Article

Molecular Rift: Virtual Reality for Drug Designers

Magnus Norrby,[†,‡] Christoph Grebner,[†] Joakim Eriksson,[§] and Jonas Boström[*,†]

[†]Department of Medicinal Chemistry, CVMD iMed, AstraZeneca, S-43183 Mölndal, Sweden
[‡]Department of Computer Sciences and [§]Department of Design Sciences, Lund University, S-223 62 Lund, Sweden

Ⓢ *Supporting Information*

ABSTRACT: Recent advances in interaction design have created new ways to use computers. One example is the ability to create enhanced 3D environments that simulate physical presence in the real world—a virtual reality. This is relevant to drug discovery since molecular models are frequently used to obtain deeper understandings of, say, ligand—protein complexes. We have developed a tool (Molecular Rift), which creates a virtual reality environment steered with hand movements. Oculus Rift, a head-mounted display, is used to create the virtual settings. The program is controlled by gesture-recognition, using the gaming sensor MS Kinect v2, eliminating the need for standard input devices. The Open Babel toolkit was integrated to provide access to powerful cheminformatics functions. Molecular Rift was developed with a focus on usability, including iterative test-group evaluations. We conclude with reflections on virtual reality's future capabilities in chemistry and education. Molecular Rift is open source and can be downloaded from GitHub.

INTRODUCTION

Molecules of the simplicity of dopamine or the complexity of proteins and the famous DNA double helix are all three-dimensional (3D) objects. Obtaining knowledge about the 3D character of such small and large molecules is important in many approaches contributing to drug discovery. One apparent way to achieve this is to visualize and interact with 3D models of the molecular objects of interest—either in the real world or in a virtual reality.

The concept of molecules as 3D objects emerged during the 19th century and was independently proven by Le Bel and van't Hoff around 1870.[1] Experimental techniques, especially X-ray crystallography, have provided models of biological molecules (e.g., penicillin and cholesterol) for almost a century. The first X-ray structure of a protein (myoglobin) was solved in the late fifties.[2] X-ray crystallography has since served as a source and inspiration for the construction of models facilitating understanding and leading to many important discoveries.[3] For example, the use of physical models was imperative for Watson and Crick in the race towards creating the first accurate 3D model of the DNA structure.[4] With the rise of computers, the focus switched from physical representations to computer generated models. An important innovation here was the ability to display stereoscopic 3D views. The TAMS (Teaching Aids for Macromolecular Structure) is a notable example of an early such project.[5] They used two polarized slide projectors mounted on a cardboard box to simulate stereoscopic 3D. Much has happened since.

Nowadays, visualizing and interacting with molecular models is a significant part of a drug designer's day job. Some common examples are (i) inspecting small molecule conformations obtained from a virtual screen, (ii) using conformational information to elucidate experimental data, such as solving X-ray crystallography structures or chemical shifts in NMR spectra, and (iii) identifying important structural features and molecular interactions in ligand—protein X-ray structures (be it experimentally determined or docked) to form design hypotheses and to obtain a deeper understanding of the system at hand. The list can go on. In short, the ability to visualize molecules is key to drug design and, of course, also in life science as a whole.

A range of stand-alone molecular viewing computer programs have been developed over the years. Some examples are Rasmol,[6] PyMol,[7] VIDA,[8] VMD,[9] and Chimera.[10] As of recently, web-based systems such as Jmol,[11] AstexViewer,[12] iView,[13] and the OpenEye 3D Web Toolkit[14] have gained traction. A large part of their appeal is the simplicity with which they integrate to build rich dynamic web sites. Most stand-alone and web-based molecular viewers support 3D-stereoscopic effects. The stereo techniques can be divided into three main classes: passive 3D (glasses that chromatically distort images[15,16]), active 3D (using shutter glasses and high frequency image representations[17]), and autostereoscopic 3D where no other headgear is required (such as the parallax
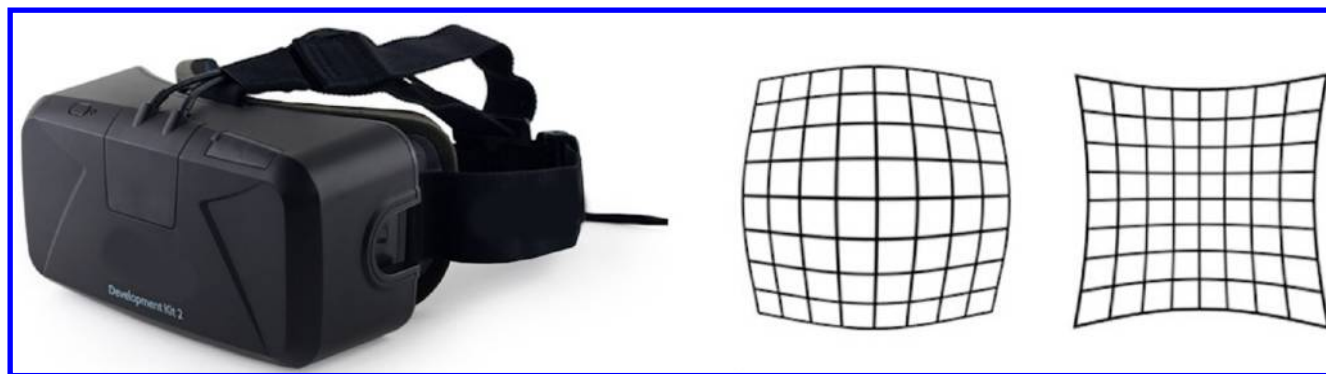
Figure 1. Oculus Rift goggles (left) used to create to the virtual reality environment in Molecular Rift. Oculus Rift applies a barrel (middle) and pincushion (right) distortion to the image, in combination with a wide FOV, to create the VR experience.

barrier) in iView.[13] Although aiming for essentially the same effect the underlying theories vary significantly, as well as their implementation, performance, and the user experience (UX).

Without a doubt, current molecular viewers (using these techniques) can generally provide adequate illusions of 3D stereo. But there are drawbacks. For example, the passive stereo approaches strip the models of their original color properties and thus hamper the analysis. Autostereoscopic methods suffer from lower resolution, and the shutter glasses required for the active 3D stereo are by far the most expensive appliance needed. But, most importantly, a full immersion into a highly visual environment and a real 3D effect is lacking for the available methods. This can be alleviated by implementing virtual reality (VR) techniques.[18] It must be emphasized that a virtual reality experience is very different from using standard molecular visualizers. In the VR environment objects such as atoms and bonds have a location in 3D-space relative to the user's position. This is not an illusion but an effect. You feel like you are really there. It should be mentioned that virtual reality is not new to life sciences (e.g. medicine, therapy) and gaming, but its improved performance and affordability certainly is.

Several different techniques can be used to implement a virtual reality, ranging from the simple Google Cardboard goggles[19] to sophisticated CAVEs (cave automatic virtual environment) requiring dedicated rooms.[20] Within chemistry and drug discovery, MolDRIVE and its Reality Cube is an exciting CAVE example of a highly interactive VR system.[21] The introduction of high-quality but relatively inexpensive head mounted VR devices such as Oculus Rift (Figure 1) has opened up new opportunities for creating mobile stereoscopic 3D visualizers, which can be easily used at home or in office spaces. Oculus Rift creates a stereoscopic 3D using two lenses and a LCD display. In brief, the VR effect is created by displaying two separate images, with slightly different points of view. The VR experience is enchanced by applying two different distortion techniques to the field of view (FOV). The pincushion distortion (Figure 1) makes the display appear to occupy a wider FOV. The (unwanted) distorted image is subsequently treated with a barrel distortion (Figure 1) to cancel out the pincushion effect. The effect of increasing the FOV, without introducing any obvious distortions, can be observed by the user in this manner. That is, the lenses make it possible to accommodate a large FOV on a small screen. The view is 3D without distortions; straight lines look straight, just like in real life.

A lightweight version of the Oculus Rift technology has been implemented in the web-based molecular viewer iView.[13]

However, the 3D stereo effect in iView merely offers an improved stereoscopic effect, without essential functionalities such as head tracking or advanced interaction techniques. Head tracking means that you can turn your head and look around and the view follows, as in real life. The prototype program Oculus Crystal[22] can display different crystal packings in an interactive environment and thus offers a full VR effect. However, Oculus Crystal is limited to a few predefined scenes. The head-tracking, allowing the user to look in all directions, and the wide field of view are key components for providing a real VR experience.

An important aspect when implementing a VR is the human−computer interaction (HCI). That is, to integrate the human body into the virtual experience in a natural way. Wearing VR-goggles, such as Oculus Rift, limits the use of many external input devices (e.g. mouse, keyboards, and touchpads) since the goggles are not transparent. As a consequence, new alternative interaction methods present an opportunity. Kinect v2,[23] developed by Microsoft for both Windows and Xbox platforms, is such a motion sensing device specifically tailored to recognize gestures. Kinect v2 offers high flexibility and is equipped with RGB color and infrared (IR) sensors to capture gestures. Although first marketed for gaming, its competitive pricing and ease of access has led to a spread into scientific fields. As an example, Alnowami et al. recently showed that Kinect sensors could be successfully used for gait tracking, an important indicator of health.[24] Another life science application is the use of Kinect sensors in exercise games to treat people suffering from Parkinson's disease.[25]

Finally, both Oculus Rift and MS Kinect v2 offer free software development kits (SDK's)[26,27] compatible with the Windows game engine Unity 5,[28] through wrappers. SDK's are prewritten modules (or libraries or packages) available at your fingertips intended to extend the capabilities of the tools. For example, packages related to image processing, changing colors, interacting with hardware, keeping track of a user's location, performing complex numerical analysis, rendering text on the screen, and so forth are available providing an excellent development platform.

The overall goal of this project was to develop a versatile tool providing the functionalities of a high-end molecular viewer together with a real VR experience to bring virtual reality into the daily work of drug designers. In addition to using the tool in drug design projects we see opportunities to use it to inspire and motivate people to learn more about life sciences (e.g. for educational purposes). Molecular Rift was developed with an emphasis on user friendliness to improve the tool and thus the

Figure 2. Start menu of Molecular Rift.

user experience (UX). For this purpose, an iterative design and development cycle including several focus groups was used to test and evaluate new functionalities and to provide instant feedback to guide further development.

IMPLEMENTATION

Development, Infrastructure, and Features of Molecular Rift. Molecular Rift was developed in an iterative fashion, where each iteration cycle consisted of a design and development step followed by a focus-group evaluation. This ensured that the UX could be monitored and that the usability of the tool was optimized toward the requirements of typical end-users: drug designers.

The development was performed in Windows 8.1 using the game engine Unity 5 Personal Edition and MonoDevelop as the primary editor. Kinect for Windows SDK v2.0[23] and its application programming interface (API) were utilized for the gesture recognition. The Oculus Rift Developer Kit 2 (DK2) with the Oculus Rift 0.4.4-beta SDK and Windows Runtime[26] were used to create the virtual reality environment. The open source cheminformatics tool-kit Open Babel[29,30] was integrated with Molecular Rift. All programming was done in C# due to its compatibility with the Windows environment.

Molecular Rift is divided into two parts: one traditional start menu (no VR) and one VR environment for the visualization and interaction with molecular models. On initialization, the start menu is launched within the graphical user interface (GUI). Options for typical settings, such as atom and bond representations, labels, color schemes etc. are shown (Figure 2) and subsequently applied. The main menu is controlled in a conventional fashion, by keyboard and mouse, facilitating the typing of text, which is still a challenge in most VR environments. Upon loading a file containing 3D coordinates of a molecular model, either by file browsing or remote

download from the protein data bank (PDB),[31,32] the VR is started and the control is shifted to gesture recognition. A summary of the visualization options implemented in the tool and thus the Table 1. Moreover, an in-game VR menu for common commands is available, see Figure 3. Keyboard shortcuts were implemented to improve the usability (Table 2). More details about implemented gestures and commands are given in the following sections.

Table 1. Summary of Molecular Rift features

category	feature
file input formats	PDB, MOL2, and SDF
structure coloring	various color schemes
primary structure rendering	lines, sticks, ball-and-sticks, and CPK
secondary structure rendering	ribbons and alpha-trace
residue labeling	three-letter code and sequence number
protein surfaces	van der Waals
hydrogen bonds	intermolecular (ligand–protein)
ions and waters	yes (ball-and-stick)
virtual reality effect	Oculus Rift
human–computer interaction	hand motions (MS Kinect v2), mouse, and keyboard
in-game manipulation modes	rotation, zooming, movement, translation, and center view
cheminformatics tool-kit	Open Babel (C#)
background color	black, white, gray, and space
pharmacophore objects	H-bond donors and acceptors, rings, positive and negative charges, lipophilic, excluded volumes
hydrogens display	all, none, or polar only
force fields	mmff94, uff, and ghemical

Figure 3. In-game virtual reality menu of Molecular Rift which allows the user to modify the general appearance of, for example, ligand–protein complexes. Different options include showing and hiding the protein atoms, ribbons, C-alpha trace, water molecules, ions, or hydrogen bonds. Two in-game hands are shown. While the in-game menu is active, a cursor is displayed steered by the right-hand.

Table 2. Overview of Keyboard Commands Implemented in Molecular Rift

action	key
move forward	W
move left	A
move right	D
move back	S
move up	T
move down	G
zoom mouse	wheel
toggle menu	N
toggle ribbons	R
center view	V
new focus target	P
display details	space
screenshot	L
reset center	X
lock atom selection	Z

Graphical Representation of the Molecular Models—3D VR Engine. The Unity 3D engine was used to develop the graphical environment for the VR experience. Unity is a cross-platform game engine offering a flexible and powerful development platform for creating interactive 3D experiences.[28]

Rendering structural features of proteins and ligands was done using standard geometric elements available via Unity, as well as using custom-made objects. Typical molecular representations such as lines, ball-and-sticks, sticks, and space-filling (CPK) for atomic representations, and ribbons and C-alpha traces for protein secondary structures were implemented. The interpretation of atom and bond types, ligands, and proteins, as well as secondary structures were performed using standard Open Babel functions; e.g. OBConversion to read PDB, MOL2, and SDF input files. The OBResidue and its GetResidueProperty and GetAtomProperty functions were used to identify proteins, ligands, ions, and water molecules. A detailed description for the various visualization representations follows.

Line representations only render bonds between atoms; atoms are thus not specifically visualized. The line representation was created using Unity's standard cylinder object. Accordingly, the Euclidean distance between two bonded

atoms was calculated and the cylinder was positioned at the center. The Unity lookAt function was subsequently used to rotate the cylinder ends toward the bonded atoms positions. The (single-) bond creation process is shown in Figure 4.

Figure 4. Three steps included in bond creation: Rendering the bond (left), scaling the bond according to atom distance (middle), and rotation toward the bonded atom positions (right).

Various bond types (single, double, triple, and aromatic) were assigned according to Open Babel Bond definitions and rendered in an identical fashion, although by four different graphical objects. The latter three (double, triple, and aromatic) using tailor-made objects. The same technique as for line representation was used for the stick representation, with the only exception of using Unity's standard capsules instead of cylinders, making the bonds appear thicker (and without sharp edges). The ball-and-stick representation employed the line representation combined with the Unity's standard sphere object to render the bonded atoms. The size of the sphere objects were scaled to match the Bondi[33] atomic radii and colored according to atom type (several color schemes are available). Finally, atoms can be represented by a solid sphere with a constant (large) radius in CPK representations. Illustrations of the four representations are shown in Figure 5.

Protein secondary structures are often visualized by C-alpha traces or ribbon-like representations in standard molecular viewers. Secondary structure custom objects were thus created
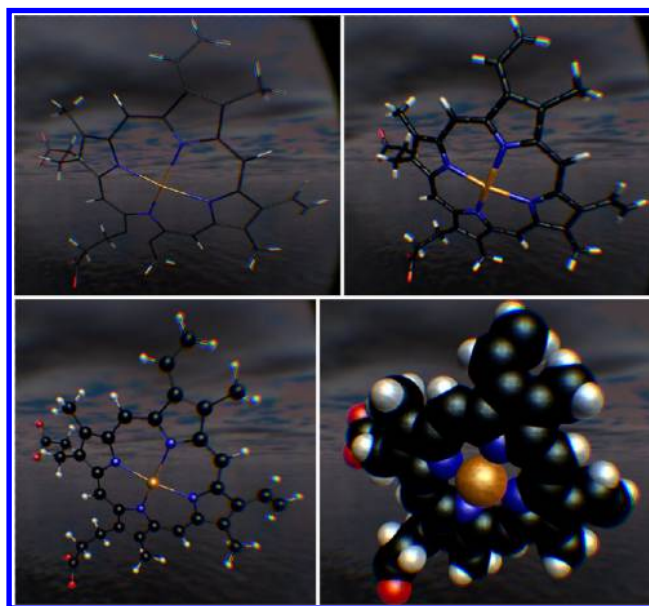

Figure 5. Different representations available for ligand and atoms: lines (top-left), sticks (top-right), ball-and-sticks (bottom-left), and CPK-spheres (bottom-right). PDB code: 2dhb.

using Unity. The ribbon-shape objects were rendered by interpolating the atom positions using mirrored Bézier splines. In order to achieve smooth curves the control points are mirrored in each other, thus making the acceleration along the curve continuous. The secondary structure ribbon creation process is described in Figure 6. The OBMol GetTorsion
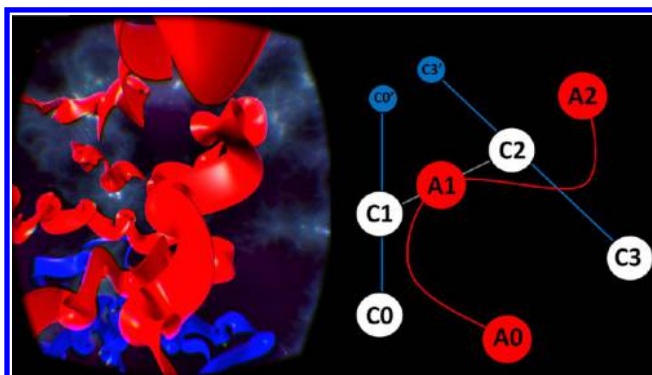

Figure 6. Example of ribbon display as a secondary structure visualization (left) and the secondary structure creation process (right). In order to achieve a smoother curve the control points are mirrored in each other (C1 into C2, C3 into C3′, and C0 into C0′), thus making the acceleration along the curve continuous.

function was used to identify alpha-helices (as defined by the torsions for two sets of four consecutive atoms (i.e., the φ- and ψ-angles) being in the range of 95−115°) when parsing PDB files. When an alpha-helix was encountered a slightly different protocol was used. That is, only alpha carbons were used as control points to create a smooth-looking spiral, as compared to the standard ribbon depiction. The ribbon mesh was created using a modified version of StarManta's TubeRenderer.[34] To obtain an elliptic, almost flat shape the original tube was modified by dividing the y-axis value by an arbitrary factor of 10. The same approach as for the alpha-helix of ribbons was used to visualize the C-alpha trace, although using a small radius, making the representation to appear as a line.

The ability to highlight intermolecular hydrogen bonds is an important feature when analyzing ligand−protein complexes and was thus implemented. Open Babel OBAtom's functions (IsHbondAcceptor and IsHbondDonor) were used to identify candidate atoms and their interatomic distances were subsequently evaluated. Hydrogen bonds (H-bonds) are rendered between all pairs of H-bond acceptor and donors within a typical H-bond distance (<3.5 Å), using Unity's particle system.[35] The particle system creates a random emission of tiny objects in a specified shape and direction. The effect of an electrical bond is achieved by placing one cylindrical particle system on each of the heavy atoms involved in the hydrogen-bond and orienting the emission toward the other atom (Figure 7). The hydrogen-bonds between protein and ligands, water molecules, and ions can be toggled on and off using the in-game menu.

Various types of free text options were implemented. For example, amino acid alpha-carbons can be labeled with their residue names as the standard three letter codes describing the amino acid type. The three-letter code is concatenated with the sequence number (separated by a dash) representing the amino acid's position in the protein chain; see Figure 8. The labels were retrieved using Open Babel (OBResidue) and imple-
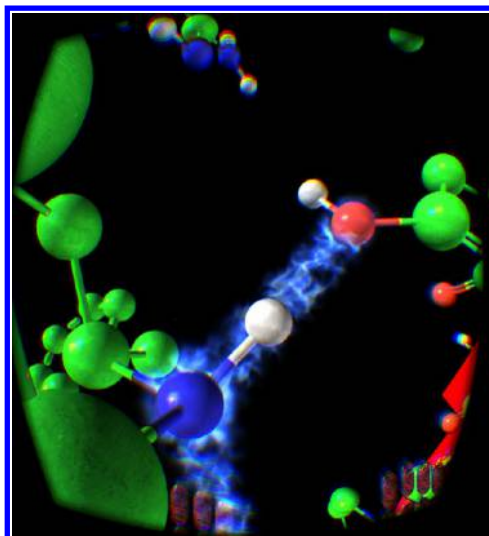

Figure 7. Intermolecular ligand−protein hydrogen bonds visualized using Unity's particle system.
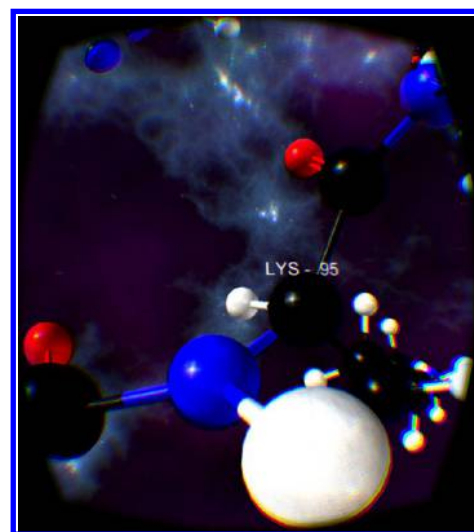

Figure 8. Residues can be labeled with residue type and number.

mented using Unity's TextMesh class, which includes functions for displaying 2D text in a 3D environment.

In addition, options for displaying pharmacophore models were added to Molecular Rift. Typical pharmacophore elements, defining essential features for specific ligand−protein interactions either located on the ligand itself or projected points (vectors) representing the protein, include hydrophobic regions, aromatic rings, H-bond acceptors and donors (including vectors), cations, anions, and excluded volumes. Figure 9 illustrates a CB1 receptor antagonist pharmacophore model,[35] for illustrational purposes. The PDB file format was used to store the information needed to render the pharmacophores (see the Supporting Information). In this context it should also be mentioned that intra- and intermolecular distances can be measured and work to allow future implementations of Open Babel available force fields (mmff94, uff, and ghemical) have been initialized.

Finally, to allow a real virtual reality experience the user position is monitored with the Oculus positional tracker and a 3-axis accelerometer. In this fashion, the virtual reality environment follows the users head movements resulting in a
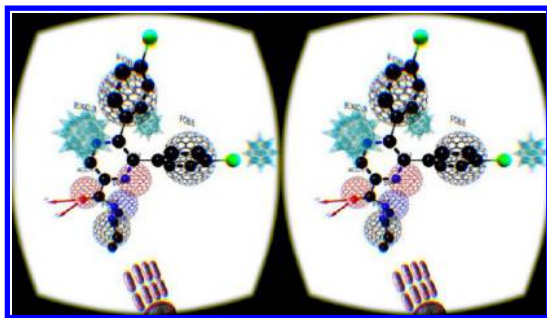
Figure 9. CB1 receptor antagonist pharmacophore model. The model includes three hydrophobic elements (black), two hydrogen-bond acceptor elements (red), one hydrogen-bond donor (blue), and three excluded/forbidden volumes (cyan).

real virtual reality experience. The "center of view" selection works analogous to eye-tracking. That is, the direction where the user is looking is predicted, by tracking the head rotation, and an atom straight in front of the head is selected. This method is also used for picking atoms when measuring distances.

Implementation of Gesture Recognition. Hand gestures have been studied extensively, especially in the field of sign language. For example, Stokoe[37] defined hand gestures by four components: hand shape, location, orientation, and movement. By accounting for these four components when interpreting gestures, we can allow users to have a natural and intuitive interaction with Molecular Rift. The implemented gesture recognition in Molecular Rift makes use of three specific steps: hand tracking, finger tracking, and thumb tracking. Gesture feedback was also added by implementing two in-game hands. These hands (Figure 3) are moved through animations to mimic the gesture interpreted by the system and help the user to control the interaction.

The hand tracking algorithm uses frames recorded from only one sensor: the infrared (IR) Kinect sensor. Initial tests using both the color and the IR sensor showed poor performance, most probably due to the huge amount of data from the two cameras that needed to be processed. An advantage of the IR sensor is that it is insensitive to colors. Furthermore, it allows depth-calculations, which was used for filtering away background noise (i.e. pixels from the IR frames). Figure 10 illustrates how the depth-map calculation using the IR sensor is performed.

The 3D position of the user's hand (the palm joint) was located using Kinect's native function (Body Basics-WPF). The palm location was combined with the depth map, which was obtained from another native Kinect function (Depth Basic-WPF). These were used to calculate the distance between the

Figure 10. Unmodified IR depth image of the scene (left), Kinect sensor detected joints (middle), and the interpreted hand (right). The extracted depth was used to filter out objects behind the identified hand. In this way, a model of the hand in the depth image could be generated.

hand and the sensor. A rectangle around the hand was subsequently defined, with the palm joint at its center, to increase performance by limiting the number of pixels needed to be analyzed. The depth of each point in the rectangle was extracted and compared to the depth of the palm. The algorithm successfully tracked the hand and required low computational resources since the number of pixel operations was narrowed down to approximately 1% of the entire IR frame. One drawback of this approach was that all objects close to the hand and at a similar depth also were selected. For instance, if a user's hand came in contact with his/her head, the head was interpreted as part of the hand, resulting in an incorrect gesture. This issue was, however, resolved by determining how well the hand was tracked, using nearby points and ignoring occasions when the hand was in the vicinity of "disturbing" objects to improve the gesture recognition. For example, by analyzing points defining the elbow angle, the location of the hand was predicted. The position of the head is easiest to predict since it is usually between the shoulders.

As done for hand tracking, the predefined Kinect joints were utilized for finger tracking by combining the palm joint with the wrist joint. An approximate direction for the identified hand was obtained by drawing a virtual line from the wrist to the palm joint, also limiting the number of pixels to be analyzed (see Figure 11). A second line was drawn at the palm joint
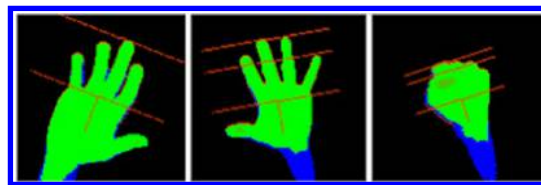

Figure 11. Direction line and two parallel lines defining the search area (left). The average edge distance for open (middle) and closed hand (right).

perpendicular to the first line, and a third line at the fingertip position farthest away from the first line. By traversing the pixels included in the search rectangle between the second and the third line in the direction from to palm joint (top) to the wrist joint (bottom), a method that successfully localizes all fingers was obtained. Open and closed hand gestures were treated by adding an average distance line (middle line in Figure 11) to the search area. Depending on the ratio between the farthest fingertip point and the average distance, the opened and closed hand status can be determined. A hand is considered to be closed if no point above 20% of the average distance is found. Finger gestures do not only involve recognizing all fingers but also identifying which finger is which. The thumb often plays a unique role in finger gestures and was tracked by a specific method. A detailed description of the approach used is available in the Supporting Information.

The MS Kinect v2 sensor tracks multiple bodies (up to six) by default and stores that information (of identified bodies) in a list. The active user (i.e. the "focus target") can be selected from this list by iterating through all recognized targets using their tracking ID. Once the end of the list is reached, the iteration starts over from the first entry. This was used to implement a function allowing switching between users. The first entry is the focus target by default.

Commands and Gestures to Control Molecular Rift. Controlling the VR in Molecular Rift is performed with

different hand gestures. The gestures implemented in Molecular Rift include movement, rotation, translation, zoom, center-view, toggle in-game menu, cursor movement, and cursor click. Various right- and left-hand combinations of four basic gestures were used; see Figure 12. For example, a closed
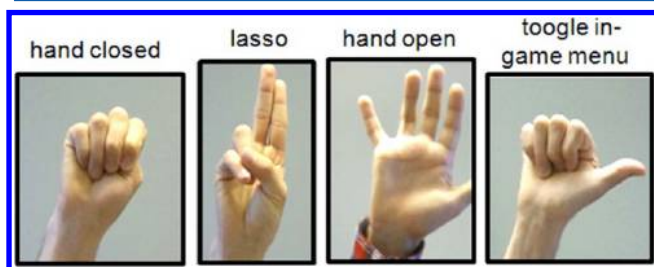

Figure 12. Overview of the four different basic gestures implemented in Molecular Rift. The user's left hand is shown.

right hand signals "movement", a left-hand lasso combined with a closed right hand is the gesture for rotating molecular objects, and the right-hand lasso gesture is used for zooming. A complete listing of the available hand gestures for controlling the system is available in the Supporting Information.

A challenge when working with HCI is to define accurate boundaries to evaluate if the recorded gesture was an intended interaction. For example, it was observed that a natural reaction when finishing an intended interaction was to lower the hand, often quite rapidly. This could cause issues such as making the molecule spin unintentionally. As a remedy, a function was added that extracted the height coordinates of the hand joints. Consequently, all gestures where the hand was not raised above the hip could be filtered out. This method successfully solved the unintended interactions issues and allowed users to lower their hands without being interpreted as an interaction gesture.

Focus Groups to Optimize Molecular Rift. User tests were employed to evaluate the virtual reality and the gestures from a usability perspective to improve Molecular Rift. For this purpose, an iterative design and development cycle including several focus groups was used to test and evaluate functionalities and thus provide instant feedback to guide the development. The first test group consisted of nine selected AstraZeneca employees, all of which had a background in medicinal chemistry and drug discovery. This smaller group of potential end-users allowed for in-depth instructions and was used to provide feedback about the usability of the program and detailed evaluations of all features. The two subsequent focus group evaluations were performed at two internal science symposiums and reached a much larger and diverse group of testers as compared to the first iteration. The larger test groups were used to gather information on the most common gestures, as well as on the general usability. These tests provided quantitative data about the UX which was found to be very useful for further improvements. The results from the three focus-group tests are given in more detail in the Supporting Information.

The user testing allowed us to identify issues, and Molecular Rift was improved considerably during the different development iterations. The main findings were that some of the gestures were difficult to perform for some of the testers, and that the sensitivity of the movement gesture needed to be fine-tuned. New modified gestures and a different sensitivity were implemented and subsequently re-evaluated, showing an improved rating by the users. To give a few examples, the

results from the first focus-group revealed that the in-game menu was accidentally launched on several occasions. The original gesture (for launching the in-game menu in the first version of Molecular Rift) was closing the right-hand above the head. It was found that this gesture could be triggered when another gesture (movement) was performed (when the user intended to move upward in the virtual reality). When comparing the usability scores for the two conflicting gestures the toggle in-game menu showed a poorer score and it was therefore re-engineered. Our solution was simply to shift the gesture from the right to the left hand (and extend the thumb). In the subsequent focus test, none of the 31 testers launched the in-game menu unintentionally, suggesting that it was an improvement. In addition, improvement ideas were collected during the symposiums. Suggestions that were implemented were the ability to measure atom—atom distances and the ability to change the center of view. That is, the center of view in the initial version of Molecular Rift was defined as the mean position of all atoms. Since a ligand—protein complex contains mostly protein atoms the rotation is effectively performed according to the protein chain positions. Many scientists expressed that they instead would like the option to rotate around the ligand; the main point of interest.

The focus-group tests revealed that some gestures were more difficult to recognize (by the sensor) than others. To improve the recognition of the gestures, machine learning via the Visual Gesture Builder was employed.[38] A range of users were recorded while performing the specific gestures, in alternation with random gestures. The recorded frames were subsequently marked as true or false, representing if the gesture was performed correctly or not correctly in the specified frame, and used to evaluate and design new gestures.

Data from surveying the third test group show that the 3D visual effects frequently exceeded the users' expectations. A summary is shown in Figure 13, revealing that the great
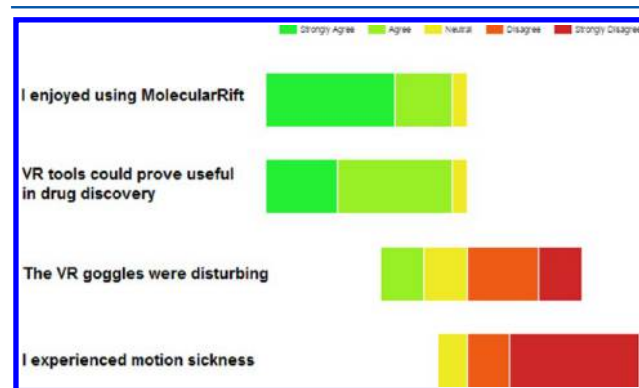

Figure 13. Most users (9 out of 14) strongly agreed with the statement that they enjoyed using Molecular Rift, revealing a very positive UX. Additionally, most users saw potential in the use of VR tools in drug discovery. None experienced motions sickness.

majority enjoyed Molecular Rift and ranked it as a useful tool in drug design for the future. Interestingly, motion-sickness, an often reported issue of VR, was not reported by any of the test users.

The focus-group evaluation showed that practice was required to perform the gestures accurately. Nevertheless, most, if not all, of the users could successfully execute all gestures and control the virtual reality system after instructions and a few minutes of training.

DISCUSSION AND OUTLOOK

We have developed an accessible virtual reality environment to interact with molecular models, such as protein–ligand complexes, using Oculus Rift and Microsoft Kinect within the framework of the Unity gaming development platform. Molecular Rift provides a unique experience and possibility of interacting in a virtual reality with molecular models. The general 3D visualization using Unity is on par with the industry standards, while also offering a new realistic VR experience. The main difference between conventional 3D computer graphics and virtual reality is that in VR you are working with things as opposed to pictures of things. Traditional molecular visualizers show molecules in front of you. With Molecular Rift you can step into ligand-protein complexes and feel like you are really there. The presence makes the difference. One reason for this is the wider field of view (FOV), another is head-tracking allowing the user to look in all directions.

There are development opportunities to improve the visualization further. First, adding a larger variety of options for rendering representation of molecular systems, often available in stand-alone tools (e.g. PyMol,[7] VIDA,[8] and VMD[9]), such as cropped surfaces (a feature requested by testers), will improve the programs usability. Second, allowing users to perform interactive molecular design, by adding options to modify molecules (adding/deleting atoms and bonds) and employing force field calculations to energy minimize structures, would increase its utility. Work along these lines is in progress. The integration of a cheminformatics toolkit (e.g., Open Babel) paves the way for achieving this goal (of adding new functionalities).

The program Molecular Rift was tested and evaluated in development cycles using customized focus groups. During the tests, Molecular Rift was found to be well accepted and often praised with enthusiasm. The word "cool" was frequently used by the users when assessing the tool. However, there is some room for improvement with respect to the human–computer interactions. The virtual reality in Molecular Rift is controlled by specific hand motions. The human–computer interaction using gestures requires initial training and the gestures we defined may be further optimized to improve the interaction efficiency and learnability. In the future, other interaction devices such as new gaming controllers may be investigated. In fact, future versions of Oculus Rift are reported to include custom optics, sensor systems, and algorithms to detect and track hand movement.[39,40] In this context, it should be noted that the first official version of Oculus Rift is expected to be released in the beginning of next year (2016).

The Kinect v2 sensor is equipped with an array of microphones. We did explore voice recognition as an option to interpret sound input. However, due to an apparent lack of precision, voice control was not added to Molecular Rift. Nevertheless, acoustic feedback using Unity's AudioSource class was added to signal activating and deactivating of the in-game menu as well as for the in-game menu selections to improve usability.

With respect to impact, who would have envisaged ten years ago that the smartphones would be in almost every person's hand? We did not. Similarly, the vision of VR as a routine tool may be considered naïve within the area of molecular visualizers. But, it should not be forgotten that a considerable part of the drug designer's day-to-day job revolves around exactly this—visualizing and interacting with molecules. The

quest toward optimizing the tools at hand, as well as developing new ones must go on. To speculate about the future, VR may go beyond merely visualizing ligands and proteins. Imagine being able to take a virtual stroll in a cell or, say, regions in the brain and observe down/up regulation of the targeted pathways.[41]

Additional applications of virtual reality could attract the general interest in life science and education. In addition for Molecular Rift to be used within drug hunting projects in a pharmaceutical setting we anticipate a role in academia. That is, chemistry as a university subject is nowadays quite unpopular among students. In many countries, such as in Sweden, a gloomy picture is reported; there is a steady decline in the number of students studying chemistry.[42] Explanations for the decline have been sought for, and studies report that chemistry education must be "more relevant" in order to better motivate and attract students.[43] The phrase more relevant may be seen as somewhat vague and what it actually entails remains largely unclear. Nevertheless, developing chemistry courses that engage students more and inspire them to further studies in the field is required, without a doubt. In the current era when kids spend a lot of time playing video and/or computer games we see opportunities to gamify chemistry.[43] When designed correctly, gamification has proven to be successful in engaging people to change behaviors, develop skills, and solve problems, in many different areas. VR and gamifying Molecular Rift might be one way to motivate students and inspire learning.

We have made the Molecular Rift code open source, and it can be downloaded for free from GitHub. We are certainly not alone making our code available. For example, Google gives away a lot of their software for free. The Apache web server as well as the Hadoop big data crunching platform have been governed by this. Thus, much of the code in the world is freely available, created by clever and generous commercial and noncommercial contributors cross the globe to serve their own needs as well as providing spring-boards for others. This give-and-take relationship is complex, but we do hope that sharing the Molecular Rift code will stimulate further development in a collaborative fashion.

CONCLUSIONS

The tool Molecular Rift represents a new generation of molecular visualization tools. It combines a real virtual reality (VR) experience with a natural and interactive way of controlling the computer, as compared to the conventional way of using artificial physical objects such as mouse and keyboards. The unique and realistic 3D effects offered by the VR demonstrate potential when investigating various molecular systems. For example, protein–ligand complexes can be viewed in a completely new way. In this case, drug designers can step inside a protein and be fully immersed in the scene. Furthermore, by employing SDK's from Unity, Oculus Rift, Microsoft Kinect, and Open Babel, a flexible, collaborative, and high-quality virtual reality development platform was achieved. In this manner, Molecular Rift could easily be used and enhanced by anyone at a relatively low cost (see Minimum Requirements for Running Molecular Rift below). The implemented MS Kinect v2 gesture control requires some training, but according to our tests Molecular Rift users quickly adapted to the new interaction system. Molecular Rift was developed with a focus on usability, and it received very positive feedback according to several focus groups.

Figure 14. Oculus Rift (HMD) and Kinect v2 sensor placement used during Molecular Rift development.

The Molecular Rift code is open source available for free from GitHub. It is designed with the aim to bring virtual reality to drug designers cross the globe. To fully understand the true potential of virtual reality, and tools like Molecular Rift, there is really one option, and that is to test them.

AVAILABILITY AND REQUIREMENTS

The code can be downloaded for free from GitHub: http://github.com/Magnusnorrby/MolecularRift.

Setup of Hardware. For optimal performance of Molecular Rift, both the Kinect Sensors and the Oculus Position Tracker need specific placements. Microsoft states that the Kinect v2 correctly tracks humans at distances between 0.5 and 4.5 m with its best performance in the 0.8–3.5 m range.[45] The sensor has a 70° horizontal and 60° vertical angle of vision. It was decided to place the Kinect sensor above the computer screen at a height of approximately 1.8 m with a slight downward tilt. This placement allowed the tracker to monitor the interacting users from the waist up when standing in the 0.8–3.5 m range. By positioning the Kinect at the center of the screen it was also ensured that users placed straight in front of the screen would be optimally tracked. The Oculus positional tracker receives IR-signals from the HMD and thus need unobscured vision of it. It was decided to place it at the top of the computer screen. The position of both sensors can be seen in Figure 14.

Minimum Requirements for Running Molecular Rift.

- Microsoft Kinect v2 for Windows
- Oculus Rift Head Mount Display DK 2
- Oculus Positional Tracker DK 2
- Unity 5
- MonoDevelop with MonoC# compiler
- PC with
  - Windows 8 or 8.1 (x64)
  - 1x USB 3.0
  - 2x USB 2.0 or higher (at least one powered)
  - DVI-D or HDMI graphics output
  - DX11 capable graphics adapter
  - 4 GB memory

Assuming you are already in the possession of a fairly standard PC (as outlined above) the cost for this VR environment is currently around 500 USD (Kinect v2 USD 150 and Oculus Rift USD 350). The MolecularRift executable requires no additional setup allowing instant testing.

ASSOCIATED CONTENT

Ⓢ Supporting Information

The Supporting Information is available free of charge on the ACS Publications website at DOI: 10.1021/acs.jcim.5b00544.

> Sample pharmacophore input file format, the thumb tracking algorithm, complete list of available hand gestures, and focus group results (PDF)

AUTHOR INFORMATION

Corresponding Author
*E-mail: jonas.bostrom@astrazeneca.com.

Notes
The authors declare no competing financial interest.

ACKNOWLEDGMENTS

ABBREVIATIONS

SDK, software development kit; PDB, protein database; UX, user experience; GUI, graphical user interface; IR, infrared; FOV, field of view; 3D, three-dimensional

REFERENCES

(1) Riddell, F. G.; Robinson, M. J. T. J. H. van't Hoff and J. A. Le Bel—Their Historical Context. *Tetrahedron* **1974**, *30*, 2001–2007.
(2) Kendrew, J. C.; Bodo, G.; Dintzis, H. M.; Parrish, R. G.; Wyckoff, H.; Phillips, D. C. A Three-dimensional Model of the Myoglobin Molecule Obtained by X-ray Analysis. *Nature* **1958**, *181*, 662–666.
(3) Anderson, A. C. The Process of Structure-Based Drug Design. *Chem. Biol.* **2003**, *10*, 787–797.

(4) Watson, J. D.; Crick, F. H. C. Molecular Structure of Nucleic Acids: a Structure for Deoxyribose Nucleid Acid. *Nature* **1953**, *171*, 737−738.

(5) Feldmann, R. J.; Bing, D. H. *TAMS: Teaching Aids for Macromolecular Structure*; Teachers manual, Division of Computer Research and Technology (DCRT), NIH/PHS/DHEW, 1980.

(6) Sayle, R. A.; Milner-White, E. J.; RASMOL. RASMOL: Biomolecular Graphics for All. *Trends Biochem. Sci.* **1995**, *20*, 374−376.

(7) PyMol. http://www.pymol.org/pymol (accessed 2015/08/27).

(8) VIDA. http://www.eyesopen.com/vida (accessed 2015/08/27).

(9) Humphrey, W.; Dalke, A.; Schulten, K. VMD: Visual Molecular Dynamics. *J. Mol. Graphics* **1996**, *14*, 33−38.

(10) Pettersen, E. F.; Goddard, T. D.; Huang, C. C.; Couch, G. S.; Greenblatt, D. M.; Meng, E. C.; Ferrin, T. E. UCSF Chimera— A Visualization System for Exploratory Research and Analysis. *J. Comput. Chem.* **2004**, *25*, 1605−1612.

(11) Jmol: An Open-source Java Viewer for Chemical Structures in 3D. http://www.jmol.org (accessed 2015/08/27).

(12) Hartshorn, M. J. AstexViewer: a Visualisation Aid for Structure-based Drug Design. *J. Comput.-Aided Mol. Des.* **2002**, *16*, 871−81.

(13) Li, H.; Leung, K.-S.; Nakane, T.; Wong, M.-H. iView: an Interactive WebGL Visualizer for Protein-Ligand Complex. *BMC Bioinf.* **2014**, *15*, 56.

(14) OEChem. http://www.eyesopen.com/oechem-tk (accessed 2015/08/27).

(15) Anaglyph 3D. http://en.wikipedia.org/wiki/Anaglyph_3D (accessed 2015/08/27).

(16) ChromaDepth. http://en.wikipedia.org/wiki/ChromaDepth (accessed 2015/08/27).

(17) NVIDIA 3D Vision. http://www.nvidia.com/object/3d-vision-main.html (accessed 2015/08/27).

(18) Virtual Reality. http://en.wikipedia.org/wiki/Virtual_reality (accessed 2015/08/27).

(19) Google Cardboard. http://www.google.com/get/cardboard (accessed 2015/08/27).

(20) Cave Automatic Virtual Environment (CAVE) https://en.wikipedia.org/wiki/Cave_automatic_virtual_environment (accessed 2015/10/14).

(21) Reality Cube. http://www.rug.nl/society-business/centre-for-information-technology/research/hpcv/faciliteiten/realitycube (accessed 2015/10/14).

(22) Sakuta, M. OculusCrystal. http://github.com/msakuta/OculusCrystal (accessed 2015/08/27).

(23) Microsoft Kinect. http://www.microsoft.com/en-us/kinectforwindows (accessed 2015/08/27).

(24) Alnowami, M.; Khan, A.; Morfeq, A. H.; Alothmany, N.; Hafez, E. A. Feasibility Study of Markerless Gait Tracking Using Kinect. *Life Sci. J.* **2014**, *11*, 514−523.

(25) Galna, B.; Jackson, D.; Schofield, G.; McNaney, R.; Webster, M.; Barry, G.; Mhiripiri, D.; Balaam, M.; Olivier, P.; Rochester, L. Retraining Function in People with Parkinson's Disease Using the Microsoft Kinect: Game Design and Pilot Testing. *J. Neuroeng. Rehabil.* **2014**, *11*, 60−60.

(26) OculusRift SDK. http://developer.oculus.com/downloads/pc/0.6.0.1-beta/Oculus_SDK_for_Windows (accessed 2015/08/27).

(27) Microsoft Kinect 2.0 SDK. http://www.microsoft.com/en-us/download/details.aspx?id=44561 (accessed 2015/08/27).

(28) Unity 5. http://unity3d.com (accessed 2015/08/27).

(29) O'Boyle, N. M.; Banck, M.; James, C. A.; Morley, C.; Vandermeersch, T.; Hutchison, G. R. Open Babel: An Open Chemical Toolbox. *J. Cheminf.* **2011**, *3*, 33−33.

(30) OpenBabel Wiki. http://openbabel.org/wiki (accessed 2015/08/27).

(31) Bernstein, F. C.; Koetzle, T. F.; Williams, G. J. B.; Meyer, E. F.; Brice, M. D.; Rodgers, J. R.; Kennard, O.; Shimanouchi, T.; Tasumi, M. The Protein Data Bank. *Eur. J. Biochem.* **1977**, *80*, 319−324.

(32) Berman, H. M.; Westbrook, J.; Feng, Z.; Gilliland, G.; Bhat, T. N.; Weissig, H.; Shindyalov, I. N.; Bourne, P. E. The Protein Data Bank. *Nucleic Acids Res.* **2000**, *28*, 235−242.

(33) Bondi, A. van der Waals Volumes and Radii. *J. Phys. Chem.* **1964**, *68*, 441−451.

(34) StarManta. TubeRenderer. http://wiki.unity3d.com/index.php?title=TubeRenderer (accessed 2015/08/27).

(35) Unity 3D: Particle system. http://docs.unity3d.com/ScriptReference/ParticleSystem.html (accessed 2015/08/27).

(36) Boström, J.; Berggren, K.; Elebring, T.; Greasley, P. J.; Wilstermann, M. Scaffold hopping, Synthesis and Structure−Activity Relationships of 5,6-diaryl-pyrazine-2-amide Derivatives: a Novel Series of CB1 Receptor Antagonists. *Bioorg. Med. Chem.* **2007**, *15*, 4077−4084.

(37) Stokoe, W. C. Sign Language Structure: an Outline of the Visual Communication Systems of the American Deaf. *J. Deaf Stud. Deaf Educ.* **2005**, *10*, 3−37.

(38) Lower, B. Custom Gestures End to End with Kinect and Visual Gesture Builder. http://channel9.msdn.com/Blogs/k4wdev/Custom-Gestures-End-to-End-with-Kinect-and-Visual-Gesture-Builder (accessed 2015/08/27).

(39) Pebbles Interfaces Joins Oculus. https://www.oculus.com/en-us/blog/pebbles-interfaces-joins-oculus (accessed 2015/06/16).

(40) "Toybox Demo for Oculus Touch" at The official Oculus YouTube account. http://www.youtube.com/watch?v=dbYP4bhKr2M (accessed 2015/10/14).

(41) Murcko, M. A.; Walters, P. W. Alpha Shock. *J. Comput.-Aided Mol. Des.* **2012**, *26*, 97−102.

(42) Broman, K.; Ekborg, M.; Johnels, D. Chemistry in Crisis? Perspectives on Teaching and Learning Chemistry in Swedish Upper Secondary Schools. *NorDiNa* **2011**, *7*, 43−60.

(43) Eilks, I.; Hofstein, A. From Some Historical Reflections on the Issue of Relevance of Chemistry Education Towards a Model and an Advance Organizer—a Prologue. *In Relevant chemistry Education - From Theory to Practice*, Eilks, I.; Hofstein, A, Eds.; Sense: Rotterdam, 2015; pp 1−10.

(44) Boström, J. A World Without New Drugs—Play It Before You Live It, Pharma IQ blog "On Drug Design". http://www.pharma-iq.com/pre-clinical-discovery-and-development/columns/a-world-without-new-drugs-play-it-before-you-live (accessed 2015/08/27).

(45) Microsoft Corporation, Human Interface Guidelines (HIG). http://go.microsoft.com/fwlink/?LinkID=403900&clcid=0x409 (accessed 2015/06/15).