device is placed on the table and tracks an eight cubic feet area vertically above it. The Leap Motion's blind spot was when the fingers were aligned vertically. Even though these blind spots are expected once a user understands the design of the devices, they nevertheless provide a barrier for novices to intuitively start using these devices effectively.

One reason for slower results of the Kinect on the selection test was participants used the voice command 'Select Residue' once the cursor was over the desired residue. The voice command has an inherent delay as it waits a moment to determine if any more words will follow.

Although the Leap Motion monitors a smaller space than the Kinect, it provides a greater resolution that allows for finger tracking with its accuracy recently evaluated and contrasted to the Kinect [34].

A desired outcome for using gesture devices for manipulating three-dimensional objects would be to create an intuitive interface to simulate operating physical objects in the real world. However, after comparing the user study times of novice users to that of an expert user (as seen on the Supplementary Video) it is clear that the gesture devices are not intuitive enough to avoid a learning curve altogether.

We witnessed it was difficult for the hand to switch between the visual channels of analog movement during translation in a three-dimensional space to a digital action, such as the poke gesture, to simulate a mouse click. The algorithm we produced to ease the X and Y translation when performing the poke gesture on the Leap Motion improved the ability to select what was intended, however it did create a glitch post-gesture when the cursor would suddenly return to the real X and Y translation. This is a minor issue that could be solved with a smooth post-gesture interpolation to return the cursor to the real X and Y position, however, the problem of switching visual channels would remain. One solution could be to allow the leap motion to be operated with one hand with the intention of using the other hand to manipulate the keyboard. This would require further testing to determine its effectiveness.

Novice users had trouble transitioning between motions and disengaging. This was more evident with the Kinect, which required the user to evenly bring their hands back towards their body to disengage. The Leap Motion's action of closing the hand for disengaging was easier to perform after practice, however it was not completely intuitive for the novice user, as some commented that they felt like gripping the object with a closed fist to drag the object. The new Kinect SDK version 1.7 supports a gesture for recognizing a closed fist to represent translation by dragging [21]. A technical advantage of tracking when the fingers are spread is it was easier for the Leap Motion to determine the hand orientation when it was open.

The pilot user study was not intended to comprehensively assess the effectiveness of the various interfaces, as the user group were novices at gesture devices, yet typically had over twenty years experience in the mouse/keyboard interface. It was the first time for many users using any gesture device, and the first time for all users for using the Leap Motion.

We found that in the user study the tasks were generally two to four times slower on the gesture devices than the mouse/keyboard, however given the very short training time, and the fact that the users had very little prior exposure to gesture devices, the results are encouraging. Indeed, most users rated the gesture devices quite highly (Figs. 5a, 5b, and 5c) and many felt that with more training these devices would be the preferred device for controlling molecular graphics, especially in presentation and installation scenarios. Also encouraging was the enthusiastic response from the freeform feedback comments, with most biologists and bioinformaticians in the pilot study wanting to examine this technology further (Supplementary Table 7).

4.1 Potential Biases in User Study

The pilot study was not to gauge between which is the superior input device, but to gain some preliminary understanding to potential barriers for adaptation for biologists, to understand intuitiveness of one-handed versus two-handed interaction, and to get an insight on the potential scenarios for which a gesture device would be useful. As the Kinect portion of this study involved two impressively large (46 inch) LCD screens this may have influenced the user experience, when compared to the 15 inch laptop used for the other two interfaces.

4.2 Future Work

Once the framework for processing speech commands was developed, it was straightforward to map new speech commands to actions in Aquaria. While the Molecular Control Toolkit has the ability to change representation from residues to atoms through voice commands, the vocabulary could be expanded to include changing color schemes to show homology or chains, and other representations such as surface rendering.

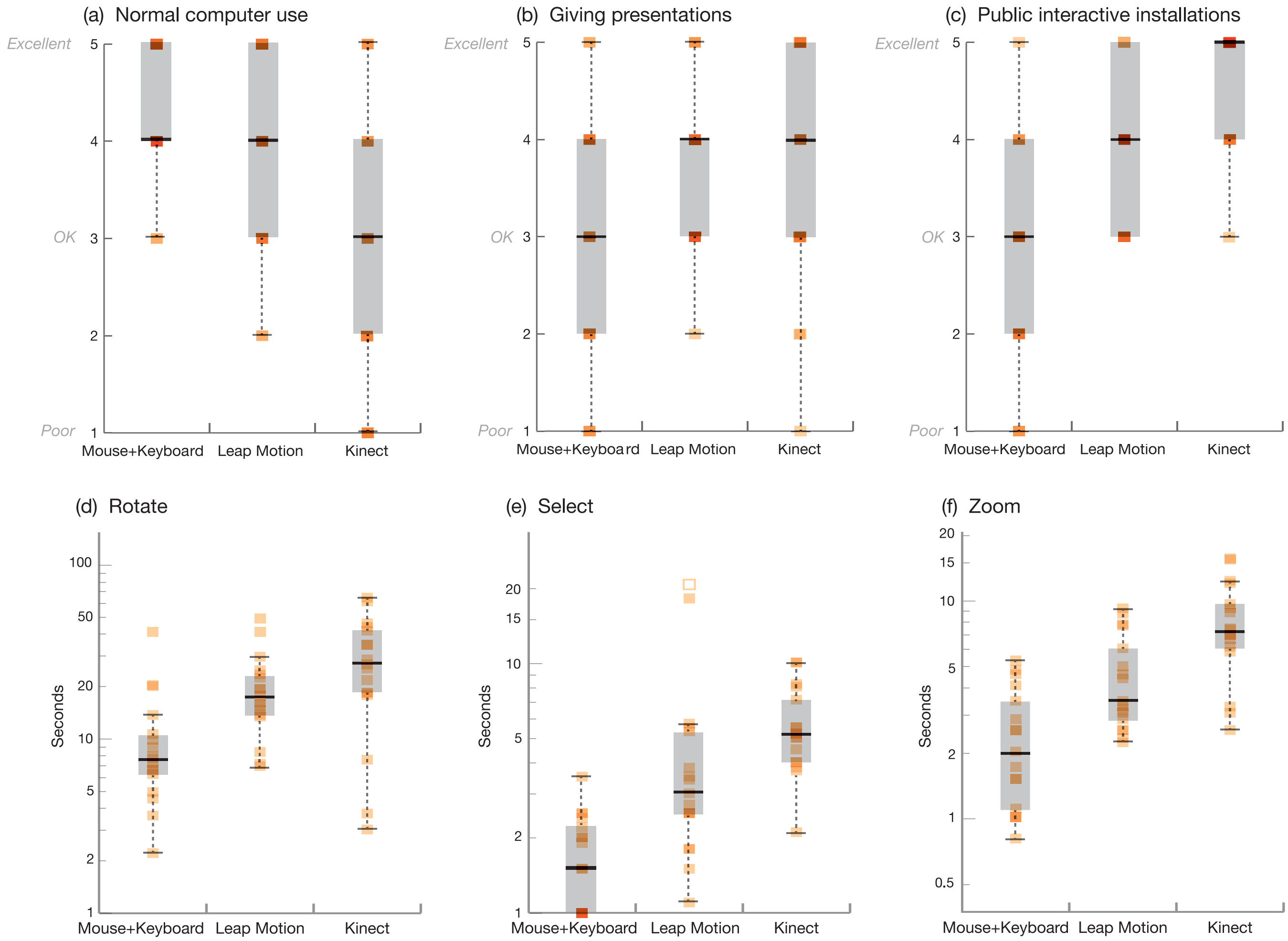

Figure 5: Pilot study comparing mouse/keyboard, Leap Motion, and Kinect controls for molecular graphics. Plots (a), (b) and (c) compare subjective assessments of how the devices are suited for different usage scenarios. For normal computer use, the mouse/keyboard and Leap Motion were both similarly rated, while the Kinect was rated lower. For presentations, all three devices were rated similarly. For public interactive installations, the Kinect and Leap Motion were rated similarly, and both better than mouse/keyboard. Plots (d), (e) and (f) compare the times taken to perform three common molecular graphics tasks. The median times for the Leap Motion and Kinect are significantly slower than for mouse/keyboard, and the median times for Leap Motion are consistently better than for Kinect. Thick horizontal black bars indicate median values; shaded regions show the interquartile ranges (estimated span of 50% of data); dotted lines show the interdecile ranges (estimated span of 99.3% of data). The red squares represent high frequency of user scores and the orange squares represent low frequency.

While the Molecular Control Toolkit has specific commands to deal with macromolecular structures, it could also be modified to support any three-dimensional non-molecular application using the same gestures. The dictionary used for the voice input could be modified to provide relevant commands for a particular domain. Another future improvement for Molecular Control Toolkit would be to support other new gesture devices coming on the market, such as the Creative Interactive Gesture Camera, and provide application adapters for popular molecular graphics packages such as Jmol and PyMOL [11, 7, 32].

Given that the Molecular Control Toolkit supports simultaneous use of multiple gesture devices, the blind-spots that inhibited intuitive use for the novice users could be alleviated by combining the gesture tracking algorithms for both the Leap Motion and Kinect at the same time. The Leap Motion will monitor the hands with a vertical camera angle and the Kinect will monitor using a horizontal angle. This could create a smoother interaction for the user by reducing glitches and blind-spots, which in turn will allow new gestures such as continuous rolling of both hands in a circular manner.

For designing a user study in the future, it would be interesting to test expert users across all devices to compare on the test times and for the device preference for various scenarios. Another study could be to work with novice users over a longer period of time to measure the rate of effectiveness and intuitiveness of using the devices.

5 Conclusion

Together with Aquaria, the Molecular Control Toolkit is the first system designed to enable life scientists to control molecular graphics using gesture and voice for multiple devices; it supports the Leap Motion and Kinect — two very affordable and practical devices. The pilot user study presented in this work suggests that such devices are already quite useful and usable for many life scientists: as these devices become cheaper and more widely available, it is likely that a large number of scientists will soon begin to use them to control molecular graphics systems, either as part of their research, while giving presentations, or for creating interactive installations. Thus, we anticipate that the Molecular Control Toolkit presented here is likely to be widely adopted; to encourage others to extend the toolkit to other devices and molecular graphics systems, the source code is released under an open source license and is available at `http://bit.ly/molecular-control-toolkit`. The full working system, integrated with the Aquaria molecular graphics system (also open source), is available at `http://aquaria.ws`.

ACKNOWLEDGMENTS

We are grateful to Leap Motion for providing early access to the device and developer toolkit. The authors are not official Leap Motion or Microsoft representatives and are speaking not as a spokesperson of Leap Motion or Microsoft. We also thank all the participants of the user study.

REFERENCES

[1] R. Alsayegh and H. Makatsoris. Integration of Molecular Dynamic with Computer Vision. *ResCon12, School of Engineering and Design, Brunel University*, pages A8–A9, 2012.

[2] M. Barlow. Game Led HCI Improvements. *Business, Technological, and Social Dimensions of Computer Games: Multidisciplinary Developments*, 2011.

[3] K. Biswas and S. Basu. Gesture Recognition using Microsoft Kinect. In *Automation, Robotics and Applications (ICARA)*, pages 100–103, Dec. 2011.

[4] M. Fiala. ARTag, An Improved Marker System Based on ARToolkit. *NRC Publications Archive*, 2004.

[5] A. Gillet, M. Sanner, D. Stoffler, and A. Olson. Tangible interfaces for structural molecular biology. *Structure (London, England : 1993)*, 13(3):483–91, Mar. 2005.

[6] T. Herman, J. Morris, S. Colton, A. Batiza, M. Patrick, M. Franzen, and D. S. Goodsell. Tactile teaching: Exploring protein structure/function using physical models. *Biochemistry and Molecular Biology Education*, 34(4):247–254, Nov. 2006.

[7] A. Herráez. Biomolecules in the computer: Jmol to the rescue. *Biochemistry and Molecular Biology Education*, Nov. 2006.

[8] H. Hodson. Leap Motion hacks show potential of new gesture tech. *New Scientist*, 218(2911):21, Apr. 2013.

[9] M. Hollander and D. Wolfe. *Nonparametric statistical methods*. New York: John Wiley & Sons, 1973.

[10] M. Hollander and D. Wolfe. *Nonparametric statistical methods*. New York: John Wiley & Sons, 2nd edition, 1999.

[11] Intel. The Intel Perceptual Computing SDK 2013: Production Ready! `http://www.intel.com/software/perceptual`.

[12] I. Jamie and C. McRae. Manipulating molecules: Using kinect for immersive learning in chemistry. In *Proceedings of the Australian Conference on Science and Mathematics Education*, Sept. 2011.

[13] M. Karam and M. Schraefel. A taxonomy of gestures in human computer interactions. *Technical report, Electronics and Computer Science, University of Southampton*, pages 1–45, Aug. 2005.

[14] KinVi 3D. Interactive Molecular Manipulation Using Kinect. `http://tinyurl.com/kinvi3d`, 2013.

[15] O. Kreylos. Low-Cost VR. `http://tinyurl.com/LowCostVR`, 2008.

[16] O. Kulyk and I. Wassink. Getting to know bioinformaticians: Results of an exploratory user study. In *In Proceedings of BCS HCI 06 Int, Workshop on Combining Visualisation and Interaction to Facilitate Scientific Exploration and Discovery*, page 37, Sept. 2006.

[17] Leap Motion Inc. Leap Motion. `http://www.leapmotion.com`, 2013.

[18] J. Liu, O. K.-C. Au, H. Fu, and C.-L. Tai. Two-Finger Gestures for 6DOF Manipulation of 3D Objects. *Computer Graphics Forum*, 31(7):2047–2055, Sept. 2012.

[19] J. Logtenberg. Designing interaction with molecule visualizations on a multi-touch table. *Research for Capita Selecta, a Twente University course*, Dec. 2008.

[20] K. Lyons, H. Brashear, and T. Westeyn. GART: the gesture and activity recognition toolkit. *Human-Computer Interaction. HCI Intelligent Multimodal Interaction Environments*, July 2007.

[21] Microsoft. The latest Kinect for Windows SDK is here. `http://tinyurl.com/windows-kinect-sdk`.

[22] R. Moncrief and W. Gobber. A motion sensor interactive interface for viewing and manipulating protein structural data in 3D. *ACM SIGGRAPH 2012 Posters*, Aug. 2012.

[23] T. Moscovich and J. F. Hughes. Indirect mappings of multi-touch input using one and two hands. *Proceeding of the twenty-sixth annual CHI conference on Human factors in computing systems - CHI '08*, page 1275, Apr. 2008.

[24] N. O'Boyle. Jmol-kinect. `http://tinyurl.com/jkinect`, 2011.

[25] S. I. O'Donoghue, D. S. Goodsell, A. S. Frangakis, F. Jossinet, R. a. Laskowski, M. Nilges, H. R. Saibil, A. Schafferhans, R. C. Wade, E. Westhof, and A. J. Olson. Visualization of macromolecular structures. *Nature Methods*, 7(3 Suppl):S42–55, Mar. 2010.

[26] S. I. O'Donoghue, J. E. W. Meyer, A. Schafferhans, and K. Fries. The SRS 3D module: integrating structures, sequences and features. *Bioinformatics*, 20(15):2476–8, Oct. 2004.

[27] V. Pavlovic, R. Sharma, and T. Huang. Gestural interface to a visual computing environment for molecular biologists. *Automatic Face and Gesture Recognition*, pages 30–35, Oct. 1996.

[28] Z. Ren, J. Yuan, and Z. Zhang. Robust hand gesture recognition based on finger-earth mover's distance with a commodity depth camera. *Proceedings of the 19th ACM international conference on Multimedia*, Nov. 2011.

[29] M. Rossmann. Molecular replacement-historical background. *Acta Crystallographica Section D: Biological Crystallography*, 57(10):1360–1366, Oct. 2001.

[30] G. Sankaranarayanan, S. Weghorst, M. Sanner, A. Gillet, and A. Olson. Role of haptics in teaching structural molecular biology. *11th Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems, 2003. HAPTICS 2003. Proceedings.*, pages 363–366, Mar. 2003.

[31] T. Schiphorst, R. Lovell, and N. Jaffe. Using a gestural interface toolkit for tactile input to a dynamic virtual space. *CHI '02 Extended Abstracts on Human Factors in Computing Systems*, Apr. 2002.

[32] Schrodinger LLC. The PyMOL Molecular Graphics System, Version 1.3r1, 2010.

[33] D. Vyas, O. Kulyk, and P. V. D. Vet. Structures of life: the role of molecular structures in scientists' work. In *Proceedings of the 15th European conference on Cognitive ergonomics: the ergonomics of cool interaction*, Sept. 2008.

[34] F. Weichert, D. Bachmann, B. Rudak, and D. Fisseler. Analysis of the accuracy and robustness of the leap motion controller. *Sensors*, 13(5):6380–93, Jan. 2013.

[35] T. Westeyn and H. Brashear. Georgia tech gesture toolkit: supporting experiments in gesture recognition. *Proceedings of the 5th international conference on Multimodal interfaces*, pages 85–92, Nov. 2003.